aws

Guide de l'utilisateur

Amazon Relational Database Service



Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon Relational Database Service: Guide de l'utilisateur

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Les marques et la présentation commerciale d’Amazon ne peuvent être utilisées en relation avec un produit ou un service qui n’est pas d’Amazon, d’une manière susceptible de créer une confusion parmi les clients, ou d’une manière qui dénigre ou discrédite Amazon. Toutes les autres marques commerciales qui ne sont pas la propriété d'Amazon sont la propriété de leurs propriétaires respectifs, qui peuvent ou non être affiliés ou connectés à Amazon, ou sponsorisés par Amazon.

Table of Contents

Qu'est-ce que Amazon RDS ? 1
Avantages d'Amazon RDS 1
Comparaison des responsabilités 2
Modèle de responsabilité partagée Amazon RDS 3
Instances de base de données 4
Architecture d'application Amazon RDS : exemple 5
Moteurs de base de données 6
Classes d'instances de base de données 7
Stockage d'instance de base de données 8
Instances de base de données dans un Amazon Virtual Private Cloud (Amazon VPC) 9
Régions AWS et zones de disponibilité 9
Zones de disponibilité 10
Déploiements multi-AZ 10
Contrôle d'accès avec des groupes de sécurité 12
Surveillance Amazon RDS 14
Interfaces utilisateur pour Amazon RDS 15
AWS Management Console 15
interface de ligne de commande 16
Amazon RDS APIs 16
Comment fonctionne la facturation pour Amazon RDS 17
Niveau gratuit d'AWS sur Amazon RDS 17
Quelle est la prochaine étape ? 18
Mise en route 18
Sujets spécifiques aux moteurs de bases de données 18
Instances DB 20
Classes d'instances de base de données 23
Types de classes d'instance de base de données 23
Moteurs de base de données pris en charge 33
Déterminer le support des classes d'instance de base de données dans Régions AWS 103
Modification d'une classe d'instance de base de données 109
Configuration du processeur pour RDS for Oracle 109
Spécifications matérielles 134
Stockage d'instance de base de données 183
Types de stockage 183

Stockage sur volumes IOPS provisionnés 184
Stockage à usage général 188
Caractéristiques de performance du SSD 193
Stockage magnétique (obsolète) 198
Volume de journal dédié (DLV) 198
Surveillance des performances des bases de données 199
Facteurs influant sur les performances des bases de données 201
Régions, zones de disponibilité et zones locales 205
AWS Régions 206
Zones de disponibilité 212
Zones locales 212
RDSFonctionnalités Amazon prises en charge par région et par moteur 215
Conventions de tableau 216
Référence rapide des fonctionnalités 216
Déploiements bleu/vert 219
Sauvegardes automatiques interrégionales 220
Réplicas en lecture entre Régions 222
Flux d'activité de base de données. 224
Mode double pile 232
Exporter les instantanés vers S3 254
Authentification de base de données IAM 268
Authentification Kerberos 273
Clusters de base de données multi-AZ 289
Performance Insights 299
RDS Custom 300
Proxy Amazon RDS 314
Intégration de Secrets Manager 332
Intégrations zéro ETL 332
Fonctions natives du moteur 338
Facturation des instances de base de données pour Amazon RDS 339
Instances de base de données à la demande 341
Instances de base de données réservées 343
Configuration 359
Inscrivez-vous pour un Compte AWS 359
Création d’un utilisateur doté d’un accès administratif 360
Octroi d’un accès par programmation 361

Déterminer les exigences 363
Fournir un accès à votre instance de base de données 366
Mise en route 370
Création d'une instance de base de données MariaDB et connexion à cette instance 371
Prérequis 372
Création d'une EC2 instance 373
Création d'une instance de base de données MariaDB 379
(Facultatif) Créez un VPC, une instance et une EC2 instance MariaDB en utilisant AWS CloudFormation 385
Se connecter à une instance de base de données MariaDB 387
Supprimer l' EC2 instance et l'instance de base de données 391
(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation 392
(Facultatif) Connecter votre instance de base de données à une fonction Lambda 393
Création et connexion à une instance de base de données Microsoft SQL Server 394
Prérequis 396
Création d'une instance EC2 396
Création d'une instance de base de données SQL Server 401
(Facultatif) Créez un VPC, une instance et une EC2 instance SQL Server à l'aide de AWS CloudFormation 407
Connexion à votre instance de base de données SQL Server 409
Exploration de votre exemple d'instance de base de données 413
Supprimer l' EC2instance et l'instance de base de données 414
(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation 415
(Facultatif) Connecter votre instance de base de données à une fonction Lambda 416
Création d'une instance de base de données MySQL et connexion à cette instance 417
Prérequis 418
Création d'une instance EC2 419
Création d'une instance de base de données MySQL 425
(Facultatif) Créez un VPC, une instance et une EC2 instance MySQL en utilisant AWS CloudFormation 431
Se connecter à une instance de base de données MySQL 433
Supprimer l' EC2 instance et l'instance de base de données 437
(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation 438

(Facultatif) Connecter votre instance de base de données à une fonction Lambda 439
Création et connexion à une instance de base de données Oracle 440
Prérequis 441
Étape 1 : créer une EC2 instance 442
Étape 2 : Créer une instance de base de données Oracle 448
(Facultatif) Créez un VPC, une instance et une EC2 instance de base de données Oracle à l'aide de AWS CloudFormation 454
Étape 3 : Connecter votre client SQL à une instance de base de données Oracle 456
Étape 4 : Supprimer l' EC2 instance et l'instance de base de données 460
(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation 461
(Facultatif) Connecter votre instance de base de données à une fonction Lambda 461
Création et connexion à une instance de base de données PostgreSQL 463
Prérequis 464
Création d'une EC2 instance 465
Création d'une instance de base de données PostgreSQL 471
(Facultatif) Créez un VPC, une instance et une EC2 instance PostgreSQL à l'aide de AWS CloudFormation 477
Se connecter à une instance DB PostgreSQL 479
Supprimer l' EC2instance et l'instance de base de données 483
(Facultatif) Supprimez l'EC2 instance et l'instance de base de données créées avec CloudFormation 484
(Facultatif) Connecter votre instance de base de données à une fonction Lambda 485
Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS 486
Lancez une EC2 instance pour vous connecter à votre instance de base de données de de bases de données 488
Créez une instance de base de données. 494
Installer un serveur web 515
Tutoriel : Créer une fonction Lambda pour accéder à votre instance de base de données Amazon RDS 527
Prérequis 528
Créer une instance de base de données Amazon RDS 528
Création d'une fonction Lambda et d'un proxy 530
Pour créer un rôle d'exécution de fonction 531
Création d'un package de déploiement Lambda 532
Mise à jour de la fonction Lambda 535

Test de votre fonction Lambda dans la console 536
Créez une file d’attente Amazon SQS. 538
Création d'un mappage des sources d'événements pour invoquer votre fonction Lambda 538
Test et surveillance de votre configuration 539
Nettoyage de vos ressources 540
Tutoriels et exemple de code 542
Tutoriels dans ce guide 542
Tutoriels dans d'autres AWS guides 544
AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS 544
AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS 545
Tutoriels et exemples de code dans GitHub 545
Travailler avec AWS SDKs 545
Bonnes pratiques relatives à Amazon RDS. 547
Directives opérationnelles de base Amazon RDS 547
Recommandations RAM d'une instance de base de données 549
Maintien à jour des versions du moteur de base de données 549
AWS pilotes de base de données 550
Utilisation de la surveillance améliorée 550
Utilisation de métriques 550
Consultation des métriques de performances 551
Évaluation des métriques de performances 554
Réglage des requêtes 556
Bonnes pratiques d'utilisation de MySQL 557
Taille des tables 557
Nombre de tables 558
Moteur de stockage 559
Bonnes pratiques d'utilisation de MariaDB 560
Taille des tables 560
Nombre de tables 561
Moteur de stockage 561
Bonnes pratiques d'utilisation d'Oracle 562
Bonnes pratiques pour utiliser les moteurs de stockage PostgreSQL 562
Chargement des données dans une instance de base de données PostgreSQL 562
Utilisation de la fonction autovacuum de PostgreSQL 563
Vidéo des bonnes pratiques Amazon RDS for PostgreSQL 565
Bonnes pratiques pour l'utilisation de SQL Server 565

Vidéo des bonnes pratiques Amazon RDS pour SQL Server 566
Utilisation des groupes de paramètres DB 566
Automatisation de la création d'instances de base de données 567
Vidéo sur les nouvelles fonctionnalités 568
Accès programmatique à Amazon RDS Amazon 569
Console-to-Code 570
Configuration d'une instance de base de données 571
Création d'une instance de base de données 572
Prérequis 572
Création d'une instance de base de données 579
Paramètres disponibles 587
Création de ressources avec AWS CloudFormation 634
RDS et AWS CloudFormation modèles 634
En savoir plus sur AWS CloudFormation 634
Connexion à une instance de base de données 635
Recherche des informations de connexion 635
Scénarios d'accès à une instance de base de données 639
Connexion aux instances de base de données avec les AWS pilotes 640
Connexion à une instance de base de données exécutant un moteur de base de données spécifique 642
Gestion des connexions avec RDS Proxy 642
Options d'authentification de base de données 642
Connexions chiffrées 643
Utilisation de groupes d'options 644
Présentation des groupes d'options 644
Création d'un groupe d'options 647
Copie d'un groupe d'options 649
Ajout d'une option à un groupe d'options 650
Liste des options et des paramètres d'options pour un groupe d'options 657
Modification d'un paramètre d'option 658
Suppression d'une option d'un groupe d'options 662
Suppression d'un groupe d'options 664
Groupes de paramètres 668
Présentation des groupes de paramètres 668
Groupes de paramètres DB 673
Groupes de paramètres de cluster DB 691

Comparaison des groupes de paramètres de bases de données 705
Spécification des paramètres de base de données 706
Création d'un ElastiCache cache depuis Amazon RDS 714
Vue d'ensemble de la création du ElastiCache cache avec les paramètres de l'instance de base de données du RDS cluster 714
Création d'un ElastiCache cache avec les paramètres d'une instance de base de données du cluster de RDS base 716
Migration automatique EC2 des bases de données 719
Présentation 719
Prérequis 720
Limites 722
Création de ressources IAM 722
Configurer la migration des données 730
Gérer les migrations 732
Surveillance 735
Tutoriel : Création d'un de base de données MySQL avec un paramètre personnalisé et un nouveau groupe d'options 737
Prérequis 738
Création d'un groupe de paramètres personnalisé 738
Modifier la des paramètres 739
Créer un groupe d'options 739
Ajouter une option 740
Créez votre de base de données 741
Gestion d'une instance de base de données 743
Arrêt d'une instance de base de données 744
Cas d’utilisation 745
Moteurs, classes et régions de base de données pris en charge 746
Prise en charge d'un déploiement multi-AZ 746
Comment ça marche 746
Limites 748
Groupes d'options et de paramètres 749
Adresses IP publiques 749
Arrêt d'une instance de base de données 749
Démarrage d'une instance de base de données 751
Redémarrage d'une instance DB 753

Cas d'utilisation pour le redémarrage d'une instance de base de données cluster de base de données ... 753
Comment fonctionne le redémarrage ... 754
Redémarrage en mode multi-AZ ... 755
Considérations ... 756
Prérequis ... 756
Redémarrage d'une instance de base de données de base ... 756
Connecter une EC2 instance ... 759
Présentation ... 760
Connecter une EC2 instance ... 765
Affichage des ressources de calcul connectées ... 768
Connexion à une instance de base de données exécutant un moteur de base de données spécifique ... 769
Connexion d'une fonction Lambda ... 770
Présentation ... 772
Connexion d'une fonction Lambda ... 783
Affichage des ressources de calcul connectées ... 785
Modification d'une instance de base de données ... 787
Modifications de planification ... 789
Paramètres disponibles ... 790
Entretien d'une instance de base de données ... 839
Vue d'ensemble des mises à jour relatives à la maintenance de bases de données ... 839
Affichage de la maintenance en attente ... 841
Maintenance pour les déploiements multi-AZ ... 843
Fenêtre de maintenance ... 844
Application des mises à jour ... 850
Mises à jour du système d'exploitation ... 852
Mise à niveau de la version du moteur ... 857
Mise à niveau manuelle de la version du moteur ... 858
Mise à niveau automatique de la version mineure du moteur ... 860
Affectation d'un nouveau nom à une instance DB ... 866
Renommer pour remplacer une instance de base de données existante ... 867
Utilisation des réplicas en lecture d'instance de base de données ... 870
Présentation ... 872
Différences entre les moteurs de base de données ... 876
Création d'un réplica en lecture ... 880

Promotion d'un réplica en lecture 884
Supervision de la réplication en lecture 890
Réplicas en lecture entre Régions 893
Marquage d' RDS de ses ressources 908
Pourquoi utiliser des RDS tags ? 909
Comment fonctionnent les RDS tags 910
Bonnes pratiques 913
Copier des balises dans des instantanés de base de données 914
Ajouter et supprimer des tags sur Amazon RDS 915
Tutoriel : Spécifiez les instances de base de données à arrêter à l'aide de balises 920
ARNssur Amazon RDS 924
Construction d'un ARN 924
Obtenir un existant ARN 932
Utilisation du stockage 935
Augmentation de la capacité de stockage d'une instance de base de données 935
Gestion automatique de la capacité avec le dimensionnement automatique du stockage 938
Mise à niveau du système de fichiers de stockage 948
Modification des paramètres provisionnés IOPS 950
Modifications du stockage à forte intensité d'I/O 952
Modification des paramètres de stockage à usage général (gp3) 953
Utilisation d'un volume dédié aux journaux (DLV) 956
Suppression d'une instance DB 963
Conditions préalables pour la suppression d'une instance de base de données 963
Considérations lors de la suppression d'une instance de base de données 963
Suppression d'une instance DB 965
Tutoriel : Gestion d'une instance My SQL DB 969
Introduction 969
Prérequis 970
Ajoutez des balises à votre instance de base de données 970
Augmenter le stockage de base de données 970
Créer des réplicas en lecture 971
Mettre à jour des balises 972
Configuration et gestion d'un déploiement multi-AZ 976
Déploiements d'instances de base de données multi-AZ 978
Conversion d'une instance de base de données en déploiement multi-AZ 980
Défaillance d'une instance de base de données multi-AZ 982

Déploiements de clusters de base de données multi-AZ 989
Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ 990
Architecture de cluster de bases de données multi-AZ 991
Groupes de paramètres pour les clusters de bases de données multi-AZ 993
RDSProxy avec clusters de bases de données multi-AZ 993
Retard de réplica et clusters de base de données multi-AZ 993
Instantanés de clusters de bases de données multi-AZ 997
Création d'un cluster de base de données multi-AZ 998
Connexion à un cluster de base de données multi-AZ 1031
Connexion d'une ressource AWS de calcul et d'un cluster de base de données multi-AZ ... 1038
Modification d'un cluster de base de données multi-AZ 1066
Mise à niveau d'un cluster de base de données multi-AZ 1094
Renommage d'un cluster de bases de données multi-AZ 1097
Redémarrage d'un cluster de base de données multi-AZ 1100
Défaillance d'un cluster de base de données multi-AZ 1102
Réplication SQL logique Postgre avec des clusters de bases de données multi-AZ 1106
Utilisation des réplicas en lecture d'un cluster de base de données multi-AZ 1111
Configuration de la réplication externe à partir de clusters de bases de données multi-AZ . 1125
Suppression d'un cluster de base de données multi-AZ 1127
Limites des clusters de bases de données multi-AZ 1130
Support étendu RDS 1132
Présentation du support étendu RDS 1133
RDSFrais de Support étendu 1134
Éviter les frais liés RDS au Support étendu 1135
Versions avec support étendu RDS 1135
Dénomination des versions de RDS Extended Support 1135
Responsabilités liées RDS au Support étendu 1136
Responsabilités RDS 1137
Vos responsabilités 1137
Création d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster 1138
RDSComportement du Support étendu 1138
Considérations relatives RDS au Support étendu 1139
Créez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support 1140
Afficher l'inscription au RDS Extended Support 1141

Afficher les dates de support 1144
Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster 1147
RDSComportement du Support étendu 1148
Considérations relatives RDS au Support étendu 1148
Restaurez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support 1149
Utilisation des déploiements bleu/vert pour les mises à jour de base de données 1152
Vue d'ensemble des déploiements bleu/vert 1153
Disponibilité des régions et des versions 1154
Avantages 1154
Flux de travail 1155
Autorisation de l’accès 1159
Limites et considérations 1160
Bonnes pratiques 1169
Méthodes de SQL réplication Postgre 1174
Création d'un déploiement bleu/vert 1175
Préparation d'un déploiement bleu/vert 1176
Spécification des modifications 1179
Chargement différé et initialisation du stockage 1182
Création d'un déploiement bleu/vert 1184
Paramètres disponibles 1186
Affichage d'un déploiement bleu/vert 1192
Basculement d'un déploiement bleu/vert 1196
Délai de commutation 1197
Barrières de protection de commutation 1197
Actions de commutation 1198
Bonnes pratiques de commutation 1199
Vérification des CloudWatch métriques avant le passage au numérique 1201
Surveillance du délai de réplication avant le passage au numérique 1201
Basculement d'un déploiement bleu/vert 1202
Après la commutation 1205
Suppression d'un déploiement bleu/vert 1207
Sauvegarde, restauration et exportation de données 1211
Présentation des sauvegardes 1212
Stockage de sauvegarde 1213

Gestion des sauvegardes automatisées 1214
Fenêtre de sauvegarde 1214
Période de rétention des sauvegardes 1217
Activation des sauvegardes automatiques 1218
Conservation des sauvegardes automatiques 1221
Suppression des sauvegardes automatisées conservées 1223
Mes moteurs de SQL stockage non pris en charge 1227
Moteurs de stockage MariaDB non pris en charge 1228
Sauvegardes automatiques interrégionales 1230
Gestion des sauvegardes manuelles 1249
Création d'un instantané de base de données pour une instance de base de données mono-AZ 1250
Création d'un instantané de cluster de bases de données multi-AZ 1253
Suppression d'un instantané de base de données 1255
Restauration vers une instance de base de données 1258
Restaurer à partir d'un instantané 1259
Considérations 1261
Point-in-time rétablissement 1266
Restauration d'un cluster de base de données multi-AZ à une date définie 1272
Restauration d'un instantané dans un cluster de base de données multi-AZ 1276
Restauration d'un instantané de cluster de bases de données multi-AZ dans une instance de base de données 1280
Didacticiel : restaurer une instance de base de données à partir d'un instantané de base de données 1283
Copie d'un instantané de base de données 1288
Copie d'un instantané de base de données 1288
Limites 1299
Considérations 1300
Partage d'un instantané de de base de données 1310
Partage d'un instantané 1312
Partage d'instantanés publics 1316
Partage d'instantanés chiffrés 1318
Arrêter le partage de snapshots 1322
Exportation de données d'instantanés de bases de données vers Amazon S3 1325
Présentation de l'exportation des données d'instantané 1326
Configuration de l'accès à un compartiment S3 1327

Exportation d'un instantané de base de données ... 1333
Disponibilité des régions et des versions ... 1337
Limites ... 1337
Surveillance des exportations d'instantanés ... 1339
Annulation d'une exportation d'instantané ... 1341
Messages d'échec ... 1342
Résolution des erreurs liées aux SQL autorisations Postgre ... 1344
Conventions de dénomination des fichiers ... 1345
Conversion des données ... 1346
En utilisant AWS Backup ... 1357
Surveillance des métriques dans une instance de base de données ... 1358
Plan de surveillance ... 1358
Référence des performances ... 1359
Instructions sur les performances ... 1359
Outils de surveillance ... 1361
Outils de surveillance automatique ... 1361
Outils de surveillance manuelle ... 1363
Affichage de l'état ... 1365
Affichage de l'état de l'instance de base de données dans un cluster Aurora ... 1366
Recommandations d'Amazon RDS ... 1374
Affichage des recommandations ... 1376
Appliquer les recommandations ... 1384
Rejet des recommandations ... 1390
Modifier les recommandations rejetées en recommandations actives ... 1392
Référence aux recommandations ... 1393
Afficher les métriques dans la RDS console Amazon ... 1425
Consulter le tableau de bord Performance Insights ... 1426
Choix de la nouvelle vue de surveillance dans l'onglet Surveillance ... 1427
Choix de la nouvelle vue de surveillance sur la page Performance Insights ... 1429
Création d'un tableau de bord personnalisé ... 1431
Choix du tableau de bord préconfiguré ... 1434
Surveillance du RDS avec CloudWatch ... 1436
Présentation d'Amazon RDS, d' et d'Amazon CloudWatch ... 1437
Afficher CloudWatch les métriques ... 1439
Exportation des indicateurs de Performance Insights vers CloudWatch ... 1445
Création d' CloudWatch alarmes ... 1451

Tutoriel : Création d'une CloudWatch alarme pour le retard de réplication du cluster de bases de données ... 1451
Surveillance avec Database Insights ... 1459
Tarification ... 1459
Support du moteur, de la région et de la classe d'instance ... 1459
Activation du mode avancé ... 1465
Activation du mode standard ... 1469
Surveillez les requêtes lentes ... 1475
Considérations ... 1477
Surveillance de la charge de la base de données avec Performance Insights ... 1479
Présentation de Performance Insights ... 1480
Activation ou désactivation de l'Analyse des performances ... 1496
Schéma de performance pour MariaDB ou MySQL ... 1502
Politiques de Performance Insights ... 1507
Analyse des métriques à l'aide du tableau de bord de Performance Insights ... 1520
Consulter les recommandations proactives de Performance Insights ... 1571
Récupération de métriques avec les Performance Insights API ... 1574
Journalisation des appels Performance Insights à l'aide de AWS CloudTrail ... 1600
VPCpoints de terminaison ()AWS PrivateLink ... 1603
Analyser les performances avec DevOps Guru pour RDS ... 1607
Les avantages de DevOps Guru pour RDS ... 1607
Comment RDS fonctionne DevOps Guru for ... 1609
Configuration de DevOps Guru pour RDS ... 1610
Surveillance du système d'exploitation à l'aide de la surveillance améliorée ... 1619
Vue d'ensemble de la surveillance améliorée ... 1619
Configuration et activation de la surveillance améliorée ... 1621
Afficher les métriques du système d'exploitation dans la RDS console ... 1627
Afficher les métriques du système d'exploitation à l'aide CloudWatch de ... 1631
RDSRéférence des métriques ... 1633
CloudWatch métriques pour RDS ... 1633
CloudWatch dimensions pour RDS ... 1658
CloudWatch métriques pour Performance Insights ... 1658
Métrique de compteur pour Performance Insights ... 1662
Statistiques SQL pour Performance Insights ... 1697
Métriques du système d'exploitation dans la surveillance améliorée ... 1714
Surveillance des événements, des journaux et des flux d'activité de base de données ... 1731

Afficher les journaux, les événements et les flux dans la RDS console Amazon 1732
Surveillance des RDSévénements 1736
Vue d'ensemble des événements pour Amazon RDS 1736
Consulter les RDS événements Amazon 1738
Utilisation des notifications d'RDSévénements Amazon 1742
Création d'une règle qui se déclenche sur un évènement Amazon RDS 1768
Catégories d'événements Amazon RDS et messages d'événements 1774
Surveillance des journaux RDS 1830
Liste et affichage des fichiers journaux de base de données 1830
Téléchargement d'un fichier journal de base de données 1831
Consultation d'un fichier journal de base de données 1833
Publication dans des CloudWatch journaux 1835
Lire le contenu du fichier journal à l'aide de REST 1838
Fichiers journaux de base de données DB2 1840
Fichiers journaux de base de données MariaDB 1845
Fichiers journaux SQL de base de données Microsoft Server 1861
Fichiers journaux de base de données MySQL 1867
Fichiers journaux de base de données Oracle 1886
Fichiers journaux de base de données PostgreSQL 1897
Surveillance des RDSAPIappels CloudTrail 1912
CloudTrail intégration avec Amazon RDS 1912
Entrées du fichier RDS journal Amazon 1913
Surveillance de RDS à l'aide des flux d'activité de base de données 1918
Présentation 1918
Configuration d'audit unifié Oracle 1925
Configuration de l'audit des SQL serveurs 1926
Démarrage d'un flux d'activité de base de données 1928
Modification d'un flux d'activité de base de données 1931
Obtention du statut de flux d'activité 1934
Arrêt d'un flux d'activité de base de données 1935
Surveillance des flux d'activité 1937
Exemples de politiques IAM pour les flux d'activité 1981
Surveillance des menaces avec GuardDuty RDS Protection 1985
Amazon RDS Custom 1987
Défi de la personnalisation des bases de données 1987
Modèle de gestion et avantages de RDS Custom 1989

Modèle de responsabilité partagée dans RDS Custom 1990
Périmètre de prise en charge et configurations non prises en charge dans RDS Custom ... 1992
Principaux avantages de RDS Custom 1993
Architecture RDS Custom 1994
VPC 1994
Automatisation et surveillance RDS Custom 1995
Amazon S3 2000
AWS CloudTrail 2001
Sécurité de RDS Custom 2002
Comment RDS Custom gère les tâches en votre nom en toute sécurité 2002
Certificats SSL 2003
Protégez votre compartiment Amazon S3 contre le problème de confusion lié aux adjoints 2003
Rotation des informations d'identification pour RDS Custom for Oracle 2005
Utilisation de RDS Custom for Oracle 2011
Flux de travail RDS Custom for Oracle 2011
Architecture de base de données pour Amazon RDS Custom for Oracle 2017
Disponibilité des fonctionnalités et support pour RDS Custom pour Oracle 2019
Exigences et limites de RDS Custom for Oracle 2022
Configuration de votre environnement RDS Custom for Oracle 2026
Utilisation de CEV pour RDS Custom for Oracle 2047
Configuration d'une instance de base de données RDS Custom for Oracle 2081
Gestion d'une instance de base de données RDS personnalisée pour Oracle 2102
Gestion d'un déploiement multi-AZ pour RDS Custom for Oracle 2121
Utilisation de répliques RDS personnalisées pour Oracle 2142
Sauvegarde et restauration d'une instance de base de données RDS personnalisée pour Oracle 2153
Utilisation de groupes d'options dans RDS Custom for Oracle 2164
Migration vers RDS Custom pour Oracle 2174
Mise à niveau d'une instance de base de données RDS personnalisée pour Oracle 2176
Résolution des problèmes liés à RDS Custom for Oracle 2190
Problèmes connus relatifs à RDS Custom for Oracle 2216
Utilisation de RDS Custom for SQL Server 2220
RDSFlux de travail personnalisé pour SQL le serveur 2220
Exigences et limites de RDS Custom for SQL Server 2223
Configuration de votre environnement RDS Custom for SQL Server 2303
Apportez votre propre contenu multimédia avec RDS Custom for SQL Server 2330

Utilisation de versions CEV pour RDS Custom for SQL Server 2333
Création et connexion à une instance de base de données RDS personnalisée pour le SQL serveur 2358
Gestion d'une instance de base de données RDS personnalisée pour le SQL serveur 2371
Utilisation de Microsoft Active Directory avec RDS Custom for SQL Server 2387
Gestion d'un déploiement multi-AZ pour RDS Custom for Server SQL 2415
Sauvegarde et restauration d'une instance de base de données RDS personnalisée pour le SQL serveur 2432
Copier un instantané de base de données RDS personnalisé pour SQL Server 2450
Migration d'une base de données sur site vers RDS Custom for SQL Server 2462
Mise à niveau d'une instance de base de données RDS Custom pour SQL Server 2466
Résolution des problèmes liés à Amazon RDS Custom for SQL Server 2471
Amazon RDS sur AWS Outposts 2513
Prérequis 2514
Support pour les RDS fonctionnalités d'Amazon 2515
Classes d'instances de bases de données prises en charge 2523
Adresses IP clients 2526
Utilisation des CoIP 2526
Limites 2528
Déploiements multi-AZ 2530
Utiliser le modèle de responsabilité partagée 2530
Amélioration de la disponibilité 2531
Prérequis 2531
Utilisation des opérations API pour les autorisations Amazon EC2 2533
Création d'instances de base de données pour RDS on Outposts 2535
Création de réplicas en lecture pour RDS sur Outposts 2546
Considérations relatives à la restauration des instances de base de données 2550
Proxy Amazon RDS 2551
Disponibilité des régions et des versions 2552
Quotas et limites 2552
Limites de RDS for MariaDB 2554
Limites de RDS for SQL Server 2555
Limites de MySQL 2556
Limitations de PostgreSQL 2557
Planification Où utiliser RDS Proxy 2558
RDSConcepts et terminologie relatifs aux proxys 2560

Vue d'ensemble des concepts de RDS proxy 2561
Regroupement de connexions 2562
Sécurité 2562
Basculement 2565
Transactions 2566
Commencer à utiliser RDS Proxy 2567
Configuration d'un réseau proxy 2567
Configuration des informations d'identification de base de données 2570
Configuration de l’authentification IAM 2574
Création d'un proxy 2578
Afficher un proxy 2588
Connexion via RDS Proxy 2590
Gestion d'un RDS proxy 2594
Modification d'un RDS Proxy 2594
Ajout d'un utilisateur de base de données 2602
Considérations relatives à la connexion au proxy RDS 2603
Évitez d'épingler le proxy RDS 2607
Supprimer un RDS proxy 2614
Utilisation des points de terminaison du proxy RDS 2615
Présentation des points de terminaison proxy 2616
Limites pour les points de terminaison proxy 2616
Points de terminaison proxy pour les clusters de bases de données multi-AZ 2617
Accès aux bases de données RDS via VPCs 2619
Création d'un point de terminaison proxy 2621
Affichage des points de terminaison proxy 2623
Modification d'un point de terminaison proxy 2625
Suppression d'un point de terminaison proxy 2626
Surveillance du proxy RDS avec CloudWatch 2628
Utilisation des événements RDS proxy 2638
RDSÉvénements liés au proxy 2638
Résolution des problèmes de RDS Proxy 2641
Vérification de la connectivité pour un proxy 2641
Problèmes courants et solutions correspondantes 2643
Résolution des problèmes liés à RDS pour MySQL 2645
Résolution des problèmes liés à RDS pour PostgreSQL 2647
Utilisation de RDS Proxy avec AWS CloudFormation 2652

Intégrations zéro ETL 2654
Avantages 2656
Concepts clés 2656
Limites 2657
Limitations générales 2657
Limitations propres à RDS for MySQL 2658
Limites de RDS pour PostgreSQL 2659
Limitations RDS for Oracle 2659
Limitations propres à Amazon Redshift 2660
Amazon SageMaker AI limites du lakehouse 2660
Quotas 2660
Régions prises en charge 2661
Bien démarrer avec les intégrations zéro ETL 2661
Étape 1 : Créer un groupe de paramètres de base de données personnalisé 2662
Étape 2 : sélectionner ou créer un de base de données source 2663
Étape 3a : créer un entrepôt de données cible 2664
Configurez une intégration à l'aide du AWS SDKs 2665
Étape 3b : Création d'un AWS Glue catalogue pour l'intégration Amazon SageMaker AI Zero-ETL 2670
Étapes suivantes 2674
Création d'intégrations sans ETL avec Amazon Redshift 2674
Prérequis 2674
Autorisations requises 2674
Création d'intégrations zéro ETL 2677
Intégrations de chiffrement 2682
Étapes suivantes 2684
Création d'intégrations sans ETL avec un lakehouse Amazon SageMaker 2684
Prérequis 2684
Autorisations requises 2685
Création d'intégrations sans ETL avec un lakehouse Amazon SageMaker 2688
Intégrations de chiffrement 2693
Étapes suivantes 2695
Filtrage des données pour les intégrations sans ETL 2695
Format d'un filtre de données 2696
Logique de filtrage 2699
Priorité du filtre 2699

Exemples d'de RDS pour MySQL 2700
Exemples d'RDS pour PostgreSQL 2701
Exemples de RDS pour Oracle 2702
Ajouter des filtres de données 2703
Supprimer les filtres de données 2705
Ajout et interrogation de données 2705
Création d'une base de données cible 2706
Ajout de données au de base de données source 2706
Interrogation de vos données Amazon RDS dans Amazon Redshift 2708
Différences de type de données 2709
..... 2729
Affichage et surveillance des intégrations zéro ETL 2731
Affichage des intégrations 2732
Surveillance à l'aide des tables système 2733
Surveillance avec EventBridge 2734
Modifier les intégrations Zero-ETL 2734
Suppression d'intégrations zéro ETL 2736
Résolution des problèmes liés aux intégrations zéro ETL 2737
Je ne parviens pas à créer une intégration zéro ETL 2738
Mon intégration est bloquée dans un état de `Syncing` 2738
Mes tables ne sont pas répliquées sur Amazon Redshift 2739
Une ou plusieurs de mes tables Amazon Redshift nécessitent une resynchronisation 2739
Problèmes d'intégration pour les intégrations Amazon SageMaker AI Lakehouse Zero-ETL 2742
DB2 sur Amazon RDS 2743
Vue d'ensemble de Db2 2744
Fonctionnalités DB2 2745
Versions DB2 2749
Licences DB2 2754
Classes d'instances DB2 2772
Rôles par défaut DB2 2774
Paramètres DB2 2775
EBCDICcollation 2782
Fuseau horaire local Db2 2783
Prérequis pour l'instance de base de données 2786
Compte administrateur 2786

Considérations supplémentaires 2787
Plusieurs bases de données DB2 2788
Connexion à votre instance de base de données DB2 2790
Recherche du point de terminaison 2790
IBM Db2 CLP 2792
IBM CLPPlus 2797
DBeaver 2800
IBM Db2 Data Management Console 2804
Considérations relatives aux groupes de sécurité 2814
Sécurisation des connexions DB2 2815
Chiffrement avec SSL/TLS 2815
Utilisation de l'Kerberosauthentification 2823
Administration de votre instance de base de données RDS pour DB2 2840
Tâches système 2842
Tâches de base de données 2855
Intégration à S3 2880
Création d'une stratégie IAM 2880
Créez un IAM rôle et associez votre IAM politique 2883
Ajoutez votre IAM rôle à votre instance de base de données 2886
Migration des données vers RDS for DB2 2889
Migration des données à l'aide de services AWS 2889
Migration de données avec des outils Db2 natifs 2902
Fédération 2916
Fédération homogène 2916
Fédération hétérogène 2921
Utilisation de répliques DB2 2926
Répliques de secours 2926
Activations de bases de données 2926
Configurations HADR 2927
Conservation des journaux d'archivage 2927
Pannes lors de la réplication DB2 2927
Exigences et considérations relatives aux répliques DB2 2928
Préparation à la création d'une réplique DB2 2930
Création d'une réplique DB2 en veille 2931
Utilisation des répliques de sauvegarde DB2 2933
Résolution des problèmes de réplication DB2 2936

Options RDS pour les instances de base de données DB2 2937
Journalisation des audits DB2 2938
Procédures stockées externes 2954
Procédures stockées externes basées sur Java 2954
Limites et problèmes connus 2963
Limitation de l'authentification 2963
Routines non clôturées 2963
Tablespaces de stockage non automatiques pendant la migration 2963
Définition du paramètre db2_compatibility_vector 2964
Migration de bases de données contenant des packages INVALIDES 2964
Procédures stockées RDS pour DB2 2966
Considérations relatives aux procédures stockées 2975
Octroi et révocation de privilèges 2976
Politiques d'audit 2994
Piscines tampons 3000
Bases de données 3006
Accès au stockage 3036
Espaces de table 3039
RDSpour les fonctions définies par l'utilisateur de DB2 3050
rdsadmin.get_task_status 3050
rdsadmin.list_databases 3055
Résolution des problèmes 3057
Erreur de connexion de base de données 3057
I/O Erreur de fichier 3057
Erreurs de procédure stockée 3062
MariaDB sur Amazon RDS 3073
Prise en charge des fonctions MariaDB 3075
Versions majeures de MariaDB 3076
Moteurs de stockage pris en charge 3084
Préparation du cache 3086
Fonctions non prises en charge 3088
Versions de MariaDB 3090
Versions de MariaDB mineures prises en charge 3090
Versions de MariaDB majeures prises en charge 3098
Environnement de prévisualisation de base de données 3099
MariaDB version 11.8 dans l'environnement de prévisualisation de la base de données 3103

MariaDB version 11.7 dans l'environnement de prévisualisation de la base de données 3103
MariaDB version 11.4 dans l'environnement de prévisualisation de la base de données 3104
Versions MariaDB rendues obsolètes 3104
Connexion à une instance de base de données exécutant MariaDB 3105
Recherche des informations de connexion 3106
Connexion depuis le client de ligne de commande 3110
Communiquer avec les AWS conducteurs 3110
Résolution des problèmes 3111
Sécurisation des connexions MariaDB 3113
Sécurité MariaDB 3113
Plug-ins de validation de mots 3115
Chiffrer avec/SSLTLS 3116
Utilisation de nouveaux TLS certificatsSSL/ 3121
Amélioration des performances des requêtes grâce à RDS Optimized Reads 3127
Présentation 3127
Cas d’utilisation 3128
Bonnes pratiques 3129
Utilisation 3129
Surveillance 3130
Limites 3131
Amélioration des performances d'écriture avec Écritures optimisées pour RDS for MariaDB .. 3132
Présentation 3132
Utilisation avec une nouvelle base de données 3134
Activation sur une base de données existante 3138
Limites 3139
Mises à niveau du moteur de base de données MariaDB 3140
Considérations 3141
Trouver des cibles de mise à niveau valides 3142
Numéros de version de MariaDB 3143
RDSnuméros de version 3146
Mises à niveau de version majeure. 3146
Mise à niveau d'une instance de base de données MariaDB 3147
Mises à niveau automatiques des versions mineures 3147
Mise à niveau avec réduction des temps d'arrêt 3151
Mise à niveau d'une version du moteur de snapshots de base de données MariaDB 3156
Options de mise à niveau pour les versions de moteur non prises en charge 3159

Importation de données dans une instance de base de données RDS pour MariaDB 3161
Considérations sur l'importation de données 3166
Importation de données à partir d'une base de données externe 3172
Importation de données avec un temps d'arrêt réduit 3177
Importation de données à partir de n'importe quelle source 3199
Réplication MariaDB 3206
MariaDB lit les répliques 3207
Configuration de la réplication GTID basée sur une instance source externe 3223
Configuration d'une réplication de position de fichier journal binaire avec une instance source externe 3228
Options pour MariaDB 3234
Prise en charge du plugin d'audit MariaDB 3234
Paramètres pour MariaDB 3242
Affichage des paramètres MariaDB 3242
Paramètres MySQL qui ne sont pas disponibles 3244
Migration des données d'un instantané My SQL DB vers une instance de base de données MariaDB 3247
Exécution de la migration 3247
Incompatibilités entre MariaDB et My SQL 3249
Référence MariaDB sur SQL Amazon RDS 3251
mysql.rds_replica_status 3251
mysql.rds_set_external_master_gtid 3253
mysql.rds_kill_query_id 3256
Fuseau horaire local 3258
Limites et problèmes connus pour MariaDB 3261
Limites de taille des fichiers 3261
Mot réservé InnoDB 3263
Ports personnalisés 3263
Performance Insights 3263
Microsoft SQL Server sur Amazon RDS 3264
Tâches courantes de gestion 3266
Limites 3270
Assistance pour les classes d'instance de base de données 3273
Sécurité 3278
SSLsoutien 3279
Utilisation SSL avec une instance de base de données de SQL serveur 3280

Configuration des protocoles de sécurité et des chiffrements du SQL serveur 3286
Mise à jour des applications pour les SSL nouveaux/ TLS certificats 3293
Programmes de conformité 3297
HIPAA 3298
Prise en charge des versions 3299
Gestion des versions 3303
Prise en charge des fonctionnalités 3305
SQLFonctionnalités de Server 2022 3306
SQLFonctionnalités de Server 2019 3306
SQLFonctionnalités de Server 2017 3307
SQLFonctionnalités de Server 2016 3308
SQLFonctionnalités de Server 2014 3308
SQLFin du support de Server 2012 sur Amazon RDS 3308
SQLFin du support de Server 2008 R2 sur Amazon RDS 3309
CDCsoutien 3309
Fonctionnalités non prises en charge et prises en charge limitées 3310
Déploiements multi-AZ 3311
Utilisation de TDE 3312
Fonctions et procédures stockées 3312
Fuseau horaire local 3319
Fuseaux horaires pris en charge 3320
Gestion des licences SQL Server sur Amazon RDS 3333
Restauration des instances de bases de données résiliées faute de licence 3333
SQL Server Developer Edition 3334
Connexion à une instance de base de données exécutant SQL Server 3335
Avant de vous connecter 3335
Recherche du point de terminaison de l'instance de base de données et du numéro de port 3336
Connexion à votre instance de base de données avec SSMS 3338
Connexion à votre instance de base de données avec SQL Workbench/J 3341
Considérations relatives aux groupes de sécurité 3343
Résolution des problèmes 3344
Utilisation d'Active Directory avec RDS for SQL Server 3346
Utilisation d'Active Directory autogéré avec une instance de base de données de SQL serveur 3347
Utilisation d'Active Directory AWS géré avec RDS pour SQL Server 3369

Mises à niveau du moteur de base de données SQL Server 3386
Mises à niveau de version majeure. 3387
Considérations relatives aux mises à niveau 3389
Test d'une mise à niveau 3393
Mise à niveau d'une instance de base de données SQL Server 3394
Mise à niveau des instances de base de données obsolètes avant la fin de la prise en charge 3394
Importation et exportation de bases de données SQL Server 3396
Limitations et recommandations 3399
Configuration de 3401
Utilisation des sauvegarde et restauration natives 3407
Compression des fichiers de sauvegarde 3425
Résolution des problèmes 3426
Importation et exportation de données SQL Server à l'aide d'autres méthodes 3431
Utilisation de BCP depuis Linux 3446
SQL Server lit les répliques 3452
Configuration des réplicas en lecture pour SQL Server 3452
Limites des réplicas en lecture avec SQL Server 3453
Considérations relatives aux options 3454
Synchronisation des utilisateurs et des objets de base de données 3456
Résolution des problèmes 3458
Multi-AZ RDS pour serveur SQL 3460
Ajout de Multi-AZ à une instance de base de données SQL de serveur 3461
Suppression de Multi-AZ d'une instance de base de données SQL de serveur 3462
Limitations, notes et recommandations 3462
Détermination de l'emplacement du réplica secondaire 3467
Migration vers Always On AGs 3467
Fonctionnalités supplémentaires pour le SQL serveur 3469
Utilisation d'une politique de mot de passe avec une instance de base de données SQL Server 3470
Intégration Amazon S3 3479
Utilisation de Database Mail 3503
Prise en charge du stockage d'instance pour tempdb 3519
Utilisation d'événements étendus 3523
Accès aux sauvegardes des journaux de transactions 3527
Options pour SQL Server 3572

Liste des options disponibles pour les versions et éditions de SQL Server 3574
Serveurs liés avec Oracle OLEDB 3577
Serveurs liés avec Teradata ODBC 3590
Sauvegarde et restauration natives 3600
Transparent Data Encryption 3605
SQLAudit du serveur 3619
SQLServices d'analyse de serveurs 3630
SQL Server Integration Services 3661
SQLServices de rapports sur les serveurs 3685
Microsoft Distributed Transaction Coordinator 3707
Tâches d'administrateur de base de données courantes 3725
Accès à la base de données tempdb 3727
Analyse de la charge de travail d'une base de données avec l'Assistant Paramétrage du moteur de base de données 3731
Remplacement de db_owner par le compte rdsa pour votre base de données 3736
Gestion des classements et des jeux de caractères 3737
Création d'un utilisateur de base de données 3746
Détermination d'un modèle de récupération 3747
Détermination de l'heure du dernier basculement 3748
Résoudre les défaillances du PITR dues à des lacunes du LSN 3749
Refuser ou autoriser l'affichage des noms de base de données 3750
Désactivation des insertions rapides 3751
Suppression d'une base de données SQL Server 3751
Modification du nom d'une base de données Multi-AZ 3752
Réinitialisation de l'appartenance au db_owner rôle pour l'utilisateur principal 3753
Restauration des instances de bases de données résiliées faute de licence 3754
Transition d'une base de données de vers OFFLINE ONLINE 3755
Utilisation de CDC 3755
Utilisation de SQL Server Agent 3759
Utilisation des journaux SQL Server 3766
Utilisation des fichiers de trace et de vidage 3768
Mon SQL sur Amazon RDS 3770
Prise en charge des fonctionnalités MySQL 3773
Versions majeures de MySQL 3773
Moteurs de stockage pris en charge 3776
Utilisation de memcached et d'autres options 3777

Préparation du cache InnoDB 3777
Changements linguistiques inclusifs pour MySQL 8.4 3779
Fonctions non prises en charge 3782
Versions MySQL 3783
Versions mineures 3783
Versions majeures 3791
Versions de support étendu RDS 3792
Environnement d'aperçu de base de données 3797
9.3 (aperçu) 3801
9.2 (aperçu) 3801
9.1 (aperçu) 3801
Versions déconseillées 3802
Connexion à une instance de base de données exécutant MySQL 3803
Recherche des informations de connexion 3804
Installation du client en ligne de commande 3808
Connexion depuis le client de ligne de commande 3808
Connexion depuis My SQL Workbench 3809
Communiquer avec les AWS conducteurs 3811
Résolution des problèmes 3812
Sécurisation des connexions MySQL 3814
validation du mot de passe 3817
Chiffrer avec/SSLTLS 3818
Utilisation de nouveaux TLS certificatsSSL/ 3822
Utilisation de Kerberos l'authentification pour MySQL 3828
Améliorer les performances des requêtes grâce à des lectures RDS optimisées 3844
Présentation 3844
Cas d’utilisation 3845
Bonnes pratiques 3846
Utilisation 3847
Surveillance 3848
Limites 3848
Améliorer les performances d'écriture grâce à RDS Optimized Writes for My SQL 3850
Présentation 3132
Utilisation avec une nouvelle base de données 3852
Activation sur une base de données existante 3856
Limites 3857

Mises à niveau du moteur de base de données MySQL 3858
Considérations 3860
Trouver des cibles de mise à niveau valides 3860
Numéros de version de MySQL 3861
Numéros de version RDS 3863
Mises à niveau de version majeure. 3864
Test d'une mise à niveau 3871
Mise à niveau d'une instance de base de données MySQL 3872
Mises à niveau automatiques des versions mineures 3872
Mise à niveau avec réduction des temps d'arrêt 3875
Mise à niveau d'une version du moteur de snapshots de base de données MySQL 3881
Options de mise à niveau pour les versions de moteur non prises en charge 3884
Importation de données dans une instance de base de données RDS pour MySQL 3887
Considérations sur l'importation de données 3893
Restauration d'une sauvegarde dans une instance de base de données MySQL 3899
Importation de données à partir d'une base de données externe 3916
Importation de données avec un temps d'arrêt réduit 3920
Importation de données à partir de n'importe quelle source 3943
Réplication MySQL 3950
Mes répliques de SQL lecture 3951
GTIDréplication basée sur la réplication 3969
Configuration d'une réplication de position de fichier journal binaire avec une instance source externe 3978
Configuration de la réplication multi-sources 3983
Configuration de clusters actifs-actifs 3992
Cas d’utilisation 3992
Limites et considérations relatives aux clusters actifs-actifs 3993
Préparation à un cluster à VPC activités croisées 3997
Réglages de paramètres requis pour les clusters actifs-actifs 3999
Conversion d'une instance de base de données en cluster actif-actif 4002
Configuration d'un nouveau cluster actif-actif 4008
Ajouter une instance de base de données à un cluster actif-actif 4015
Surveillance des clusters actifs-actifs 4019
Arrêt de la réplication de groupe dans un cluster actif-actif 4019
Modification du nom d'une instance de base de données dans un cluster actif-actif 4020
Supprimer une instance de base de données d'un cluster actif-actif 4021

Exportation de données depuis une instance My SQL DB 4023
Préparer une base de SQL données My externe 4023
Préparer l'instance My SQL DB source 4024
Copier la base de données 4026
Terminer l'exportation 4027
Options pour moi SQL 4030
Plug-in d'audit MariaDB 4031
memcached 4040
Paramètres pour My SQL 4046
DBATâches courantes pour My SQL 4049
Comprendre les utilisateurs prédéfinis 4049
Modèle de privilège basé sur les rôles 4050
Privilèges dynamiques 4053
Mettre fin à une session ou à une requête 4057
Ignorer une erreur de réplication 4058
Améliorez les temps de reprise après un crash 4060
Gestion de l'historique global des statuts (GoSH) 4063
Configuration de la taille du pool de mémoire tampon et de la capacité du redo log 4065
Fuseau horaire local 4067
Limites et problèmes connus 4070
Mot réservé InnoDB 4070
Comportement de stockage plein 4070
Taille de pool de mémoires tampons InnoDB incohérente 4071
L'optimisation de la fusion d'index renvoie des résultats incorrects 4072
Exceptions des paramètres MySQL pour les instances de base de données Amazon RDS 4073
Limites de taille des fichiers MySQL dans Amazon RDS 4074
Plug-in MySQL Keyring non pris en charge 4077
Ports personnalisés 4077
Limitations des procédures stockées MySQL 4077
Réplication basée sur GTID avec une instance source externe 4077
Plugin d'authentification par défaut MySQL 4078
Remplacer innodb_buffer_pool_size 4078
Mise à niveau de MySQL 5.7 vers MySQL 8.4 4079
Compression de page InnoDB 4079
RDSpour Mes procédures SQL stockées 4080
Collecte et mise à jour de l'historique des statuts mondiaux 4081

Configuration, démarrage et arrêt de la réplication du journal binaire (binlog) 4084
Mettre fin à une session ou à une requête 4127
Gestion des clusters actifs-actifs 4129
Gestion de la réplication multi-sources 4134
Répliquer des transactions à l'aide de GTIDs 4157
Rotation des journaux de requêtes 4160
Configuration et affichage de la configuration du journal binaire 4162
Réchauffement du cache InnoDB 4167
Oracle sur Amazon RDS 4169
Présentation d'Oracle 4170
Fonctions Oracle 4171
Versions d'Oracle 4175
Licences Oracle 4182
Utilisateurs et privilèges Oracle 4187
Classes d'instances Oracle 4188
Architecture de base de données Oracle 4197
Paramètres Oracle 4199
Jeux de caractères Oracle 4201
Limitations Oracle 4205
Connexion à votre instance de base de données Oracle 4209
Recherche du point de terminaison 4209
SQL Developer 4212
SQL*Plus 4215
Considérations relatives aux groupes de sécurité 4216
Processus serveur dédiés et partagés 4217
Dépannage 4217
Modification des paramètres Oracle sqlnet.ora 4219
Sécurisation des connexions Oracle 4225
Chiffrement avec SSL 4225
Utilisation de nouveaux certificats SSL/TLS 4226
Chiffrement avec NNE 4230
Configuration de l'authentification Kerberos 4231
Configuration de l'accès UTL_HTTP 4252
Utilisation des CDB 4265
Vue d'ensemble de CDBs 4265
Configuration d'une CDB 4272

Sauvegarde et restauration d'une CDB 4279
Conversion d'une base de données non-CDB en CDB 4280
Conversion d'un locataire unique en locataire multiple 4283
Ajout d'une base de données locataire RDS for Oracle à votre instance de CDB 4286
Modification d'une base de données locataire RDS for Oracle 4290
Suppression d'une base de données locataire RDS for Oracle de votre CDB 4292
Affichage des détails de la base de données locataire 4295
Mise à niveau de votre CDB 4300
Administration de votre instance de base de données Oracle 4301
Tâches système 4316
Tâches de base de données 4343
Tâches de journal 4376
RMANtâches 4390
Tâches Oracle Scheduler 4429
Diagnostic des problèmes 4439
Autres tâches 4449
Configuration des fonctions avancées RDS for Oracle 4465
Configuration du stockage d'instances 4465
Activation de HugePages 4478
Activation des types de données étendus 4482
Importation de données dans Oracle 4486
Importation à l'aide d'Oracle SQL Developer 4487
Migration à l'aide des espaces de table transportables Oracle 4487
Importation à l'aide d'Oracle Data Pump 4505
Importation avec les utilitaires d’importation/importation d’Oracle 4524
Importation avec Oracle SQL*Loader 4525
Migration avec les vues matérialisées d'Oracle 4526
Utilisation des réplicas Oracle 4529
Présentation des réplicas Oracle 4529
Exigences et considérations relatives aux réplicas Oracle 4532
Préparation de la création d'un réplica Oracle 4536
Création d'un réplica Oracle monté 4538
Modification du mode réplica 4539
Utilisation des sauvegardes de réplica Oracle 4541
Exécution d'un basculement d'Oracle Data Guard 4544
Dépannage des réplicas Oracle 4552

Options pour Oracle 4555
Présentation des options de base de données Oracle 4555
Intégration Amazon S3 4558
Oracle Application Express (APEX) 4586
EFSIntégration avec Amazon 4611
Java Virtual Machine (JVM) 4629
Enterprise Manager 4634
Label Security 4660
Locator 4664
Chiffrement réseau natif (NNE) 4669
OLAP 4686
couche de sockets sécurisée (SSL) 4690
Spatial 4702
SQLT 4707
Statspack 4717
Fuseau horaire 4721
Mise à niveau automatique du fichier sur le fuseau horaire 4727
Chiffrement transparent des données (TDE) 4738
UTL_MAIL 4743
XML DB 4747
Mise à niveau du moteur de base de données Oracle 4748
Présentation des mises à niveau Oracle 4748
Mises à niveau de version majeure. 4756
Mises à niveau de version mineure. 4758
Considérations relatives aux mises à niveau 4764
Test d'une mise à niveau 4768
Mise à niveau d'une instance de base de données RDS pour Oracle 4770
Mise à niveau d'un instantané de base de données Oracle 4772
Outils et des logiciels tiers pour Oracle 4775
Utilisation d'Oracle GoldenGate 4776
Utilisation de l'utilitaire Oracle Repository Creation Utility 4796
Configuration de CMAN 4804
Installation d'une base de données Siebel sur Oracle sur Amazon RDS 4807
Versions du moteur de base de données Oracle 4812
PostgreSQL sur Amazon RDS 4813
Tâches courantes de gestion 4815

Utilisation de l'environnement de prévisualisation de base de données 4820
Fonctions non prises en charge dans l'environnement de prévisualisation de base de données 4821
Postgre SQL version 17 dans l'environnement de prévisualisation de la base de données . 4821
Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données 4822
Versions de PostgreSQL 4824
Versions obsolètes de PostgreSQL 4824
Processus de publication de RDS pour PostgreSQL 4826
Avantages du processus de publication incrémentielle de RDS pour PostgreSQL 4826
Gestion des mises à jour des versions 4826
Versions de l'SQLextension Postgre 4828
Restreindre l'installation des extensions Postgre SQL 4828
Extensions SQL fiables de Postgre 4830
Fonctionnalités de Postgre SQL 4832
Types de données et énumérations personnalisés 4833
Événements déclencheurs 4833
D'énormes pages 4834
Exécution d'une réplication logique 4835
RAMdisque pour le répertoire stats_temp_directory 4843
Espaces de table 4844
Collations pour EBCDIC les migrations de mainframe et autres 4844
Gestion de la synchronisation des emplacements logiques 4850
Connexion à une instance PostgreSQL 4852
Installation du client PSQL 4853
Recherche des informations de connexion 4853
Utilisation pgAdmin pour se connecter à une instance RDS de SQL base de données Postgre 4856
Utilisation de psql pour vous connecter à votre instance de base RDS de données Postgre SQL 4858
Connexion RDS à ou Postgre SQL avec le pilote AWS JDBC 4860
Connexion RDS à Postgre SQL avec le pilote AWS Python 4860
Résolution des problèmes de connexion à votre instance RDS for PostgreSQL 4860
Sécurisation des connexions avecSSL/TLS 4863
Utilisation de SSL avec une instance de base de données PostgreSQL 4863
Mise à jour des applications pour utiliser les nouveaux TLS certificatsSSL/ 4871

Utilisation de l'authentification Kerberos 4876
Disponibilité des régions et des versions 4877
Présentation de l'authentification Kerberos 4877
Configuration 4878
Gestion d'une RDS pour PostgreSQL dans un domaine Active Directory 4893
Connexion avec l'authentification Kerberos 4894
Utilisation d'un DNS serveur personnalisé pour l'accès au réseau sortant 4898
Activer la DNS résolution personnalisée 4898
Désactiver la DNS résolution personnalisée 4898
Configuration d'un DNS serveur personnalisé 4898
Mises à niveau du moteur de base de données PostgreSQL 4901
Considérations 4903
Trouver des cibles de mise à niveau valides 4904
Numéros de SQL version de Postgre 4905
RDSnuméros de version 4906
Choix d'une mise à niveau de version majeure 4907
Comment effectuer une mise à niveau de version majeure 4917
Mises à niveau automatiques des versions mineures 4926
Mise à niveau des extensions Postgre SQL 4929
Mise à niveau d'une version du moteur de snapshots SQL de base de données Postgre 4931
Utilisation de réplicas en lecture pour RDS for PostgreSQL 4934
Décodage logique sur une réplique lue 4934
Limites des réplicas en lecture avec PostgreSQL 4938
Configuration de réplicas en lecture avec PostgreSQL 4939
Utilisation de réplicas en lecture en cascade 4942
Création de répliques de lecture en cascade entre régions 4944
Fonctionnement de la réplication pour différentes versions de RDS for PostgreSQL 4946
Surveillance et réglage du processus de réplication 4950
Résolution des problèmes liés à la réplication en lecture de RDS pour PostgreSQL 4953
Amélioration des performances des requêtes grâce à RDS Optimized Reads 4955
Présentation de Lectures optimisées pour RDS dans PostgreSQL 4955
Cas d’utilisation 4956
Bonnes pratiques 4957
Utilisation 4957
Surveillance 4958
Limites 4959

Importer des données dans Postgre SQL 4960
Importation d'une SQL base de données Postgre depuis une instance Amazon EC2 4963
Utilisation de la commande \ copy pour importer des données dans une table sur une instance de SQL base de données Postgre 4965
Importation de données depuis Amazon S3 vers Amazon RDS for PostgreSQL 4967
Transport de bases de données Postgre entre de bases de données 4987
Exportation de données PostgreSQL vers Amazon S3 4997
Installation de l'extension 4998
Présentation de l'exportation vers S3 4999
Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation .. 5000
Configuration de l'accès à un compartiment Amazon S3 5002
Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3 5007
Références de fonctions 5010
Résolution des problèmes d'accès à Amazon S3 5014
Invocation d'une fonction Lambda depuis RDS for PostgreSQL 5016
Étape 1 : configurer les connexions sortantes 5017
Étape 2 : configurer IAM pour votre instance et Lambda 5018
Étape 3 : installer l'extension 5020
Étape 4 : utiliser les fonctions d'assistance Lambda 5021
Étape 5 : appeler une fonction Lambda. 5022
Étape 6 : accorder des autorisations aux utilisateurs 5023
Exemples : appel de fonctions Lambda 5024
Messages d'erreur de fonction Lambda 5026
Référence de fonction et de paramètre Lambda 5028
Tâches courantes d'administration de bases de données pour RDS for PostgreSQL 5033
Collations prises en charge dans RDS Postgre SQL 5034
Comprendre les rôles et les autorisations PostgreSQL 5035
Gestion des connexions mortes dans PostgreSQL 5051
Utilisation de la fonction autovacuum de PostgreSQL 5054
Mécanismes de journalisation 5103
Gestion des fichiers temporaires avec PostgreSQL 5105
Utilisation de pgBadger pour l'analyse de journal serveur avec PostgreSQL 5111
Utilisation PGSnapper pour la surveillance de PostgreSQL 5111
Utilisation de paramètres 5111
Réglage avec les événements d'attente pour RDS for PostgreSQL 5138
Concepts essentiels à connaître pour le réglage de RDS for PostgreSQL 5139

Événements d'attente RDS for PostgreSQL 5144
Cliente : ClientRead 5146
Cliente : ClientWrite 5150
CPU 5153
IO:BufFileRead et IO:BufFileWrite 5159
IO : DataFileRead 5168
IO : WALWrite 5177
IPC : événements d'attente parallèles 5180
Lock:advisory 5187
Lock:extend 5190
Lock:Relation 5193
Lock:transactionid 5196
Lock:tuple 5199
LWLock:BufferMapping (LWLock:buffer_mapping) 5204
LWLock:BufferIO (IPC:BufferIO) 5206
LWLock:buffer_content (BufferContent) 5208
LWLock:lock_manager (:lockmanager) LWLock 5211
LWLock:pg_stat_statements 5217
Timeout:PgSleep 5221
Timeout:VacuumDelay 5221
Optimisation RDSpour Postgre SQL Aurora Postgre d'Amazon DevOps Guru 5225
La base de données a une connexion de longue durée à l'état Transaction inactive 5225
Utilisation des extensions PostgreSQL 5229
Utilisation des fonctions d'orafce 5230
Utilisation de la prise en charge RDS déléguée des extensions Amazon pour Postgre SQL 5232
Gestion des partitions avec l'extension pg_partman 5246
Utilisation pgAudit pour enregistrer l'activité de la base de données 5253
Planification de la maintenance avec l'extension pg_cron 5267
Utilisation de pglogical pour synchroniser les données 5278
Utilisation de pgactive pour créer une réplication active-active 5293
Réduction du ballonnement avec l'extension pg_repack 5370
Mise à niveau et utilisation PLV8 5376
Utilisation de PL/Rust pour écrire des fonctions dans le langage Rust 5379
Gestion des données spatiales avec Post GIS 5384
Enveloppeurs de données étrangers pris en charge dans SQL 5394

Utilisation de l'extension log_fdw 5394
Utilisation de postgres_fdw pour accéder à des données externes 5397
Utilisation d'une base de SQL données My 5398
Utilisation d'une base de données Oracle 5402
Utilisation d'une base SQL de données serveur 5406
Utilisation de Trusted Language Extensions pour PostgreSQL 5410
Terminologie 5411
Exigences relatives à l'utilisation de Trusted Language Extensions 5412
Configuration de Trusted Language Extensions 5415
Présentation de Trusted Language Extensions 5419
Création d'TLEextensions 5421
Supprimer vos TLE extensions d'une base de données 5426
Désinstallation de Trusted Language Extensions 5428
Utilisation des hooks PostgreSQL avec vos extensions TLE 5429
Utilisation de types de données personnalisés dans Trusted Language Extensions 5435
Référence fonctionnelle pour les extensions linguistiques fiables 5435
Référence des hooks pour Trusted Language Extensions 5450
Exemples de code 5453
Principes de base 5464
Hello Amazon RDS 5465
Principes de base 5475
Actions 5595
Scénarios 5731
Créer un outil de suivi des éléments de travail sans serveur Aurora 5732
Exemples sans serveur 5736
Connexion à une base de données Amazon RDS dans une fonction Lambda 5736
Sécurité 5756
Authentification de base de données 5758
Authentification par mot de passe 5759
Authentification de base de données IAM 5760
Authentification Kerberos 5760
Gestion des mots de passe avec RDS et Secrets Manager 5762
Limites 5762
Présentation 5763
Avantages 5764
Autorisations 5765

.... 5766
Gestion du mot de passe pour une instance de base de 5767
Gestion du mot de passe pour la base de données des 5771
Gestion du mot de passe pour un cluster de base de données multi-AZ 5775
Secrets de rotation pour une instance de base de données 5779
Secrets de rotation pour les clusters de bases de données multi-AZ 5781
Afficher les détails secrets 5783
Affichage des détails des secrets pour les clusters de bases de données multi-AZ 5787
Afficher les détails des secrets pour les bases de données des locataires 5791
Disponibilité des régions et des versions 5794
Protection des données 5794
Chiffrement des données 5796
Confidentialité du trafic inter-réseau 5828
Gestion des identités et des accès 5830
Public ciblé 5830
Authentification par des identités 5831
Gestion des accès à l’aide de politiques 5835
Comment Amazon RDS travaille avec IAM 5837
Exemples de politiques basées sur l’identité 5846
AWS politiques gérées 5870
Mises à jour des politiques 5878
Prévention du problème de l’adjoint confus entre services 5903
Authentification de base de données IAM 5906
Dépannage 5961
Journalisation et surveillance 5963
Validation de conformité 5966
Résilience 5967
Sauvegarde et restauration 5967
Réplication 5967
Basculement 5968
Sécurité de l’infrastructure 5969
Groupes de sécurité 5969
Accessible publiquement 5970
Points de terminaison d'un VPC (AWS PrivateLink) 5971
Considérations 1603
Disponibilité 1603

Création d'un point de terminaison d'un VPC d'interface 1604
Création d'une politique de point de terminaison de VPC 1604
Bonnes pratiques de sécurité 5975
Contrôle d'accès par groupe de sécurité 5976
Présentation des groupes de sécurité VPC 5976
Scénario de groupes de sécurité 5977
Création d'un groupe de sécurité VPC 5978
Association à une instance de bases de données 5979
Privilèges du compte utilisateur principal 5979
Rôles liés à un service 5984
Autorisations des rôles liés à un service pour Amazon RDS 5984
Autorisations du rôle lié à un service pour Amazon RDS Custom 5987
Autorisations de rôle liées à un service pour Amazon RDS Beta 5989
Rôle lié à un service pour Amazon RDS Preview 5990
Utiliser Amazon RDS avec Amazon VPC 5992
Utilisation d'un(e) instance de base de données dans un VPC 5992
Mettre à jour le VPC pour une instance de base de données 6013
Scénarios d'accès à un d'instances de base de données dans un VPC 6014
Tutoriel : Création d'un VPC à utiliser avec un) IPv4 6021
Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile) 6029
Déplacement d'une instance de base de données vers un VPC. 6041
Quotas et contraintes 6044
Quotas dans Amazon RDS 6044
Contraintes d'affectation de noms dans Amazon RDS 6052
Nombre maximal de connexions à une base de données 6053
Limites de taille des fichiers dans Amazon RDS 6056
Résolution des problèmes 6057
Impossible de se connecter à l'instance de base de données 6057
Test de connexion d'une instance de base de données 6060
Dépannage des problèmes d'authentification de connexion 6061
Problèmes de sécurité 6061
Message d'erreur « Échec de l'extraction des attributs du compte. Certaines fonctions de la console sont peut être dégradées. » 6062
Résolution des problèmes liés à l'état de réseau incompatible 6062
Causes 6062

Résolution 6062
Réinitialisation du mot de passe du propriétaire de l'instance de base de données 6064
Panne ou redémarrage d'une instance de base de données 6065
Modifications de paramètre n'entrant pas en vigueur 6066
Instance de base de données à court de stockage 6067
Instances de base de données disponibles insuffisantes 6069
Problèmes liés à la mémoire libérable dans RDS 6069
Problèmes MySQL et MariaDB 6070
Maximum de connexions MySQL et MariaDB 6070
Diagnostic et résolution d'un état de paramètres incompatibles pour une limite de mémoire 6071
Diagnostic et résolution du retard entre réplicas en lecture 6073
Diagnostic et résolution d'une défaillance de la réplication en lecture MySQL ou MariaDB . 6076
La création de déclencheurs avec la journalisation binaire activée requiert le privilège SUPER 6077
Diagnostic et résolution des défaillances de point-in-time restauration 6079
Erreur d'arrêt de réplication 6080
La création de réplica en lecture échoue ou la réplication s'arrête avec l'erreur irrécupérable 1236 6081
La réplication de la réplique en lecture ne parvient pas à initialiser la structure des métadonnées 6082
Impossible de définir la période de rétention des sauvegardes sur 0 6082
Référence d'API Amazon RDS 6083
Utilisation de l'API Query 6083
Paramètres Query (Requête) 6083
Authentification de demande Query 6084
Applications de dépannage 6084
Récupération d'erreurs 6084
Conseils pour le dépannage 6085
Historique du document 6086
Mises à jour antérieures 6304
AWS Glossaire 6342
..... 6343

Qu'est-ce que Amazon Relational Database Service (Amazon RDS) ?

Amazon Relational Database Service (Amazon RDS) est un service web qui facilite la configuration, l'exploitation et la mise à l'échelle d'une base de données relationnelle dans le AWS Cloud. Il fournit des capacités redimensionnables, à faible coût, pour les bases de données relationnelles classiques, et gère les tâches courantes d'administration de base de données.

> **Note**
>
> Ce guide traite des moteurs de base de données Amazon RDS autres que Amazon Aurora. Pour plus d'informations sur l'utilisation d'Amazon Aurora, consultez le Guide de l'utilisateur Amazon Aurora.

Si vous débutez dans le domaine des AWS produits et services, commencez à en apprendre davantage à l'aide des ressources suivantes :

- Pour un aperçu de tous les AWS produits, voir Qu'est-ce que le cloud computing ?
- Amazon Web Services fournit un certain nombre de services de base de données. Pour en savoir plus sur les différentes options de bases de données disponibles sur AWS, consultez Choix d'un service de base de données AWS et Exécution de bases de données sur AWS.

Avantages d'Amazon RDS

Amazon RDS est un service de base de données géré. Il est responsable de la plupart des tâches de gestion. En éliminant les processus manuels fastidieux, Amazon RDS vous permet de vous concentrer sur votre application et vos utilisateurs.

Amazon RDS offre les principaux avantages suivants par rapport aux déploiements de bases de données qui ne sont pas entièrement gérés :

- Vous pouvez utiliser des moteurs de base de données que vous connaissez déjà : IBM Db2, MariaDB, Microsoft SQL Server, MySQL, Oracle Database et PostgreSQL.
- Amazon RDS gère les sauvegardes, les correctifs logiciels, la détection automatique des pannes et la récupération.

- Vous pouvez activer des sauvegardes automatiques ou créer manuellement vos propres instantanés de sauvegarde. Vous pouvez utiliser ces sauvegardes pour restaurer une base de données. Le processus de restauration Amazon RDS est fiable et efficace.
- Vous pouvez obtenir une haute disponibilité avec une instance de base de données principale et une instance de base de données secondaire synchrone vers laquelle vous pouvez basculer en cas de problème. Vous pouvez aussi utiliser les réplicas en lecture pour augmenter la mise à l'échelle de la lecture.
- Outre la sécurité de votre package de base de données, vous pouvez contrôler l'accès en utilisant AWS Identity and Access Management (IAM) pour définir les utilisateurs et les autorisations. Vous pouvez aussi aider à protéger vos bases de données en les plaçant dans un virtual private cloud (VPC).

Comparaison des responsabilités avec Amazon EC2 et les déploiements sur site

Nous recommandons Amazon RDS comme choix par défaut pour la plupart des déploiements de bases de données relationnelles. Les alternatives suivantes présentent l'inconvénient de vous faire passer plus de temps à gérer les logiciels et le matériel :

Déploiement sur site

Quand vous achetez un serveur sur site, l'UC, la mémoire, le stockage et les IOPS sont tous regroupés ensemble. Vous assumez l'entière responsabilité du serveur, du système d'exploitation et du logiciel de base de données.

Amazon EC2

Amazon Elastic Compute Cloud (Amazon EC2) fournit une capacité de calcul évolutive dans le AWS Cloud. Contrairement à un serveur sur site, le processeur, la mémoire, le stockage et les IOPS sont séparés afin que vous puissiez les dimensionner indépendamment. AWS gère les couches matérielles, ce qui élimine une partie du fardeau lié à la gestion d'un serveur de base de données sur site.

L'inconvénient de gérer une base de données sur Amazon EC2 est que vous êtes plus vulnérable aux erreurs des utilisateurs. Par exemple, lorsque vous mettez à jour manuellement le système d'exploitation ou le logiciel de base de données, vous risquez de provoquer accidentellement des interruptions d'application. Vous pouvez passer des heures à vérifier chaque modification pour identifier et résoudre un problème.

Le tableau suivant compare les modèles de gestion pour les bases de données sur site EC2, Amazon et Amazon RDS.

Fonctionnalité	Gestion sur site	EC2 Gestion d'Amazon	Gestion Amazon RDS
Optimisation des applications	Client	Client	Client
Mise à l'échelle	Client	Client	AWS
Haute disponibilité	Client	Client	AWS
Sauvegardes de base de données	Client	Client	AWS
Correction de logiciel de base de données	Client	Client	AWS
Installation de logiciels de base de données	Client	Client	AWS
Correction du système d'exploitation (OS)	Client	Client	AWS
Installation du système d'exploitation	Client	Client	AWS
Maintenance des serveurs	Client	AWS	AWS
Cycle de vie du matériel	Client	AWS	AWS
Alimentation, réseau et refroidissement	Client	AWS	AWS

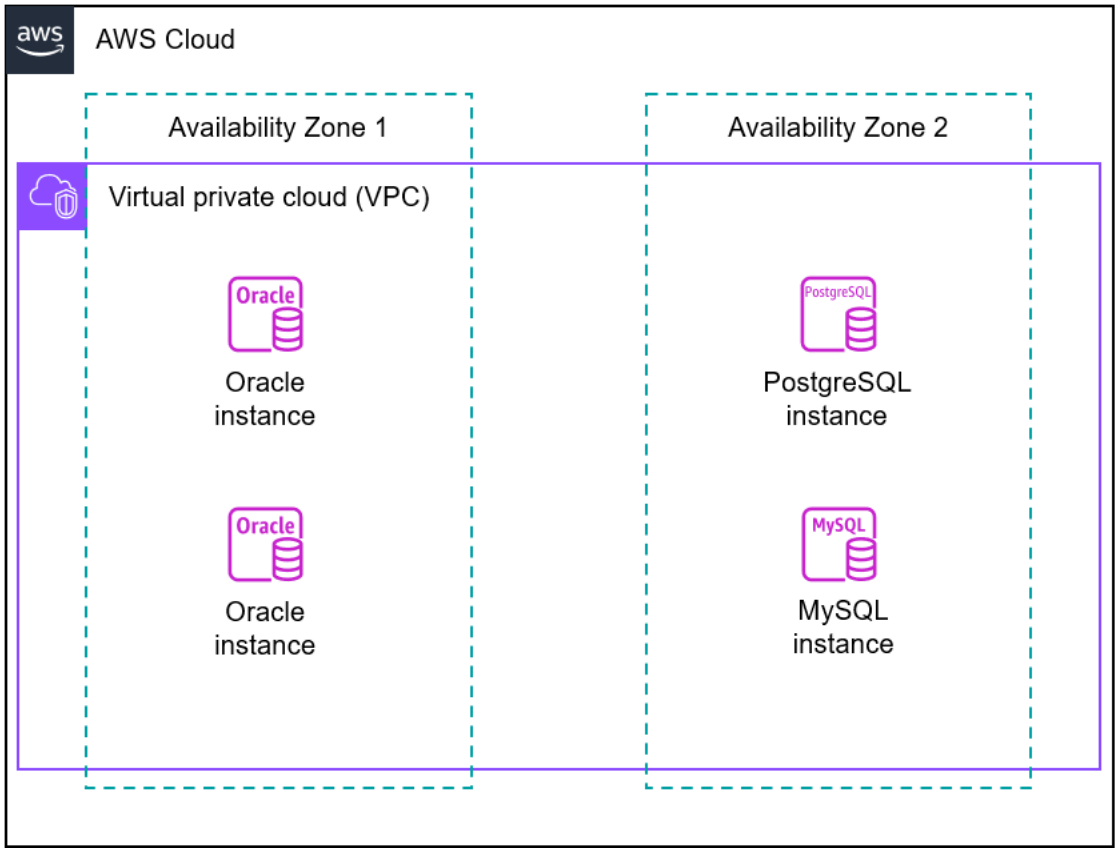
Modèle de responsabilité partagée Amazon RDS

Amazon RDS est responsable de l'hébergement des composants logiciels et de l'infrastructure des instances de base de données et des clusters de bases de données. Vous êtes responsable du

réglage des requêtes, qui consiste à ajuster les requêtes SQL afin d'améliorer les performances. Les performances des requêtes dépendent fortement de la conception de la base de données, de la taille des données, de la distribution des données, de la charge de travail des applications et des modèles de requêtes, qui peuvent varier considérablement. La surveillance et le réglage sont des processus hautement individualisés que vous possédez pour vos bases de données RDS. Vous pouvez utiliser l'analyse des performances d'Amazon RDS et d'autres outils pour identifier des requêtes problématiques.

Instances de base de données Amazon RDS

Une instance de base de données est un environnement de base de données isolé s'exécutant dans AWS Cloud. La fondation de base d'Amazon RDS est l'instance de base de données. Votre instance de base de données peut comporter plusieurs bases de données créées par l'utilisateur. Le schéma suivant montre un cloud privé virtuel (VPC) qui contient deux zones de disponibilité, chaque AZ contenant deux instances de base de données.



Vous pouvez accéder à vos instances de base de données à l'aide des mêmes outils et applications que ceux que vous utilisez avec une instance de base de données autonome. Vous pouvez créer

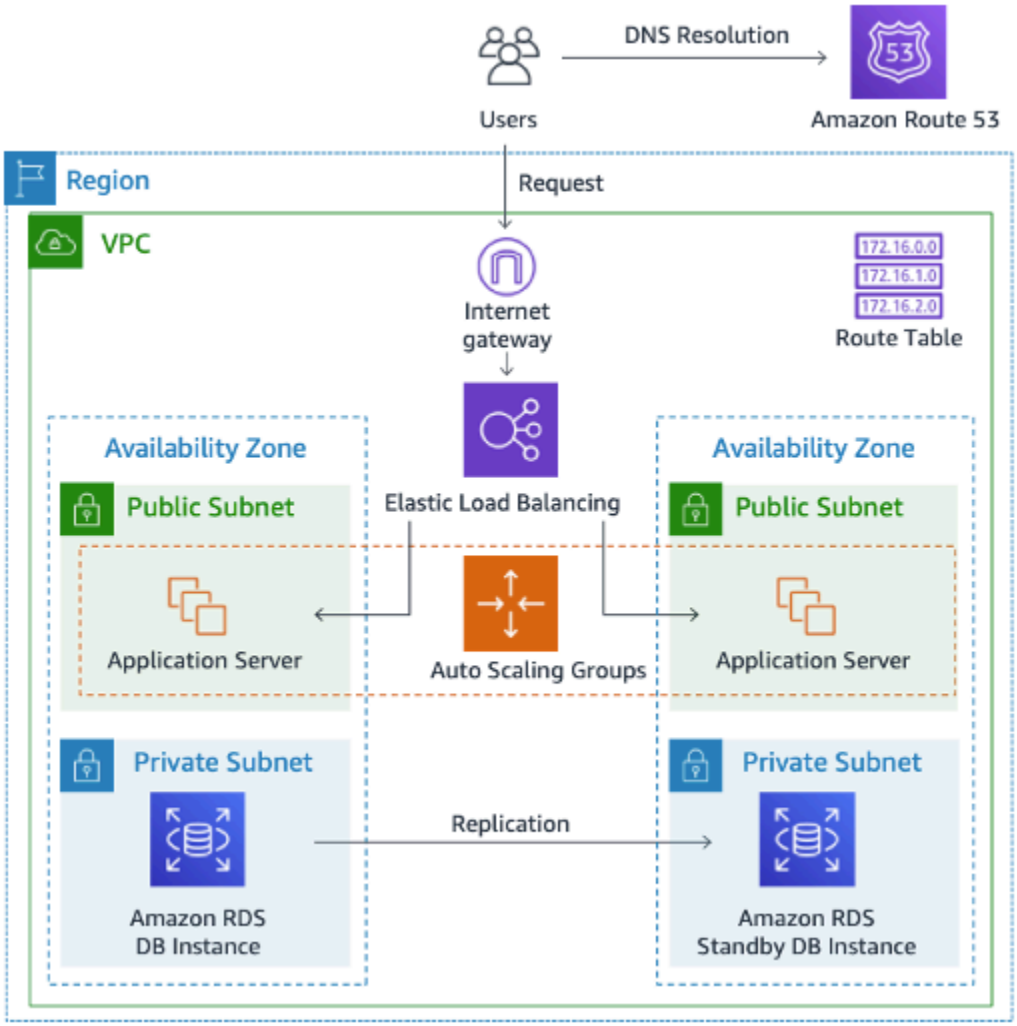
et modifier une instance de base de données en utilisant AWS Command Line Interface (AWS CLI), l'API Amazon RDS ou la AWS Management Console.

Rubriques

- Architecture d'application Amazon RDS : exemple
- Moteurs de base de données
- Classes d'instances de base de données
- Stockage d'instance de base de données
- Instances de base de données dans un Amazon Virtual Private Cloud (Amazon VPC)

Architecture d'application Amazon RDS : exemple

L'image suivante montre un cas d'utilisation typique d'un site Web dynamique qui utilise des instances de base de données Amazon RDS pour le stockage de bases de données :



Les principaux composants de l'architecture précédente sont les suivants :

Elastic Load Balancing

AWS achemine le trafic utilisateur via Elastic Load Balancing. Un équilibreur de charge répartit les charges de travail sur plusieurs ressources de calcul, telles que des serveurs virtuels. Dans cet exemple de cas d'utilisation, Elastic Load Balancer transmet les demandes des clients aux serveurs d'applications.

Serveurs d'applications

Les serveurs d'applications interagissent avec les instances de base de données RDS. Un serveur d'applications AWS est généralement hébergé sur des EC2 instances, qui fournissent une capacité de calcul évolutive. Les serveurs d'applications résident dans des sous-réseaux publics dotés de différentes zones de disponibilité (AZs) au sein du même Virtual Private Cloud (VPC).

Instances de base de données RDS

Les serveurs EC2 d'applications interagissent avec les instances de base de données RDS. Les instances de base de données résident dans des sous-réseaux privés au sein de différentes zones de disponibilité (AZs) au sein du même Virtual Private Cloud (VPC). Les sous-réseaux étant privés, aucune demande provenant d'Internet n'est autorisée.

L'instance de base de données principale se réplique vers une autre instance de base de données, appelée réplique de lecture. Les deux instances de base de données se trouvent dans des sous-réseaux privés au sein du VPC, ce qui signifie que les utilisateurs d'Internet ne peuvent pas y accéder directement.

Moteurs de base de données

Un moteur de base de données est le logiciel de base de données relationnelle spécifique qui s'exécute sur votre instance de base de données. Amazon RDS prend en charge les moteurs de base de données suivants :

- IBM Db2

  Pour de plus amples informations, veuillez consulter Amazon RDS pour DB2.

- MariaDB

  Pour de plus amples informations, veuillez consulter Amazon RDS for MariaDB.

- Microsoft SQL Server

  Pour de plus amples informations, veuillez consulter Amazon RDS for Microsoft SQL Server.

- MySQL

  Pour de plus amples informations, veuillez consulter Amazon RDS pour moi SQL.

- Oracle Database

  Pour de plus amples informations, veuillez consulter Amazon RDS for Oracle.

- PostgreSQL

  Pour de plus amples informations, veuillez consulter Amazon RDS for PostgreSQL.

Chaque moteur de base de données a ses propres fonctions prises en charge et chaque version d'un moteur de base de données peut inclure plusieurs fonctions. Support pour les fonctionnalités Amazon RDS varie selon Régions AWS les versions spécifiques de chaque moteur de base de données. Pour vérifier la prise en charge des fonctions dans différentes versions de moteurs et régions, consultez Fonctionnalités prises en charge dans Amazon RDS by Région AWS et dans le moteur de base de données.

En outre, chaque moteur DB possède un ensemble de paramètres au sein d'un groupe de paramètres DB qui contrôle le comportement des bases de données qu'il gère. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Classes d'instances de base de données

Une classe d'instance de base de données détermine la capacité de calcul et de mémoire d'une instance de base de données. Une classe d'instance de base de données comprend à la fois le type de classe d'instance de base de données et la taille. Amazon RDS prend en charge les types de classes d'instance suivants, où l'astérisque (*) représente la génération, l'attribut facultatif et la taille :

- Usage général — db.m*
- Mémoire optimisée : db.z*, db.x*, db.r*
- Optimisé pour le calcul — db.c*
- Performances éclatantes — db.t*

Chaque classe d'instance offre des capacités de calcul, de mémoire et de stockage différentes. Par exemple, db.m7g est un type de classe d'instance de base de données à usage général de 7e génération alimenté par des processeurs Graviton3. AWS Lorsque vous créez une instance de base de données, vous spécifiez une classe d'instance de base de données telle que db.m7g.2xlarge, où 2xlarge est la taille. Pour plus d'informations sur les spécifications matérielles des différentes classes d'instances, consultezSpécifications matérielles pour les classes d'instance de base de données .

Vous pouvez sélectionner la classe d'instance de base de données qui répond le mieux à vos besoins. Si vos exigences changent au fil du temps, vous pouvez modifier votre classe d'instance de base de données. Par exemple, vous pouvez étendre votre instance db.m7g.2xlarge à db.m7g.4xlarge. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .

> **Note**
>
> Pour les informations de tarification des classes d'instance de base de données, consultez la section Tarification de la page produit Amazon RDS.

Stockage d'instance de base de données

Amazon EBS fournit des volumes de stockage de niveau bloc que vous pouvez attacher à une instance en cours d'exécution. Le stockage d'instance de base de données est disponible dans les types suivants :

- Usage général (SSD)

  Ce type de stockage économique est idéal pour un large éventail de charges de travail exécutées sur des instances de base de données de taille moyenne. Le stockage à usage général convient le mieux aux environnements de développement et de test.

- IOPS provisionnées (PIOPS)

  Ce type de stockage est conçu pour répondre aux besoins de I/O-intensive workloads, particularly database workloads, that require low I/O latency and consistent I/O débit. Le stockage d'IOPS provisionnés convient le mieux aux environnements de production.

- Magnétique

Amazon RDS prend en charge le stockage magnétique à des fins de rétrocompatibilité. Nous vous recommandons d'utiliser le stockage SSD à usage général ou à IOPS provisionnés pour tout nouveau besoin de stockage.

Les types de stockage diffèrent en termes de performances et de prix. Vous pouvez adapter les performances et les coûts de stockage aux exigences de votre base de données.

Chaque instance de base de données dispose d'exigences de stockage minimal et maximal en fonction du type de stockage et du moteur de base de données qu'elle prend en charge. Il est important de disposer d'un stockage suffisant pour que vos bases de données puissent devenir plus importantes. De plus, grâce à un stockage suffisant, les fonctionnalités du moteur de base de données ont de l'espace pour écrire des contenus ou des entrées de journal. Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.

Instances de base de données dans un Amazon Virtual Private Cloud (Amazon VPC)

Vous pouvez exécuter une instance de base de données sur un Virtual Private Cloud (VPC) à l'aide du service Amazon Virtual Private Cloud (Amazon VPC). Lorsque vous utilisez un VPC, vous disposez d'un contrôle total sur l'environnement de réseau virtuel. Vous pouvez choisir votre propre plage d'adresses IP, créer des sous-réseaux et configurer le routage et les listes de contrôle d'accès.

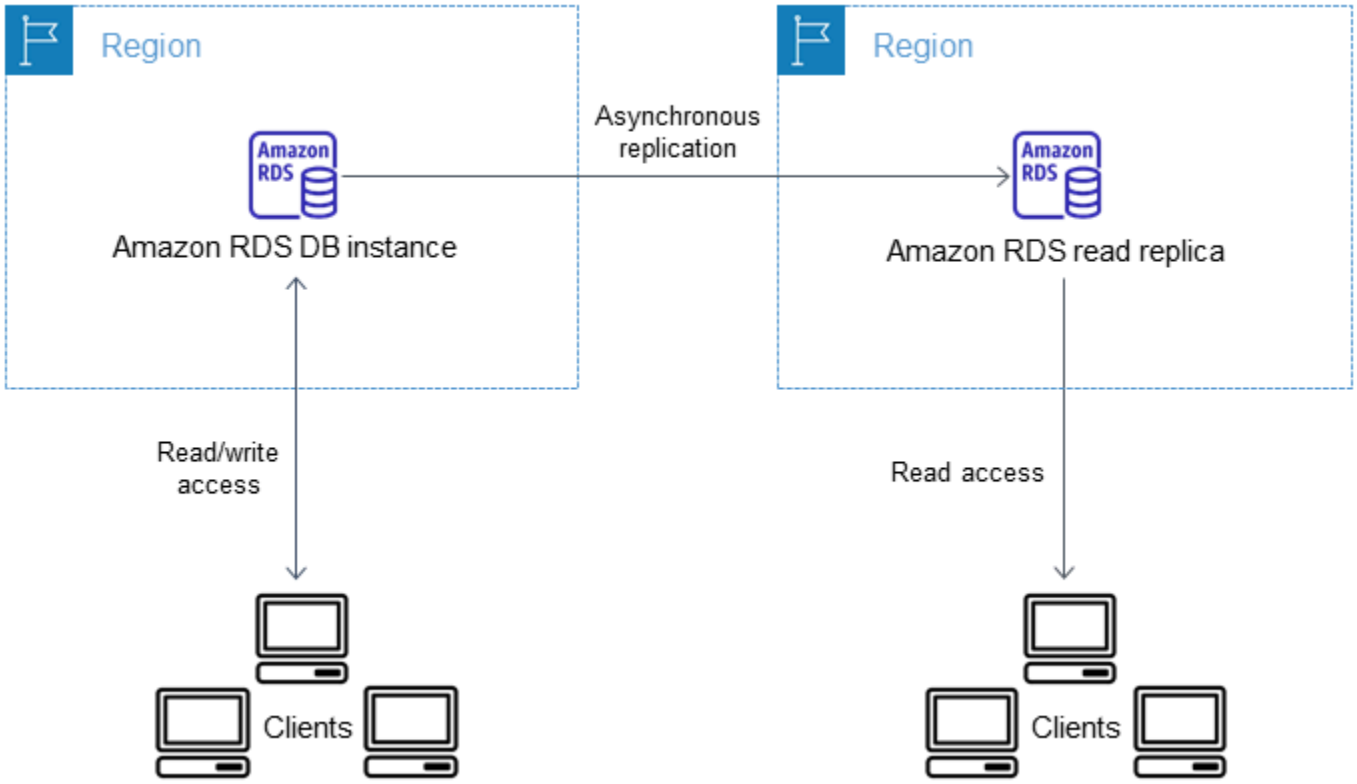
Les fonctionnalités de base d'Amazon RDS sont les mêmes, qu'il s'exécute ou non dans un VPC. Amazon RDS gère les sauvegardes, les correctifs logiciels, la détection automatique des pannes et la récupération. Il n'y a pas de frais supplémentaires pour exécuter votre instance de base de données dans un VPC. Pour de plus amples informations sur l'utilisation de Amazon VPC avec RDS, veuillez consulter Amazon VPC et Amazon RDS.

Amazon RDS utilise le protocole NTP (Network Time Protocol) pour synchroniser l'heure sur les instances de base de données.

Régions AWS et zones de disponibilité

Les ressources du cloud computing Amazon sont hébergées dans des installations de centres de données hautement disponible de différentes régions du monde (par exemple, Amérique du Nord, Europe et Asie). Chaque emplacement de centre de données est appelé Région AWS. Avec Amazon RDS, vous pouvez créer vos instances de base de données dans plusieurs régions.

Le scénario suivant montre une instance de base de données RDS dans une région qui se réplique de manière asynchrone vers une instance de base de données de secours dans une autre région. Si une région devient indisponible, l'instance de l'autre région est toujours disponible.



Zones de disponibilité

Chaque AWS région contient plusieurs emplacements distincts appelés zones de disponibilité, ou AZs. Chaque zone de disponibilité est conçue pour être isolée des défaillances dans d'autres zones de disponibilité. Chacune est conçue pour fournir une connectivité réseau peu coûteuse et à faible latence aux autres zones de disponibilité de la même AWS région. En lançant des instances de base de données dans des zones de disponibilité distinctes, vous pouvez protéger vos applications contre la défaillance d'un seul emplacement. Pour de plus amples informations, veuillez consulter Régions, zones de disponibilité et zones locales.

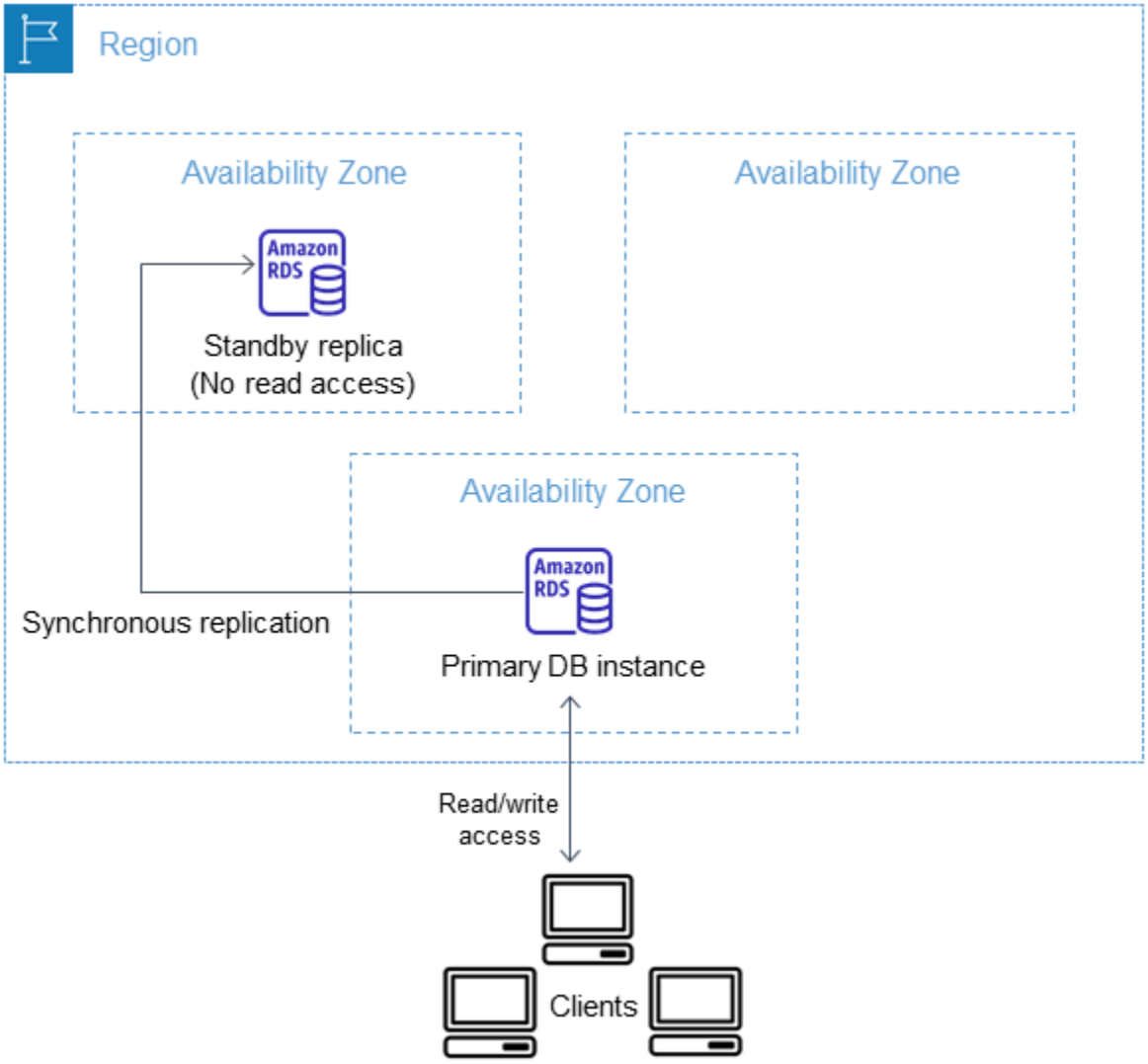
Déploiements multi-AZ

Vous pouvez exécuter votre instance de base de données dans plusieurs zones de disponibilité, option appelée déploiement Multi-AZ. Lorsque vous choisissez cette option, Amazon provisionne et gère automatiquement une ou plusieurs instances de base de données de secours secondaires dans une autre zone de disponibilité. Votre instance de base de données principale est répliquée de manière synchrone entre les zones de disponibilité dans l'instance de base de données secondaire.

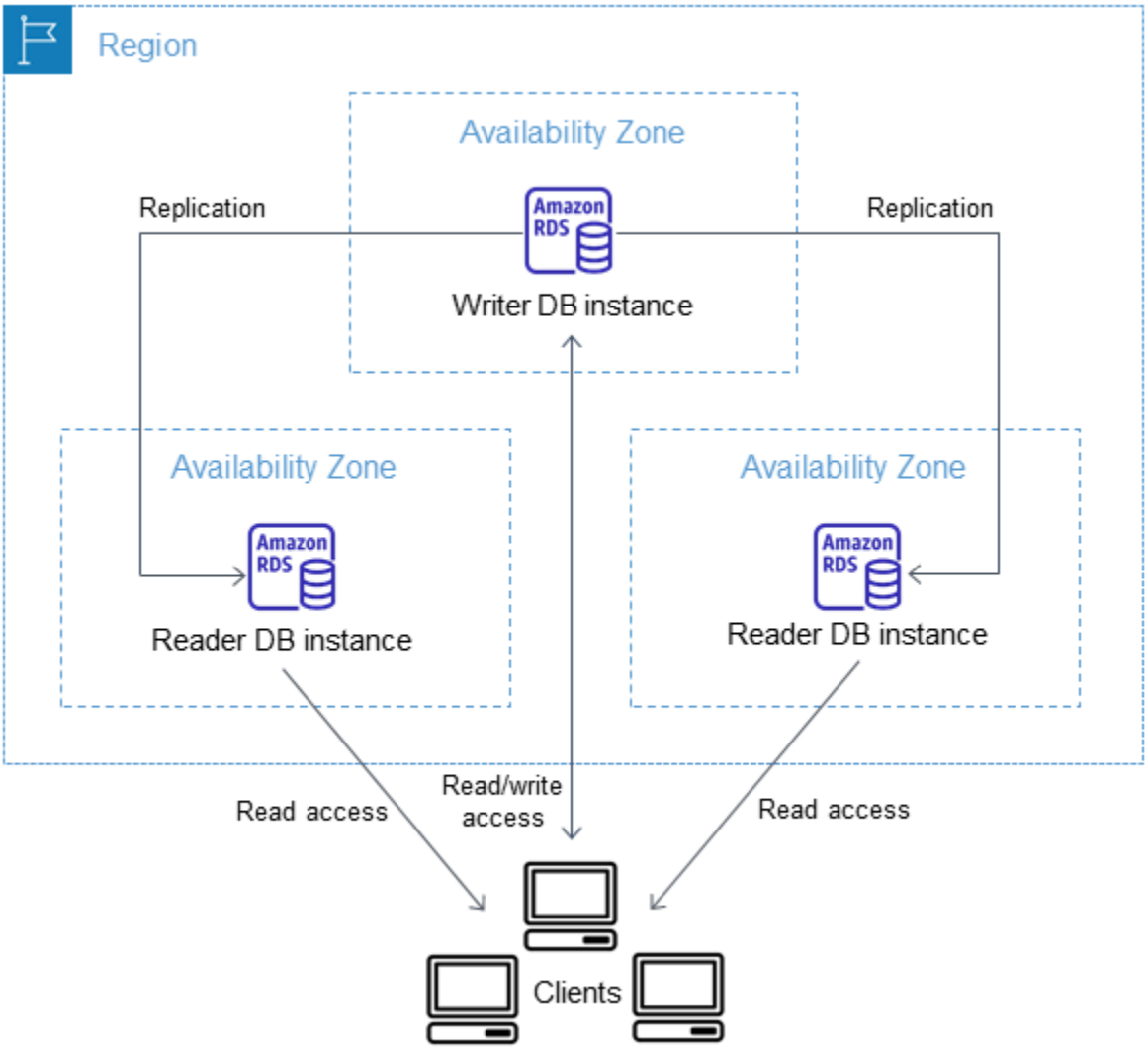
Un déploiement multi-AZ présente les avantages suivants :

- Fournir une assistance en matière de redondance et de basculement des données
- Élimination du I/O gel
- Minimiser les pics de latence lors des sauvegardes du système
- Gestion du trafic de lecture sur les instances de base de données secondaires (déploiement de clusters de base de données multi-AZ uniquement)

Le schéma suivant décrit un déploiement d'instance de base de données multi-AZ, dans lequel Amazon RDS provisionne et gère automatiquement une réplique de secours synchrone dans une zone de disponibilité différente. La base de données de répliques ne gère pas le trafic de lecture.



Le schéma suivant décrit un déploiement de cluster de base de données multi-AZ, qui comprend une instance de base de données d'écriture et deux instances de base de données de lecteur dans trois zones de disponibilité distinctes. Région AWS Les trois instances de base de données peuvent traiter le trafic de lecture.

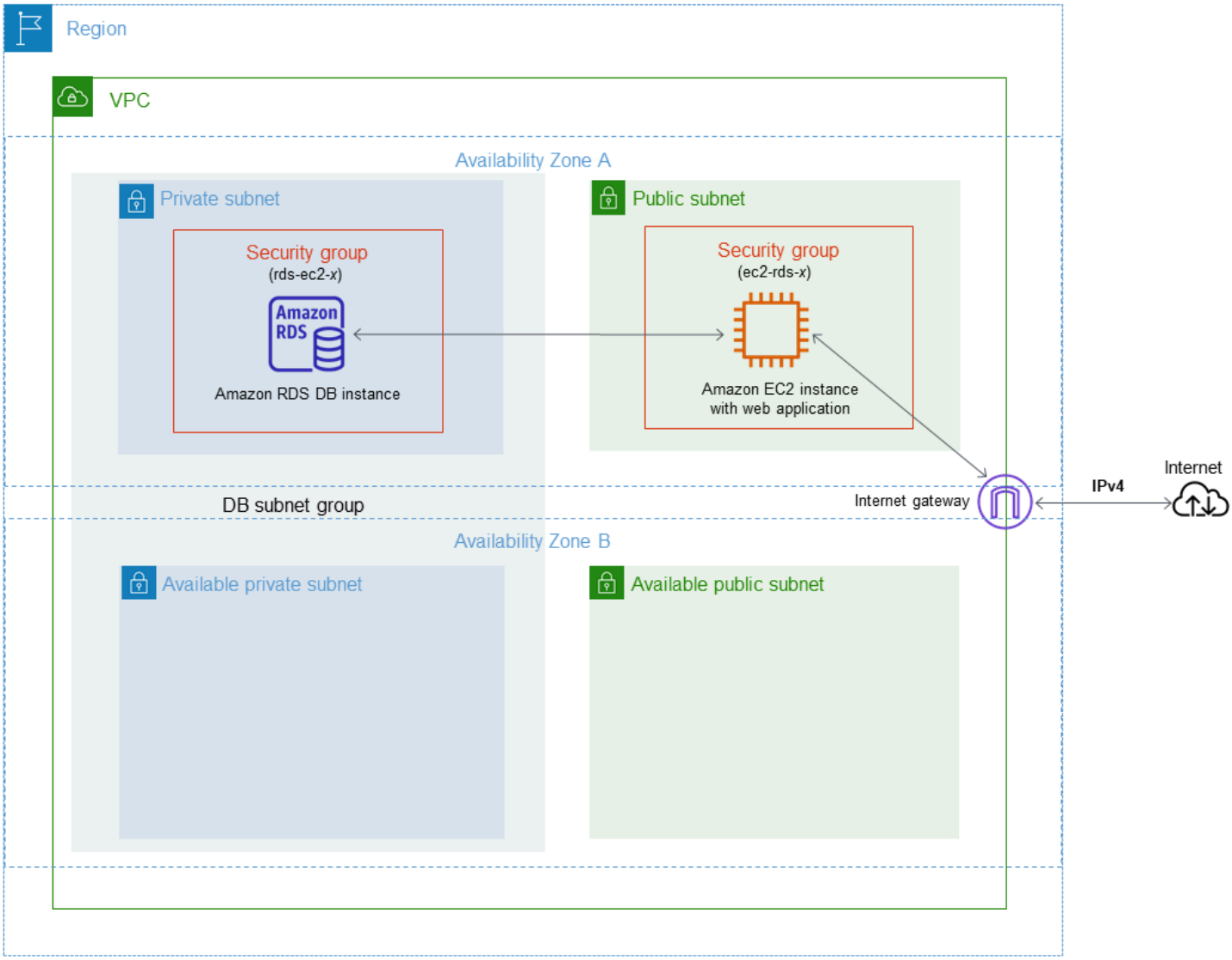


Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Contrôle d'accès avec des groupes de sécurité

Un groupe de sécurité contrôle l'accès à une instance de base de données en autorisant l'accès aux plages d'adresses IP ou aux EC2 instances Amazon que vous spécifiez. Vous pouvez appliquer un groupe de sécurité à une ou plusieurs instances de base de données.

L'utilisation courante d'une instance de base de données dans un VPC consiste à partager des données avec un serveur d'applications du même VPC. L'exemple suivant utilise le groupe de sécurité VPC `ec2-rds-x` pour définir des règles entrantes qui utilisent les adresses IP de l'application cliente comme source. Le serveur d'applications appartient à ce groupe de sécurité. Un deuxième groupe de sécurité nommé `rds-ec2-x` spécifie `ec2-rds-x` comme source et s'attache à une instance de base de données RDS. Selon les règles du groupe de sécurité, les applications clientes ne peuvent pas accéder directement à l'instance de base de données, mais l' EC2 instance peut accéder à l'instance de base de données.



Pour plus d'informations sur les groupes de sécurité, consultez Sécurité dans Amazon RDS.

Surveillance Amazon RDS

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances d'Amazon RDS, d' et de vos autres AWS solutions. AWS fournit divers outils de surveillance pour surveiller Amazon RDS, signaler un problème et prendre des mesures automatiques le cas échéant.

Vous pouvez suivre les performances et l'état de santé de vos instances de base de données à l'aide de divers outils automatisés et manuels :

État et recommandations de l'instance de base de données Amazon RDS

Consultez les informations relatives à l'état actuel de votre instance à l'aide de la console Amazon RDS ou de l'API RDS. AWS CLI Vous pouvez également répondre aux recommandations automatisées pour les ressources de base de données, telles que les instances de base de données, les répliques de lecture et les groupes de paramètres de base de données. Pour de plus amples informations, veuillez consulter Recommandations d'Amazon RDS.

CloudWatch Métriques Amazon pour Amazon RDS

Vous pouvez utiliser le CloudWatch service Amazon pour surveiller les performances et l'état d'une instance de base de données. CloudWatch les graphiques de performances sont affichés dans la console Amazon RDS. Amazon RDS envoie automatiquement des métriques CloudWatch toutes les minutes pour chaque base de données active. Vous n'avez pas à payer de frais supplémentaires pour les métriques Amazon RDS dans CloudWatch.

À l'aide des CloudWatch alarmes Amazon, vous pouvez consulter une seule métrique Amazon RDS sur une période donnée. Vous pouvez ensuite effectuer une ou plusieurs actions en fonction de la valeur de la métrique selon le seuil que vous définissez. Pour de plus amples informations, veuillez consulter Surveillance des métriques Amazon RDS () avec Amazon CloudWatch.

Amazon RDS Performance Insights et surveillance du système d'exploitation

Performance Insights évalue la charge de votre base de données et détermine quand et où prendre des mesures. Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS. Amazon RDS Enhanced Monitoring examine les métriques en temps réel du système d'exploitation. Pour de plus amples informations, veuillez consulter Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Intégré Services AWS

Amazon RDS est intégré à Amazon EventBridge, Amazon CloudWatch Logs et Amazon DevOps Guru. Pour de plus amples informations, veuillez consulter Surveillance des métriques dans une RDS instance Amazon.

Interfaces utilisateur pour Amazon RDS

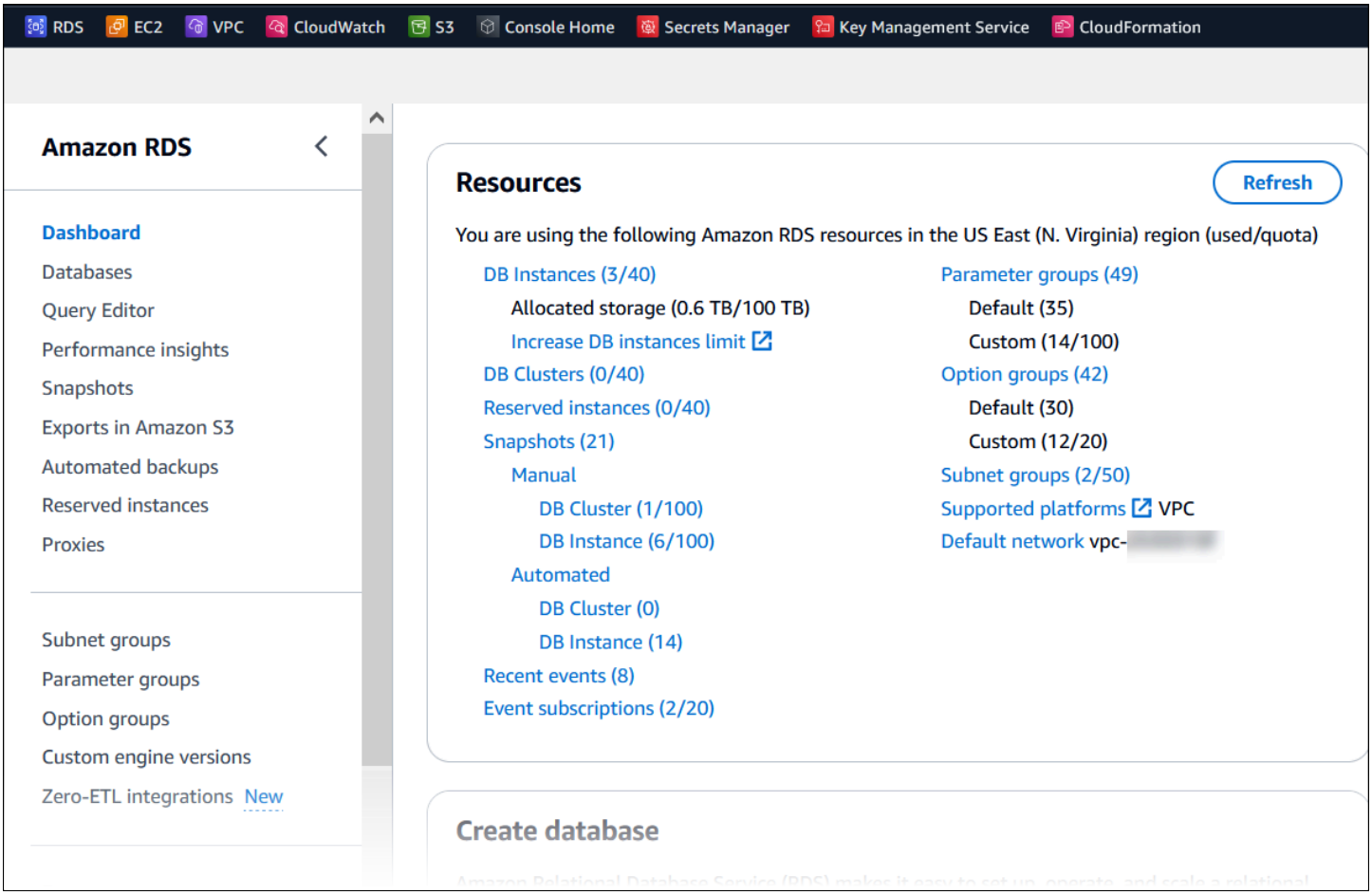
Vous pouvez interagir avec Amazon RDS de différentes manières.

Rubriques

- AWS Management Console
- interface de ligne de commande
- Amazon RDS APIs

AWS Management Console

AWS Management Console Il s'agit d'une interface utilisateur Web simple. Vous pouvez gérer vos instances de bases de données à partir de la console sans programmation requise. Pour accéder à la console Amazon RDS, connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse. https://console.aws.amazon.com/rds/



interface de ligne de commande

Vous pouvez utiliser le AWS Command Line Interface (AWS CLI) pour accéder à l'API Amazon RDS de manière interactive. Pour l'installer AWS CLI, reportez-vous à la section Installation de l'interface de ligne de AWS commande. Pour commencer à utiliser AWS CLI for RDS, consultez la AWS Command Line Interface référence relative à Amazon RDS.

Amazon RDS APIs

Si vous êtes développeur, vous pouvez accéder à Amazon RDS par programmation à l'aide de. APIs Pour de plus amples informations, veuillez consulter Référence d'API Amazon RDS.

Pour le développement d'applications, nous vous recommandons d'utiliser l'un des kits de développement AWS logiciel (SDKs). Ils AWS SDKs gèrent les détails de bas niveau tels que l'authentification, la logique des nouvelles tentatives et la gestion des erreurs, afin que vous puissiez vous concentrer sur la logique de votre application. AWS SDKs sont disponibles dans une grande

variété de langues. Pour de plus amples informations, veuillez consulter Outils pour Amazon Web Services.

AWS fournit également des bibliothèques, des exemples de code, des didacticiels et d'autres ressources pour vous aider à démarrer plus facilement. Pour de plus amples informations, veuillez consulter Exemples de codes et bibliothèques.

Comment fonctionne la facturation pour Amazon RDS

Lorsque vous utilisez Amazon RDS, vous pouvez choisir d'utiliser des instances de base de données à la demande ou réservées. Pour plus d'informations, consultez Facturation d'instance de base de données pour Amazon.

Pour plus d'informations sur la tarification de Amazon RDS, consultez la page produit de Amazon RDS.

Niveau gratuit d'AWS sur Amazon RDS

Vous pouvez l'utiliser Niveau gratuit d'AWS sur Amazon RDS avec les moteurs et classes d'instances de base de données suivants :

- Types de moteurs : MariaDB, MySQL, PostgreSQL ou SQL Server Express Edition
- classes d'instances de base de données
  - t3.micro — Pour tous les types de moteurs
  - t4g.micro — Pour tous les types de moteurs sauf SQL Server Express Edition

Vous ne pouvez pas utiliser les fonctionnalités, ressources et actions suivantes Niveau gratuit d'AWS sur Amazon RDS :

- Options de moteur personnalisées Amazon RDS
- Instances de base de données réservées
- Options de déploiement autres que le mode mono-AZ
- Migration de snapshots
- Éditeur de requête
- Création de répliques de lecture Aurora

Ces limites sont susceptibles d'être modifiées au fur et à mesure de l'évolution Niveau gratuit d'AWS du programme. Pour plus d'informations Niveau gratuit d'AWS, voir Explorer les AWS services avec Niveau gratuit d'AWS. Pour plus d'informations sur Amazon RDS Free Tier, consultez Amazon RDS Free Tier.

Si vous étiez Niveau gratuit d'AWS client avant le 17 juillet 2025, vous pouvez continuer à utiliser le niveau gratuit d'Amazon RDS jusqu'à l'expiration des 12 mois de votre période d'utilisation gratuite. Pendant cette période, vous restez éligible aux utilisations suivantes :

- 750 heures par mois de classes d'instance mono-AZ db.t3.micro et db.t4g.micro exécutant MySQL, MariaDB ou PostgreSQL sur Amazon RDS
- 750 heures par mois d'une classe d'instance db.t3.micro exécutant SQL Server Express Edition sur Amazon RDS

Après le 17 juillet 2025, toutes les nouvelles Niveau gratuit d'AWS ressources que vous AWS créerez utiliseront la nouvelle Niveau gratuit d'AWS offre.

Quelle est la prochaine étape ?

La section précédente vous a présenté les composants de base de l'infrastructure que propose RDS. Qu'allez-vous faire ensuite ?

Mise en route

Créez une instance de base de données à l'aide des instructions dans Mise en route avec Amazon RDS.

Sujets spécifiques aux moteurs de bases de données

Vous pourrez vérifier les informations spécifiques à un moteur DB particulier dans les sections suivantes :

- Amazon RDS pour DB2
- Amazon RDS for MariaDB
- Amazon RDS for Microsoft SQL Server
- Amazon RDS pour moi SQL
- Amazon RDS for Oracle

- Amazon RDS for PostgreSQL

Instances de base de données Amazon RDS

Une instance de bases de données est un environnement de base de données isolé s'exécutant dans le cloud. Elle constitue la composante de base d'Amazon RDS. Une instance de base de données peut comporter plusieurs bases de données créées par l'utilisateur et est accessible avec les mêmes applications et outils clients que ceux que vous utiliseriez pour accéder à une instance de base de données autonome. Les instances de base de données sont simples à créer et à modifier à l'aide des outils de ligne de AWS commande, des opérations d'API Amazon RDS ou du AWS Management Console.

> **Note**
>
> Amazon RDS prend en charge l'accès aux bases de données à l'aide de toute application cliente SQL standard. Amazon RDS n'autorise pas l'accès direct à l'hôte, sauf avec RDS Custom.

Vous pouvez avoir jusqu'à 40 instances de base de données Amazon RDS, avec les limitations suivantes :

- 10 instances de chaque édition SQL Server (Enterprise, Standard, Web et Express) sous le modèle « license-included (licence incluse) »
- 10 instances pour Oracle sous le modèle « license-included (licence incluse) »
- 40 pour Db2 dans le cadre du modèle de bring-your-own-license licence « » (BYOL)
- 40 instances pour MySQL, MariaDB ou PostgreSQL
- 40 pour Oracle dans le cadre du modèle de licence bring-your-own-license « » (BYOL)

> **Note**
>
> Si votre application nécessite plusieurs instances de bases de données, vous pouvez demander des instances de bases de données supplémentaires à l'aide de ce formulaire.

Chaque instance de base de données possède un identifiant d'instance de base de données. Ce nom fourni par le client identifie de manière unique l'instance de base de données lors de l'interaction avec

l'API et les commandes Amazon RDS. AWS CLI L'identifiant de l'instance de base de données doit être unique pour ce client dans une AWS région.

L'identifiant d'instance de base de données fait partie intégrante du nom d'hôte DNS alloué à votre instance par RDS. Par exemple, si vous spécifiez db1 comme identifiant d'instance de base de données, RDS attribue automatiquement un point de terminaison DNS pour votre instance. *db1*.*abcdefghijkl*.*us-east-1*.rds.amazonaws.com est un exemple de point de terminaison, où *db1* est votre ID d'instance.

Dans l'exemple de point de terminaison*db1*.*abcdefghijkl*.*us-east-1*.rds.amazonaws.com, la chaîne *abcdefghijkl* est un identifiant unique pour une combinaison spécifique de Région AWS et Compte AWS. L'identifiant *abcdefghijkl* de cet exemple est généré en interne par RDS et ne change pas pour la combinaison spécifiée d'une région et d'un compte. Ainsi, toutes vos instances de base de données figurant dans cette région partagent le même identifiant fixe. Tenez compte des caractéristiques suivantes de cet identifiant fixe :

- Si vous renommez votre instance de base de données, le point de terminaison est différent mais l'identifiant fixe est le même. Par exemple, si vous renommez *db1* en *renamed-db1*, le nouveau point de terminaison de l'instance est *renamed-db1*.*abcdefghijkl*.*us-east-1*.rds.amazonaws.com.
- Si vous supprimez et recréez une instance de base de données avec le même identifiant d'instance de base de données, le point de terminaison est le même.
- Si vous utilisez le même compte pour créer une instance de base de données dans une autre région, l'identifiant généré en interne est différent, car la région est différente, comme dans *db2*.*mnopqrstuvwx*.*us-west-1*.rds.amazonaws.com.

Chaque instance de base de données prend en charge un moteur de base de données. Amazon RDS prend actuellement en charge les moteurs de base de données DB2, MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server et Amazon Aurora.

Lorsque vous créez une instance de base de données, certains moteurs de base de données nécessitent qu'un nom de base de données soit spécifié. Une instance de base de données peut héberger plusieurs bases de données ou une seule base de données Oracle avec plusieurs schémas. La valeur du nom de base de données dépend du moteur de base de données :

- Pour le moteur de base de données DB2, le nom de la base de données est le nom de la base de données hébergée dans votre instance de base de données. Ce champ est facultatif. Vous pouvez créer une base de données ultérieurement en appelant la procédure

`rdsadmin.create_database` stockée. Pour de plus amples informations, veuillez consulter the section called “Création d’une base de données”.

- Pour les moteurs de base de données MySQL et MariaDB, le nom de base de données est le nom d'une base de données hébergée dans votre instance de base de données. Les bases de données hébergées par la même instance de base de données doivent avoir un nom unique au sein de cette instance.
- Pour le moteur de base de données Oracle, le nom de base de données est utilisé pour définir la valeur d'ORACLE_SID, qui doit être fournie lors de la connexion à l'instance Oracle RDS.
- Pour le moteur de base de données Microsoft SQL Server, le nom de base de données n'est pas un paramètre pris en charge.
- Pour le moteur de base de données PostgreSQL, le nom de base de données est le nom d'une base de données hébergée dans votre instance de base de données. Un nom de base de données n'est pas obligatoire lors de la création d'une instance de base de données. Les bases de données hébergées par la même instance de base de données doivent avoir un nom unique au sein de cette instance.

Amazon RDS crée un compte utilisateur principal pour votre instance de base de données comme partie intégrante du processus de création. Cet utilisateur principal dispose des autorisations pour créer des bases de données et exécuter des opérations de création, suppression, sélection, mise à jour et insertion sur les tables créées par l'utilisateur principal. Vous devez définir le mot de passe de l'utilisateur principal lorsque vous créez une instance de base de données, mais vous pouvez le modifier à tout moment à l' AWS CLI aide des opérations d'API Amazon RDS ou du AWS Management Console. Vous pouvez aussi modifier le mot de passe de l'utilisateur principal et gérer les utilisateurs à l'aide des commandes SQL standard.

> **Note**
>
> Ce guide traite des moteurs de base de données Amazon RDS non-Aurora. Pour plus d'informations sur l'utilisation d'Amazon Aurora, consultez le Guide de l'utilisateur Amazon Aurora.

Classes d'instances de base de données

La classe d'instance de base de données détermine la capacité de calcul et de mémoire d'une instance de base de données Amazon RDS . La classe d'instance de base de données dont vous avez besoin varie selon vos exigences en mémoire et en puissance de traitement.

Une classe d'instance de base de données comprend à la fois le type de classe d'instance de base de données et la taille. Par exemple, db.r6g est un type de classe d'instance de base de données optimisé pour la mémoire et alimenté par les processeurs Graviton2. AWS Dans le type de classe d'instance db.m6g, db.r6g.2xlarge est une classe d'instance de base de données. La taille de cette classe est 2xlarge.

Pour de plus amples informations sur la tarification des classes d'instance, veuillez consulter Tarification Amazon RDS.

Pour plus d'informations sur les types de classes d'instances de base de données, les moteurs de base de données pris en charge Régions AWS, la modification de votre classe d'instance de base de données, la configuration du processeur pour RDS pour Oracle ou les spécifications matérielles des classes d'instances de base de données, consultez les sections suivantes.

Rubriques

- Types de classes d'instance de base de données
- Moteurs de base de données pris en charge pour les classes d'instance de base de données
- Déterminer le support des classes d'instance de base de données dans Régions AWS
- Modification d'une classe d'instance de base de données
- Configuration du processeur pour une classe d'instances de base de données dans RDS for Oracle
- Spécifications matérielles pour les classes d'instance de base de données

Types de classes d'instance de base de données

Amazon RDS prend en charge les classes d'instances de base de données pour les cas d'utilisation suivants :

- Usage général
- Optimisé pour la mémoire
- Optimisé pour le calcul
- Capacité extensible

- Optimized Reads

Pour plus d'informations sur les types d' EC2 instances Amazon, consultez la section Types d'instances dans la EC2 documentation Amazon.

Types de classes d'instances à usage général

Les classes d'instances de base de données à usage général disponibles sont les suivantes :

- db.m8g — Classes d'instance de base de données à usage général alimentées par les processeurs Graviton4. AWS Ces classes d'instances fournissent des capacités de calcul, de mémoire et de réseau équilibrées pour une large gamme de charges de travail à usage général. Par rapport aux instances m7G AWS basées sur Graviton3 de septième génération, ces nouvelles classes offrent des tailles d'instance plus importantes avec jusqu'à 3 fois plus de v et de mémoire. CPUs

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton4. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.m7i — Classes d'instances de base de données à usage général alimentées par des processeurs Intel Xeon Scalable de 4e génération. Les instances db.m7i sont certifiées SAP et sont idéales pour prendre en charge les applications d'entreprise nécessitant des tailles d'instance plus importantes ou une utilisation continue élevée du processeur. Ces classes d'instances fournissent des capacités de calcul, de mémoire et de réseau équilibrées pour une large gamme de charges de travail à usage général. Ce type de classe d'instance fournit jusqu'à 40 000 Mbits/s de bande passante EBS et jusqu'à 50 Gbit/s de bande passante réseau. Les instances db.m7i offrent un rapport prix/performances jusqu'à 15 % supérieur à celui des instances db.m6i.
- db.m7g — Classes d'instance de base de données à usage général alimentées par les processeurs Graviton3. AWS Ces classes d'instances fournissent des capacités de calcul, de mémoire et de réseau équilibrées pour une large gamme de charges de travail à usage général.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton3. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.m6g — Classes d'instance de base de données à usage général alimentées par les processeurs Graviton2. AWS Ces instances fournissent des capacités de calcul, de mémoire et de réseau équilibrées pour une large gamme de charges de travail à usage général. Les classes

d'instance db.m6gd disposent d'un stockage SSD local NVMe au niveau des blocs pour les applications nécessitant un stockage local à haut débit et à faible latence.

Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton2. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.

- db.m6i : classes d'instances de base de données à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération. Ces instances sont certifiées SAP et sont idéales pour les charges de travail telles que les serveurs backend prenant en charge les applications d'entreprise, les serveurs de jeu, les flottes de mise en cache et les environnements de développement d'applications. Les classes d'instance db.m6id et db.m6idn offrent jusqu'à 7,6 To de stockage SSD local, tandis que db.m6in propose un stockage EBS uniquement NVMe. Les classes db.m6in et db.m6idn offrent jusqu'à 200 Gbits/s de bande passante réseau.
- db.m5 : classes d'instances de base de données à usage général qui assurent l'équilibre entre ressources de calcul, de mémoire et de réseau et qui constituent le choix idéal pour de nombreuses applications. La classe d'instance db.m5d propose un stockage SSD NVMe basé sur une connexion physique au serveur hôte. Les classes d'instances db.m5 offrent une plus grande capacité de calcul que les précédentes classes d'instance db.m4. Elles sont alimentées par le système AWS Nitro, qui allie un matériel dédié et un hyperviseur léger.
- db.m4 : classes d'instances de base de données à usage général qui offrent une plus grande capacité de calcul que les précédentes classes d'instances db.m3.

  Pour les moteurs de base de données RDS for Oracle, Amazon RDS ne prend plus en charge les classes d'instances de base de données db.m4. Si vous avez précédemment créé des instances de base de données RDS for Oracle db.m4, Amazon RDS met automatiquement à niveau ces instances de base de données vers des classes d'instances de base de données db.m5 équivalentes.

  Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL, RDS pour SQL Server et RDS pour les moteurs de base de données PostgreSQL, Amazon RDS a lancé le processus pour cette classe d'instance de base de données selon le calendrier suivant. end-of-support Pour toutes les instances de base de données RDS qui utilisent cette classe d'instance, nous vous recommandons de passer à une classe d'instance de base de données de nouvelle génération dès que possible.

Action ou recommandation	Date
À compter de cette date, Amazon RDS a commencé à mettre à niveau automatiquement les instances utilisant db.m4 vers la classe d'instance db.m5 de nouvelle génération. La création d'instanc es de base de données à l'aide de la classe d'instance db.m4 n'est plus prise en charge.	1er juin 2024
Amazon RDS met fin au support de db.m4.	31 décembre 2024

- db.m3 : classes d'instances de base de données à usage général qui offrent une plus grande capacité de calcul que les précédentes classes d'instances db.m1.

  Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL, Amazon RDS a lancé le processus pour les classes d'instances de base de données db.m3 selon end-of-life le calendrier suivant, qui inclut des recommandations de mise à niveau. Pour toutes les instances de base de données RDS qui utilisent les classes d'instance de base de données db.m3, nous vous recommandons de passer à une classe d'instance de base de données de génération supérieure dès que possible.

Action ou recommandation	Dates
Vous ne pouvez plus créer d'instances de base de données RDS qui utilisent des classes d'instances de base de données db.m3.	Maintenant
Amazon RDS a lancé des mises à niveau automatiq ues des instances de base de données RDS qui utilisent des classes d'instances de base de données db.m3 vers des classes d'instances de base de données équivalentes db.m5.	1er février 2023

Types de classes d'instances à mémoire optimisée

La famille Z à mémoire optimisée prend en charge les classes d'instances suivantes :

- db.z1d – Classes d'instance optimisées pour les applications gourmandes en mémoire. Ces classes d'instances offrent une forte capacité de calcul et une forte empreinte mémoire. Les instances z1d à haute fréquence fournissent une fréquence intégrale soutenue allant jusqu'à 4,0. GHz

La famille X à mémoire optimisée prend en charge les classes d'instances suivantes :

- db.x2g — Classes d'instance optimisées pour les applications gourmandes en mémoire et alimentées par les processeurs Graviton2. AWS Ces classes d'instances offrent un faible coût par Gio de mémoire.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton2. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.x2i : classes d'instances optimisées pour les applications gourmandes en mémoire. Les types de classes d'instances db.x2iedn et db.x2idn fonctionnent avec des processeurs Intel Xeon Scalable de troisième génération (Ice Lake). Ils incluent jusqu'à 3,8 To de stockage NVMe SSD local, jusqu'à 100 Gbit/s de bande passante réseau et jusqu'à 4 TiB (db.x2iden) ou 2 TiB (db.x2idn) de mémoire. Le type db.x2iezn est alimenté par des processeurs Intel Xeon Scalable de deuxième génération (Cascade Lake) avec une fréquence turbo intégrale allant jusqu'à 4,5 et GHz jusqu'à 1,5 TiB de mémoire.
- db.x1 – Classes d'instance optimisées pour les applications gourmandes en mémoire. Ces classes d'instances offrent l'un des prix les moins élevés par Gio de RAM parmi les classes d'instances de base de données et jusqu'à 1 952 Gio de mémoire d'instance basée sur DRAM. Le type de classe d'instance db.x1e offre jusqu'à 3 904 Gio de mémoire d'instance DRAM.

La famille R à mémoire optimisée prend en charge les types de classe d'instance suivants :

- db.r8g — Classes d'instances alimentées par les processeurs Graviton4. AWS Ces classes d'instances sont idéales pour exécuter des charges de travail exigeantes en mémoire dans des bases de données open source telles que MySQL et PostgreSQL. Ces instances offrent des tailles d'instance plus importantes avec jusqu'à 3 fois plus de v CPUs et de mémoire que les instances db.r7g basées sur AWS Graviton3 de septième génération.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton4. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.

- db.r7g — Classes d'instances alimentées par les processeurs Graviton3. AWS Ces classes d'instances sont idéales pour exécuter des charges de travail exigeantes en mémoire dans des bases de données open source telles que MySQL et PostgreSQL.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton3. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.r7i — Classes d'instances alimentées par des processeurs Intel Xeon Scalable de 4e génération. Ces classes d'instance sont certifiées SAP et sont idéales pour exécuter des charges de travail gourmandes en mémoire dans des bases de données open source telles que MySQL et PostgreSQL. Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instances de base de données alimentées par les processeurs Intel Xeon Scalable de 4e génération. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.r6g — Classes d'instances alimentées par les processeurs Graviton2. AWS Ces classes d'instances sont idéales pour exécuter des charges de travail exigeantes en mémoire dans des bases de données open source telles que MySQL et PostgreSQL. Le type db.r6gd offre un stockage SSD local NVMe au niveau des blocs pour les applications nécessitant un stockage local à haut débit et à faible latence.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton2. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.r6i : classes d'instances alimentées par des processeurs Intel Xeon Scalable de 3e génération. Ces classes d'instance sont certifiées SAP et conviennent parfaitement aux charges de travail gourmandes en mémoire dans les bases de données open source telles que MySQL et PostgreSQL. Les classes d'instance db.r6id, db.r6in et db.r6idn ont un ratio de 8:1 et une mémoire maximale de 1 TiB. memory-to-vCPU Les classes db.r6id et db.r6idn offrent jusqu'à 7,6 To de stockage SSD à connexion directe NVMe, tandis que db.r6in propose un stockage EBS uniquement. Les classes db.r6idn et db.r6in offrent jusqu'à 200 Gbits/s de bande passante réseau.
- db.r5b – Classes d'instances optimisées pour la mémoire pour les applications à débit élevé. Alimentées par le système AWS Nitro, les instances db.r5b fournissent une bande passante allant jusqu'à 60 Gbit/s et 260 000 IOPS de performances EBS. Il s'agit de la performance de stockage par blocs la plus rapide jamais enregistrée EC2.
- db.r5d — Classes d'instances optimisées pour une faible latence, des I/O performances aléatoires très élevées et un débit de lecture séquentiel élevé.

- db.r4 – Classes d'instance optimisées pour les applications gourmandes en mémoire. Ces classes d'instances offrent une amélioration de la mise en réseau et des performances . Ils sont alimentés par le système AWS Nitro, une combinaison de matériel dédié et d'hyperviseur léger.
- db.r4 – Classes d'instance qui fournissent une mise en réseau améliorée par rapport aux classes d'instance db.r3 précédentes.

  Pour les moteurs de base de données RDS pour Oracle, Amazon RDS a lancé le end-of-life processus pour les classes d'instance de base de données db.r4 selon le calendrier suivant, qui inclut des recommandations de mise à niveau. Pour les instances de base de données RDS pour Oracle qui utilisent les classes d'instance db.r4, nous vous recommandons de passer à une classe d'instance de génération supérieure dès que possible.

Action ou recommandation	Dates
Vous ne pouvez plus créer d'instances de base de données RDS for Oracle qui utilisent des classes d'instance de base de données db.r4.	Maintenant
Amazon RDS a lancé des mises à niveau automatiq ues des instances de base de données RDS for Oracle qui utilisent des classes d'instance de base de données db.r4 vers des classes d'instance de base de données équivalentes db.r5.	17 avril 2023

Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL, RDS pour SQL Server et RDS pour les moteurs de base de données PostgreSQL, Amazon RDS a lancé le processus pour cette classe d'instance de base de données selon le calendrier suivant. end-of-support Pour toutes les instances de base de données RDS qui utilisent cette classe d'instance, nous vous recommandons de passer à une classe d'instance de base de données de nouvelle génération dès que possible.

Action ou recommandation	Dates
À compter de cette date, Amazon RDS a commencé à mettre à niveau automatiquement les	1er juin 2024

Action ou recommandation	Dates
instances utilisant db.r4 vers la classe d'instance db.r5 de nouvelle génération. La création d'instanc es de base de données à l'aide de la classe d'instance db.m4 n'est plus prise en charge.	
Amazon RDS met fin au support de db.r4.	31 décembre 2024

- db.r3 – Classes d'instances fournissant une optimisation de la mémoire.

  Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL, Amazon RDS a lancé le processus pour les classes d'instances de base de données db.r3 selon end-of-life le calendrier suivant, qui inclut des recommandations de mise à niveau. Pour toutes les instances de base de données RDS qui utilisent les classes d'instance de base de données db.r3, nous vous recommandons de passer à une classe d'instance de base de données de génération supérieure dès que possible.

Action ou recommandation	Dates
Vous pouvez maintenant créer des instances de base de données RDS qui utilisent les classes d'instances de base de données db.r3.	Maintenant
Amazon RDS a lancé des mises à niveau automatiq ues des instances de base de données RDS qui utilisent des classes d'instances de base de données db.r3 vers des classes d'instances de base de données équivalentes db.r5.	1er février 2023

Type de classe d'instance optimisé pour le calcul

Les types de classes d'instance optimisés pour le calcul suivants sont disponibles :

- db.c6gd — Classes d'instance idéales pour exécuter des charges de travail avancées intensives en calcul. Alimentées par des processeurs AWS Graviton2, ces classes d'instances offrent un stockage SSD local NVMe au niveau des blocs pour les applications nécessitant un stockage local à haut débit et à faible latence.

> **Note**
>
> Les classes d'instance c6gd ne sont prises en charge que pour les déploiements de clusters de bases de données multi-AZ. Il s'agit de la seule classe d'instance prise en charge pour les clusters de bases de données multi-AZ qui offrent la même taille d'`medium`instance. Pour de plus amples informations, veuillez consulter the section called “Déploiements de clusters de base de données multi-AZ”.

Types de classes d'instance à capacité extensible

Les types de classes d'instances de base de données à capacité extensible disponibles sont les suivants :

- db.t4g — Classes d'instance à usage général alimentées par des processeurs Graviton2 basés sur ARM. AWS Ces classes d'instances offrent de meilleures performances de prix que les précédentes classes d'instances de base de données de performance à capacité extensible pour un large ensemble de charges de travail extensibles à usage général. Les instances Amazon RDS db.t4g sont configurées pour le mode illimité. Cela signifie qu'elles peuvent dépasser le niveau de base d'utilisation de l'UC sur une période de 24 heures moyennant des frais supplémentaires.

  Vous pouvez modifier une instance de base de données pour utiliser l'une des classes d'instance de base de données alimentées par les processeurs AWS Graviton2. Pour ce faire, suivez les mêmes étapes que pour toute autre modification d'une instance de base de données.
- db.t2 – Classes d'instances qui fournissent un niveau de performance de base, avec la possibilité de transmission étendue jusqu'à une utilisation intégrale de l'UC. Les instances db.t3 sont configurées pour le mode Illimité. Ces classes d'instances offrent une plus grande capacité de calcul que les précédentes classes d'instance db.t2. Elles sont alimentées par le système AWS Nitro, qui allie un matériel dédié et un hyperviseur léger.
- db.t2 – Classes d'instances qui fournissent un niveau de performance de base, avec la possibilité de transmission étendue jusqu'à une utilisation intégrale de l'UC. Les instances db.t2 sont configurées pour le mode illimité. Nous recommandons d'utiliser ces classes d'instances uniquement pour les serveurs de développement et de test, ou pour d'autres serveurs non dédiés à la production.

  Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL, RDS pour SQL Server et RDS pour les moteurs de base de données PostgreSQL, Amazon RDS a lancé le

processus pour cette classe d'instance de base de données selon le calendrier suivant. end-of-support Pour toutes les instances de base de données RDS qui utilisent cette classe d'instance, nous vous recommandons de passer à une classe d'instance de base de données de nouvelle génération dès que possible.

Action ou recommandation	Dates
À compter de cette date, Amazon RDS a commencé à mettre à niveau automatiquement les instances utilisant db.t2 vers la classe d'instance db.t3 de nouvelle génération. La création d'instanc es de base de données à l'aide de la classe d'instance db.t2 n'est plus prise en charge.	1er juin 2024
Amazon RDS met fin à la prise en charge de db.t2.	31 décembre 2024

> **Note**
>
> Les classes d'instances de base de données qui utilisent le système AWS Nitro (db.m5, db.r5, db.t3) sont limitées en cas de charge de travail combinée en lecture et en écriture.

Pour de plus amples informations sur les spécifications matérielles de classe d'instance de base de données, veuillez consulter Spécifications matérielles pour les classes d'instance de base de données .

Types de classes d'instance Optimized Reads

Les types de classe d'instances Optimized Reads disponibles sont les suivants :

- db.r6gd — Classes d'instances alimentées par les processeurs Graviton2. AWS Ces classes d'instances sont idéales pour exécuter des charges de travail gourmandes en mémoire et offrent un stockage SSD local au niveau NVMe des blocs pour les applications nécessitant un stockage local à haut débit et à faible latence.
- db.r6i : classes d'instances alimentées par des processeurs Intel Xeon Scalable de 3e génération. Ces classes d'instances sont certifiées SAP et sont idéales pour les charges de travail qui

demandent beaucoup de mémoire. Ils offrent une mémoire maximale de 1 TiB et jusqu'à 7,6 To de stockage SSD à connexion directe. NVMe

Moteurs de base de données pris en charge pour les classes d'instance de base de données

Vous trouverez ci-après des considérations spécifiques au moteur de base de données pour les classes d'instances de base de données :

Db2

La prise en charge des classes d'instance de base de données varie en fonction de la version et de l'édition de Db2. Pour une prise en charge des classes d'instances par version et édition, consultez Amazon RDS pour les classes d'instance DB2.

Microsoft SQL Server

La prise en charge des classes d'instance de base de données varie selon la version et l'édition de SQL Server. Pour une prise en charge des classes d'instances par version et édition, consultez prise en charge des classes d'instance de base de données pour Microsoft SQL Server.

Oracle

La prise en charge des classes d'instance de base de données varie selon la version et l'édition d'Oracle Database. RDS for Oracle prend en charge d'autres classes d'instances optimisées en mémoire. Ces classes ont des noms de la forme db.r5. *instance_size*.tpc *threads_per_core* .mem. *ratio* Pour connaître le nombre de vCPU et l'allocation de mémoire pour chaque classe optimisée, consultez RDS pris en charge pour les classes d'instance de base de données Oracle.

RDS Custom

Pour obtenir des informations sur les classes d'instances de base de données prises en charge dans RDS Custom, consultez Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle et Support de classe d'instance de base de données pour RDS Custom for SQL Server.

Le tableau suivant vous présente des détails sur les classes d'instances de base de données Amazon RDS prises en charge pour chaque moteur de base de données Amazon RDS. La cellule de chaque moteur contient l'une des valeurs suivantes :

Oui

La classe d'instances est prise en charge pour toutes les versions du moteur de base de données.

Non

La classe d'instances de base de données n'est pas prise en charge pour le moteur de base de données.

specific-versions

La classe d'instances est prise en charge pour les versions de base de données spécifiées du moteur de base de données.

Amazon RDS déconseille régulièrement les versions majeures et mineures du moteur de base de données. Tous ne sont Régions AWS peut-être pas compatibles avec les versions antérieures du moteur. Pour plus d'informations sur les versions actuellement prises en charge, consultez les rubriques relatives aux moteurs de base de données individuels : versions DB2, versionsMariaDB, versions Microsoft SQL Server, versions MySQL, versions Oracle et versions PostgreSQL.

Rubriques

- Moteurs de base de données pris en charge pour les classes d'instance à usage général
- Moteurs de base de données pris en charge pour les classes d'instance optimisées pour la mémoire
- Moteurs de base de données pris en charge pour les classes d'instances optimisées pour le calcul
- Moteurs de base de données pris en charge pour les classes d'instance aux performances évolutives
- Moteurs de base de données compatibles pour les classes d'instance Optimized Reads

Moteurs de base de données pris en charge pour les classes d'instance à usage général

Les tableaux suivants indiquent les bases de données et les versions de base de données prises en charge pour les classes d'instances à usage général.

db.m8g — classes d'instance à usage général alimentées par les processeurs Graviton4 AWS

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m8g.48 x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.24 x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.16 xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.12 x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
						supérieure, 13.11 et supérieure
db.m8g.8x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.4x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.2x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m8g.xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m8g.large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et supérieur	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

db.m7i : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 4e génération

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m7i.48x large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, EE uniquel t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m7i 24 x large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, EE uniquel t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i.16 x large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, EE uniquel t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i 12 x large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, EE uniquel t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i.8x large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, EE uniquel t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m7i.4xlarge	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, toutes les édition	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i.2xlarge	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, toutes les édition	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i.xlarge	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et supérieur	BYOL uniquel t, toutes les édition	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.m7i.large	DB2 11,5	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquel t, toutes les édition	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

db.m7g : classes d'instances à usage général optimisées par des processeurs AWS Graviton3

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m7g.1(xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieur es 14, et les versions 13.4 et supérieures 13
db.m7g.1: xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieur es 14, et les versions 13.4 et supérieures 13
db.m7g.8: large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieur es 14, et les versions 13.4 et supérieures 13
db.m7g.4: large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieur es 14, et les

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
						versions 13.4 et supérieures 13
db.m7g.2x large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.m7g.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.m7g.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13

db.m6g : classes d'instances polyvalentes optimisées par des processeurs AWS Graviton2

Classe d'instanc e	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.m6g.1(xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieur es
db.m6g.1: xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieur es
db.m6g.8: large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieur es
db.m6g.4: large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieur es
db.m6g.2: large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieur es
db.m6g.xl arge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						12 versions ultérieures
db.m6g.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

db.m6gd — classes d'instance à usage général alimentées par des processeurs Graviton2 et un stockage SSD AWS

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.m6g.16xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4
db.m6g.12xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4
db.m6g.8xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.m6g.4xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4
db.m6g.2xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4
db.m6gd.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4
db.m6gd.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16, 15 et 14 ; 13.7 et versions ultérieures 13 ; et 13.4

db.m6id : classes d'instances à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération et un stockage SSD

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m6id.32xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL	Toutes les versions 17, 16 et 15 de

Classe d'instance	Db	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
					uniquement	PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.24xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.16xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.12xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m6id.8xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.4xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.2xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6id.xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m6id.large	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

db.m6idn – classes d'instances à usage général avec processeurs Intel Xeon Scalable de 3e génération, stockage SSD et optimisation réseau

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m6idn.32xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.24xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.16xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
						les versions 13.7 et supérieures 13
db.m6idn.12xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.8xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.4xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.2xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m6idn.xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.m6idn.large	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

db.m6in : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération et optimisations du réseau

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.m6in.32xlarge	No	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

Classe d'instance	Db	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.m6in.24xlarge	No	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.m6in.16xlarge	No	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.m6in.12xlarge	No	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

Classe d'instance	Db	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.m6in.8xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYC unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.m6in.4xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYC unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.m6in.2xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYC unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.m6in.xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.m6in.large	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

db.m6i : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m6i.32xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibles
db.m6i.24xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibles

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.m6i.16 xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.12 xlarge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.8x large	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.4x large	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.2x large	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.xl arge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es
db.m6i.la rge	Ou	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oracle Databa	Toutes versions disponibl es

db.m5d : classes d'instances à usage général alimentées par des processeurs Intel Xeon Platinum et un stockage SSD

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.m5d.24 xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m5d.16xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.m5d.12xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.m5d.8xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.m5d.4xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.m5d.2xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m5d.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.m5d.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

db.m5 — processeurs Intel Xeon Platinum de classe 2.5 à usage général GHz

Classe d'instance	Db2	Maria	Microsoft SQL Server	MyS	Oracl	PostgreSQL
db.m5.24xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.16xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.12xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.8xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9

Classe d'instance	Db2	Maria	Microsoft SQL Server	MyS	Oracl	PostgreSQL
db.m5.4xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.2xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.m5.large	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9

db.m4 — classes d'instances à usage général dotées de processeurs Intel Xeon

Classe d'instance	Db	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m4.16large	No	Obsolète	Obsolète	Obsolète	Obsol	Obsolète
db.m4.10large	No	Obsolète	Obsolète	Obsolète	Obsol	Obsolète
db.m4.4xarge	No	Obsolète	Obsolète	Obsolète	Obsol	Obsolète
db.m4.2xarge	No	Obsolète	Obsolète	Obsolète	Obsol	Obsolète

Classe d'instanc e	Dl	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m4.xla rge	Nc	Obsolète	Obsolète	Obsolète	Obsol	Obsolète
db.m4.lar ge	Nc	Obsolète	Obsolète	Obsolète	Obsol	Obsolète

db.m3 : classes d'instances polyvalentes

Classe d'instanc e	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.m3.2xl arge	Non	Non	Obsolète	Oui	Obsolète	Obsolète
db.m3.xla rge	Non	Non	Obsolète	Oui	Obsolète	Obsolète
db.m3.lar ge	Non	Non	Obsolète	Oui	Obsolète	Obsolète
db.m3.med ium	Non	Non	Obsolète	Oui	Obsolète	Obsolète

Moteurs de base de données pris en charge pour les classes d'instance optimisées pour la mémoire

Les tableaux suivants indiquent les bases de données et les versions de base de données prises en charge pour les classes d'instances optimisées pour la mémoire.

db.z1d : classes d'instance à mémoire optimisée

Classe d'instanc e	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.z1d.1 xlarge	Non	Non	Oui	Non	Oui	Non
db.z1d.6 large	Non	Non	Oui	Non	Oui	Non
db.z1d.3 large	Non	Non	Oui	Non	Oui	Non
db.z1d.2 large	Non	Non	Oui	Non	Oui	Non
db.z1d.xl arge	Non	Non	Oui	Non	Oui	Non
db.z1d.la rge	Non	Non	Oui	Non	Oui	Non

db.x2g — classes d'instance optimisées pour la mémoire alimentées par les processeurs Graviton2 AWS

Classe d'instanc e	Db	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.x2g.1 xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.x2g.1 xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

Classe d'instanc e	Db	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.x2g.8 large	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.x2g.4 large	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.x2g.2 large	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.x2g.x arge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.x2g.la rge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

db.x2idn : classes d'instances à mémoire optimisée et alimentées par des processeurs Intel Xeon Scalable de 3e génération

Classe d'instanc e	Db	MariaDB	Micros SQL Serve	MySQL	Oracle	PostgreSQL
db.x2idn. 32xlarge	Nc	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Enterp e Editi unique t	Versions de PostgreSQL 15, 14.6 et 13.9

Classe d'instance	Dl	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.x2idn.24xlarge	N(	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Enterp e Editi unique t	Versions de PostgreSQL 15, 14.6 et 13.9
db.x2idn.16xlarge	N(	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Enterp e Editi unique t	Versions de PostgreSQL 15, 14.6 et 13.9

db.x2iedn : classes d'instance optimisées pour la mémoire avec une NVMe base locale, alimentées par des processeurs Intel Xeon Scalable de SSDs 3e génération

Classe d'instance	Dl	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x2iedn.32xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Enterp e Editi unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.x2iedn.24xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014	MySQL 8.4 et 8.0	Enterp e Editi unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les

Classe d'instance	DI	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
			version 12.00 et supérieures			versions 13.7 et supérieures 13 et 13.4
db.x2iedn.16xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Enterprise Editi unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.x2iedn.8xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Enterprise Editi unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.x2iedn.4xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Éditior Enterprise et éditior Standa (2SE2	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

Classe d'instance	DI	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x2iedn.2xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Édition Enterprise et édition Standa (2SE2	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.x2iedn.xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Éditions Enterprise et Standard uniquement, SQL Server 2014 version 12.00 et supérieures	MySQL 8.4 et 8.0	Édition Enterprise et édition Standa (2SE2	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

db.x2iezn : classes d'instances à mémoire optimisée et alimentées par des processeurs Intel Xeon Scalable de 2e génération

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x2iezn.8xlarge	Non	Non	Non	Non	Enterprise Edition uniquement	Non
db.x2iezn.6xlarge	Non	Non	Non	Non	Enterprise Edition uniquement	Non

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x2iezn.4xlarge	Non	Non	Non	Non	Édition Enterprise et édition Standard (2SE2)	Non
db.x2iezn.2xlarge	Non	Non	Non	Non	Édition Enterprise et édition Standard (2SE2)	Non

db.x1e – Classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x1e.32xlarge	Non	Non	Oui	Non	Obsolète 1	Non
db.x1e.16xlarge	Non	Non	Oui	Non	Obsolète 1	Non
db.x1e.8xlarge	Non	Non	Oui	Non	Obsolète 1	Non
db.x1e.4xlarge	Non	Non	Oui	Non	Obsolète 1	Non
db.x1e.2xlarge	Non	Non	Oui	Nos	Obsolète 1	Non
db.x1e.xlarge	Non	Non	Oui	Non	Obsolète 1	Non

[1] Vous ne pouvez plus créer de RDS pour les instances de base de données Oracle à l'aide de la famille de classes d'instances X1. Si vous utilisez actuellement des classes X1, passez à une classe d'instance de nouvelle génération dès que possible. À compter du 22 janvier 2025, RDS lance les

mises à niveau automatisées dans le créneau de maintenance que vous avez défini. Au cours de la mise à niveau, RDS choisit le type d'instance X2iEDN équivalent et le met à niveau. Pour plus d'informations, consultez l'article Re:post Amazon RDS for Oracle mettra fin au support des instances de base de données X1 le 22 janvier 2025.

db.x1 – Classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.x1.32x large	Non	Non	Oui	Non	Obsolète 1	Non
db.x1.16x large	Non	Non	Oui	Non	Obsolète 1	Non

[1] Vous ne pouvez plus créer de RDS pour les instances de base de données Oracle à l'aide de la famille de classes d'instances X1. Si vous utilisez actuellement des classes X1, passez à une classe d'instance de nouvelle génération dès que possible. À compter du 22 janvier 2025, RDS lance les mises à niveau automatisées dans le créneau de maintenance que vous avez défini. Au cours de la mise à niveau, RDS choisit le type d'instance X2iEDN équivalent et le met à niveau. Pour plus d'informations, consultez l'article Re:post Amazon RDS for Oracle mettra fin au support des instances de base de données X1 le 22 janvier 2025.

db.r8g — classes d'instance optimisées pour la mémoire alimentées par les processeurs Graviton4 AWS

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.r8g.48 x large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
						supérieure, 13.11 et supérieure
db.r8g.24xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r8g.16xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r8g.12xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.r8g.8xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r8g.4xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r8g.2xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.r8g.xlarge	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r8g.large	Non	MariaDB 11.4.3 et supérieur, 10.11.7 et supérieur, 10.6.13 et supérieur, 10.5.20 et supérieur, et 10.4.29 et supérieur	Non	MySQL 8.0.32 et versions ultérieures	Non	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

db.r7i : classes d'instances optimisées pour la mémoire alimentées par des processeurs Intel Xeon Scalable de 4e génération

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.r7i.48x large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.r7i 24 x large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i.16 x large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i 12 x large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i.8x large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Oracle	PostgreSQL
db.r7i.4xlarge	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t, toutes les éditions	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i.2xlarge	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t, toutes les éditions	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i.xlarge	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t, toutes les éditions	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure
db.r7i.large	DE 11	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.0.32 et versions ultérieures	BYOL uniquer t, toutes les éditions	PostgreSQL version 17.1 et supérieure, 16.1 et supérieure, 15.4 et supérieure, 14.9 et supérieure, 13.11 et supérieure

db.r7g — classes d'instance optimisées pour la mémoire alimentées par les processeurs Graviton3 AWS

Classe d'instanc e	DI	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r7g.1 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.1 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.8 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.4 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.2 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5

Classe d'instanc e	Dl	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.xl arge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13
db.r7g.la rge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, et les versions 13.4 et supérieures 13

db.r6g — classes d'instance optimisées pour la mémoire alimentées par les processeurs Graviton2 AWS

Classe d'instanc e	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6g.16 xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r6g.12 xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						de PostgreSQL, et 12 versions ultérieures
db.r6g.8x large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r6g.4x large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r6g.2x large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r6g.xl arge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r6g.la rge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

db.r6gd — classes d'instance optimisées pour la mémoire alimentées par les processeurs Graviton2 AWS

Classe d'instance	DI	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6g.16 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.12 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.8x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.4x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.2x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14,

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						les versions 13.7 et supérieures 13 et 13.4
db.r6gd.x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6gd.large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

db.r6id : classes d'instances à mémoire optimisée, alimentées par des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	Db2	MariaDB	Microso SQL Server	MySQL	Oracle	PostgreSQL
db.r6id.32xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.24xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et	Toutes les versions 17, 16 et 15 de PostgreSQ

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
					BYOL uniquement	L, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.16xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.12xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.8xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.4xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL uniquement	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r6id.2xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

db.r6idn – classes d'instances à mémoire optimisée fonctionnant avec des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Ora	PostgreSQL
db.r6idn.32xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						les versions 13.7 et supérieures 13
db.r6idn.24xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6idn.16xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6idn.12xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6idn.8xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6idn.4xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6idn.2xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6idn.xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

db.r6in – classes d'instances à mémoire optimisée fonctionnant avec des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6in.32xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.24xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.16xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.12xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieures 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6in.8xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieur es 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.4xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieur es 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.2xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieur es 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

Classe d'instanc e	Db2	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6in.x large	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieur es 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11
db.r6in.l arge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYO unic t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.3 et supérieur es 14, les versions 13.7 et supérieures 13, les versions 12.11 et supérieures 12, et les versions 11.16 et supérieures 11

db.r6i : classes d'instance optimisées pour la mémoire, préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQ L
db.r6i.8x large.tpc 2.mem4x	Non	Non	Non	Non	EE uniquemen t	Non
db.r6i.8x large.tpc 2.mem3x	Non	Non	Non	Non	EE uniquemen t	Non

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r6i.6xlarge.tpc2.mem4x	Non	Non	Non	Non	EE uniquement	Non
db.r6i.4xlarge.tpc2.mem4x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.4xlarge.tpc2.mem3x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.4xlarge.tpc2.mem2x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.2xlarge.tpc2.mem8x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.2xlarge.tpc2.mem4x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.2xlarge.tpc1.mem2x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.xlarge.tpc2.mem4x	Non	Non	Non	Non	EE et SE2 BYOL	Non
db.r6i.xlarge.tpc2.mem2x	Non	Non	Non	Non	EE et SE2 BYOL	Non

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r6i.large.tpc1.mem2x	Non	Non	Non	Non	EE et SE2 BYOL	Non

db.r6i : classes d'instances à mémoire optimisée

Classe d'instance	Db	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r6i.32xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.24xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.16xlarge	Oui	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions

Classe d'instance	DB	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
						11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.1 xlarge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.8 large	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.4 large	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieures 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10

Classe d'instan e	DI	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.r6i.2: large	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieur es 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.xl arge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieur es 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10
db.r6i.la rge	O	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15 et 14 de PostgreSQL, les versions 13.4 et supérieur es 13, les versions 12.8 et supérieures 12, les versions 11.13 et supérieures 11, et les versions 10.21 et supérieures 10

db.r5d : classes d'instance à mémoire optimisée

Classe d'instance	DI	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
db.r5d.2 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.1 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.1 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.8 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.4 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.2 large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur

Classe d'instance	Db	MariaDB	Micros SQL Serve	MySQL	Ora	PostgreSQL
						es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.x arge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4
db.r5d.la rge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieur es 14, les versions 13.7 et supérieures 13 et 13.4

db.r5b : classes d'instances à mémoire optimisée préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r5b.8x large.tpc 2.mem3x	Non	Non	Non	Non	Oui	Non
db.r5b.6x large.tpc 2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5b.4x large.tpc 2.mem4x	Non	Non	Non	Non	Oui	Non

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r5b.4xlarge.tpc2.mem3x	Non	Non	Non	Non	Oui	Non
db.r5b.4xlarge.tpc2.mem2x	Non	Non	Non	Non	Oui	Non
db.r5b.2xlarge.tpc2.mem8x	Non	Non	Non	Non	Oui	Non
db.r5b.2xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5b.2xlarge.tpc1.mem2x	Non	Non	Non	Non	Oui	Non
db.r5b.xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5b.xlarge.tpc2.mem2x	Non	Non	Non	Non	Oui	Non
db.r5b.large.tpc1.mem2x	Non	Non	Non	Non	Oui	Non

db.r5b – Classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.r5b.24xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.16xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.12xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.8xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	>Ou	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.4xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.2xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.r5b.xlarge	Nor	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

Classe d'instance	Db2	MariaDB	Micros SQL Server	MySQL	Orac	PostgreSQL
db.r5b.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Oui	MySQL 8.4 et 8.0	Oui	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

db.r5 : classes d'instances optimisées en mémoire préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r5.12xlarge.tpc2.mem2x	Non	Non	Non	Non	Oui	Non
db.r5.8xlarge.tpc2.mem3x	Non	Non	Non	Non	Oui	Non
db.r5.6xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5.4xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5.4xlarge.tpc2.mem3x	Non	Non	Non	Non	Oui	Non
db.r5.4xlarge.tpc2.mem2x	Non	Non	Non	Non	Oui	Non
db.r5.2xlarge.tpc2.mem8x	Non	Non	Non	Non	Oui	Non

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r5.2xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5.2xlarge.tpc1.mem2x	Non	Non	Non	Non	Oui	Non
db.r5.xlarge.tpc2.mem4x	Non	Non	Non	Non	Oui	Non
db.r5.xlarge.tpc2.mem2x	Non	Non	Non	Non	Oui	Non
db.r5.large.tpc1.mem2x	Non	Non	Non	Non	Oui	Non

db.r5 : classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaD	Microsoft SQL Server	MySC	Oracle	PostgreSQL
db.r5.24xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.16xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.12xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions

Classe d'instance	Db2	MariaD	Microsoft SQL Server	MySC	Oracle	PostgreSQL
						10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.8xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.4xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.2xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.xlarge	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9
db.r5.large	Non	Oui	Oui	Oui	Oui	Toutes les versions de PostgreSQL 17, 16, 15, 14, 13, 12 et 11 ; les versions 10.17 et supérieures 10 ; et les versions 9.6.22 et supérieures 9

db.r4 – Classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r4.16xlarge	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.r4.8xlarge	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.r4.4xlarge	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.r4.2xlarge	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.r4.xlarge	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.r4.large	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète

db.r3 – Classes d'instance à mémoire optimisée

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQ	Oracle	PostgreSQL
db.r3.8xlarge**	Non	Toutes les versions 10.6, 10.5, 10.4 et 10.3 de MariaDB	Obsolète	Oui	Obsolète	Obsolète
db.r3.4xlarge	Non	Toutes les versions 10.6, 10.5, 10.4 et 10.3 de MariaDB	Obsolète	Oui	Obsolète	Obsolète

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQ	Oracle	PostgreSQL
db.r3.2xlarge	Non	Toutes les versions 10.6, 10.5, 10.4 et 10.3 de MariaDB	Obsolète	Oui	Obsolète	Obsolète
db.r3.xlarge	Non	Toutes les versions 10.6, 10.5, 10.4 et 10.3 de MariaDB	Obsolète	Oui	Obsolète	Obsolète
db.r3.large	Non	Toutes les versions 10.6, 10.5, 10.4 et 10.3 de MariaDB	Obsolète	Oui	Obsolète	Obsolète

Moteurs de base de données pris en charge pour les classes d'instances optimisées pour le calcul

Les tableaux suivants indiquent les bases de données et les versions de base de données prises en charge pour les classes d'instances optimisées pour le calcul.

db.c6gd : classes d'instance optimisées pour le calcul (pour les déploiements de clusters de bases de données multi-AZ uniquement)

Classe d'instance	Db2	Maria	Microsoft SQL Server	MySQL	Orac	PostgreSQL
db.c6gd.16 x large	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.12xlarge	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et

Classe d'instance	Db2	Maria	Microsoft SQL Server	MySQL	Orac	PostgreSQL
						supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.8xlarge	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.4xlarge	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.2xlarge	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.xlarge	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.large	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions
db.c6gd.medium	Non	Non	Non	MySQL 8.4 et 8.0	Non	Toutes les versions de PostgreSQL 17, 16 et 15 ; versions 14.5 et supérieures 14 ; versions 13.4 et 13.7 et supérieures 13 versions

Moteurs de base de données pris en charge pour les classes d'instance aux performances évolutives

Les tableaux suivants indiquent les bases de données et les versions de base de données prises en charge pour les classes d'instances à performances évolutives.

db.t4g — classes d'instance aux performances éclatantes alimentées par les processeurs Graviton2 AWS

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Orac	PostgreSQL
db.t4g.2xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.t4g.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.t4g.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.t4g.medium	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures
db.t4g.small	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Orac	PostgreSQL
db.t4g.micro	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Non	Toutes les versions 17, 16, 15, 14 et 13 de PostgreSQL, et 12 versions ultérieures

db.t3 : classes d'instance de performance à capacité extensible

Classe d'instance	Db2	Maria	Microsoft SQL Server	MySC	Oracle	PostgreSQL
db.t3.2xlarge	Oui	Oui	Oui	Oui	Oui	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures
db.t3.xlarge	Oui	Oui	Oui	Oui	Oui	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures
db.t3.large	Oui	Oui	Oui	Oui	Oui	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures
db.t3.medium	Oui	Oui	Oui	Oui	Oui	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures
db.t3.small	Oui	Oui	Oui	Oui	Oui	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures

Classe d'instance	Db2	Maria	Microsoft SQL Server	MySC	Oracle	PostgreSQL
db.t3.micro	Non	Oui	Oui	Oui	Uniquement sur Oracle Database 12c version 1 (12.1.0.2), qui est rendue obsolète	Toutes les versions 17, 16, 15, 14, 13, 12, 11 et 10 de PostgreSQL ; 9 versions 9.6.22 et supérieures

db.t2 : classes d'instance de performance à capacité extensible

Classe d'instance	Db2	MariaDB	Microsof SQL Server	MySQL	Oracle	PostgreSQL
db.t2.2xlarge	Non	Obsolète	Non	Obsolète	Obsolèt	Obsolète
db.t2.xlarge	Non	Obsolète	Non	Obsolète	Obsolèt	Obsolète
db.t2.large	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.t2.medium	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.t2.small	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète
db.t2.micro	Non	Obsolète	Obsolète	Obsolète	Obsolèt	Obsolète

Moteurs de base de données compatibles pour les classes d'instance Optimized Reads

Les tableaux suivants indiquent les bases de données et les versions de base de données prises en charge pour les classes d'instance Optimized Reads.

db.r6gd — classes d'instance optimisées pour la mémoire qui prennent en charge les lectures optimisées et sont alimentées par des processeurs Graviton2 AWS

Classe d'instanc e	DI	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
db.r6g.16 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.12 xlarge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.8x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6g.4x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQ L, les versions 14.5 et supérieures 14,

Classe d'instance	Db	MariaDB	Micros SQL Server	MySQL	Ora	PostgreSQL
						les versions 13.7 et supérieures 13 et 13.4
db.r6g.2x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6gd.x large	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4
db.r6gd.l arge	N	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	Nor	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, les versions 13.7 et supérieures 13 et 13.4

db.r6id : classes d'instance optimisées pour la mémoire qui prennent en charge les lectures optimisées et sont alimentées par des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	Db2	MariaDB	Microso SQL Server	MySQL	Oracle	PostgreSQL
db.r6id.32xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et	Toutes les versions 17, 16 et 15 de PostgreSQ

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
					BYOL unique	L, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.24xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL unique	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.16xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL unique	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.12xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL unique	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.8xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	EE et BYOL unique	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Classe d'instance	Db2	MariaDB	Microsoft SQL Server	MySQL	Oracle	PostgreSQL
db.r6id.4xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.2xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.xlarge	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13
db.r6id.large	Non	MariaDB 11.4, 10.11, 10.6, 10.5 et 10.4	Non	MySQL 8.4 et 8.0	BYOL unique t	Toutes les versions 17, 16 et 15 de PostgreSQL, les versions 14.5 et supérieures 14, et les versions 13.7 et supérieures 13

Déterminer le support des classes d'instance de base de données dans Régions AWS

Pour déterminer les classes d'instance de base de données prises en charge par chaque moteur de base de données dans une Région AWS spécifique, vous pouvez adopter l'une des différentes approches. Vous pouvez utiliser la AWS Management Console page de tarification Amazon RDS ou la commande describe-orderable-db-instance-options pour le AWS Command Line Interface ()AWS CLI.

> **Note**
>
> Lorsque vous effectuez des opérations avec le AWS Management Console, il affiche automatiquement les classes d'instance de base de données prises en charge pour un moteur de base de données, une version de moteur de base de données et Région AWS. Parmi les opérations que vous pouvez effectuer, citons la création et la modification d'une instance de base de données.

Table des matières

- Utilisation de la page de tarification d'Amazon RDS pour déterminer la prise en charge des classes d'instances de base de données dans Régions AWS
- Utilisation du AWS CLI pour déterminer la prise en charge des classes d'instances de base de données dans Régions AWS
  - Répertorier les classes d'instance de base de données prises en charge par une version de moteur de base de données spécifique dans une Région AWS
  - Répertorier les versions de moteur de base de données qui prennent en charge une classe d'instance de base de données spécifique dans une Région AWS
  - Liste Régions AWS prenant en charge un moteur de base de données et une classe d'instance spécifiques

Utilisation de la page de tarification d'Amazon RDS pour déterminer la prise en charge des classes d'instances de base de données dans Régions AWS

Vous pouvez utiliser la page Tarification Amazon RDS pour déterminer les classes d'instance de base de données prises en charge par chaque moteur de base de données dans une Région AWS spécifique.

Pour utiliser la page de tarification pour déterminer les classes d'instance de base de données prises en charge par chaque moteur dans une région

1. Accédez à Tarification d'Amazon RDS.
2. Dans la section Calculateur de tarification AWS pour Amazon RDS, choisissez Créer votre estimation personnalisée maintenant.
3. Dans Choisir une région, choisissez une Région AWS.
4. Dans Trouver un service, saisissez **`Amazon RDS`**.
5. Choisissez Configurer pour votre option de configuration et votre moteur de base de données.
6. Dans la section où figurent les instances compatibles, examinez les classes d'instances de base de données prises en charge.
7. (Facultatif) Choisissez d'autres options dans le calculateur, puis sélectionnez Enregistrer et afficher le récapitulatif ou Enregistrer et ajouter un service.

Utilisation du AWS CLI pour déterminer la prise en charge des classes d'instances de base de données dans Régions AWS

Vous pouvez utiliser le AWS CLI pour déterminer quelles classes d'instances de base de données sont prises en charge pour des moteurs de base de données et des versions de moteurs de base de données spécifiques dans un Région AWS. Le tableau suivant présente les valeurs du moteur de base de données valides.

Noms de moteur	Valeurs du moteur dans les commandes de la CLI	Plus d'informations sur les versions
Db2	`db2-ae` `db2-se`	Versions de DB2 sur Amazon RDS
MariaDB	`mariadb`	Versions de MariaDB sur Amazon RDS

Microsoft SQL Server	`sqlserver-ee`
	`sqlserver-se`
	`sqlserver-ex`
		Versions de Microsoft SQL Server sur Amazon RDS

Noms de moteur	Valeurs du moteur dans les commandes de la CLI	Plus d'informations sur les versions
	`sqlserver-web`	
MySQL	`mysql`	Versions de MySQL sur Amazon RDS
Oracle	`oracle-ee` `oracle-se2`	Notes de mise à jour d'Amazon RDS for Oracle
PostgreSQL	`postgres`	Versions de base de données PostgreSQL disponibles

Pour plus d'informations sur Région AWS les noms, consultezAWS Régions.

Les exemples suivants montrent comment déterminer la prise en charge des classes d'instance de base de données à Région AWS l'aide de la AWS CLI commande describe-orderable-db-instance-options.

> **Note**
>
> Pour limiter la sortie, les exemples suivants montrent les résultats uniquement pour le type de stockage SSD à usage général (`gp2`). Si nécessaire, vous pouvez modifier le type de stockage en SSD à usage général (`gp3`), Provisioned IOPS (`io1`), Provisioned IOPS Block Express (`io2`) ou magnétique (`standard`) dans les commandes.

Rubriques

- Répertorier les classes d'instance de base de données prises en charge par une version de moteur de base de données spécifique dans une Région AWS
- Répertorier les versions de moteur de base de données qui prennent en charge une classe d'instance de base de données spécifique dans une Région AWS
- Liste Régions AWS prenant en charge un moteur de base de données et une classe d'instance spécifiques

Répertorier les classes d'instance de base de données prises en charge par une version de moteur de base de données spécifique dans une Région AWS

Pour répertorier les classes d'instance de base de données prises en charge par une version spécifique du moteur de base de données dans un Région AWS, exécutez la commande suivante.

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options --engine engine --engine-version version
 \
    --query "*[].{DBInstanceClass:DBInstanceClass,StorageType:StorageType}|[?
StorageType=='gp2']|[].{DBInstanceClass:DBInstanceClass}" \
    --output text \
    --region region
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options --engine engine --engine-version version
 ^
    --query "*[].{DBInstanceClass:DBInstanceClass,StorageType:StorageType}|[?
StorageType=='gp2']|[].{DBInstanceClass:DBInstanceClass}" ^
    --output text ^
    --region region
```

Par exemple, la commande suivante répertorie les classes d'instance de base de données prises en charge pour la version 13.6 du moteur de base de données RDS for PostgreSQL dans la région USA Est (Virginie du Nord).

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options --engine postgres --engine-version 15.4
 \
    --query "*[].{DBInstanceClass:DBInstanceClass,StorageType:StorageType}|[?
StorageType=='gp2']|[].{DBInstanceClass:DBInstanceClass}" \
    --output text \
    --region us-east-1
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options --engine postgres --engine-version 15.4
 ^
```

```
    --query "*[].{DBInstanceClass:DBInstanceClass,StorageType:StorageType}|[?
StorageType=='gp2']|[].{DBInstanceClass:DBInstanceClass}" ^
    --output text ^
    --region us-east-1
```

Répertorier les versions de moteur de base de données qui prennent en charge une classe d'instance de base de données spécifique dans une Région AWS

Pour répertorier les versions de moteur de base de données qui prennent en charge une classe d'instance de base de données spécifique dans une Région AWS, exécutez la commande suivante.

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options --engine engine --db-instance-
class DB_instance_class \
    --query "*[].{EngineVersion:EngineVersion,StorageType:StorageType}|[?
StorageType=='gp2']|[].{EngineVersion:EngineVersion}" \
    --output text \
    --region region
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options --engine engine --db-instance-
class DB_instance_class ^
    --query "*[].{EngineVersion:EngineVersion,StorageType:StorageType}|[?
StorageType=='gp2']|[].{EngineVersion:EngineVersion}" ^
    --output text ^
    --region region
```

Par exemple, la commande suivante répertorie les versions du moteur de base de données du moteur de base de données RDS for PostgreSQL qui prennent en charge la classe d'instance de base de données db.r5.large dans US East (N. Virginia).

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options --engine postgres --db-instance-class
 db.m7g.large \
    --query "*[].{EngineVersion:EngineVersion,StorageType:StorageType}|[?
StorageType=='gp2']|[].{EngineVersion:EngineVersion}" \
    --output text \
    --region us-east-1
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options --engine postgres --db-instance-class
 db.m7g.large ^
    --query "*[].{EngineVersion:EngineVersion,StorageType:StorageType}|[?
StorageType=='gp2']|[].{EngineVersion:EngineVersion}" ^
    --output text ^
    --region us-east-1
```

Liste Régions AWS prenant en charge un moteur de base de données et une classe d'instance spécifiques

Le script bash suivant répertorie tous ceux Régions AWS qui prennent en charge la combinaison spécifiée de moteur de base de données et de classe d'instance.

```
#!/usr/bin/env bash
# Usage: check_region_support.sh <db-engine> <db-instance-class>

if [ $# -ne 2 ]; then
  echo "Usage: $0 <db-engine> <db-instance-class>"
  exit 1
fi
ENGINE="$1"
INSTANCE_CLASS="$2"
REGIONS=$(aws ec2 describe-regions --query "Regions[].RegionName" --output text)
for region in $REGIONS; do
  supported_count=$(aws rds describe-orderable-db-instance-options \
    --region "$region" \
    --engine "$ENGINE" \
    --db-instance-class "$INSTANCE_CLASS" \
    --query 'length(OrderableDBInstanceOptions)' \
    --output text 2>/dev/null || echo "0")
  if [ "$supported_count" -gt 0 ]; then
    echo "$region supports $INSTANCE_CLASS for $ENGINE."
  else
    echo "$region doesn't support $INSTANCE_CLASS for $ENGINE."
  fi
done
```

L'exemple de sortie suivant vérifie le support régional pour à l'aide de la classe d'instance db.r8g.large.

```
./check_region_support.sh mysql db.r8g.large
ap-south-1 doesn't support db.r8g.large for mysql.
eu-north-1 doesn't support db.r8g.large for mysql.
eu-west-3 doesn't support db.r8g.large for mysql.
eu-west-2 doesn't support db.r8g.large for mysql.
eu-west-1 doesn't support db.r8g.large for mysql.
ap-northeast-3 doesn't support db.r8g.large for mysql.
ap-northeast-2 doesn't support db.r8g.large for mysql.
ap-northeast-1 doesn't support db.r8g.large for mysql.
ca-central-1 doesn't support db.r8g.large for mysql.
sa-east-1 doesn't support db.r8g.large for mysql.
ap-southeast-1 doesn't support db.r8g.large for mysql.
ap-southeast-2 doesn't support db.r8g.large for mysql.
eu-central-1 supports db.r8g.large for mysql.
us-east-1 supports db.r8g.large for mysql.
us-east-2 supports db.r8g.large for mysql.
us-west-1 doesn't support db.r8g.large for mysql.
us-west-2 supports db.r8g.large for mysql.
```

Modification d'une classe d'instance de base de données

Vous pouvez modifier la capacité de calcul et de mémoire d'une instance de base de données en modifiant sa classe d'instance de base de données. Pour modifier une classe d'instance de base de données, modifiez l'instance de base de données en suivant les instructions fournies dans Modification d'une RDS instance de base de données Amazon.

Configuration du processeur pour une classe d'instances de base de données dans RDS for Oracle

Les classes d'instances de base de données Amazon RDS prennent en charge la technologie hyper-threading d'Intel, qui permet l'exécution simultanée de plusieurs threads sur un seul cœur d'UC Intel Xeon. Chaque thread est représenté comme UC virtuelle (vCPU) sur l'instance de base de données. Par défaut, une instance de base de données possède un certain nombre de cœurs d'UC, qui varie en fonction de la classe d'instance de base de données. Par exemple, une classe d'instance de base de données db.m4.xlarge possède deux cœurs de processeur et deux threads par cœur par défaut, soit quatre v au total. CPUs

> **Note**
>
> Chaque vCPU est un hyperthread d'un cœur d'UC Intel Xeon.

Rubriques

- Présentation de la configuration de processeur pour RDS for Oracle
- Classes d'instances DB prenant en charge la configuration du processeur
- Définition des cœurs d'UC et des threads par cœur d'UC pour une classe d'instance de base de données

Présentation de la configuration de processeur pour RDS for Oracle

Lorsque vous utilisez RDS pour Oracle, vous pouvez généralement trouver une classe d'instance de base de données qui possède une combinaison de mémoire et de nombre de v adaptée CPUs à vos charges de travail. Toutefois, vous pouvez également spécifier les fonctionnalités de processeur suivantes afin d'optimiser votre instance de base de données RDS pour Oracle en fonction de charges de travail ou de besoins commerciaux spécifiques :

- Nombre de cœurs d'UC – Vous pouvez personnaliser le nombre de cœurs d'UC pour l'instance de base de données. Vous pourriez agir ainsi pour optimiser potentiellement les coûts de licence de vos logiciels avec une instance de base de données ayant une quantité suffisante de RAM pour les charges de travail exigeantes en mémoire, mais moins de cœurs d'UC.
- Threads par cœur – Vous pouvez désactiver la technologie hyper-threading d'Intel en spécifiant une seul thread par cœur d'UC. Vous pourriez agir ainsi pour certaines charges de travail, telles que les charges de travail de calcul haute performance (HPC).

Vous pouvez contrôler le nombre de cœurs d'UC et de threads pour chaque cœur séparément. Vous pouvez définir l'un ou les deux dans une demande. Une fois qu'un paramètre est associé à une instance de base de données, il persiste jusqu'à ce que vous le changiez.

Les paramètres du processeur pour une instance de base de données sont associés aux instantanés de l'instance de base de données. Lorsqu'un instantané est restauré, son instance de base de données restaurée utilise les paramètres des fonctionnalités du processeur qui ont servi lors de la prise de l'instantané.

Si vous modifiez la classe d'une instance de base de données avec des paramètres de processeur autres que ceux définis par défaut, spécifiez les paramètres de processeur par défaut ou spécifiez explicitement les paramètres de processeur par défaut lors de la modification de l'instance de base de données. Cela vous permet d'avoir connaissance des coûts de licence tiers susceptibles d'être encourus lorsque vous modifiez l'instance de base de données.

Il n'y a pas de frais supplémentaires ou réduits pour la spécification des fonctionnalités du processeur sur une instance de base de données RDS for Oracle. Le même montant vous est facturé pour les instances de base de données qui sont lancées avec les configurations de l'UC par défaut.

Classes d'instances DB prenant en charge la configuration du processeur

Vous pouvez configurer le nombre de cœurs de processeur et de threads par cœur uniquement lorsque les conditions suivantes sont remplies :

- Vous configurez une instance de base de données RDS for Oracle. Pour obtenir des informations sur les classes d'instances de base de données prises en charge par les différentes éditions d'Oracle Database, consultez RDS pour les classes d'instance de base de données Oracle.
- Votre instance de base de données utilise l'option de licence Apportez votre propre licence (BYOL) de RDS for Oracle. Pour plus d'informations sur les options de licence Oracle, consultez RDSpour les options de licence Oracle.
- Votre instance de base de données n'appartient pas à l'une des classes d'instances db.r5 ou db.r5b qui possèdent des configurations de processeur prédéfinies. Ces classes d'instance ont des noms au format db.r5. *instance_size*.tpc *threads_per_core* .mem ou *ratio* db.r5b. *instance_size*.tpc *threads_per_core* .mem. *ratio* Par exemple, db.r5b.xlarge.tpc2.mem4x est préconfiguré avec deux threads par cœur (tpc2) et quatre fois plus de mémoire que la classe d'instances db.r5b.xlarge standard. Vous ne pouvez pas configurer les fonctions de processeur de ces classes d'instances optimisées. Pour de plus amples informations, veuillez consulter RDS pris en charge pour les classes d'instance de base de données Oracle.

Vous pouvez utiliser la AWS CLI commande suivante pour afficher le v par défautCPUs, les cœurs, les threads par cœur et le nombre valide de cœurs pour une classe d'instance. Remplacez *r7i.48xlarge* dans l'exemple de commande par le nom de votre classe d'instance.

```
aws ec2 describe-instance-types \
    --instance-types r7i.48xlarge \
    --query '{
        DefaultVCPUs: InstanceTypes[0].VCpuInfo.DefaultVCpus,
```

```
        DefaultCores: InstanceTypes[0].VCpuInfo.DefaultCores,
        DefaultThreadsPerCore: InstanceTypes[0].VCpuInfo.DefaultThreadsPerCore,
        ValidCores: InstanceTypes[0].VCpuInfo.ValidCores
    }' \
    --output json
```

Le tableau suivant présente les classes d'instances de base de données qui prennent en charge la définition d'un certain nombre de cœurs d'UC et de threads d'UC par cœur. Il contient également la valeur par défaut et les valeurs valides pour le nombre de cœurs d'UC et de threads d'UC par cœur pour chaque classe d'instance de base de données.

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.m6i : classes d'instances à mémoire optimisée					
db.m6i.large	2	1	2	1	1, 2
db.m6i.xlarge	4	2	2	2	1, 2
db.m6i.2xlarge	8	4	2	2, 4	1, 2
db.m6i.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.m6i.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.m6i.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.m6i.12 xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.m6i.16 xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.m6i.24 xlarge	96	48	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28,	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
				30, 32, 34, 36, 38, 40, 42, 44, 46, 48	
db.m6i.32xlarge	128	64	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64	1, 2
db.m5 : classes d'instance à usage général					
db.m5.large	2	1	2	1	1, 2
db.m5.xlarge	4	2	2	2	1, 2
db.m5.2xlarge	8	4	2	2, 4	1, 2
db.m5.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.m5.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.m5.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.m5.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.m5.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.m5d : classes d'instance à usage général					

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.m5d.large	2	1	2	1	1, 2
db.m5d.xlarge	4	2	2	2	1, 2
db.m5d.2xlarge	8	4	2	2, 4	1, 2
db.m5d.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.m5d.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.m5d.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.m5d.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.m5d.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.m4 : classes d'instance à usage général					
db.m4.10xlarge	40	20	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20	1, 2
db.m4.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r7i — classes d'instance optimisées pour la mémoire					
db.r7i.large	2	1	2	1	1, 2
db.r7i.xlarge	4	2	2	1, 2	1, 2
db.r7i.2xlarge	8	4	2	1, 2, 3, 4	1, 2
db.r7i.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
db.r7i.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
db.r7i 12 x large	48	24	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24	1, 2
db.r7i.16 x large	64	32	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32	1, 2
db.r7i 24 x large	96	48	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r7i.48 x large	192	96	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64, 66, 68, 70, 72, 74, 76, 78, 80, 82, 84, 86, 88, 90, 92, 94, 96	1, 2
db.r6i : classes d'instances à mémoire optimisée					
db.r6i.large	2	1	2	1	1, 2
db.r6i.xlarge	4	2	2	1, 2	1, 2
db.r6i.2xlarge	8	4	2	2, 4	1, 2
db.r6i.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.r6i.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.r6i.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.r6i.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.r6i.24xlarge	96	48	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r6i.32xlarge	128	64	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64	1, 2
db.r5 : classes d'instance à mémoire optimisée					
db.r5.large	2	1	2	1	1, 2
db.r5.xlarge	4	2	2	2	1, 2
db.r5.2xlarge	8	4	2	2, 4	1, 2
db.r5.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.r5.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.r5.12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.r5.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.r5.24xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.r5 : classes d'instance à mémoire optimisée					
db.r5b.large	2	1	2	1	1, 2
db.r5b.xlarge	4	2	2	2	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r5b.2xlarge	8	4	2	2, 4	1, 2
db.r5b.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.r5b.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.r5b.12 xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.r5b.16 xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.r5b.24 xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.r5d : classes d'instance à mémoire optimisée					
db.r5d.large	2	1	2	1	1, 2
db.r5d.xlarge	4	2	2	2	1, 2
db.r5d.2xlarge	8	4	2	2, 4	1, 2
db.r5d.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.r5d.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.r5d.12 xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r5d.16 xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.r5d.24 xlarge	96	48	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.r4 – Classes d'instance à mémoire optimisée					
db.r4.large	2	1	2	1	1, 2
db.r4.xlarge	4	2	2	1, 2	1, 2
db.r4.2xlarge	8	4	2	1, 2, 3, 4	1, 2
db.r4.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
db.r4.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
db.r4.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.r3 – Classes d'instance à mémoire optimisée					
db.r3.large	2	1	2	1	1, 2
db.r3.xlarge	4	2	2	1, 2	1, 2
db.r3.2xlarge	8	4	2	1, 2, 3, 4	1, 2
db.r3.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.r3.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.x2idn : classes d'instances à mémoire optimisée					
db.x2idn.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.x2idn.24xlarge	96	48	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.x2idn.32xlarge	128	64	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64	1, 2
db.x2iedn : classes d'instances à mémoire optimisée					
db.x2iedn.xlarge	4	2	2	1, 2	1, 2
db.x2iedn.2xlarge	8	4	2	2, 4	1, 2
db.x2iedn.4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.x2iedn.8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.x2iedn .16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.x2iedn .24xlarge	96	48	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48	1, 2
db.x2iedn .32xlarge	128	64	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42, 44, 46, 48, 50, 52, 54, 56, 58, 60, 62, 64	1, 2
db.x2iezn : classes d'instance à mémoire optimisée					
db.x2iezn .2xlarge	8	4	2	2, 4	1, 2
db.x2iezn .4xlarge	16	8	2	2, 4, 6, 8	1, 2
db.x2iezn .6xlarge	24	12	2	2, 4, 6, 8, 10, 12	1, 2
db.x2iezn .8xlarge	32	16	2	2, 4, 6, 8, 10, 12, 14, 16	1, 2
db.x2iezn .12xlarge	48	24	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2
db.x1 – Classes d'instance à mémoire optimisée					

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.x1.16xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.x1.32xlarge	128	64	2	4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64	1, 2
db.x1e – Classes d'instance à mémoire optimisée					
db.x1e.xlarge	4	2	2	1, 2	1, 2
db.x1e.2xlarge	8	4	2	1, 2, 3, 4	1, 2
db.x1e.4xlarge	16	8	2	1, 2, 3, 4, 5, 6, 7, 8	1, 2
db.x1e.8xlarge	32	16	2	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	1, 2
db.x1e.16 xlarge	64	32	2	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32	1, 2
db.x1e.32 xlarge	128	64	2	4, 8, 12, 16, 20, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64	1, 2
db.z1d : classes d'instance à mémoire optimisée					
db.z1d.large	2	1	2	1	1, 2
db.z1d.xlarge	4	2	2	2	1, 2
db.z1d.2xlarge	8	4	2	2, 4	1, 2

Classe d'instances de base de données	Par défaut v CPUs	Cœurs d'UC par défaut	Threads par défaut par cœur	Nombre valide de cœurs d'UC	Nombre valide de threads par cœur
db.z1d.3xlarge	12	6	2	2, 4, 6	1, 2
db.z1d.6xlarge	24	12	2	2, 4, 6, 8, 10, 12	1, 2
db.z1d.12 xlarge	48	24	2	4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	1, 2

> **Note**
>
> Vous pouvez l'utiliser AWS CloudTrail pour surveiller et auditer les modifications apportées à la configuration des processus des instances de base de données Amazon RDS for Oracle. Pour plus d'informations sur l'utilisation de CloudTrail, consultez Surveillance (Amazon RDS API appelle) AWS CloudTrail.

Définition des cœurs d'UC et des threads par cœur d'UC pour une classe d'instance de base de données

Vous pouvez configurer le nombre de cœurs de l'UC et de threads par cœur pour la classe de l'instance de base de données lorsque vous exécutez les opérations suivantes :

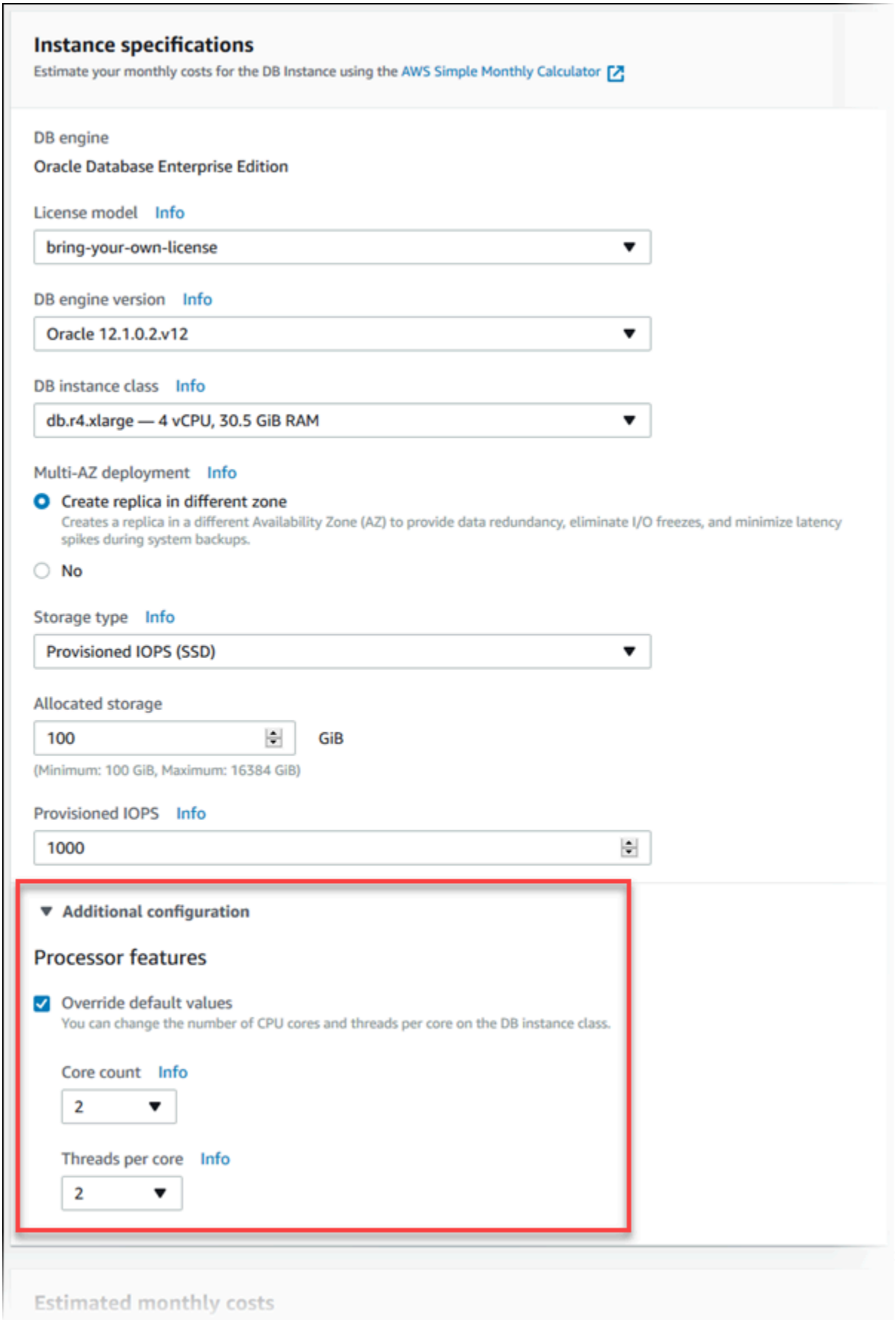
- Création d'une instance de base de données Amazon RDS
- Modification d'une RDS instance de base de données Amazon
- Restauration vers une instance de base de données
- Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

> **Note**
>
> Lorsque vous modifiez une instance de base de données pour configurer le nombre de cœurs de l'UC ou de threads par cœur, il se produit une courte interruption de l'instance de base de données.

Vous pouvez définir les cœurs de processeur et les threads par cœur de processeur pour une classe d'instance de base de données à l'aide de l' AWS Management Console API AWS CLI, de ou de l'API RDS.

console

Lorsque vous créez, modifiez ou restaurez une instance de base de données, vous définissez la classe d'instance de base de données dans l AWS Management Console. La section Spécifications de l'instance comporte les options du processeur. L'image suivante montre les options relatives aux fonctionnalités du processeur.

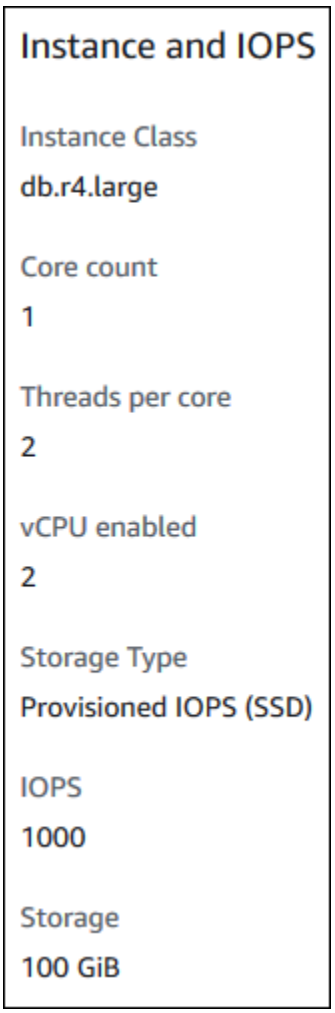

Instance specifications
Estimate your monthly costs for the DB Instance using the AWS Simple Monthly Calculator
DB engine
Oracle Database Enterprise Edition
License model Info
bring-your-own-license
DB engine version Info
Oracle 12.1.0.2.v12
DB instance class Info
db.r4.xlarge — 4 vCPU, 30.5 GiB RAM
Multi-AZ deployment Info
Create replica in different zone
Creates a replica in a different Availability Zone (AZ) to provide data redundancy, eliminate I/O freezes, and minimize latency spikes during system backups.
No
Storage type Info
Provisioned IOPS (SSD)
Allocated storage
100
GiB
(Minimum: 100 GiB, Maximum: 16384 GiB)
Provisioned IOPS Info
1000
Additional configuration
Processor features
Override default values
You can change the number of CPU cores and threads per core on the DB instance class.
Core count Info
2
Threads per core Info
2
Estimated monthly costs

Définissez les options suivantes sur les valeurs appropriées pour votre classe d'instance de base de données sous Fonctions du processeur :

- Nombre de cœurs – Définissez le nombre de cœurs d'UC à l'aide de cette option. La valeur doit être égale ou inférieure au nombre maximum de cœurs d'UC pour la classe d'instance de base de données.
- Threads par cœur – Spécifiez 2 pour activer plusieurs threads par cœur ou spécifiez 1 pour désactiver plusieurs threads par cœur.

Lorsque vous modifiez ou restaurez une instance de base de données, vous pouvez également définir les cœurs d'UC et les threads par cœur d'UC sur les valeurs par défaut pour la classe d'instance.

Lorsque vous affichez les détails d'une instance de base de données dans la console, vous pouvez afficher les informations de processeur pour sa classe d'instance de base de données dans l'onglet Configuration. L'image suivante montre une classe d'instance de base de données avec un cœur d'UC et plusieurs threads par cœur activés.



En ce qui concerne les instances de base de données Oracle, les informations du processeur apparaissent uniquement pour les instances de base de données Réutilisez vos licences (BYOL).

AWS CLI

Vous pouvez définir les fonctions de processeur pour une instance de base de données lorsque vous exécutez l'une des commandes d' AWS CLI suivantes :

- create-db-instance
- modify-db-instance
- restore-db-instance-from-instantané de base de données
- restore-db-instance-from-s3
- restore-db-instance-to-point-in-time

Pour configurer le processeur d'une classe d'instance de base de données pour une instance de base de données à l'aide de l'option AWS CLI, incluez l'`--processor-features`option dans la commande. Spécifiez le nombre de cœurs d'UC avec le nom de fonction `coreCount`, et spécifiez si plusieurs threads par cœur sont activés avec le nom de fonction `threadsPerCore`.

L'option a la syntaxe suivante.

```
--processor-features "Name=coreCount,Value=<value>" "Name=threadsPerCore,Value=<value>"
```

Vous trouverez ci-après des exemples de configuration du processeur :

Exemples

- Définition du nombre de cœurs d'UC pour une instance de base de données
- Définition du nombre de cœurs d'UC et désactivation de plusieurs threads pour une instance de base de données
- Affichage des valeurs de processeur valides pour une classe d'instance de base de données
- Réinitialiser les paramètres de processeur par défaut pour une instance de base de données
- Rétablissement du nombre de cœurs d'UC par défaut pour une instance de base de données
- Rétablissement du nombre de threads par cœur par défaut pour une instance de base de données

Définition du nombre de cœurs d'UC pour une instance de base de données

Example

L'exemple suivant modifie `mydbinstance` en définissant le nombre de cœurs d'UC sur 4. Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Si vous souhaitez appliquer les modifications pendant la fenêtre de maintenance planifiée, omettez l'option `--apply-immediately`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --processor-features "Name=coreCount,Value=4" \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --processor-features "Name=coreCount,Value=4" ^
    --apply-immediately
```

Définition du nombre de cœurs d'UC et désactivation de plusieurs threads pour une instance de base de données

Example

L'exemple suivant modifie `mydbinstance` en définissant le nombre de cœurs d'UC sur `4` et en désactivant plusieurs threads par cœur. Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Si vous souhaitez appliquer les modifications pendant la fenêtre de maintenance planifiée, omettez l'option `--apply-immediately`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --processor-features "Name=coreCount,Value=4" "Name=threadsPerCore,Value=1" \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --processor-features "Name=coreCount,Value=4" "Name=threadsPerCore,Value=1" ^
    --apply-immediately
```

Affichage des valeurs de processeur valides pour une classe d'instance de base de données

Example

Vous pouvez afficher les valeurs de processeur valides pour une classe d'instance de base de données particulière en exécutant la commande describe-orderable-db-instance-options et en spécifiant la classe d'instance pour l'`--db-instance-class`option. Par exemple, la sortie de la commande suivante montre les options de processeur pour la classe d'instance db.r3.large.

```
aws rds describe-orderable-db-instance-options --engine oracle-ee --db-instance-class
 db.r3.large
```

Voici un exemple de sortie pour la commande au format JSON.


```
    {
                "SupportsIops": true,
                "MaxIopsPerGib": 50.0,
                "LicenseModel": "bring-your-own-license",
                "DBInstanceClass": "db.r3.large",
                "SupportsIAMDatabaseAuthentication": false,
                "MinStorageSize": 100,
                "AvailabilityZones": [
                    {
                        "Name": "us-west-2a"
                    },
                    {
                        "Name": "us-west-2b"
                    },
                    {
                        "Name": "us-west-2c"
                    }
                ],
                "EngineVersion": "12.1.0.2.v2",
                "MaxStorageSize": 32768,
                "MinIopsPerGib": 1.0,
                "MaxIopsPerDbInstance": 40000,
                "ReadReplicaCapable": false,
                "AvailableProcessorFeatures": [
                    {
                        "Name": "coreCount",
                        "DefaultValue": "1",
                        "AllowedValues": "1"
                    },
                    {
                        "Name": "threadsPerCore",
                        "DefaultValue": "2",
                        "AllowedValues": "1,2"
                    }
                ],
                "SupportsEnhancedMonitoring": true,
                "SupportsPerformanceInsights": false,
                "MinIopsPerDbInstance": 1000,
                "StorageType": "io1",
                "Vpc": false,
                "SupportsStorageEncryption": true,
                "Engine": "oracle-ee",
```

```
                "MultiAZCapable": true
    }
```

De plus, vous pouvez exécuter les commandes suivantes pour les informations de processeur de la classe d'instance de base de données :

- describe-db-instances— Affiche les informations du processeur pour l'instance de base de données spécifiée.
- describe-db-snapshots— Affiche les informations du processeur pour le snapshot de base de données spécifié.
- describe-valid-db-instance-modifications — Affiche les modifications valides apportées au processeur pour l'instance de base de données spécifiée.

Dans la sortie des commandes précédentes, les fonctions du processeur n'ont pas la valeur null seulement si les conditions suivantes sont remplies :

- Vous utilisez une instance de base de données RDS for Oracle.
- Votre instance de base de données RDS for Oracle prend en charge des valeurs de processeur changeantes.
- Les paramètres actuels de cœurs de processeur et de threads sont définis sur des valeurs personnalisées.

Si les conditions précédentes ne sont pas remplies, vous pouvez obtenir le type d'instance en utilisant describe-db-instances. Vous pouvez obtenir les informations sur le processeur pour ce type d'instance en exécutant l' EC2 opération describe-instance-types.

Réinitialiser les paramètres de processeur par défaut pour une instance de base de données

Example

L'exemple suivant modifie `mydbinstance` en rétablissant les valeurs de processeur par défaut pour la classe d'instance de base de données. Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Si vous souhaitez appliquer les modifications pendant la fenêtre de maintenance planifiée, omettez l'option `--apply-immediately`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
```

```
    --db-instance-identifier mydbinstance \
    --use-default-processor-features \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --use-default-processor-features ^
    --apply-immediately
```

Rétablissement du nombre de cœurs d'UC par défaut pour une instance de base de données

Example

L'exemple suivant modifie `mydbinstance` en rétablissant le nombre de cœurs d'UC par défaut pour la classe d'instance de base de données. Le paramètre des threads par cœur n'est pas modifié. Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Si vous souhaitez appliquer les modifications pendant la fenêtre de maintenance planifiée, omettez l'option `--apply-immediately`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --processor-features "Name=coreCount,Value=DEFAULT" \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --processor-features "Name=coreCount,Value=DEFAULT" ^
    --apply-immediately
```

Rétablissement du nombre de threads par cœur par défaut pour une instance de base de données

Example

L'exemple suivant modifie `mydbinstance` en rétablissant le nombre de threads par cœur par défaut pour la classe d'instance de base de données. Le nombre de cœurs d'UC n'est pas modifié.

Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Si vous souhaitez appliquer les modifications pendant la fenêtre de maintenance planifiée, omettez l'option `--apply-immediately`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --processor-features "Name=threadsPerCore,Value=DEFAULT" \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --processor-features "Name=threadsPerCore,Value=DEFAULT" ^
    --apply-immediately
```

API RDS

Vous pouvez définir les fonctions de processeur pour une instance de base de données lorsque vous appelez l'une des opérations d'API Amazon RDS suivantes :

- CréerDBInstance
- ModifyDBInstance
- Restaurer à DBInstance partir de DBSnapshot
- Restaurer DBInstance depuis S3
- RestaurerDBInstanceToPointInTime

Pour configurer les fonctions de processeur d'une classe d'instance de base de données pour une instance de base de données en utilisant l'API Amazon RDS, incluez le paramètre `ProcessFeatures` dans l'appel.

Le paramètre a la syntaxe suivante.

```
ProcessFeatures "Name=coreCount,Value=<value>" "Name=threadsPerCore,Value=<value>"
```

Spécifiez le nombre de cœurs d'UC avec le nom de fonction `coreCount`, et spécifiez si plusieurs threads par cœur sont activés avec le nom de fonction `threadsPerCore`.

Vous pouvez afficher les valeurs de processeur valides pour une classe d'instance de base de données particulière en exécutant l'opération DescribeOrderableDBInstanceOptions et en spécifiant la classe d'instance pour le `DBInstanceClass` paramètre. Vous pouvez également utiliser les opérations suivantes :

- Décrire DBInstances — Affiche les informations du processeur pour l'instance de base de données spécifiée.
- Décrire DBSnapshots — Affiche les informations du processeur pour le snapshot de base de données spécifié.
- DescribeValidDBInstanceModifications — Affiche les modifications valides apportées au processeur pour l'instance de base de données spécifiée.

Dans la sortie des opérations précédentes, les fonctions du processeur n'ont pas la valeur null seulement si les conditions suivantes sont remplies :

- Vous utilisez une instance de base de données RDS for Oracle.
- Votre instance de base de données RDS for Oracle prend en charge des valeurs de processeur changeantes.
- Les paramètres actuels de cœurs de processeur et de threads sont définis sur des valeurs personnalisées.

Si les conditions précédentes ne sont pas remplies, vous pouvez obtenir le type d'instance à l'aide de Describe DBInstances. Vous pouvez obtenir les informations sur le processeur pour ce type d'instance en exécutant l' EC2 opération DescribeInstanceTypes.

Spécifications matérielles pour les classes d'instance de base de données

Dans les de cette section, vous trouverez des informations matérielles sur les classes d'instances de base de données Amazon RDS .

Pour plus d'informations sur le moteur de base de données Amazon RDS pris en charge pour chaque classe d'instance de base de données, veuillez consulter Moteurs de base de données pris en charge pour les classes d'instance de base de données.

Rubriques

- Terminologie matérielle pour les classes d'instances de base de données

- Spécifications matérielles pour les classes d'instances à usage général
- Spécifications matérielles pour les classes d'instances optimisées pour la mémoire
- Spécifications matérielles pour les classes d'instances optimisées pour le calcul
- Spécifications matérielles pour les classes d'instances à performances évolutives

Terminologie matérielle pour les classes d'instances de base de données

La terminologie suivante est utilisée pour décrire les spécifications matérielles des classes d'instances de base de données :

vCPU

Nombre d'unités centrales virtuelles (CPUs). Un processeur virtuel est une unité de capacité que vous pouvez utiliser pour comparer les classes d'instances de base de données. Au lieu d'acheter ou de louer un processeur particulier pour l'utiliser pendant plusieurs mois ou plusieurs années, vous louez la capacité à l'heure. Notre but est de fournir une quantité constante et spécifique de capacité CPU, dans les limites du matériel sous-jacent.

ECU

Mesure relative de la puissance de traitement entière d'une EC2 instance Amazon. Pour permettre aux développeurs de comparer facilement la capacité du processeur entre différentes classes d'instances, nous avons défini une unité de EC2 calcul Amazon. La quantité de CPU allouée à une instance particulière est exprimée en termes de ces unités de EC2 calcul. Un ECU fournit actuellement une capacité de processeur équivalente à celle d'un processeur Opteron GHz 2007 ou Xeon 2007 1,0—1,2.

Mémoire (Gio)

Mémoire RAM, en gibioctets (Gio), allouée à l'instance de base de données. Il existe souvent un ratio cohérent entre la mémoire et le processeur virtuel. Citons, par exemple, la classe d'instance db.r4, qui a un ratio mémoire/processeur virtuel similaire à celui de la classe db.r5. Toutefois, dans la plupart des cas d'utilisation, la classe d'instance db.r5 fournit de meilleures performances, plus cohérentes, que la classe d'instance db.r4.

Optimisé pour EBS

Une instance de base de données utilise une pile de configuration optimisée et fournit une capacité supplémentaire dédiée aux I/O. Cette optimisation offre les meilleures performances

en minimisant les conflits entre les I/O et le trafic en provenance de votre instance. Pour plus d'informations sur les instances optimisées pour Amazon EBS, consultez la section Instances optimisées pour Amazon EBS dans le guide de l'utilisateur Amazon. EC2

Les instances optimisées pour EBS ont un taux d'IOPS de base et un taux d'IOPS maximal. Le taux maximal d'IOPS est appliqué au niveau de l'instance de base de données. Un ensemble de volumes EBS dont la combinaison donne un taux d'IOPS supérieur au maximum ne peut pas dépasser le seuil au niveau de l'instance. Par exemple, si le nombre maximal d'IOPS pour une classe d'instance de base de données précise est de 40 000 et que vous attachez quatre volumes EBS de 64 000 IOPS, le nombre maximal d'IOPS est de 40 000 au lieu de 256 000. Pour connaître le nombre maximal d'IOPS spécifique à chaque type d' EC2 instance, consultez la section Types d'instances pris en charge dans le Guide de EC2 l'utilisateur Amazon pour les instances Linux.

Taille max. Bande passante EBS (Mbit/s)

Bande passante EBS maximale en mégabits par seconde. Divisez cette valeur par 8 pour calculer le débit attendu en mégaoctets par seconde.

> **Important**
>
> Les volumes à usage général SSD (gp2) pour les instances de base de données Amazon RDS possèdent une limite de débit de 250 Mio/s dans la plupart des cas. Toutefois, cette limite peut varier en fonction de la taille du volume. Pour plus d'informations, consultez les types de volumes Amazon EBS dans le guide de l' EC2 utilisateur Amazon.

Bande passante réseau

Vitesse du réseau par rapport à d'autres classes d'instance de base de données.

Spécifications matérielles pour les classes d'instances à usage général

Les tableaux suivants présentent les spécifications de calcul, de mémoire, de stockage et de bande passante pour les classes d'instances à usage général.

db.m8g — classes d'instance à usage général alimentées par les processeurs Graviton4 AWS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m8g.48xlarge	192	—	768	Optimisé pour EBS uniquement	40 000	50
db.m8g. 24 x large	96	—	512	Optimisé pour EBS uniquement	30 000	40
db.m8g.16xlarge	64	—	384	Optimisé pour EBS uniquement	20 000	30
db.m8g. 12 x large	48	—	256	Optimisé pour EBS uniquement	15 000	22,5
db.m8g.8xlarge	32	—	128	Optimisé pour EBS uniquement	10 000	15
db.m8g.4xlarge*	16	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.m8g.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.m8g.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m8g.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.m7i : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 4e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m7i.48 x large	192	—	768	Optimisé pour EBS uniquement	40 000	50
db.m7i 24 x large	96	—	384	Optimisé pour EBS uniquement	30 000	37,5
db.m7i.16 x large	64	—	256	Optimisé pour EBS uniquement	20 000	25
db.m7i 12 x large	48	—	192	Optimisé pour EBS uniquement	15 000	18,75
db.m7i.8xlarge	32	—	128	Optimisé pour EBS uniquement	10 000	12,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m7i.4xlarge	16	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m7i.2xlarge	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m7i.xlarge	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m7i.large	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.m7g : classes d'instances à usage général optimisées par des processeurs AWS Graviton3

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m7g.16xlarge	64	—	256	Optimisé pour EBS uniquement	20 000	30
db.m7g.12xlarge	48	—	192	Optimisé pour EBS uniquement	15 000	22,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m7g.8xlarge	32	—	128	Optimisé pour EBS uniquement	10 000	15
db.m7g.4xlarge	16	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.m7g.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.m7g.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m7g.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.m6g : classes d'instances polyvalentes optimisées par des processeurs AWS Graviton2

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6g.16xlarge	64	—	256	Optimisé pour EBS uniquement	19 000	25

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6g.12xlarge	48	—	192	Optimisé pour EBS uniquement	13 500	20
db.m6g.8xlarge	32	—	128	Optimisé pour EBS uniquement	9 000	12
db.m6g.4xlarge	16	—	64	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.m6g.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.m6g.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.m6g.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.m6gd — classes d'instance à usage général alimentées par des processeurs Graviton2 et un stockage SSD AWS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6g.16xlarge	64	—	256	2 disques NVMe SSD 1900	19 000	25
db.m6g.12xlarge	48	—	192	2 disques SSD 1425 NVMe	13 500	20
db.m6g.8xlarge	32	—	128	1 NVMe disque SSD de 1900	9 000	12
db.m6gd.4xlarge*	16	—	64	1 NVMe disque SSD 950	4 750	Jusqu'à 10
db.m6gd.2xlarge*	8	—	32	1 disque SSD 474 NVMe	Jusqu'à 4 750	Jusqu'à 10
db.m6gd.xlarge*	4	—	16	1 NVMe disque SSD 237	Jusqu'à 4 750	Jusqu'à 10
db.m6gd.large*	2	—	8	1 x 118 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10

db.m6id : classes d'instances à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération et un stockage SSD

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6id.32xlarge	128	—	512	4 disques NVMe SSD 1900	40 000	50
db.m6id.24xlarge	96	—	384	4 disques SSD 1425 NVMe	30 000	37,5
db.m6id.16xlarge	64	—	256	2 disques NVMe SSD 1900	20 000	25
db.m6id.12xlarge	48	—	192	2 disques SSD 1425 NVMe	15 000	18,75
db.m6id.8xlarge	32	—	128	1 NVMe disque SSD de 1900	10 000	12,5
db.m6id.4xlarge*	16	—	64	1 NVMe disque SSD 950	Jusqu'à 10 000	Jusqu'à 12,5
db.m6id.2xlarge*	8	—	32	1 disque SSD 474 NVMe	Jusqu'à 10 000	Jusqu'à 12,5
db.m6id.xlarge*	4	—	16	1 NVMe disque SSD 237	Jusqu'à 10 000	Jusqu'à 12,5
db.m6id.large*	2	—	8	1 x 118 NVMe SSD	Jusqu'à 10 000	Jusqu'à 12,5

db.m6idn – classes d'instances à usage général avec processeurs Intel Xeon Scalable de 3e génération, stockage SSD et optimisation réseau

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6idn.32xlarge	128	—	512	4 disques NVMe SSD 1900	80 000	200
db.m6idn.24xlarge	96	—	384	4 disques SSD 1425 NVMe	60 000	150
db.m6idn.16xlarge	64	—	256	2 disques NVMe SSD 1900	40 000	100
db.m6idn.12xlarge	48	—	192	2 disques SSD 1425 NVMe	30 000	75
db.m6idn.8xlarge	32	—	128	1 NVMe disque SSD de 1900	20 000	50
db.m6idn.4xlarge*	16	—	64	1 NVMe disque SSD 950	Jusqu'à 20 000	Jusqu'à 50
db.m6idn.2xlarge*	8	—	32	1 disque SSD 474 NVMe	Jusqu'à 20 000	Jusqu'à 40
db.m6idn.xlarge*	4	—	16	1 NVMe disque SSD 237	Jusqu'à 20 000	Jusqu'à 30

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6idn.large*	2	—	8	1 x 118 NVMe SSD	Jusqu'à 20 000	Jusqu'à 25

db.m6in : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération et optimisations du réseau

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6in.32xlarge	128	—	512	Optimisé pour EBS uniquement	80 000	200
db.m6in.24xlarge	96	—	384	Optimisé pour EBS uniquement	60 000	150
db.m6in.16xlarge	64	—	256	Optimisé pour EBS uniquement	40 000	100
db.m6in.12xlarge	48	—	192	Optimisé pour EBS uniquement	30 000	75
db.m6in.8xlarge	32	—	128	Optimisé pour EBS uniquement	20 000	50

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6in.4xlarge*	16	—	64	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 50
db.m6in.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 40
db.m6in.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 30
db.m6in.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 25

db.m6i : classes d'instance à usage général alimentées par des processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6i.32xlarge	128	—	512	Optimisé pour EBS uniquement	40 000	50
db.m6i.24xlarge	96	—	384	Optimisé pour EBS uniquement	30 000	37,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m6i.16xlarge	64	—	256	Optimisé pour EBS uniquement	20 000	25
db.m6i.12xlarge	48	—	192	Optimisé pour EBS uniquement	15 000	18,75
db.m6i.8xlarge	32	—	128	Optimisé pour EBS uniquement	10 000	12,5
db.m6i.4xlarge*	16	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m6i.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m6i.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.m6i.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.m5d : classes d'instances à usage général alimentées par des processeurs Intel Xeon Platinum et un stockage SSD

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m5d.24xlarge	96	345	384	4 disques NVMe SSD 900	19 000	25
db.m5d.16xlarge	64	262	256	4 disques NVMe SSD 600	13 600	20
db.m5d.12xlarge	48	173	192	2 disques NVMe SSD 900	9 500	10
db.m5d.8xlarge	32	131	128	2 disques NVMe SSD 600	6 800	10
db.m5d.4xlarge	16	61	64	2 disques NVMe SSD de 300	4 750	Jusqu'à 10
db.m5d.2xlarge*	8	31	32	1 x 300 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10
db.m5d.xlarge*	4	15	16	1 NVMe disque SSD de 150	Jusqu'à 4 750	Jusqu'à 10
db.m5d.large*	2	10	8	1 NVMe disque SSD de 75 pouces	Jusqu'à 4 750	Jusqu'à 10

db.m5 – classes d'instances à usage général avec processeurs Intel Xeon Platinum

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m5.24xlarge	96	345	384	Optimisé pour EBS uniquement	19 000	25
db.m5.16xlarge	64	262	256	Optimisé pour EBS uniquement	13 600	20
db.m5.12xlarge	48	173	192	Optimisé pour EBS uniquement	9 500	10
db.m5.8xlarge	32	131	128	Optimisé pour EBS uniquement	6 800	10
db.m5.4xlarge	16	61	64	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.m5.2xlarge*	8	31	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.m5.xlarge*	4	15	16	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.m5.large*	2	10	8	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.m4 — classes d'instances à usage général dotées de processeurs Intel Xeon Scalable

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m4.16xlarge	64	188	256	Optimisé pour EBS uniquement	10 000	25
db.m4.10xlarge	40	124,5	160	Optimisé pour EBS uniquement	4 000	10
db.m4.4xlarge	16	53,5	64	Optimisé pour EBS uniquement	2 000	Élevée
db.m4.2xlarge	8	25,5	32	Optimisé pour EBS uniquement	1 000	Élevée
db.m4.xlarge	4	13	16	Optimisé pour EBS uniquement	750	Élevée
db.m4.large	2	6,5	8	Optimisé pour EBS uniquement	450	Modérée

db.m3 : classes d'instances polyvalentes

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.m3.2xlarge	8	26	30	Optimisé pour EBS uniquement	1 000	Élevée
db.m3.xlarge	4	13	15	Optimisé pour EBS uniquement	500	Élevée
db.m3.large	2	6,5	7,5	EBS uniquement	—	Modérées
db.m3.medium	1	3	3,75	EBS uniquement	—	Modérée

* Ces classes d'instance de base de données peuvent prendre en charge des performances maximales pendant 30 minutes au moins une fois toutes les 24 heures. Pour plus d'informations sur les performances de base des types d' EC2 instances sous-jacents, consultez la section Instances optimisées pour Amazon EBS dans le guide de EC2 l'utilisateur Amazon.

Spécifications matérielles pour les classes d'instances optimisées pour la mémoire

Les tableaux suivants présentent les spécifications de calcul, de mémoire, de stockage et de bande passante pour les classes d'instances optimisées pour la mémoire.

db.z1d : classes d'instance à mémoire optimisée

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.z1d.12xlarge	48	271	384	2 disques NVMe SSD 900	14 000	25
db.z1d.6xlarge	24	134	192	1 NVMe disque SSD 900	7 000	10
db.z1d.3xlarge	12	75	96	1 NVMe disque SSD 450	3 500	Jusqu'à 10
db.z1d.2xlarge	8	53	64	1 x 300 NVMe SSD	2 333	Jusqu'à 10
db.z1d.xlarge*	4	28	32	1 NVMe disque SSD de 150	Jusqu'à 2 333	Jusqu'à 10
db.z1d.large*	2	15	16	1 NVMe disque SSD de 75 pouces	Jusqu'à 2 333	Jusqu'à 10

db.x2g — classes d'instance optimisées pour la mémoire avec processeurs Graviton2 AWS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2g.16xlarge	64	—	1 024	Optimisé pour EBS uniquement	19 000	25
db.x2g.12xlarge	48	—	768	Optimisé pour EBS uniquement	14 250	20
db.x2g.8xlarge	32	—	512	Optimisé pour EBS uniquement	9 500	12
db.x2g.4xlarge	16	—	256	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.x2g.2xlarge	8	—	128	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.x2g.xlarge	4	—	64	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.x2g.large	2	—	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.x2idn : classes d'instances optimisées pour la mémoire avec processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2idn.32xlarge	128	—	2 048	2 disques NVMe SSD 1900	80 000	100
db.x2idn.24xlarge	96	—	1 536	2 disques SSD 1425 NVMe	60 000	75
db.x2idn.16xlarge	64	—	1,024	1 NVMe disque SSD de 1900	40 000	50

db.x2iedn : classes d'instance optimisées pour la mémoire avec des processeurs évolutifs Intel Xeon de 3e génération NVMe basés sur SSDs des processeurs locaux

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2iedn.32xlarge	128	—	4 096	2 disques NVMe SSD 1900	80 000	100
db.x2iedn.24xlarge	96	—	3 072	2 disques SSD 1425 NVMe	60 000	75
db.x2iedn.16xlarge	64	—	2 048	1 NVMe disque SSD de 1900	40 000	50

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2iedn.8xlarge	32	—	1,024	1 NVMe disque SSD 950	20 000	25
db.x2iedn.4xlarge	16	—	512	1 NVMe disque SSD 475	Jusqu'à 20 000	Jusqu'à 25
db.x2iedn.2xlarge	8	—	256	1 NVMe disque SSD 237	Jusqu'à 20 000	Jusqu'à 25
db.x2iedn.xlarge	4	—	128	1 x 118 NVMe SSD	Jusqu'à 20 000	Jusqu'à 25

db.x2iezn : classes d'instances optimisées pour la mémoire avec processeurs Intel Xeon Scalable de 2e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2iezn.12xlarge	>48	—	1 536	Optimisé pour EBS uniquement	19 000	100
db.x2iezn.8xlarge	32	—	1,024	Optimisé pour EBS uniquement	12 000	75

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x2iezn.6xlarge	24	—	768	Optimisé pour EBS uniquement	Jusqu'à 9 500	50
db.x2iezn.4xlarge	16	—	512	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 25
db.x2iezn.2xlarge	8	—	256	Optimisé pour EBS uniquement	Jusqu'à 3 170	Jusqu'à 25

db.x1e – Classes d'instance à mémoire optimisée

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x1e.32xlarge	128	340	3 904	Optimisé pour EBS uniquement	14 000	25
db.x1e.16xlarge	64	179	1 952	Optimisé pour EBS uniquement	7 000	10
db.x1e.8xlarge	32	91	976	Optimisé pour EBS uniquement	3 500	Jusqu'à 10

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x1e.4xlarge	16	47	488	Optimisé pour EBS uniquement	1 750	Jusqu'à 10
db.x1e.2xlarge	8	23	244	Optimisé pour EBS uniquement	1 000	Jusqu'à 10
db.x1e.xlarge	4	12	122	Optimisé pour EBS uniquement	500	Jusqu'à 10

db.x1 – Classes d'instance à mémoire optimisée

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.x1.32xlarge	128	349	1 952	Optimisé pour EBS uniquement	14 000	25
db.x1.16xlarge	64	174,5	976	Optimisé pour EBS uniquement	7 000	10

db.r8g — classes d'instance optimisées pour la mémoire avec processeurs Graviton4 AWS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r8g.48xlarge	192	—	1536	Optimisé pour EBS uniquement	40 000	50
db.r8g.24xlarge	96	—	768	Optimisé pour EBS uniquement	30 000	40
db.r8g.16xlarge	64	—	512	Optimisé pour EBS uniquement	20 000	30
db.r8g.12xlarge	48	—	384	Optimisé pour EBS uniquement	15 000	22,5
db.r8g.8xlarge	32	—	256	Optimisé pour EBS uniquement	10 000	15
db.r8g.4xlarge*	16	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.r8g.2xlarge*	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.r8g.xlarge*	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r8g.large*	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.r7i : classes d'instances optimisées pour la mémoire alimentées par des processeurs Intel Xeon Scalable de 4e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r7i.48 x large	192	—	1536	Optimisé pour EBS uniquement	40 000	50
db.r7i 24 x large	96	—	768	Optimisé pour EBS uniquement	30 000	37,5
db.r7i.16 x large	64	—	512	Optimisé pour EBS uniquement	20 000	25
db.r7i 12 x large	48	—	384	Optimisé pour EBS uniquement	15 000	18,75
db.r7i.8xlarge	32	—	256	Optimisé pour EBS uniquement	10 000	12,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r7i.4xlarge	16	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r7i.2xlarge	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r7i.xlarge	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r7i.large	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.r7g – classes d'instances à mémoire optimisée avec processeurs AWS Graviton3

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r7g.16xlarge	64	—	512	Optimisé pour EBS uniquement	20 000	30
db.r7g.12xlarge	48	—	384	Optimisé pour EBS uniquement	15 000	22,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r7g.8xlarge	32	—	256	Optimisé pour EBS uniquement	10 000	15
db.r7g.4xlarge	16	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.r7g.2xlarge*	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 15
db.r7g.xlarge*	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r7g.large*	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.r6g : classes d'instances à mémoire optimisée avec processeurs AWS Graviton2

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6g.16xlarge	64	—	512	Optimisé pour EBS uniquement	19 000	25

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6g.12xlarge	48	—	384	Optimisé pour EBS uniquement	13 500	20
db.r6g.8xlarge	32	—	256	Optimisé pour EBS uniquement	9 000	12
db.r6g.4xlarge	16	—	128	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.r6g.2xlarge*	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.r6g.xlarge*	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.r6g.large*	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.r6gd — classes d'instance optimisées pour la mémoire avec processeurs Graviton2 et stockage SSD AWS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6g.16xlarge	64	—	512	2 disques NVMe SSD 1900	19 000	25
db.r6g.12xlarge	48	—	384	2 disques SSD 1425 NVMe	13 500	20
db.r6g.8xlarge	32	—	256	1 NVMe disque SSD de 1900	9 000	12
db.r6g.4xlarge	16	—	128	1 NVMe disque SSD 950	4 750	Jusqu'à 10
db.r6g.2xlarge	8	—	64	1 disque SSD 474 NVMe	Jusqu'à 4 750	Jusqu'à 10
db.r6gd.xlarge	4	—	32	1 NVMe disque SSD 237	Jusqu'à 4 750	Jusqu'à 10
db.r6gd.large	2	—	16	1 x 118 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10

db.r6id : classes d'instances optimisées pour la mémoire avec processeurs Intel Xeon Scalable de 3e génération et stockage SSD

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6id.32xlarge	128	—	1,024	SSD 4 x 1900 NVMe	40 000	50
db.r6id.24xlarge	96	—	768	SSD 4 x 1425 NVMe	30 000	37,5
db.r6id.16xlarge	64	—	512	SSD 2 x 1900 NVMe	20 000	25
db.r6id.12xlarge	48	—	384	SSD 2 x 1425 NVMe	15 000	18,75
db.r6id.8xlarge	32	—	256	1 x 1900 SSD NVMe	10 000	12,5
db.r6id.4xlarge*	16	—	128	1 x 950 SSD NVMe	Jusqu'à 10 000	Jusqu'à 12,5
db.r6id.2xlarge*	8	—	64	1 x 474 SSD NVMe	Jusqu'à 10 000	Jusqu'à 12,5
db.r6id.xlarge*	4	—	32	1 x 237 SSD NVMe	Jusqu'à 10 000	Jusqu'à 12,5
db.r6id.large*	2	—	16	1 x 118 SSD NVMe	Jusqu'à 10 000	Jusqu'à 12,5

db.r6idn : classes d'instances optimisées pour la mémoire avec processeurs Intel Xeon Scalable de 3e génération, stockage SSD et optimisation du réseau

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6idn.32xlarge	128	—	1,024	SSD 4 x 1900 NVMe	80 000	200
db.r6idn.24xlarge	96	—	768	SSD 4 x 1425 NVMe	60 000	150
db.r6idn.16xlarge	64	—	512	SSD 2 x 1900 NVMe	40 000	100
db.r6idn.12xlarge	48	—	384	SSD 2 x 1425 NVMe	30 000	75
db.r6idn.8xlarge	32	—	256	1 x 1900 SSD NVMe	20 000	50
db.r6idn.4xlarge*	16	—	128	1 x 950 SSD NVMe	Jusqu'à 20 000	Jusqu'à 50
db.r6idn.2xlarge*	8	—	64	1 x 474 SSD NVMe	Jusqu'à 20 000	Jusqu'à 40
db.r6idn.xlarge*	4	—	32	1 x 237 SSD NVMe	Jusqu'à 20 000	Jusqu'à 30
db.r6idn.large*	2	—	16	1 x 118 SSD NVMe	Jusqu'à 20 000	Jusqu'à 25

db.r6in : classes d'instances optimisées pour la mémoire avec processeurs Intel Xeon Scalable de 3e génération et optimisation du réseau

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6in.32xlarge	128	—	1,024	Optimisé pour EBS uniquement	80 000	200
db.r6in.24xlarge	96	—	768	Optimisé pour EBS uniquement	60 000	150
db.r6in.16xlarge	64	—	512	Optimisé pour EBS uniquement	40 000	100
db.r6in.12xlarge	48	—	384	Optimisé pour EBS uniquement	30 000	75
db.r6in.8xlarge	32	—	256	Optimisé pour EBS uniquement	20 000	50
db.r6in.4xlarge*	16	—	128	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 50
db.r6in.2xlarge*	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 40
db.r6in.xlarge*	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 30

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6in.large*	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 20 000	Jusqu'à 25

db.r6i — Classes d'instances optimisées pour la mémoire Oracle préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6i.8xlarge.tpc2.mem4x	32	—	1 024	Optimisé pour EBS uniquement	40 000	50
db.r6i.8xlarge.tpc2.mem3x	32	—	768	Optimisé pour EBS uniquement	30 000	37,5
db.r6i.6xlarge.tpc2.mem4x	24	—	768	Optimisé pour EBS uniquement	30 000	37,5
db.r6i.4xlarge.tpc2.mem4x	16	—	512	Optimisé pour EBS uniquement	20 000	25
db.r6i.4xlarge.tpc2.mem3x	16	—	384	Optimisé pour EBS uniquement	15 000	18,75

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6i.4xlarge.tpc2.mem2x	16	—	256	Optimisé pour EBS uniquement	10 000	12,5
db.r6i.2xlarge.tpc2.mem8x	8	—	512	Optimisé pour EBS uniquement	20 000	12,5
db.r6i.2xlarge.tpc2.mem4x	8	—	256	Optimisé pour EBS uniquement	10 000	12,5
db.r6i.2xlarge.tpc1.mem2x	8	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	12,5
db.r6i.xlarge.tpc2.mem4x	4	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	12,5
db.r6i.xlarge.tpc2.mem2x	4	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	12,5
db.r6i.large.tpc1.mem2x	2	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	12,5

db.r6id – classes d'instances à mémoire optimisée avec processeurs Intel Xeon Scalable de 3e génération

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6i.32xlarge	128	—	1,024	Optimisé pour EBS uniquement	40 000	50
db.r6i.24xlarge	96	—	768	Optimisé pour EBS uniquement	30 000	37,5
db.r6i.16xlarge	64	—	512	Optimisé pour EBS uniquement	20 000	25
db.r6i.12xlarge	48	—	384	Optimisé pour EBS uniquement	15 000	18,75
db.r6i.8xlarge	32	—	256	Optimisé pour EBS uniquement	10 000	12,5
db.r6i.4xlarge*	16	—	128	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r6i.2xlarge*	8	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5
db.r6i.xlarge*	4	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r6i.large*	2	—	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 12,5

db.r5d – classes d'instances à mémoire optimisée avec processeurs Intel Xeon Platinum et stockage SSD

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5d.24xlarge	96	347	768	4 disques NVMe SSD 900	19 000	25
db.r5d.16xlarge	64	264	512	4 disques NVMe SSD 600	13 600	20
db.r5d.12xlarge	48	173	384	2 disques NVMe SSD 900	9 500	10
db.r5d.8xlarge	32	132	256	2 disques NVMe SSD 600	6 800	10
db.r5d.4xlarge	16	71	128	2 disques NVMe SSD de 300	4 750	Jusqu'à 10

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5d.2xlarge*	8	38	64	1 x 300 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10
db.r5d.xlarge*	4	19	32	1 NVMe disque SSD de 150	Jusqu'à 4 750	Jusqu'à 10
db.r5d.large*	2	10	16	1 NVMe disque SSD de 75 pouces	Jusqu'à 4 750	Jusqu'à 10

db.r5b : classes d'instances Oracle à mémoire optimisée préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5b.8xlarge.tpc2.mem3x	32	—	768	Optimisé pour EBS uniquement	60 000	25
db.r5b.6xlarge.tpc2.mem4x	24	—	768	Optimisé pour EBS uniquement	60 000	25
db.r5b.4xlarge.tpc2.mem4x	16	—	512	Optimisé pour EBS uniquement	40 000	20

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5b.4xlarge.tpc2.mem3x	16	—	384	Optimisé pour EBS uniquement	30 000	10
db.r5b.4xlarge.tpc2.mem2x	16	—	256	Optimisé pour EBS uniquement	20 000	10
db.r5b.2xlarge.tpc2.mem8x	8	—	512	Optimisé pour EBS uniquement	40 000	20
db.r5b.2xlarge.tpc2.mem4x	8	—	256	Optimisé pour EBS uniquement	20 000	10
db.r5b.2xlarge.tpc1.mem2x	8	—	128	Optimisé pour EBS uniquement	10 000	Jusqu'à 10
db.r5b.xlarge.tpc2.mem4x	4	—	128	Optimisé pour EBS uniquement	10 000	Jusqu'à 10
db.r5b.xlarge.tpc2.mem2x	4	—	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 10
db.r5b.large.tpc1.mem2x	2	—	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 10

db.r5b – classes d'instances à mémoire optimisée avec processeurs Intel Xeon Platinum et optimisation EBS

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5b.24xlarge	96	347	768	Optimisé pour EBS uniquement	60 000	25
db.r5b.16xlarge	64	264	512	Optimisé pour EBS uniquement	40 000	20
db.r5b.12xlarge	48	173	384	Optimisé pour EBS uniquement	30 000	10
db.r5b.8xlarge	32	132	256	Optimisé pour EBS uniquement	20 000	10
db.r5b.4xlarge	16	71	128	Optimisé pour EBS uniquement	10 000	Jusqu'à 10
db.r5b.2xlarge*	8	38	64	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 10
db.r5b.xlarge*	4	19	32	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 10

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5b.large*	2	10	16	Optimisé pour EBS uniquement	Jusqu'à 10 000	Jusqu'à 10

db.r5 : classes d'instances Oracle à mémoire optimisée préconfigurées pour une mémoire, un stockage et des E/S élevés

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5.12xlarge.tpc2.mem2x	48	—	768	Optimisé pour EBS uniquement	19 000	25
db.r5.8xlarge.tpc2.mem3x	32	—	768	Optimisé pour EBS uniquement	19 000	25
db.r5.6xlarge.tpc2.mem4x	24	—	768	Optimisé pour EBS uniquement	19 000	25
db.r5.4xlarge.tpc2.mem4x	16	—	512	Optimisé pour EBS uniquement	13 600	20
db.r5.4xlarge.tpc2.mem3x	16	—	384	Optimisé pour EBS uniquement	9 500	10

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5.4xlarge.tpc2.mem2x	16	—	256	Optimisé pour EBS uniquement	6 800	10
db.r5.2xlarge.tpc2.mem8x	8	—	512	Optimisé pour EBS uniquement	13 600	20
db.r5.2xlarge.tpc2.mem4x	8	—	256	Optimisé pour EBS uniquement	6 800	10
db.r5.2xlarge.tpc1.mem2x	8	—	128	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.r5.xlarge.tpc2.mem4x	4	—	128	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.r5.xlarge.tpc2.mem2x	4	—	64	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.r5.large.tpc1.mem2x	2	—	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.r5 : classes d'instance à mémoire optimisée

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r5.24xlarge	96	347	768	Optimisé pour EBS uniquement	19 000	25
db.r5.16xlarge	64	264	512	Optimisé pour EBS uniquement	13 600	20
db.r5.12xlarge	48	173	384	Optimisé pour EBS uniquement	9 500	12
db.r5.8xlarge	32	132	256	Optimisé pour EBS uniquement	6 800	10
db.r5.4xlarge	16	71	128	Optimisé pour EBS uniquement	4 750	Jusqu'à 10
db.r5.2xlarge*	8	38	64	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.r5.xlarge*	4	19	32	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10
db.r5.large*	2	10	16	Optimisé pour EBS uniquement	Jusqu'à 4 750	Jusqu'à 10

db.r4 – classes d'instances à mémoire optimisée avec processeurs Intel Xeon Scalable

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r4.16xlarge	64	195	488	Optimisé pour EBS uniquement	14 000	25
db.r4.8xlarge	32	99	244	Optimisé pour EBS uniquement	7 000	10
db.r4.4xlarge	16	53	122	Optimisé pour EBS uniquement	3 500	Jusqu'à 10
db.r4.2xlarge	8	27	61	Optimisé pour EBS uniquement	1 700	Jusqu'à 10
db.r4.xlarge	4	13,5	30,5	Optimisé pour EBS uniquement	850	Jusqu'à 10
db.r4.large	2	7	15,25	Optimisé pour EBS uniquement	425	Jusqu'à 10

db.r3 – Classes d'instance à mémoire optimisée

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.r3.8xlarge**	32	104	244	EBS uniquement	—	10
db.r3.4xlarge	16	52	122	Optimisé pour EBS uniquement	2 000	Élevée
db.r3.2xlarge	8	26	61	Optimisé pour EBS uniquement	1 000	Élevée
db.r3.xlarge	4	13	30,5	Optimisé pour EBS uniquement	500	Modérée
db.r3.large	2	6,5	15,25	Optimisé pour EBS uniquement	—	Modérée

* Ces classes d'instance de base de données peuvent prendre en charge des performances maximales pendant 30 minutes au moins une fois toutes les 24 heures. Pour plus d'informations sur les performances de base des types d' EC2 instances sous-jacents, consultez la section Instances optimisées pour Amazon EBS dans le guide de EC2 l'utilisateur Amazon.

** La classe d'instance de base de données r3.8xlarge ne dispose pas de bande passante EBS dédiée et n'offre donc pas d'optimisation EBS. Pour cette classe d'instance, le trafic réseau et le trafic Amazon EBS partagent la même interface réseau de 10 gigabits.

Spécifications matérielles pour les classes d'instances optimisées pour le calcul

Les tableaux suivants présentent les spécifications de calcul, de mémoire, de stockage et de bande passante pour les classes d'instances optimisées pour le calcul.

db.c6gd : classes d'instance optimisées pour le calcul (pour les déploiements de clusters de bases de données multi-AZ uniquement)

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.c6gd.16 x large	64	—	128	2 disques NVMe SSD 1900	19 000	25
db.c6gd.12xlarge	48	—	96	2 disques SSD 1425 NVMe	13 500	20
db.c6gd.8xlarge	32	—	64	1 NVMe disque SSD de 1900	9 000	12
db.c6gd.4xlarge	16	—	32	1 NVMe disque SSD 950	4 750	Jusqu'à 10
db.c6gd.2xlarge	8	—	16	1 disque SSD 474 NVMe	Jusqu'à 4 750	Jusqu'à 10
db.c6gd.xlarge	4	—	8	1 NVMe disque SSD 237	Jusqu'à 4 750	Jusqu'à 10
db.c6gd.large	2	—	4	1 x 118 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10
db.c6gd.medium	1	—	2	1 x 59 NVMe SSD	Jusqu'à 4 750	Jusqu'à 10

Spécifications matérielles pour les classes d'instances à performances évolutives

Les tableaux suivants présentent les spécifications de calcul, de mémoire, de stockage et de bande passante pour les classes d'instances à performances évolutives.

db.t4g : classes d'instance de performance à capacité extensible alimentées par des processeurs AWS Graviton2

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.t4g.2xlarge*	8	—	32	Optimisé pour EBS uniquement	Jusqu'à 2 780	Jusqu'à 5
db.t4g.xlarge*	4	—	16	Optimisé pour EBS uniquement	Jusqu'à 2 780	Jusqu'à 5
db.t4g.large*	2	—	8	Optimisé pour EBS uniquement	Jusqu'à 2 780	Jusqu'à 5
db.t4g.medium*	2	—	4	Optimisé pour EBS uniquement	Jusqu’à 2 085	Jusqu'à 5
db.t4g.small*	2	—	2	Optimisé pour EBS uniquement	Jusqu’à 2 085	Jusqu'à 5
db.t4g.micro*	2	—	1	Optimisé pour EBS uniquement	Jusqu’à 2 085	Jusqu'à 5

db.t3 : classes d'instance de performance à capacité extensible

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.t3.2xlarge*	8	Variab	32	Optimisé pour EBS uniquement	Jusqu'à 2 048	Jusqu'à 5
db.t3.xlarge*	4	Variab	16	Optimisé pour EBS uniquement	Jusqu'à 2 048	Jusqu'à 5
db.t3.large*	2	Variab	8	Optimisé pour EBS uniquement	Jusqu'à 2 048	Jusqu'à 5
db.t3.medium*	2	Variab	4	Optimisé pour EBS uniquement	Jusqu'à 1 536	Jusqu'à 5
db.t3.small*	2	Variab	2	Optimisé pour EBS uniquement	Jusqu'à 1 536	Jusqu'à 5
db.t3.micro*	2	Variab	1	Optimisé pour EBS uniquement	Jusqu'à 1 536	Jusqu'à 5

db.t2 : classes d'instance de performance à capacité extensible

Classe d'instance	vCPU	ECU	Mémoire (Gio)	Stockage d'instances (Gio)	Taille max. Bande passante EBS (Mbit/s)	Bande passante du réseau (Gbit/s)
db.t2.2xlarge	8	Variab	32	EBS uniquement	—	Modérée
db.t2.xlarge	4	Variab	16	EBS uniquement	—	Modérée
db.t2.large	2	Variab	8	EBS uniquement	—	Modérée
db.t2.medium	2	Variab	4	EBS uniquement	—	Modérée
db.t2.small	1	Variab	2	EBS uniquement	—	Faible
db.t2.micro	1	Variab	1	EBS uniquement	—	Faible

* Ces classes d'instance de base de données peuvent prendre en charge des performances maximales pendant 30 minutes au moins une fois toutes les 24 heures. Pour plus d'informations sur les performances de base des types d' EC2 instances sous-jacents, consultez la section Instances optimisées pour Amazon EBS dans le guide de EC2 l'utilisateur Amazon.

Stockage d'instance de base de données Amazon RDS

Les instances de base de données pour Amazon RDS pour DB2, MariaDB, MySQL, PostgreSQL, Oracle et Microsoft SQL Server utilisent les volumes Amazon Elastic Block Store (Amazon EBS) pour le stockage des bases de données et des journaux.

Dans certains cas, la charge de travail de votre base de données ne sera peut-être pas capable d'atteindre 100 % des IOPS que vous avez provisionnés. Pour de plus amples informations, veuillez consulter Facteurs influant sur les performances des bases de données.

Pour de plus amples informations sur la tarification du stockage d'instance, veuillez consulter Tarification Amazon RDS.

Rubriques

- Types de stockage Amazon RDS
- Stockage SSD d'IOPS par seconde provisionnées
- Stockage SSD à usage général
- Caractéristiques de performance des types de stockage SSD
- Stockage magnétique (obsolète)
- Volume de journal dédié (DLV)
- Surveillance des performances des bases de données
- Facteurs influant sur les performances des bases de données

Types de stockage Amazon RDS

Amazon RDS propose trois types de stockage : un SSD IOPS provisionné (également appelé io1 et io2 Block Express), un SSD à usage général (également appelé gp2 et gp3) et un SSD magnétique (également appelé standard). Ils se distinguent par leurs caractéristiques de performance et leur tarif, ce qui signifie que vous avez la possibilité d'adapter vos performances de stockage et vos coûts à vos besoins en matière de charge de travail de base de données. Vous pouvez créer des instances de base de données DB2, MySQL, MariaDB, Oracle, SQL Server et PostgreSQL RDS avec un maximum de 64 tebioctets (TiB) de stockage. RDS pour Db2 ne prend pas en charge les types de stockage gp2 et magnétique.

La liste suivante décrit rapidement les trois types de stockage :

- SSD IOPS provisionné : le stockage IOPS provisionné est conçu pour répondre aux besoins de débit. I/O-intensive workloads, particularly database workloads, that require low I/O latency and consistent I/O Le stockage d'IOPS provisionnés convient le mieux aux environnements de production.

  Pour plus d'informations sur le stockage d'IOPS provisionnés, y compris les plages de tailles de stockage, consultez Stockage SSD d'IOPS par seconde provisionnées.
- SSD à usage général : les volumes SSD à usage général offrent un stockage économique, idéal pour un large éventail de charges de travail exécutées sur des instances de base de données de taille moyenne. Le stockage à usage général convient le mieux aux environnements de développement et de test.

  Pour plus d'informations sur le stockage SSD à usage général, y compris les plages de tailles de stockage, consultez Stockage SSD à usage général.
- Magnétique – Amazon RDS prend également en charge le stockage magnétique pour assurer la rétrocompatibilité. Nous vous recommandons d'utiliser le stockage SSD à usage général ou à IOPS provisionnés pour tout nouveau besoin de stockage. La quantité maximale de stockage autorisée pour les instances de base de données sur le stockage magnétique est de 3 TiB. Pour de plus amples informations, veuillez consulter Stockage magnétique (obsolète).

Stockage SSD d'IOPS par seconde provisionnées

Pour une application de production nécessitant des I/O performances rapides et constantes, nous recommandons le stockage IOPS provisionné. Le stockage des IOPS provisionnés est un type de stockage qui offre des performances de débit prévisibles et une faible latence homogène. Le stockage IOPS provisionnées est optimisé pour les charges de travail de traitement transactionnel en ligne (OLTP) ayant des exigences de performances régulières. Les IOPS provisionnés aident à ajuster ces charges de travail.

Amazon RDS propose deux types de stockage SSD IOPS provisionnés : io2 et io1. Lorsque vous créez une instance de bases de données, vous spécifiez le taux d'IOPS et la taille du volume. Amazon RDS fournit ce taux d'IOPS pour l'instance de base de données jusqu'à ce que vous le changiez.

Rubriques

- Stockage io2 Block Express (recommandé)
- stockage io1 (génération précédente)

- Combinaison du stockage des IOPS provisionnés aux déploiements multi-AZ ou aux réplicas en lecture
- Coûts du stockage IOPS provisionnées
- Tirer le meilleur parti du stockage IOPS provisionné d'Amazon RDS

Stockage io2 Block Express (recommandé)

Pour les I/O-intensive and latency-sensitive workloads, we recommend that you use Provisioned IOPS SSD io2 Block Express storage to achieve up to 256,000 I/O operations per second (IOPS). The throughput of io2 Block Express volumes varies based on the amount of IOPS provisioned per volume and on the size of the I/O opérations en cours.

Tous les volumes RDS io2 basés sur le système AWS Nitro sont des volumes io2 Block Express et offrent une latence moyenne inférieure à la milliseconde. Les instances de base de données non basées sur le système AWS Nitro sont des volumes io2.

Le tableau suivant indique la plage d'IOPS provisionnées et le débit maximal pour chaque moteur de base de données et plage de tailles de stockage.

Moteur de base de données	Plage de tailles de stockage	Plage des IOPS provisionnées	Débit maximal
DB2, MariaDB, MySQL et PostgreSQL	100 à 65 536 GiB	1 000–256 000 IOPS	16 000 Mbits/s
Oracle	100 à 199 GiB	1 000 À 19 000 IOPS	4 000 Mio/s
Oracle	200—65 536 GiB	1 000–256 000 IOPS	16 000 Mbits/s
SQL Server	20 à 65 536 GiB	1 000–256 000 IOPS	4 000 Mio/s

Les plages d'IOPS et de taille de stockage obéissent aux contraintes suivantes :

- Le rapport entre le nombre d'IOPS et le stockage alloué (en GiB) doit être compris entre 0,5 et 1 000. Pour les instances de base de données non basées sur le système AWS Nitro, le ratio doit être compris entre 0,5 et 500.

- Les IOPS maximaux peuvent être provisionnés avec des volumes de 256 Gio et plus (1 000 IOPS × 256 Gio = 256 000 IOPS). Pour les instances de base de données non basées sur le système AWS Nitro, le nombre maximal d'IOPS est atteint à 512 GiB (500 IOPS x 512 GiB = 256 000 IOPS).
- Le débit augmente proportionnellement jusqu'à 0,256 MiB/s par IOPS provisionnée. Un débit maximal de 4 000 IOPS MiB/s peut être atteint à 256 000 IOPS avec une taille de 16 Ko et de 16 000 IOPS ou plus avec une I/O taille de 256 Ko. I/O Pour les instances de base de données non basées sur le système AWS Nitro, un débit maximal de 2 000 MiB/s peut être atteint à 128 000 IOPS avec une taille de 16 Ko. I/O
- Si vous utilisez la scalabilité automatique du stockage, les mêmes rapports entre les IOPS et le seuil de stockage maximum (en Go) s'appliquent également. Pour plus d'informations sur la scalabilité automatique du stockage, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.

Les volumes Amazon RDS io2 Block Express sont disponibles dans toutes les AWS régions commerciales et dans toutes les régions AWS GovCloud (États-Unis). Ces volumes ne sont pas disponibles dans la région Chine.

stockage io1 (génération précédente)

Pour les I/O-intensive workloads, you can use Provisioned IOPS SSD io1 storage and achieve up to 256,000 I/O operations per second (IOPS). The throughput of io1 volumes varies based on the amount of IOPS provisioned per volume and on the size of the I/O opérations en cours. Nous vous recommandons d'utiliser le stockage io2 Block Express lorsqu'il est disponible.

Le tableau suivant indique la plage d'IOPS provisionnées et le débit maximal pour chaque moteur de base de données et plage de tailles de stockage.

Moteur de base de données	Plage de tailles de stockage	Plage des IOPS provisionnées	Débit maximal
DB2, MariaDB, MySQL et PostgreSQL	100 à 399 GiB	Entre 1 000 et 19 950 IOPS	500 Mio/s
DB2, MariaDB, MySQL et PostgreSQL	400—65 536 GiB	1 000–256 000 IOPS	4 000 Mio/s

Moteur de base de données	Plage de tailles de stockage	Plage des IOPS provisionnées	Débit maximal
Oracle	100 à 199 GiB	Entre 1 000 et 9 950 IOPS	500 Mio/s
Oracle	200—65 536 GiB	1 000 À 256 000 IPS[1]	4 000 Mio/s
SQL Server	20 à 16 384 GiB	1 000 À 64 000 IPS[2]	1,000 Mio/s

> **Note**
>
> [1] Pour Oracle, vous pouvez fournir un maximum de 256 000 IOPS uniquement sur le type d'instance r5b.
> [2] Pour SQL Server, le maximum de 64 000 IOPS est garanti uniquement sur les instances basées sur Nitro appartenant aux types d'instance m5*, m6i, r5*, r6i et z1d. D'autres types d'instances garantissent des performances allant jusqu'à 32 000 IOPS.

Les plages d'IOPS et de taille de stockage obéissent aux contraintes suivantes :

- Le rapport entre les IOPS et le stockage alloué (en GiO) doit être de 1 à 50 sur RDS for SQL Server et de 0,5 à 50 sur les autres moteurs de base de données RDS.
- Si vous utilisez la scalabilité automatique du stockage, les mêmes rapports entre les IOPS et le seuil de stockage maximum (en Go) s'appliquent également.

  Pour plus d'informations sur la scalabilité automatique du stockage, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.

Combinaison du stockage des IOPS provisionnés aux déploiements multi-AZ ou aux réplicas en lecture

Pour les cas d'utilisation de traitement de transaction en ligne (OLTP) de production, nous vous recommandons d'utiliser des déploiements multi-AZ, pour profiter d'une meilleure tolérance aux pannes et d'un meilleur stockage des IOPS provisionnés, et ainsi bénéficier de performances rapides et prévisibles.

Vous pouvez également utiliser le stockage IOPS provisionné avec des répliques de lecture pour MySQL, MariaDB ou PostgreSQL. Le type de stockage pour un réplica en lecture est indépendant de celui de l'instance de base de données principale. Par exemple, vous pouvez utiliser le stockage SSD à usage général pour les réplicas en lecture avec une instance de base de données principale qui utilise le stockage SSD d'IOPS provisionnés afin de réduire les coûts. Toutefois, les performances de votre réplique de lecture dans ce cas peuvent différer de celles d'une configuration dans laquelle l'instance de base de données principale et les répliques de lecture utilisent le stockage IOPS provisionné.

Coûts du stockage IOPS provisionnées

Avec le stockage d'IOPS provisionnés, vous devez payer pour les ressources provisionnées, que vous les utilisiez ou non au cours d'un mois donné.

Pour plus d'informations sur la tarification, veuillez consulter Tarification Amazon RDS.

Tirer le meilleur parti du stockage IOPS provisionné d'Amazon RDS

Si votre charge de travail est I/O limitée, l'utilisation du stockage IOPS provisionné peut augmenter le nombre de I/O demandes que le système peut traiter simultanément. L'augmentation de la simultanéité permet de réduire le temps de latence, car I/O les demandes passent moins de temps dans une file d'attente. La réduction de la latence permet des validations de base de données plus rapides, ce qui améliore le temps de réponse et augmente le débit de la base de données.

Le stockage IOPS provisionné permet de réserver de la I/O capacité en spécifiant les IOPS. Toutefois, comme avec tout autre attribut de capacité système, le débit maximal sous charge sera limité par la ressource qui sera utilisée en premier. Cette ressource peut être la bande passante réseau, l'UC, la mémoire ou les ressources internes de la base de données.

Stockage SSD à usage général

Le stockage à usage général offre un stockage rentable qui convient à la plupart des charges de travail de base de données qui ne sont pas sensibles à la latence ou aux performances.

> **Note**
>
> Les instances de base de données qui utilisent le stockage à usage général peuvent connaître une latence beaucoup plus longue que les instances qui utilisent le stockage IOPS provisionné. Si vous avez besoin d'une instance de base de données avec une latence

minimale après ces opérations, nous vous recommandons d'utiliser Stockage SSD d'IOPS par seconde provisionnées.

Amazon RDS propose deux types de stockage à usage général : Stockage GP3 (recommandé) etstockage GP2 (génération précédente).

Stockage GP3 (recommandé)

En utilisant les volumes de stockage GP3 à usage général, vous pouvez personnaliser les performances de stockage indépendamment de la capacité de stockage. Les performances de stockage sont la combinaison des I/O opérations par seconde (IOPS) et de la rapidité avec laquelle le volume de stockage peut effectuer des lectures et des écritures (débit de stockage). Sur les volumes de stockage gp3, Amazon RDS fournit des performances de stockage de base de 3 000 IOPS et 125 Mio/s.

Pour tous les moteurs de base de données RDS, à l'exception de RDS pour SQL Server, lorsque la taille de stockage des volumes gp3 atteint un certain seuil, les performances de stockage de base augmentent. Cela est dû à la répartition en bandes des volumes, selon laquelle le stockage utilise quatre volumes à la place d'un seul. RDS for SQL Server ne prend pas en charge la répartition en bandes des volumes et n'a donc pas de valeur de seuil. Pour les volumes répartis par bandes, Amazon RDS fournit des performances de stockage de base de 12 000 IOPS et 500 Mbits/s.

Les performances de stockage des volumes gp3 sur les moteurs de base de données Amazon RDS, y compris le seuil, sont présentées dans le tableau suivant.

Moteur de base de données	Taille de stockage	Performances de stockage de base	Plage des IOPS provisionnées	Plage de débits de stockage provisionnés
DB2, MariaDB, MySQL et PostgreSQL	20 à 399 GiB	3 000 IOPS/125 Mi s	N/A	N/A
DB2, MariaDB, MySQL et PostgreSQL	400—65 536 GiB	12 000 IOPS/500 M s	12 000–64 000 IOPS	Entre 500 et 4 000 Mio/s

Moteur de base de données	Taille de stockage	Performances de stockage de base	Plage des IOPS provisionnées	Plage de débits de stockage provisionnés
Oracle	20 à 199 GiB	3 000 IOPS/125 Mi s	N/A	N/A
Oracle	200—65 536 GiB	12 000 IOPS/500 M s	12 000–64 000 IOPS	Entre 500 et 4 000 Mio/s
SQL Server	20 à 16 384 GiB	3 000 IOPS/125 Mi s	3 000–16 000 IOPS	Entre 125 et 1 000 Mio/s

Pour chaque moteur de base de données, à l'exception de RDS for SQL Server, vous pouvez allouer des IOPS et un débit de stockage supplémentaires lorsque la taille de stockage est égale ou supérieure à la valeur seuil. Pour RDS for SQL Server, vous pouvez allouer des IOPS et un débit de stockage supplémentaires pour n'importe quelle taille de stockage disponible. Pour tous les moteurs de base de données, vous ne payez que pour les performances de stockage provisionnées supplémentaires. Pour plus d'informations, consultez Tarification d'Amazon RDS.

Bien que les IOPS provisionnés et le débit de stockage ajoutés ne dépendent pas de la taille de stockage, ils sont liés les uns aux autres. Lorsque vous augmentez les IOPS au-dessus de 32 000 pour MariaDB et MySQL, la valeur du débit de stockage passe automatiquement de 500. MiBps Par exemple, lorsque vous définissez les IOPS sur 40 000 sur RDS pour MySQL, le débit de stockage doit être d'au moins 625. MiBps L'augmentation automatique ne se produit pas pour les instances de base de données DB2, Oracle, PostgreSQL et SQL Server.

Pour les clusters de bases de données multi-AZ, Amazon RDS définit automatiquement la valeur du débit en fonction des IOPS que vous fournissez. Vous ne pouvez pas modifier la valeur du débit.

Les valeurs de performances de stockage pour les volumes gp3 sur RDS sont soumises aux contraintes suivantes :

- Le rapport maximal entre le débit de stockage et les IOPS est de 0,25 pour tous les moteurs de base de données pris en charge.
- Le rapport minimal entre les IOPS et le stockage alloué (en Gio) est de 0,5 sur RDS for SQL Server. Il n'y a pas de rapport minimal pour les autres moteurs de base de données pris en charge.

- Le rapport maximal entre les IOPS et le stockage alloué est de 500 pour tous les moteurs de base de données pris en charge.
- Si vous utilisez la scalabilité automatique du stockage, les mêmes rapports entre les IOPS et le seuil de stockage maximum (en Go) s'appliquent également.

  Pour plus d'informations sur la scalabilité automatique du stockage, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.

stockage GP2 (génération précédente)

Lorsque vos applications n'ont pas besoin de performances de stockage élevées, vous pouvez utiliser le stockage SSD à usage général gp2. I/O Les performances de base du stockage gp2 sont de 3 IOPS pour chaque GiB, avec un minimum de 100 IOPS. Cette relation signifie que des volumes plus importants obtiennent de meilleures performances. Par exemple, les performances de base pour un volume de 100 GiB sont de 300 IOPS. Les performances de base pour un volume de 1 000 Gio sont de 3 000 IOPS.

Les volumes gp2 individuels inférieurs à 1 000 Gio peuvent également atteindre 3 000 IOPS pendant des périodes de temps étendues. Le solde I/O créditeur du volume détermine les performances en rafale. Pour une description plus détaillée de l'impact des performances de référence et du solde créditeur d'E/S sur les performances, consultez le billet Comprendre les performances en rafale par rapport aux performances de référence avec Amazon RDS et gp2 sur le AWS blog de base de données.

De nombreuses charges de travail n'épuisent jamais le solde de rafale. Toutefois, certaines charges de travail peuvent épuiser le solde créditeur de stockage maximal de 3 000 IOPS. Planifiez donc votre capacité de stockage en fonction des besoins de vos charges de travail.

Pour les volumes gp2 supérieurs à 4 000 GiB, les performances de base sont supérieures aux performances en rafale. Pour de tels volumes, le mode rafale est sans intérêt dans la mesure où les performances de références dépassent les performances en rafale (3 000 IOPS). Toutefois, pour les instances de base de données de certains moteurs et de certaines tailles, le stockage est réparti sur quatre volumes, ce qui fournit quatre fois le débit de base et quatre fois le débit d'IOPS en rafale d'un seul volume.

Les performances de stockage pour les volumes gp2 de différentes tailles de stockage sur les moteurs de base de données Amazon RDS sont présentées dans le tableau suivant.

Moteur de base de données	Taille de stockage RDS	Plage d'IOPS de base	Plage de débit de base	IOPS en rafale
MariaDB, MySQL et PostgreSQL	5 à 399 GiB	Entre 100 et 1 197 IOPS	Entre 128 et 250 Mio/s	3 000
MariaDB, MySQL et PostgreSQL	400 à 1 335 GiB	Entre 1 200 et 4 005 IOPS	512 à 1 000 Mbits/s	12 000
MariaDB, MySQL et PostgreSQL	1 336 à 3 999 GiB	Entre 4 008 et 11 997 IOPS	1,000 Mio/s	12 000
MariaDB, MySQL et PostgreSQL	4 000 à 65 536 GiB	Entre 12 000 et 64 000 IOPS	1,000 Mio/s	N/A[1]
Oracle	20 à 199 GiB	Entre 100 et 597 IOPS	Entre 128 et 250 Mio/s	3 000
Oracle	200—1 335 GiB	Entre 600 et 4 005 IOPS	512 à 1 000 Mbits/s	12 000
Oracle	1 336 à 3 999 GiB	Entre 4 008 et 11 997 IOPS	1,000 Mio/s	12 000
Oracle	4 000 à 65 536 GiB	Entre 12 000 et 64 000 IOPS	1,000 Mio/s	N/A[1]
SQL Server	20 à 333 GiB	Entre 100 et 999 IOPS	Entre 128 et 250 Mio/s	3 000
SQL Server	334 à 999 GiB	Entre 1 002 et 2 997 IOPS	250 Mio/s	3 000
SQL Server	1 000 à 16 384 GiB	Entre 3 000 et 16 000 IOPS	250 Mio/s	N/A[1]

[1] Les performances de base du volume dépassent les performances de rafale maximales.

Caractéristiques de performance des types de stockage SSD

Le tableau suivant décrit les cas d'utilisation et les caractéristiques de performance par volume pour les volumes de stockage SSD utilisés par Amazon RDS.

Caractéristiques	IOPS provisionnées (io2 Block Express)	IOPS provisionnés (io1)	Usage général (gp3)	Usage général (gp2)
Description	Les meilleures performances du portefeuille de stockage RDS (IOPS, débit, latence) Conçu pour les charges de travail transactionnelles sensibles à la latence	Performances de stockage constantes (IOPS, débit, latence) Conçu pour les charges de travail transactionnelles sensibles à la latence	Flexibilité d'allocation indépendante du stockage, des IOPS et du débit Équilibre les performances de prix pour un large éventail de charges de travail transactionnelles	Fournit des IOPS pouvant être émis en rafale Équilibre les performances de prix pour un large éventail de charges de travail transactionnelles
Cas d'utilisation	Charges de travail transactionnelles critiques nécessitant une latence inférieure à la milliseconde et des performances IOPS soutenues pouvant atteindre 256 000 IOPS	Charges de travail de travail transactionnelles nécessitant des performances d'IOPS soutenues allant jusqu'à 256 000 IOPS	Large gamme de charges de travail exécutées sur des bases de données relationnelles de taille moyenne dans des environnements development/test	Large gamme de charges de travail exécutées sur des bases de données relationnelles de taille moyenne dans des environnements development/test
Latence	Inférieur à une milliseconde,	Moins de 10 milliseco	Moins de 10 milliseco	Moins de 10 milliseco

Caractéristiques	IOPS provisionnées (io2 Block Express)	IOPS provisionnés (io1)	Usage général (gp3)	Usage général (gp2)
	fourni régulièrement 99,9 % du temps	ndes, fournies de manière constante 99,9 % du temps	ndes, fournies de manière constante 99 % du temps	ndes, fournies de manière constante 99 % du temps
Taille du volume	100 à 65 536 GiB	100 à 65 536 GiB (20 à 16 384 GiB sur RDS pour SQL Server)	20 à 65 536 GiB (16 384 GiB sur RDS pour SQL Server)	20 à 65 536 GiB (16 384 GiB sur RDS pour SQL Server)

Caractéristiques	IOPS provisionnées (io2 Block Express)	IOPS provisionnés (io1)	Usage général (gp3)	Usage général (gp2)
Nombre maximal d'IOPS	256 000	256 000 (64 000 sur RDS for SQL Server)	64 000 (16 000 sur RDS for SQL Server)	64 000 (16 000 sur RDS for SQL Server) **Note** Vous ne pouvez pas allouer les IOPS directement sur le stockage gp2. Le nombre d'IOPS varie en fonction de la taille de stockage allouée.

Caractéristiques	IOPS provisionnées (io2 Block Express)	IOPS provisionnés (io1)	Usage général (gp3)	Usage général (gp2)
Débit maximal	Évolue en fonction des IOPS provisionnés jusqu'à 4 000 Mo/s Le débit augmente proportionnellement jusqu'à 0,256 MiB/s par IOPS provisionnée. Un débit maximal de 4 000 IOPS MiB/s peut être atteint à 256 000 IOPS avec une taille de 16 Ko et de 16 000 IOPS ou plus avec une I/O taille de 256 Ko. I/O Pour les instances non basées sur le système AWS Nitro, un débit maximal de 2 000 MiB/s peut être atteint à 128 000 IOPS avec une taille de 16 Ko. I/O	Évolue en fonction des IOPS provisionnés jusqu'à 4 000 Mo/s	Fournir un débit supplémentaire allant jusqu'à 4 000 MB/s (1 000 MB/s sur RDS pour SQL Server)	1000 MB/s (250 MB/s sur RDS pour SQL Server)

Caractéristiques	IOPS provisionnées (io2 Block Express)	IOPS provisionnés (io1)	Usage général (gp3)	Usage général (gp2)
AWS CLI et nom de l'API RDS	io2	io1	gp3	gp2

Redécoupage automatique entre les volumes SSD

Lorsque vous sélectionnez un SSD à usage général ou un SSD à IOPS provisionnés, en fonction du moteur sélectionné et de la quantité de stockage demandée, Amazon RDS répartit automatiquement plusieurs volumes pour améliorer les performances, comme indiqué dans le tableau suivant.

Moteur de base de données	Taille de stockage Amazon RDS	Nombre de volumes provisionnés
Db2	Moins de 400 Gio	1
Db2	400—65 536 GiB	4
MariaDB, MySQL et PostgreSQL	Moins de 400 Gio	1
MariaDB, MySQL et PostgreSQL	400—65 536 GiB	4
Oracle	Moins de 200 Gio	1
Oracle	200—65 536 GiB	4
SQL Server	N’importe quel compte	1

Impact sur les performances lorsque vous modifiez un volume SSD

Lorsque vous modifiez un volume SSD à usage général ou SSD à IOPS provisionnés, il passe par différents états. Lorsque le volume est dans `optimizing` cet état, ses performances se situent entre les spécifications de configuration source et cible. Les performances de volume de transition ne seront pas inférieures à la plus faible des deux spécifications.

Lorsque vous modifiez le stockage d'une instance afin qu'il passe d'un volume à quatre volumes, ou lorsque vous modifiez une instance à l'aide du stockage magnétique, Amazon RDS n'utilise pas la fonctionnalité Elastic Volumes. Amazon RDS provisionne plutôt de nouveaux volumes et déplace de manière transparente les données de l'ancien volume vers les nouveaux. Cette opération consomme une quantité importante d'IOPS et de débit des anciens et des nouveaux volumes. En fonction de la taille du volume et de la charge de travail de base de données présente lors de la modification, cette opération peut consommer une grande quantité d'IOPS, augmenter considérablement le temps de I/O latence et prendre plusieurs heures, alors que l'instance RDS reste dans son état. `Modifying`

Taux d'IOPS de référence et maximum pour les instances optimisées pour EBS

Les instances optimisées pour EBS ont un taux d'IOPS de base et un taux d'IOPS maximal. Le taux maximal d'IOPS est appliqué au niveau de l'instance de base de données. Un ensemble de volumes EBS dont la combinaison donne un taux d'IOPS supérieur au maximum ne peut pas dépasser le seuil au niveau de l'instance. Par exemple, si le nombre maximal d'IOPS pour une classe d'instance de base de données précise est de 40 000 et que vous attachez quatre volumes EBS de 64 000 IOPS, le nombre maximal d'IOPS est de 40 000 au lieu de 256 000. Pour connaître le nombre maximal d'IOPS spécifique à chaque type d' EC2 instance, consultez la section Types d'instances pris en charge dans le Guide de EC2 l'utilisateur Amazon pour les instances Linux.

Stockage magnétique (obsolète)

Amazon RDS ne prend plus en charge le stockage magnétique. Nous vous recommandons d'utiliser un SSD à usage général ou un SSD IOPS provisionné pour vos besoins de stockage.

Volume de journal dédié (DLV)

Vous pouvez utiliser un volume de journal dédié (DLV) pour une instance de base de données qui utilise le stockage PIOPS (Provisioned IOPS) à l'aide de la console Amazon RDS AWS CLI ou de l'API Amazon RDS. Un DLV déplace les journaux de transactions, les journaux de restauration MySQL/MariaDB et les journaux binaires des bases de données PostgreSQL vers un volume de stockage distinct du volume contenant les tables de base de données. Un DLV rend l'enregistrement

des écritures de transactions plus efficace et plus cohérent. DLVs sont idéaux pour les bases de données présentant un stockage alloué important, des exigences élevées I/O par seconde (IOPS) ou des charges de travail sensibles à la latence.

DLVs sont pris en charge pour le stockage PIOPS (io1 et io2 Block Express) et sont créés avec une taille fixe de 1 024 GiB et 3 000 IOPS provisionnées.

> **Note**
>
> DLVs ne sont pas pris en charge pour le stockage à usage général (gp2 et gp3).

Amazon RDS prend DLVs en charge toutes Régions AWS les versions suivantes :

- MariaDB 10.6.7 et versions 10 ultérieures
- MySQL 8.0.28 et versions 8.0 supérieures, MySQL 8.4.3 et versions supérieures 8.4
- PostgreSQL 13.10 et supérieur 13 versions, 14.7 et supérieur 14 versions, 15.2 et supérieur 15 versions, et 16.1 et supérieur 16 versions

RDS prend en charge les DLVs déploiements multi-AZ. Lorsque vous modifiez ou créez une instance multi-AZ, un DLV est créé à la fois pour l'instance principale et pour l'instance secondaire.

RDS prend en charge les DLVs répliques en lecture. Si un DLV est activé sur l'instance de base de données principale, tous les réplicas en lecture créés après l'activation du DLV auront également un DLV. Il ne sera pas activé sur les réplicas en lecture créés avant le passage au DLV, sauf s'il est explicitement modifié à cet effet. Nous recommandons que tous les réplicas en lecture attachés à une instance principale avant l'activation du DLV soient également modifiés manuellement pour avoir un DLV.

Après la modification du paramètre DLV d'une instance de base de données, l'instance doit être redémarrée.

Pour plus d'informations sur l'activation d'un DLV, consultezUtilisation d'un volume dédié aux journaux (DLV).

Surveillance des performances des bases de données

Amazon RDS propose différentes métriques pour déterminer les performances de votre instance de bases de données. Vous pouvez consulter les métriques sur la page de résumé de votre instance

sur l'Amazon RDS Management Console. Vous pouvez également utiliser Amazon CloudWatch pour surveiller ces statistiques. Pour de plus amples informations, veuillez consulter Afficher les métriques dans la RDS console Amazon. La surveillance améliorée fournit des I/O mesures plus détaillées ; pour plus d'informations, voirSurveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Les métriques suivantes sont utiles pour surveiller les performances de votre instance de base de données :

- `DiskQueueDepth`— Le nombre de I/O demandes en attente de traitement dans la file d'attente. Il s'agit de I/O demandes qui ont été soumises par l'application mais qui n'ont pas été envoyées à l'appareil car celui-ci est occupé à traiter d'autres I/O demandes. Le temps passé dans une file d'attente est un élément de la latence et du temps de service (non disponible en tant que métrique). Cette métrique est rapportée comme étant la profondeur de file d'attente moyenne pour un intervalle de temps donné. Amazon RDS indique la profondeur de la file d'attente à intervalles d'une minute. Les valeurs habituelles pour la longueur de file d'attente vont de zéro à plusieurs centaines.
- `EBSByteBalance%`— Le pourcentage de crédits de débit restant dans le bucket burst de votre base de données RDS. Cette métrique est disponible uniquement pour la surveillance basique. La valeur de la métrique est basée sur le débit de tous les volumes, y compris le volume racine, plutôt que sur les seuls volumes contenant des fichiers de base de données.

  Lorsque cette métrique approche zéro, cela signifie que la capacité de calcul de votre instance de base de données est épuisée. Si cela se produit régulièrement, envisagez de passer à une taille de classe d'instance plus grande, par exemple de db.r6g.large à db.r6g.xlarge. Pour de plus amples informations, veuillez consulter Classe d'instances de base de données.
- `ReadIOPSet WriteIOPS` — Le nombre d' I/O opérations effectuées chaque seconde. Cette métrique est rapportée comme étant le nombre moyen d'IOPS pour un intervalle de temps donné. Amazon RDS rapporte les IOPS en lecture et en écriture séparément à intervalles d'une minute. `TotalIOPS`est la somme des IOPS en lecture et en écriture. Les valeurs habituelles pour les IOPS vont de zéro à des dizaines de milliers par seconde.

  Si vos `TotalIOPS` valeurs se rapprochent régulièrement de la valeur d'IOPS provisionnées que vous avez définie pour votre instance de base de données, envisagez d'augmenter les IOPS provisionnées (types de stockage io1, io2 Block Express et gp3).

Les valeurs d'IOPS mesurées sont indépendantes de la taille des I/O operation. This means that when you measure I/O performance, make sure to look at the throughput of the instance, not simply the number of I/O opérations individuelles.

- `ReadLatency`et `WriteLatency` — Le temps écoulé entre la soumission d'une I/O demande et son achèvement. Cette métrique est rapportée comme étant la latence moyenne pour un intervalle de temps donné. Amazon RDS fait état de la latence de lecture et d'écriture séparément par intervalles d'une minute. Les valeurs de latence sont généralement exprimées en millisecondes (ms).
- `ReadThroughput`et `WriteThroughput` — Le nombre d'octets transférés chaque seconde vers ou depuis le disque. Cette métrique est rapportée comme étant le débit moyen pour un intervalle de temps donné. Amazon RDS indique le débit de lecture et d'écriture séparément à intervalles d'une minute en utilisant des unités d'octets par seconde (bande passante maximale du B/s). Typical values for throughput range from zero to the I/O canal).

  Si vos valeurs de débit approchent régulièrement le débit maximal de votre instance de base de données, envisagez de fournir un débit de stockage supérieur si vous utilisez le type de stockage gp3.

Facteurs influant sur les performances des bases de données

Les activités du système, la charge de travail de base de données et la classe d'instance de base de données peuvent affecter les performances des bases de données.

Rubriques

- Activités du système
- Charge de travail d'une base de données
- Classe d'instances de base de données

Activités du système

Les activités suivantes liées au système consomment de la I/O capacité et peuvent réduire les performances de l'instance de base de données en cours :

- Création de veille Multi-AZ
- Création d'un réplica en lecture

- Modification des types de stockage

Charge de travail d'une base de données

Dans certains cas, la conception de votre base de données ou application entraîne des problèmes de simultanéité, des verrouillages ou d'autres formes de conflit de base de données. Vous pouvez alors rencontrer des difficultés pour utiliser toute la bande passante provisionnée directement. De plus, vous pouvez rencontrer les situations suivantes liées aux charges de travail :

- La limite de débit du type d'instance sous-jacent a été atteinte.
- La profondeur de la file d'attente est constamment inférieure à 1 car votre application ne génère pas suffisamment d' I/O opérations.
- Vous êtes confronté à des conflits de requêtes dans la base de données même si une partie de la I/O capacité n'est pas utilisée.

Dans certains cas, aucune ressource du système n'a atteint la limite ou n'en est proche, et l'ajout de threads n'augmente pas le taux de transaction de la base de données. Dans de tels cas, le goulot d'étranglement s'apparente très probablement à un conflit dans la base de données. Les formes les plus courantes sont des conflits de verrous de ligne et de verrous de page d'index, mais il existe bien d'autres possibilités. Si vous vous trouvez dans cette situation, demandez conseil à une personne experte en réglage des performances de bases de données.

Classe d'instances de base de données

Afin d'optimiser les performances de votre instance de base de données Amazon RDS, choisissez un type d'instance de la génération actuelle avec suffisamment de bande passante pour prendre en charge votre type de stockage. Par exemple, vous pouvez choisir des instances optimisées Amazon EBS et des instances avec une connectivité réseau de 10 gigabits.

> **Important**
>
> Selon la classe d'instance que vous utilisez, la performance des IOPS pourrait être inférieure au maximum que RDS vous permet d'allouer. Pour des informations spécifiques sur les performances IOPS pour les classes d'instances de base de données, consultez la section Instances optimisées pour Amazon EBS dans le guide de l'utilisateur Amazon EC2 . Nous vous recommandons de déterminer le nombre maximal d'IOPS pour la classe d'instance avant de définir une valeur d'IOPS provisionnés pour votre instance de base de données.

Pour obtenir des performances optimales, nous vous encourageons à utiliser la dernière génération d'instances. Les instances de base de données de la génération précédente peuvent avoir une limite de stockage d'instance plus faible.

Sur certains systèmes de fichiers 32 bits anciens, les capacités de stockage peuvent être inférieures. Pour déterminer la capacité de stockage de votre instance de base de données, vous pouvez utiliser la AWS CLI commande describe-valid-db-instance-modifications.

La liste suivante montre le stockage maximum que la plupart des classes d'instance de base de données peuvent mettre à l'échelle pour chaque moteur de base de données :

- DB2 — 64 TiB
- MariaDB : 64 Tio
- Microsoft SQL Server — 64 TiB
- MySQL : 64 Tio
- Oracle : 64 Tio
- PostgreSQL : 64 Tio

Le tableau suivant présente quelques exceptions pour le stockage maximum (en Tio). Toutes les instances de base de données RDS pour Microsoft SQL Server, à l'exception du stockage io2 Block Express, ont une capacité de stockage maximale de 16 TiB, il n'y a donc aucune entrée pour SQL Server.

Classe d'instance	Db2	MariaDB	MySQL	Oracle	PostgreSQL
db.m3 – Classes d'instance standard					
db.t4g : classes d'instance de performance à capacité extensible					
db.t4g.medium	N/A	16	16	N/A	32
db.t4g.small	N/A	16	16	N/A	16
db.t4g.micro	N/A	6	6	N/A	6
db.t3 : classes d'instance de performance à capacité extensible					

Classe d'instance	Db2	MariaDB	MySQL	Oracle	PostgreSQL
db.t3.medium	32	16	16	32	32
db.t3.small	32	16	16	32	16
db.t3.micro	N/A	6	6	32	6
db.t2 : classes d'instance de performance à capacité extensible					

Pour de plus amples détails sur les classes d'instances prises en charge, consultez Instances de base de données de la génération précédente.

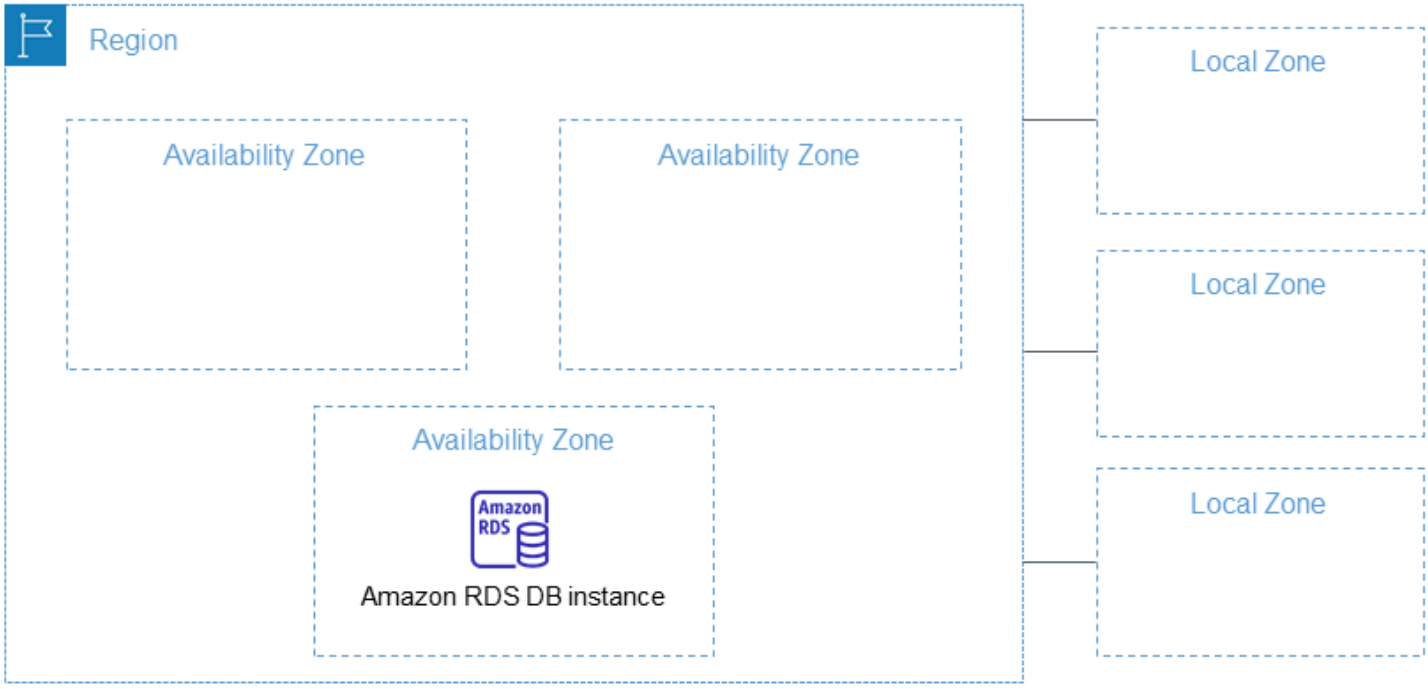
Régions, zones de disponibilité et zones locales

Les ressources de cloud computing Amazon sont hébergées dans plusieurs emplacements à travers le monde. Ces emplacements sont composés de AWS régions, de zones de disponibilité et de zones locales. Chaque AWS région constitue une zone géographique séparée. Chaque AWS région possède plusieurs emplacements isolés appelés zones de disponibilité.

> **Note**
>
> Pour plus d'informations sur la recherche des zones de disponibilité d'une AWS région, consultez la section Décrire vos zones de disponibilité dans la EC2 documentation Amazon.

Les zones locales vous permettent de placer des ressources, par exemple, de calcul et le stockage, dans plusieurs emplacements plus proches de vos utilisateurs finaux. Amazon RDS vous permet de placer des ressources telles que des instances de base de données, et des données dans plusieurs emplacements. Les ressources ne sont pas répliquées entre AWS les régions, sauf si vous le faites spécifiquement.

Amazon exploite state-of-the-art des centres de données hautement disponibles. Bien qu'elles soient rares, des pannes touchant la disponibilité des instances de base de données se trouvant au même emplacement peuvent se produire. Si vous hébergez toutes vos instances de base de données dans un seul emplacement touché par une panne de ce type, aucune de vos instances de base de données ne sera disponible.



Il est important de se rappeler que chaque AWS région est totalement indépendante. Toute activité Amazon RDS que vous lancez (par exemple, la création d'instances de base de données ou la liste des instances de base de données disponibles) s'exécute uniquement dans votre AWS région par défaut actuelle. La AWS région par défaut peut être modifiée dans la console ou en définissant la variable d'`AWS_DEFAULT_REGION`environnement. Il peut également être remplacé en utilisant le `--region` paramètre avec le AWS Command Line Interface ()AWS CLI. Pour de plus amples informations, veuillez consulter Configuration de l' AWS Command Line Interface, plus précisément les sections sur les variables d'environnement et les options de ligne de commande.

Amazon RDS prend en charge les AWS régions spéciales appelées AWS GovCloud (US). Elles sont conçues pour permettre aux agences gouvernementales et aux clients américains de déplacer des charges de travail plus sensibles vers le cloud. Les régions AWS GovCloud (US) aux exigences spécifiques du gouvernement américain en matière de réglementation et de conformité. Pour plus d'informations, voir Qu'est-ce que c'est AWS GovCloud (US) ?

Pour créer ou utiliser une instance de base de données Amazon RDS dans une AWS région spécifique, utilisez le point de terminaison de service régional correspondant.

AWS Régions

Chaque AWS région est conçue pour être isolée des autres AWS régions. Cette conception permet d'atteindre la plus grande tolérance aux pannes possible et une stabilité optimale.

Lorsque vous consultez vos ressources, seules les ressources liées à la AWS région que vous avez spécifiée s'affichent. Cela est dû au fait que les AWS régions sont isolées les unes des autres et que nous ne répliquons pas automatiquement les ressources entre AWS les régions.

Disponibilité dans les Régions

Le tableau suivant indique les AWS régions dans lesquelles Amazon RDS est actuellement disponible et le point de terminaison pour chaque région.

Nom de la région	Région	Point de terminaison	Protocole
US East (Ohio)	us-east-2	rds.us-east-2.amazonaws.com	HTTPS
		rds-fips.us-east-2.api.aws	HTTPS
		rds.us-east-2.api.aws	HTTPS
		rds-fips.us-east-2.amazonaws.com	HTTPS
US East (N. Virginia)	us-east-1	rds.us-east-1.amazonaws.com	HTTPS
		rds-fips.us-east-1.api.aws	HTTPS
		rds-fips.us-east-1.amazonaws.com	HTTPS
		rds.us-east-1.api.aws	HTTPS
USA Ouest (Californie du Nord)	us-west-1	rds.us-west-1.amazonaws.com	HTTPS
		rds.us-west-1.api.aws	HTTPS
		rds-fips.us-west-1.amazonaws.com	HTTPS
		rds-fips.us-west-1.api.aws	HTTPS
US West (Oregon)	us-west-2	rds.us-west-2.amazonaws.com	HTTPS
		rds-fips.us-west-2.amazonaws.com	HTTPS
		rds.us-west-2.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
		rds-fips.us-west-2.api.aws	HTTPS
Afrique (Le Cap)	af-south-1	rds.af-south-1.amazonaws.com	HTTPS
		rds.af-south-1.api.aws	HTTPS
Asie-Pacifique (Hong Kong)	ap-east-1	rds.ap-east-1.amazonaws.com	HTTPS
		rds.ap-east-1.api.aws	HTTPS
Asie-Pacifique (Hyderabad)	ap-south-2	rds.ap-south-2.amazonaws.com	HTTPS
		rds.ap-south-2.api.aws	HTTPS
Asie-Pacifique (Jakarta)	ap-southeast-3	rds.ap-southeast-3.amazonaws.com	HTTPS
		rds.ap-southeast-3.api.aws	HTTPS
Asie-Pacifique (Malaisie)	ap-southeast-5	rds.ap-southeast-5.amazonaws.com	HTTPS
Asie-Pacifique (Melbourne)	ap-southeast-4	rds.ap-southeast-4.amazonaws.com	HTTPS
		rds.ap-southeast-4.api.aws	HTTPS
Asia Pacific (Mumbai)	ap-south-1	rds.ap-south-1.amazonaws.com	HTTPS
		rds.ap-south-1.api.aws	HTTPS
Asie-Pacifique (Osaka)	ap-northeast-3	rds.ap-northeast-3.amazonaws.com	HTTPS
		rds.ap-northeast-3.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asia Pacific (Seoul)	ap-northeast-2	rds.ap-northeast-2.amazonaws.com	HTTPS
		rds.ap-northeast-2.api.aws	HTTPS
Asie-Pacifique (Singapour)	ap-southeast-1	rds.ap-southeast-1.amazonaws.com	HTTPS
		rds.ap-southeast-1.api.aws	HTTPS
Asia Pacific (Sydney)	ap-southeast-2	rds.ap-southeast-2.amazonaws.com	HTTPS
		rds.ap-southeast-2.api.aws	HTTPS
Asie-Pacifique (Taipei)	ap-est-2	rds.ap-east-2.amazonaws.com	HTTPS
Asie-Pacifique (Thaïlande)	ap-southeast-7	rds.ap-southeast-7.amazonaws.com	HTTPS
Asie-Pacifique (Tokyo)	ap-northeast-1	rds.ap-northeast-1.amazonaws.com	HTTPS
		rds.ap-northeast-1.api.aws	HTTPS
Canada (Central)	ca-central-1	rds.ca-central-1.amazonaws.com	HTTPS
		rds.ca-central-1.api.aws	HTTPS
		rds-fips.ca-central-1.api.aws	HTTPS
		rds-fips.ca-central-1.amazonaws.com	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Canada Ouest (Calgary)	ca-west-1	rds.ca-west-1.amazonaws.com rds-fips.ca-west-1.amazonaws.com	HTTPS HTTPS
Europe (Francfort)	eu-central-1	rds.eu-central-1.amazonaws.com rds.eu-central-1.api.aws	HTTPS HTTPS
Europe (Irlande)	eu-west-1	rds.eu-west-1.amazonaws.com rds.eu-west-1.api.aws	HTTPS HTTPS
Europe (Londres)	eu-west-2	rds.eu-west-2.amazonaws.com rds.eu-west-2.api.aws	HTTPS HTTPS
Europe (Milan)	eu-south-1	rds.eu-south-1.amazonaws.com rds.eu-south-1.api.aws	HTTPS HTTPS
Europe (Paris)	eu-west-3	rds.eu-west-3.amazonaws.com rds.eu-west-3.api.aws	HTTPS HTTPS
Europe (Espagne)	eu-south-2	rds.eu-south-2.amazonaws.com rds.eu-south-2.api.aws	HTTPS HTTPS
Europe (Stockholm)	eu-north-1	rds.eu-north-1.amazonaws.com rds.eu-north-1.api.aws	HTTPS HTTPS
Europe (Zurich)	eu-central-2	rds.eu-central-2.amazonaws.com rds.eu-central-2.api.aws	HTTPS HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Israël (Tel Aviv)	il-central-1	rds.il-central-1.amazonaws.com rds.il-central-1.api.aws	HTTPS HTTPS
Mexique (centre)	mx-central-1	rds.mx-central-1.amazonaws.com	HTTPS
Moyen-Orient (Bahreïn)	me-south-1	rds.me-south-1.amazonaws.com rds.me-south-1.api.aws	HTTPS HTTPS
Moyen-Orient (EAU)	me-central-1	rds.me-central-1.amazonaws.com rds.me-central-1.api.aws	HTTPS HTTPS
Amérique du Sud (São Paulo)	sa-east-1	rds.sa-east-1.amazonaws.com rds.sa-east-1.api.aws	HTTPS HTTPS
AWS GovCloud (USA Est)	us-gov-east-1	rds.us-gov-east-1.amazonaws.com rds.us-gov-east-1.api.aws	HTTPS HTTPS
AWS GovCloud (US-Ouest)	us-gov-west-1	rds.us-gov-west-1.amazonaws.com rds.us-gov-west-1.api.aws	HTTPS HTTPS

Si vous ne spécifiez pas explicitement de point de terminaison, le point de terminaison USA Ouest (Oregon) est la valeur par défaut.

Lorsque vous travaillez avec une instance de base de données à l'aide des opérations d'API AWS CLI or, assurez-vous de spécifier son point de terminaison régional.

Zones de disponibilité

Lorsque vous créez une instance de base de données, vous pouvez sélectionner une zone de disponibilité ou demander à Amazon RDS de le faire pour vous. Une zone de disponibilité est représentée par un code de AWS région suivi d'une lettre d'identification (par exemple,`us-east-1a`).

Utilisez la EC2 commande describe-availability-zonesAmazon comme suit pour décrire les zones de disponibilité activées pour votre compte dans la région spécifiée.

```
aws ec2 describe-availability-zones --region region-name
```

Par exemple, pour décrire les zones de disponibilité de la région USA Est (Virginie du Nord) (us-east-1) qui sont activées pour votre compte, exécutez la commande suivante :

```
aws ec2 describe-availability-zones --region us-east-1
```

Dans un déploiement de bases de données multi-AZ, vous ne pouvez pas sélectionner les zones de disponibilité des instances de base de données principales et secondaires. Amazon RDS les choisit pour vous au hasard. Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

> **Note**
>
> La sélection aléatoire des zones de disponibilité par RDS ne garantit pas une distribution uniforme des instances de base de données entre les zones de disponibilité dans un seul compte ou un groupe de sous-réseaux de bases de données. Vous pouvez demander une AZ spécifique lorsque vous créez ou modifiez une instance AZ unique, et vous pouvez utiliser des groupes de sous-réseaux de bases de données plus spécifiques pour les instances multi-AZ. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS et Modification d'une RDS instance de base de données Amazon.

Zones locales

Une zone locale est une extension d'une AWS région géographiquement proche de vos utilisateurs. Vous pouvez étendre n'importe quel VPC de la région AWS parente dans des zones locales. Pour ce faire, créez un nouveau sous-réseau et affectez-le à la zone locale AWS . Lorsque vous créez un

sous-réseau dans une zone locale, votre VPC est étendu à cette zone locale. Le sous-réseau de la zone locale fonctionne de la même manière que les autres sous-réseaux de votre VPC.

Lorsque vous créez une instance de base de données, vous pouvez choisir un sous-réseau dans une zone locale. Les zones locales ont leurs propres connexions à Internet et prennent en charge AWS Direct Connect. Ainsi, les ressources créées dans une zone locale peuvent servir les utilisateurs locaux avec des communications à très faible latence. Pour de plus amples informations, veuillez consulter AWS Local Zones.

Une zone locale est représentée par un code de AWS région suivi d'un identifiant indiquant l'emplacement, par exemple`us-west-2-lax-1a`.

> **Note**
>
> Une zone locale ne peut pas être incluse dans un déploiement multi-AZ.

Pour utiliser une zone locale

1. Activez la zone locale dans la EC2 console Amazon.

   Pour plus d'informations, consultez la section Activation des zones locales dans le guide de EC2 l'utilisateur Amazon.
2. Créez un sous-réseau dans la zone locale.

   Pour de plus amples informations, veuillez consulter Création d'un sous-réseau dans votre VPC dans le Amazon VPC Guide de l'utilisateur.
3. Créez un groupe de sous-réseaux de base de données dans la zone locale.

   Lorsque vous créez un groupe de sous-réseaux de base de données, choisissez le groupe de zone de disponibilité pour la zone locale.

   Pour plus d'informations, consultez Création d'un(e) instance de base de données dans un VPC.
4. Créez une instance de base de données qui utilise le groupe de sous-réseaux de base de données dans la zone locale.

   Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

Important

Actuellement, la seule zone AWS locale dans laquelle Amazon RDS est disponible est Los Angeles, dans la région de l'ouest des États-Unis (Oregon).

Fonctionnalités prises en charge dans Amazon RDS by Région AWS et dans le moteur de base de données

Support pour les RDS fonctionnalités et options d'Amazon varie selon Régions AWS les versions spécifiques de chaque moteur de base de données. Pour identifier la prise en charge et la disponibilité des versions du moteur de RDS base de données dans une version donnée Région AWS, vous pouvez utiliser les sections suivantes.

Les RDS fonctionnalités d'Amazon sont différentes des fonctionnalités et options natives du moteur. Pour obtenir plus d'informations sur les fonctionnalités et les options natives du moteur, consultez la section Engine-native features. (Fonctionnalités natives du moteur.)

Régions et moteurs de base de données pris en charge

- Conventions de tableau
- Référence rapide des fonctionnalités
- Régions et moteurs de base de données pris en charge pour les déploiements Amazon RDS Blue/Green
- Régions et moteurs de base de données pris en charge pour les sauvegardes automatisées entre régions dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour les répliques de lecture entre régions dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour les flux d'activité des bases de données sur Amazon RDS
- Régions et moteurs de base de données pris en charge pour le mode double pile dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour l'exportation d'instantanés vers S3 dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour l'authentification de base de données IAM dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour les clusters de bases de données multi-AZ dans Amazon RDS

- Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS
- Régions et moteurs de base de données pris en charge pour RDS Custom
- Régions et moteurs de base de données pris en charge pour Amazon RDS Proxy
- Régions et moteurs de base de données pris en charge pour l'intégration de Secrets Manager à Amazon RDS
- Régions et moteurs de base de données pris en charge pour les intégrations Amazon RDS Zero-ETL
- Fonctionnalités natives du moteur dans Amazon RDS

Conventions de tableau

Les tables dans les sections utilisent ces modèles pour spécifier les numéros de version et le niveau de disponibilité :

- Version x.y : seule cette version spécifique est disponible.
- Version x.y et ultérieures : la version spécifiée et toutes les versions mineures ultérieures de cette version majeure sont prises en charge. Par exemple, « version 10.11 et ultérieures » signifie que les versions 10.11, 10.11.1 et 10.12 sont disponibles.

Référence rapide des fonctionnalités

Le tableau de référence rapide suivant répertorie chaque fonctionnalité et le moteur de RDS base de données disponibles. La disponibilité des régions et des versions spécifiques apparaît dans les sections de fonctionnalités ultérieures.

Fonctionnalité	RDSpour Db2	RDS for MariaDB	RDSpour My SQL	RDS for Oracle	RDSpour Postgre SQL	RDSpour SQL serveur
Déploiements bleu/vert	Non disponible	Disponible	Disponible	Non disponible	Disponible	Non disponible

Fonctionnalité	RDS pour Db2	RDS for MariaDB	RDS pour My SQL	RDS for Oracle	RDS pour Postgre SQL	RDS pour SQL serveur
Sauvegardes automatiques interrégionales	Disponible	Disponible	Disponible	Disponible	Disponible	Disponible
Réplication en lecture entre Régions	Non disponible	Disponible	Disponible	Disponible	Disponible	Disponible
Flux d'activité de base de données	Non disponible	Non disponible	Non disponible	Disponible	Non disponible	Disponible
Mode double pile	Non disponible	Disponible	Disponible	Disponible	Disponible	Disponible
Exportation d'instantanés vers Amazon S3	Non disponible	Disponible	Disponible	Non disponible	Disponible	Non disponible

Fonctionnalité	RDS pour Db2	RDS for MariaDB	RDS pour My SQL	RDS for Oracle	RDS pour Postgre SQL	RDS pour SQL serveur
AWS Identity and Access Manag t (IAM) authentication de base de donné	Non disponible	Disponible	Disponible	Non disponible	Disponible	Non disponible
Authentication Kerber	Disponible	Non disponible	Disponible	Disponible	Disponible	Disponible
Cluste de base de donné multi-AZ	Non disponible	Non disponible	Disponible	Non disponible	Disponible	Non disponible
Perfor ce Insight	Non disponible	Disponible	Disponible	Disponible	Disponible	Disponible
RDSP nalisé	Non disponible	Non disponible	Non disponible	Disponible	Non disponible	Disponible

Fonctionnalité	RDSpour Db2	RDS for MariaDB	RDSpour My SQL	RDS for Oracle	RDSpour Postgre SQL	RDSpour SQL serveur
RDSP	Non disponible	Disponible	Disponible	Non disponible	Disponible	Disponible
Intégration de Secret Manag	Disponible	Disponible	Disponible	Disponible	Disponible	Disponible

Régions et moteurs de base de données pris en charge pour les déploiements Amazon RDS Blue/Green

Lors blue/green deployment copies a production database environment in a separate, synchronized staging environment. By using Amazon RDS Blue/Green des déploiements, vous pouvez apporter des modifications à la base de données dans l'environnement intermédiaire sans affecter l'environnement de production. Par exemple, vous pouvez mettre à niveau la version majeure ou mineure du moteur de base de données, modifier les paramètres de la base de données ou apporter des changements au schéma dans l'environnement intermédiaire. Lorsque vous êtes prêt, vous pouvez promouvoir l'environnement intermédiaire en tant que nouvel environnement de base de données de production. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Les déploiements bleu/vert sont pris en charge dans tous les cas. Régions AWS

Les déploiements bleu/vert ne sont pas pris en charge avec les moteurs suivants :

- RDSpour Db2
- RDSpour le SQL serveur
- RDS for Oracle

Rubriques

- Déploiements bleu/vert avec pour MariaDB RDS

- Déploiements bleu/vert avec for My RDS SQL
- Déploiements bleu/vert avec for Postgre RDS SQL

Déploiements bleu/vert avec pour MariaDB RDS

RDSPour MariaDB, les déploiements bleu/vert sont pris en charge pour les versions suivantes :

- RDSpour MariaDB 11.4 (toutes les versions disponibles)
- RDSpour MariaDB 10.2 et versions 10 supérieures

Déploiements bleu/vert avec for My RDS SQL

Pour RDS for MySQL, les déploiements bleu/vert sont pris en charge pour les versions suivantes :

- RDSpour My SQL 8.4 (toutes les versions disponibles)
- RDSpour My SQL 8.0 (toutes les versions disponibles)
- RDSpour My SQL 5.7 (toutes les versions disponibles)

Déploiements bleu/vert avec for Postgre RDS SQL

RDSPour PostgreSQL, les déploiements bleu/vert sont pris en charge pour la version 11.1 et toutes les versions majeures et mineures supérieures.

> **Note**
>
> Dans certaines conditions, RDS Postgre SQL utilise la réplication logique au lieu de la réplication physique pour synchroniser l'environnement vert avec l'environnement bleu. Pour de plus amples informations, veuillez consulter the section called “Méthodes de SQL réplication Postgre”.

Régions et moteurs de base de données pris en charge pour les sauvegardes automatisées entre régions dans Amazon RDS

En utilisant la réplication de sauvegarde dans Amazon RDS, vous pouvez configurer votre instance de base de données RDS pour répliquer les instantanés et les journaux de transactions vers une région de destination. Lorsque la réplication des sauvegardes est configurée sur une instance

de base de données, RDS lance une copie inter-région de tous les instantanés et journaux de transactions dès qu'ils sont prêts.. Pour de plus amples informations, veuillez consulter Réplication des sauvegardes automatisées vers une autre Région AWS.

Pour plus d'informations sur Régions AWS les endroits où la réplication de sauvegarde est disponible, consultezRéplication des sauvegardes automatisées vers une autre Région AWS.

Rubriques

- Backup et réplication avec RDS pour DB2
- Réplication des sauvegardes avec RDS pour MariaDB
- Réplication des sauvegardes avec RDS pour MySQL
- Réplication des sauvegardes avec RDS for Oracle
- Réplication des sauvegardes avec RDS for PostgreSQL
- Réplication des sauvegardes avec RDS for SQL Server

Backup et réplication avec RDS pour DB2

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS pour DB2.

Réplication des sauvegardes avec RDS pour MariaDB

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS pour MariaDB.

Réplication des sauvegardes avec RDS pour MySQL

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS pour MySQL.

Réplication des sauvegardes avec RDS for Oracle

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS for Oracle.

Réplication des sauvegardes avec RDS for PostgreSQL

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS for PostgreSQL.

Réplication des sauvegardes avec RDS for SQL Server

Amazon RDS prend en charge la réplication de sauvegarde pour toutes les versions actuellement disponibles de RDS for SQL Server.

Régions et moteurs de base de données pris en charge pour les répliques de lecture entre régions dans Amazon RDS

En utilisant des réplicas en lecture entre régions dans Amazon RDS, vous pouvez créer un réplica de lecture MariaDB, MySQL, Oracle, PostgreSQL ou SQL Server dans une région différente de l'instance de base de données source. Pour plus d'informations sur les réplicas en lecture entre régions, y compris sur les considérations liées aux régions source et de destination, consultez Création d'un réplica en lecture dans une autre Région AWS.

Les répliques de lecture entre régions ne sont pas disponibles pour les moteurs suivants :

- RDS pour Db2

Rubriques

- Réplicas en lecture entre régions avec RDS for MariaDB
- Réplicas en lecture entre régions avec RDS for MySQL
- Réplicas en lecture entre régions avec RDS for Oracle
- Réplicas en lecture entre régions avec RDS for PostgreSQL
- Réplicas en lecture entre régions avec RDS for SQL Server

Réplicas en lecture entre régions avec RDS for MariaDB

Des réplicas en lecture entre régions avec RDS for MariaDB sont disponibles dans toutes les régions pour les versions suivantes :

- RDS pour MariaDB 11.4 (toutes les versions disponibles)
- RDS for MariaDB 10.11 (toutes les versions disponibles)
- RDS for MariaDB 10.6 (toutes versions disponibles)
- RDS for MariaDB 10.5 (toutes versions disponibles)
- RDS for MariaDB 10.4 (toutes versions disponibles)

Réplicas en lecture entre régions avec RDS for MySQL

Des réplicas en lecture entre régions avec RDS for MySQL sont disponibles dans toutes les régions pour les versions suivantes :

- RDS pour MySQL 8.4 (toutes les versions disponibles)
- RDS for MySQL 8.0 (toutes versions disponibles)
- RDS for MySQL 5.7 (toutes versions disponibles)

Réplicas en lecture entre régions avec RDS for Oracle

Les répliques de lecture interrégionales pour RDS pour Oracle sont disponibles dans toutes les Régions AWS versions de base de données prises en charge à l'aide de l'édition Enterprise. Les répliques ne sont prises en charge que dans la configuration non locataire CDBs et dans la configuration à locataire unique de l'architecture CDB. Les répliques de lecture entre régions ne sont pas prises en charge dans la configuration multi-locataires de l'architecture CDB.

Pour obtenir plus d'informations sur les exigences supplémentaires relatives aux réplicas en lecture inter-régions avec RDS for Oracle, consultez Exigences et considérations relatives aux réplicas RDS pour Oracle.

Réplicas en lecture entre régions avec RDS for PostgreSQL

Des réplicas en lecture entre régions avec RDS for PostgreSQL sont disponibles dans toutes les régions pour les versions suivantes :

- RDS pour PostgreSQL 17 (toutes les versions disponibles)
- RDS pour PostgreSQL 16 (toutes les versions disponibles)
- RDS for PostgreSQL 15 (toutes versions disponibles)
- RDS for PostgreSQL 14 (toutes versions disponibles)
- RDS for PostgreSQL 13 (toutes versions disponibles)
- RDS for PostgreSQL 12 (toutes versions disponibles)
- RDS for PostgreSQL 11 (toutes versions disponibles)
- RDS for PostgreSQL 10 (toutes versions disponibles)

Réplicas en lecture entre régions avec RDS for SQL Server

Des réplicas en lecture entre régions avec RDS for SQL Server sont disponibles dans toutes les régions à l'exception des suivantes :

- Afrique (Le Cap)
- Asie-Pacifique (Hong Kong)
- Asie-Pacifique (Hyderabad)
- Asie-Pacifique (Jakarta)
- Asie-Pacifique (Melbourne)
- Canada-Ouest (Calgary)
- Europe (Milan)
- Europe (Espagne)
- Europe (Zurich)
- Israël (Tel Aviv)
- Moyen-Orient (Bahreïn)
- Moyen-Orient (EAU)

Des réplicas en lecture entre régions avec RDS for SQL Server sont disponibles pour les versions suivantes utilisant Microsoft SQL Server Enterprise Edition :

- RDS pour SQL Server 2022
- RDS for SQL Server 2019 (version 15.00.4073.23 et versions ultérieures)
- RDS for SQL Server 2017 (versions 14.00.3281.6 et ultérieures)
- RDS for SQL Server 2016 (versions 13.00.6300.2 et ultérieures)

Régions et moteurs de base de données pris en charge pour les flux d'activité des bases de données sur Amazon RDS

En utilisant les flux d'activité des bases de données dans AmazonRDS, vous pouvez surveiller et définir des alarmes pour l'audit des activités dans votre base de données Oracle et dans votre base de données SQL Server. Pour de plus amples informations, veuillez consulter Présentation des flux d'activité de base de données.

Les flux d'activité des bases de données ne sont pas disponibles avec les moteurs suivants :

- RDSpour Db2
- RDS for MariaDB
- RDSpour My SQL
- RDSpour Postgre SQL

Rubriques

- Flux d'activité de base de données avec RDS for Oracle
- Flux d'activité de base de données avec RDS for SQL Server

Flux d'activité de base de données avec RDS for Oracle

Les régions et versions du moteur suivantes sont disponibles pour les flux d'activité des bases RDS de données avec Oracle.

Pour plus d'informations sur les exigences supplémentaires relatives aux flux d'activité des bases de données avec RDS for Oracle, consultezPrésentation des flux d'activité de base de données.

Région	RDSpour Oracle 21c	RDSpour Oracle 19c
USA Est (Virginie du Nord)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
USA Est (Ohio)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
USA Ouest (Californie du Nord)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2

Région	RDSpour Oracle 21c	RDSpour Oracle 19c
USA Ouest (Oregon)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Afrique (Le Cap)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Hong Kong)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Hyderabad)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Jakarta)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Malaisie)	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2

Région	RDSpour Oracle 21c	RDSpour Oracle 19c
Asie-Pacifique (Osaka)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Séoul)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Singapour)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Sydney)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Asie-Pacifique (Tokyo)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Canada (Centre)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Canada Ouest (Calgary)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Chine (Beijing)	Non disponible	Non disponible

Région	RDSpour Oracle 21c	RDSpour Oracle 19c
Chine (Ningxia)	Non disponible	Non disponible
Europe (Francfort)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Irlande)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Londres)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Milan)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Paris)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Espagne)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Europe (Stockholm)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2

Région	RDSpour Oracle 21c	RDSpour Oracle 19c
Europe (Zurich)	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Moyen-Orient (UAE)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
Amérique du Sud (São Paulo)	Non disponible	Oracle Database 19.0.0.0.ru-2019-07.rur-2019-07.r1 et versions ultérieures, avec Enterprise Edition (EE) ou Standard Edition 2 () SE2
AWS GovCloud (USA Est)	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible

Flux d'activité de base de données avec RDS for SQL Server

Les régions et versions du moteur suivantes sont disponibles pour les flux d'activité de base de données avec RDS for SQL Server.

Pour plus d'informations sur les exigences supplémentaires relatives aux flux d'activité de base de données avec RDS for SQL Server, consultezPrésentation des flux d'activité de base de données.

Région	RDSpour SQL Server 2019	RDSpour SQL Server 2017	RDSpour SQL Server 2016
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDSpour SQL Server 2019	RDSpour SQL Server 2017	RDSpour SQL Server 2016
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Non disponible	Non disponible	Non disponible
Chine (Ningxia)	Non disponible	Non disponible	Non disponible
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDSpour SQL Server 2019	RDSpour SQL Server 2017	RDSpour SQL Server 2016
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (UAE)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible

Régions et moteurs de base de données pris en charge pour le mode double pile dans Amazon RDS

En utilisant le mode double pile dans RDS, les ressources peuvent communiquer avec une instance de base de données via le protocole Internet version 4 (IPv4), le protocole Internet version 6 (IPv6) ou les deux. Pour de plus amples informations, veuillez consulter Mode double pile.

Rubriques

- Mode double pile avec RDS pour DB2
- Mode double pile avec RDS for MariaDB

- Mode double pile avec RDS for MySQL
- Mode double pile avec RDS for Oracle
- Mode double pile avec RDS for PostgreSQL
- Mode double pile avec RDS for SQL Server

IPv6 n'est pris en charge pour aucun moteur de base de données Amazon RDS dans les domaines suivants : Régions AWS

- Asie-Pacifique (Hyderabad)
- Moyen-Orient (EAU)
- Asie-Pacifique (Malaisie)
- Asie-Pacifique (Melbourne)
- Canada-Ouest (Calgary)
- Europe (Zurich)
- Israël (Tel Aviv)

Mode double pile avec RDS pour DB2

Les régions et versions de moteur suivantes sont disponibles pour le mode double pile avec RDS pour DB2.

Région	RDS pour DB2 11.5				
USA Est (Virginie du Nord)	Toutes versions disponibles				
USA Est (Ohio)	Toutes versions disponibles				
USA Ouest (Californie du Nord)	Toutes versions disponibles				

Région	RDS pour DB2 11.5				
USA Ouest (Oregon)	Toutes versions disponibles				
Afrique (Le Cap)	Toutes versions disponibles				
Asie-Pacifique (Hong Kong)	Toutes versions disponibles				
Asie-Pacifique (Hyderabad)	Non disponible				
Asie-Pacifique (Jakarta)	Toutes versions disponibles				
Asie-Pacifique (Malaisie)	Non disponible				
Asie-Pacifique (Melbourne)	Non disponible				
Asie-Pacifique (Mumbai)	Toutes versions disponibles				
Asie-Pacifique (Osaka)	Toutes versions disponibles				

Région	RDS pour DB2 11.5				
Asie-Pacifique (Séoul)	Toutes versions disponibles				
Asie-Pacifique (Singapour)	Toutes versions disponibles				
Asie-Pacifique (Sydney)	Toutes versions disponibles				
Asie-Pacifique (Tokyo)	Toutes versions disponibles				
Canada (Centre)	Toutes versions disponibles				
Canada-Ouest (Calgary)	Non disponible				
Chine (Beijing)	Non disponible				
Chine (Ningxia)	Non disponible				
Europe (Francfort)	Toutes versions disponibles				
Europe (Irlande)	Toutes versions disponibles				

Région	RDS pour DB2 11.5				
Europe (Londres)	Toutes versions disponibles				
Europe (Milan)	Toutes versions disponibles				
Europe (Paris)	Toutes versions disponibles				
Europe (Espagne)	Non disponible				
Europe (Stockholm)	Toutes versions disponibles				
Europe (Zurich)	Non disponible				
Israël (Tel Aviv)	Non disponible				
Moyen-Orient (Bahreïn)	Toutes versions disponibles				
Moyen-Orient (EAU)	Non disponible				
Amérique du Sud (São Paulo)	Toutes versions disponibles				

Région	RDS pour DB2 11.5				
AWS GovCloud (USA Est)	Non disponible				
AWS GovCloud (US-Ouest)	Non disponible				

Mode double pile avec RDS for MariaDB

Les régions et les versions de moteur suivantes sont disponibles pour le mode double pile avec RDS for MariaDB.

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Mode double pile avec RDS for MySQL

Les régions et les versions de moteur suivantes sont disponibles pour le mode double pile avec RDS for MySQL.

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Mode double pile avec RDS for Oracle

Les régions et les versions de moteur suivantes sont disponibles pour le mode double pile avec RDS for Oracle.

Région	RDS for Oracle 21c	RDS for Oracle 19c
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible

Région	RDS for Oracle 21c	RDS for Oracle 19c
Asie-Pacifique (Melbourne)	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Non disponible	Non disponible
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible

Région	RDS for Oracle 21c	RDS for Oracle 19c
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Non disponible	Non disponible
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles

Mode double pile avec RDS for PostgreSQL

Les régions et les versions de moteur suivantes sont disponibles pour le mode double pile avec RDS for PostgreSQL.

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Mode double pile avec RDS for SQL Server

Les régions et les versions de moteur suivantes sont disponibles pour le mode double pile avec RDS for SQL Server.

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Régions et moteurs de base de données pris en charge pour l'exportation d'instantanés vers S3 dans Amazon RDS

Vous pouvez exporter des données d'instantanés de bases de données RDS vers un compartiment Amazon S3. Vous pouvez exporter tous les types d'instantanés de bases de données, à savoir, les instantanés manuels, les instantanés système automatisés et les instantanés créés par AWS Backup. Une fois les données exportées, vous pouvez les analyser directement via des outils tels que Amazon Athena ou Amazon Redshift Spectrum. Pour de plus amples informations, veuillez consulter Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS.

L'exportation de snapshots vers S3 n'est pas disponible pour les moteurs suivants :

- RDS pour Db2
- RDS for Oracle
- RDS for SQL Server

Rubriques

- Exporter des instantanés vers S3 avec RDS for MariaDB
- Exporter des instantanés vers S3 avec RDS for MySQL
- Exporter des instantanés vers S3 avec RDS pour PostgreSQL

Exporter des instantanés vers S3 avec RDS for MariaDB

Les régions et les versions de moteur suivantes sont disponibles pour l'export d'instantanés vers S3 avec RDS for MariaDB.

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Exporter des instantanés vers S3 avec RDS for MySQL

Les régions et versions de moteur suivantes sont disponibles pour exporter des instantanés vers S3 avec RDS for MySQL.

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (dans le cadre du support étendu RDS)
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible

Exporter des instantanés vers S3 avec RDS pour PostgreSQL

Les régions et les versions de moteur suivantes sont disponibles pour l'export d'instantanés vers S3 avec RDS for PostgreSQL.

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Virginie	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
du Nord)								
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Régions et moteurs de base de données pris en charge pour l'authentification de base de données IAM dans Amazon RDS

En utilisant l'authentification de base de données IAM dans Amazon RDS, vous pouvez vous connecter sans mot de passe lorsque vous vous connectez à une instance de base de données. En revanche, un jeton d'authentification est nécessaire. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

L'authentification de la base de données IAM n'est pas disponible avec les moteurs suivants :

- RDS pour Db2
- RDS for Oracle
- RDS for SQL Server

Rubriques

- Authentification des bases de données IAM avec RDS for MariaDB

- Authentification des bases de données IAM avec RDS pour MySQL
- Authentification des bases de données IAM avec RDS pour PostgreSQL

Authentification des bases de données IAM avec RDS for MariaDB

Les régions et les versions de moteur suivantes sont disponibles pour l'authentification des bases de données IAM avec RDS for MariaDB.

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Europe (Espagne)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Zurich)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible

Authentification des bases de données IAM avec RDS pour MySQL

L'authentification de bases de données IAM avec RDS for MySQL est disponible dans toutes les régions pour les versions suivantes :

- RDS pour MySQL 8.4 — Toutes les versions disponibles
- RDS for MySQL 8.0 : toutes versions disponibles
- RDS for MySQL 5.7 : toutes versions disponibles

Authentification des bases de données IAM avec RDS pour PostgreSQL

L'authentification de bases de données IAM avec RDS for PostgreSQL est disponible dans toutes les régions pour les versions suivantes :

- RDS pour PostgreSQL 17 — Toutes les versions disponibles
- RDS pour PostgreSQL 16 — Toutes les versions disponibles
- RDS for PostgreSQL 15 : toutes versions disponibles
- RDS for PostgreSQL 14 : toutes versions disponibles
- RDS for PostgreSQL 13 : toutes versions disponibles
- RDS for PostgreSQL 12 : toutes versions disponibles
- RDS for PostgreSQL 11 : toutes versions disponibles
- RDS for PostgreSQL 10 : toutes versions disponibles

Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS

En utilisant l'authentification Kerberos dans Amazon RDS, vous pouvez prendre en charge l'authentification externe des utilisateurs de la base de données en utilisant Kerberos et Microsoft Active Directory. L’utilisation de Kerberos et Active Directory procure les avantages d'une authentification unique et centralisée des utilisateurs de bases de données.

L'authentification Kerberos n'est pas disponible avec les moteurs suivants :

- RDS for MariaDB

Bien que la plupart des AWS régions soient actives par défaut pour votre AWS compte, certaines régions ne sont activées que lorsque vous les sélectionnez manuellement. Ces régions sont appelées « régions optionnelles ». En revanche, les régions actives par défaut, dès la création de votre AWS compte, sont appelées régions commerciales, ou simplement régions. Pour les régions optionnelles, vous devez utiliser un principal de service régionalisé du formulaire. `directoryservice.rds.region_name.amazonaws.com` Par exemple, pour l'Afrique (Cape Town), vous devez ajouter le principal de service `directoryservice.rds.af-south-1.amazonaws.com` à votre politique de confiance. Pour de plus amples informations, veuillez consulter Authentification Kerberos.

Rubriques

- Authentification Kerberos avec RDS pour DB2
- Authentification Kerberos avec RDS pour MySQL
- Authentification Kerberos avec RDS for Oracle
- Authentification Kerberos avec RDS pour PostgreSQL
- Authentification Kerberos avec RDS for SQL Server

Authentification Kerberos avec RDS pour DB2

Les régions et versions de moteur suivantes sont disponibles pour l'authentification Kerberos avec RDS pour Db2.

Région	RDS pour DB2 11.5
USA Est (Virginie du Nord)	Toutes les versions
USA Est (Ohio)	Toutes les versions
USA Ouest (Californie du Nord)	Toutes les versions
USA Ouest (Oregon)	Toutes les versions
Afrique (Le Cap)	Toutes les versions
Asie-Pacifique (Hong Kong)	Non disponible
Asie-Pacifique (Hyderabad)	Toutes les versions

Région	RDS pour DB2 11.5
Asie-Pacifique (Jakarta)	Toutes les versions
Asie-Pacifique (Malaisie)	Non disponible
Asie-Pacifique (Melbourne)	Toutes les versions
Asie-Pacifique (Mumbai)	Toutes les versions
Asie-Pacifique (Osaka)	Non disponible
Asie-Pacifique (Séoul)	Toutes les versions
Asie-Pacifique (Singapour)	Toutes les versions
Asie-Pacifique (Sydney)	Toutes les versions
Asie-Pacifique (Tokyo)	Toutes les versions
Canada (Centre)	Toutes les versions
Canada-Ouest (Calgary)	Non disponible
Chine (Beijing)	Non disponible
Chine (Ningxia)	Non disponible
Europe (Francfort)	Toutes les versions
Europe (Irlande)	Toutes les versions
Europe (Londres)	Toutes les versions
Europe (Milan)	Toutes les versions
Europe (Paris)	Non disponible
Europe (Espagne)	Toutes les versions
Europe (Stockholm)	Toutes les versions

Région	RDS pour DB2 11.5
Europe (Zurich)	Toutes les versions
Israël (Tel Aviv)	Toutes les versions
Moyen-Orient (Bahreïn)	Toutes les versions
Moyen-Orient (EAU)	Toutes les versions
Amérique du Sud (São Paulo)	Toutes les versions
AWS GovCloud (USA Est)	Non disponible
AWS GovCloud (US-Ouest)	Non disponible

Authentification Kerberos avec RDS pour MySQL

Les régions et les versions de moteur suivantes sont disponibles pour l'authentification Kerberos avec RDS for MySQL.

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (sous RDS Extended Support)
USA Est (Virginie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions
USA Est (Ohio)	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Californie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Oregon)	Toutes les versions	Toutes les versions	Toutes les versions
Afrique (Le Cap)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Hong Kong)	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (sous RDS Extended Support)
Asie-Pacifique (Hyderabad)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Jakarta)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Mumbai)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Osaka)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Séoul)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Singapour)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Sydney)	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Tokyo)	Toutes les versions	Toutes les versions	Toutes les versions
Canada (Centre)	Toutes les versions	Toutes les versions	Toutes les versions
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS pour MySQL 5.7 (sous RDS Extended Support)
Chine (Ningxia)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Francfort)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Irlande)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Londres)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Milan)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Paris)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Espagne)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Stockholm)	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Zurich)	Toutes les versions	Toutes les versions	Toutes les versions
Israël (Tel Aviv)	Toutes les versions	Toutes les versions	Toutes les versions
Moyen-Orient (Bahreïn)	Toutes les versions	Toutes les versions	Toutes les versions
Moyen-Orient (EAU)	Toutes les versions	Toutes les versions	Toutes les versions
Amérique du Sud (São Paulo)	Toutes les versions	Toutes les versions	Toutes les versions
AWS GovCloud (USA Est)	Toutes les versions	Toutes les versions	Toutes les versions
AWS GovCloud (US-Ouest)	Toutes les versions	Toutes les versions	Toutes les versions

Authentification Kerberos avec RDS for Oracle

Les régions et les versions de moteur suivantes sont disponibles pour l'authentification Kerberos avec RDS for Oracle.

Région	RDS for Oracle 21c	RDS for Oracle 19c
USA Est (Virginie du Nord)	Toutes les versions	Toutes les versions
USA Est (Ohio)	Toutes les versions	Toutes les versions
USA Ouest (Californie du Nord)	Toutes les versions	Toutes les versions
USA Ouest (Oregon)	Toutes les versions	Toutes les versions
Afrique (Le Cap) (région optionnelle)	Toutes les versions	Toutes les versions
Asie-Pacifique (Hong Kong) (région optionnelle)	Toutes les versions	Toutes les versions
Asie-Pacifique (Hyderabad) (région optionnelle)	Toutes les versions	Toutes les versions
Asie-Pacifique (Jakarta) (région optionnelle)	Toutes les versions	Toutes les versions
Asie-Pacifique (Malaisie)	Non disponible	Non disponible
Asie-Pacifique (Melbourne) (région optionnelle)	Toutes les versions	Toutes les versions
Asie-Pacifique (Mumbai)	Toutes les versions	Toutes les versions
Asie-Pacifique (Osaka)	Non disponible	Non disponible
Asie-Pacifique (Séoul)	Toutes les versions	Toutes les versions
Asie-Pacifique (Singapour)	Toutes les versions	Toutes les versions

Région	RDS for Oracle 21c	RDS for Oracle 19c
Asie-Pacifique (Sydney)	Toutes les versions	Toutes les versions
Asie-Pacifique (Tokyo)	Toutes les versions	Toutes les versions
Canada (Centre)	Toutes les versions	Toutes les versions
Canada-Ouest (Calgary)	Non disponible	Non disponible
Chine (Beijing)	Non disponible	Non disponible
Chine (Ningxia)	Non disponible	Non disponible
Europe (Francfort)	Toutes les versions	Toutes les versions
Europe (Irlande)	Toutes les versions	Toutes les versions
Europe (Londres)	Toutes les versions	Toutes les versions
Europe (Milan) (région optionnelle)	Toutes les versions	Toutes les versions
Europe (Paris)	Non disponible	Non disponible
Europe (Espagne) (région optionnelle)	Toutes les versions	Toutes les versions
Europe (Stockholm)	Toutes les versions	Toutes les versions
Europe (Zurich) (région optionnelle)	Toutes les versions	Toutes les versions
Israël (Tel Aviv) (région optionnelle)	Toutes les versions	Toutes les versions
Moyen-Orient (Bahreïn) (région optionnelle)	Toutes les versions	Toutes les versions
Moyen-Orient (Émirats arabes unis) (région optionnelle)	Toutes les versions	Toutes les versions

Région	RDS for Oracle 21c	RDS for Oracle 19c
Amérique du Sud (São Paulo)	Toutes les versions	Toutes les versions
AWS GovCloud (USA Est)	Toutes les versions	Toutes les versions
AWS GovCloud (US-Ouest)	Toutes les versions	Toutes les versions

Authentification Kerberos avec RDS pour PostgreSQL

Les régions et les versions de moteur suivantes sont disponibles pour l'authentification Kerberos avec RDS for PostgreSQL.

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Est (Virginie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Est (Ohio)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Californie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Oregon)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Afrique (Le Cap)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Hong Kong)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Hyderabad)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Jakarta)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Mumbai)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Osaka)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Séoul)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Singapour)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Sydney)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Tokyo)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Canada (Centre)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Chine (Ningxia)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Francfort)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Irlande)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Londres)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Milan)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Europe (Paris)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Espagne)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Stockholm)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Zurich)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Israël (Tel Aviv)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Moyen-Orient (Bahreïn)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Moyen-Orient (EAU)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Amérique du Sud (São Paulo)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
AWS GovCloud (USA Est)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
AWS GovCloud (US-Ouest)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Authentification Kerberos avec RDS for SQL Server

Les régions et les versions de moteur suivantes sont disponibles pour l'authentification Kerberos avec RDS for SQL Server.

Région	RDS pour SQL Server 2022	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
USA Est (Virginie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Est (Ohio)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Californie du Nord)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
USA Ouest (Oregon)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour SQL Server 2022	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Afrique (Le Cap)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Hong Kong)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Hyderabad)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Mumbai)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Osaka)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Séoul)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Singapour)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Sydney)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Asie-Pacifique (Tokyo)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Canada (Centre)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour SQL Server 2022	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Chine (Ningxia)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Francfort)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Irlande)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Londres)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Milan)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Paris)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Espagne)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Stockholm)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Europe (Zurich)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Région	RDS pour SQL Server 2022	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Moyen-Orient (EAU)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
Amérique du Sud (São Paulo)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
AWS GovCloud (USA Est)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions
AWS GovCloud (US-Ouest)	Toutes les versions	Toutes les versions	Toutes les versions	Toutes les versions

Régions et moteurs de base de données pris en charge pour les clusters de bases de données multi-AZ dans Amazon RDS

Le déploiement d'un cluster de bases de données Multi-AZ dans Amazon RDS fournit un mode de déploiement à haute disponibilité d'Amazon RDS avec deux instances de bases de données secondaires et lisibles. Un cluster de base de données multi-AZ possède une instance de base de données d'écriture et deux instances de base de données de lecture dans trois zones de disponibilité distinctes d'une même région . Les clusters de base de données multi-AZ offrent une haute disponibilité, une capacité accrue pour les charges de travail en lecture et une moindre latence en écriture par rapport aux déploiements d'instances de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.

Les clusters de base de données multi-AZ ne sont pas disponibles avec les moteurs suivants :

- RDS pour Db2
- RDS for MariaDB
- RDS for Oracle
- RDS for SQL Server

Rubriques

- Clusters de bases de données multi-AZ avec RDS pour MySQL
- Clusters de bases de données multi-AZ avec RDS pour PostgreSQL

Clusters de bases de données multi-AZ avec RDS pour MySQL

Les régions et les versions de moteur suivantes sont disponibles pour les clusters de bases de données multi-AZ avec RDS for MySQL.

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Non disponible	Non disponible
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible

Vous pouvez répertorier les versions disponibles dans une région pour une classe d'instance de base de données donnée à l'aide du AWS CLI. Modifiez la classe d'instance de base de données pour afficher les versions de moteur disponibles pour celle-ci.

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options \
--engine mysql \
--db-instance-class db.r5d.large \
--query '*[]|[?SupportsClusters == `true`].[EngineVersion]'  \
--output text
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options ^
--engine mysql ^
--db-instance-class db.r5d.large ^
--query "*[]|[?SupportsClusters == `true`].[EngineVersion]"  ^
--output text
```

Clusters de bases de données multi-AZ avec RDS pour PostgreSQL

Les régions et les versions de moteur suivantes sont disponibles pour les clusters de bases de données multi-AZ avec RDS for PostgreSQL.

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13
USA Est (Virginie du Nord)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
USA Est (Ohio)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQ L 15	RDS for PostgreSQL 14	RDS for PostgreSQ L 13
USA Ouest (Californie du Nord)	Non disponibl e	Non disponibl e	Non disponibl e	Non disponibl e	Non disponibl e
USA Ouest (Oregon)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Afrique (Le Cap)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Paci fique (Hong Kong)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Hyderabad)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Jakarta)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Malaisie)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13
Asie-Pacifique (Melbourne)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Mumbai)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Osaka)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Séoul)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Singapour)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Sydney)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Asie-Pacifique (Tokyo)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQ L 15	RDS for PostgreSQL 14	RDS for PostgreSQ L 13
Canada (Centre)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Canada-Ou est (Calgary)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Chine (Beijing)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Chine (Ningxia)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Francfort)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Irlande)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Londres)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQ L 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13
Europe (Milan)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Paris)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Espagne)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Stockholm)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Europe (Zurich)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Israël (Tel Aviv)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Moyen-Orient (Bahreïn)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13
Moyen-Orient (EAU)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
Amérique du Sud (São Paulo)	Toutes les versions de PostgreSQL 17	Toutes les versions de PostgreSQL 16	Toutes les versions PostgreSQL 15	Versions 14.5 et ultérieures	Version 13.4, version 13.7 et versions ultérieures
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible	Non disponible	Non disponible

Vous pouvez répertorier les versions disponibles dans une région pour une classe d'instance de base de données donnée à l'aide du AWS CLI. Modifiez la classe d'instance de base de données pour afficher les versions de moteur disponibles pour celle-ci.

Dans Linux, macOS, ou Unix:

```
aws rds describe-orderable-db-instance-options \
--engine postgres \
--db-instance-class db.r5d.large \
--query '*[]|[?SupportsClusters == `true`].[EngineVersion]'  \
--output text
```

Dans Windows:

```
aws rds describe-orderable-db-instance-options ^
--engine postgres ^
--db-instance-class db.r5d.large ^
```

```
--query "*[]|[?SupportsClusters == `true`].[EngineVersion]"  ^
--output text
```

Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS

> **Important**
>
> AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.
>
> Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).
>
> Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
>
> Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Performance Insights dans Amazon RDS développe les fonctions de surveillance existantes d'Amazon RDS pour illustrer et vous aider à analyser les performances de votre base de données. Avec le tableau de bord Performance Insights, vous pouvez visualiser la charge de la base de données sur votre instance de base de données Amazon RDS. Vous pouvez également filtrer la

charge par les attentes, les instructions SQL, les hôtes ou les utilisateurs. Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

Performance Insights est disponible pour tous les moteurs de base de données RDS, à l'exception de RDS pour DB2.

Pour les moteurs de base de données disponibles, Performance Insights est disponible avec toutes les versions de moteur disponibles et dans toutes les versions Régions AWS.

Pour obtenir des informations sur la prise en charge de la région, du moteur de base de données et des classes d'instance pour les fonctionnalités de Performance Insights, consultezPrise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

Régions et moteurs de base de données pris en charge pour RDS Custom

Amazon RDS Custom automatise les tâches et les opérations d'administration des bases de données. En utilisant RDS Custom, en tant qu'administrateur de base de données, vous pouvez accéder et personnaliser votre environnement de base de données et votre système d'exploitation. Avec RDS Custom, vous pouvez personnaliser pour répondre aux exigences des applications héritées, personnalisées et compilées. Pour de plus amples informations, veuillez consulter Amazon RDS Custom.

RDS Custom est uniquement pris en charge pour les moteurs de base de données suivants :

Rubriques

- Régions et moteurs de base de données pris en charge pour RDS Custom pour Oracle
- Régions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server

Régions et moteurs de base de données pris en charge pour RDS Custom pour Oracle

Les régions et les versions de moteur suivantes sont disponibles pour RDS Custom for Oracle.

Région	Oracle Database 19c	Oracle Database 18c	Oracle Database 12c
USA Est (Virginie du Nord)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de

Région	Oracle Database 19c	Oracle Database 18c	Oracle Database 12c
			janvier 2021 ou supérieur
USA Est (Ohio)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
USA Ouest (Californie du Nord)	Non disponible	Non disponible	Non disponible
USA Ouest (Oregon)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Afrique (Le Cap)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Hong Kong)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumbai)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur

Région	Oracle Database 19c	Oracle Database 18c	Oracle Database 12c
Asie-Pacifique (Osaka)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Asie-Pacifique (Séoul)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Asie-Pacifique (Singapour)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Asie-Pacifique (Sydney)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Asie-Pacifique (Tokyo)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Canada (Centre)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Non disponible	Non disponible	Non disponible
Chine (Ningxia)	Non disponible	Non disponible	Non disponible

Région	Oracle Database 19c	Oracle Database 18c	Oracle Database 12c
Europe (Francfort)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Europe (Irlande)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Europe (Londres)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Europe (Milan)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Europe (Paris)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Europe (Stockholm)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible
Moyen-Orient (Bahreïn)	Non disponible	Non disponible	Non disponible

Région	Oracle Database 19c	Oracle Database 18c	Oracle Database 12c
Moyen-Orient (EAU)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
Amérique du Sud (São Paulo)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
AWS GovCloud (USA Est)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur
AWS GovCloud (US-Ouest)	19c avec le RU/RUR de janvier 2021 ou supérieur	18c avec le RU/RUR de janvier 2021 ou supérieur	12.1 et 12.2 avec le RU/RUR de janvier 2021 ou supérieur

Régions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server

Vous pouvez déployer RDS Custom for SQL Server en utilisant une version de moteur fournie par RDS (RPEV) ou une version de moteur personnalisée (CEV) :

- Si vous utilisez une version RPEV, elle inclut l'installation par défaut d'Amazon Machine Image (AMI) et de SQL Server. Si vous personnalisez ou modifiez le système d'exploitation, vos modifications risquent de ne pas persister lors de l'application des correctifs, de la restauration d'instantané ou de la récupération automatique.
- Si vous utilisez une version CEV, vous choisissez votre propre image AMI avec Microsoft SQL Server préinstallé ou SQL Server que vous installez à l'aide de votre propre support. Lorsque vous utilisez un CEV AWS fourni, vous choisissez la dernière EC2 image Amazon (AMI) disponible par AWS, dont la mise à jour cumulative (CU) est prise en charge par RDS Custom

for SQL Server. Avec une version CEV, vous pouvez personnaliser la configuration du système d'exploitation et de SQL Server pour répondre aux besoins de votre entreprise.

Les versions suivantes Régions AWS et du moteur de base de données sont disponibles pour RDS Custom pour SQL Server. La prise en charge des versions du moteur varie selon que vous utilisez RDS Custom for SQL Server avec une version RPEV, une version CEV fournie par AWS ou une version CEV fournie par le client.

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
USA Est (Virginie du Nord)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
USA Est (Ohio)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26,	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26,	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26,

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
	CU28 -GDR, CU29 0 et. CU3 CU32	CU28 -GDR, CU29 0 et. CU3 CU32	CU28 -GDR, CU29 0 et. CU3 CU32
USA Ouest (Californie du Nord)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
USA Ouest (Oregon)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Afrique (Le Cap)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Asie-Pacifique (Hong Kong)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Hyderabad)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakarta)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Malaisie)	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbourne)	Non disponible	Non disponible	Non disponible

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Asie-Pacifique (Mumbai)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Asie-Pacifique (Osaka)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Asie-Pacifique (Séoul)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Asie-Pacifique (Singapour)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Asie-Pacifique (Sydney)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Asie-Pacifique (Tokyo)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Canada (Centre)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Canada-Ouest (Calgary)	Non disponible	Non disponible	Non disponible
Chine (Beijing)	Non disponible	Non disponible	Non disponible
Chine (Ningxia)	Non disponible	Non disponible	Non disponible
Europe (Francfort)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Europe (Irlande)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Europe (Londres)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Europe (Milan)	Non disponible	Non disponible	Non disponible

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Europe (Paris)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Europe (Espagne)	Non disponible	Non disponible	Non disponible
Europe (Stockholm)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
Europe (Zurich)	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible

Région	Version RPEV	AWS CEV fourni	Version CEV fournie par le client
Moyen-Orient (Bahreïn)	Non disponible	Non disponible	Non disponible
Moyen-Orient (EAU)	Non disponible	Non disponible	Non disponible
Amérique du Sud (São Paulo)	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32	SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 - GDR, CU15, CU16, CU17 et. CU18 CU19 SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
AWS GovCloud (USA Est)	Non disponible	Non disponible	Non disponible
AWS GovCloud (US-Ouest)	Non disponible	Non disponible	Non disponible

Régions et moteurs de base de données pris en charge pour Amazon RDS Proxy

Le proxy Amazon RDS est un proxy de base de données entièrement géré et hautement disponible qui rend les applications plus évolutives en regroupant et en partageant les connexions de base de données établies. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .

Le proxy RDS n'est pas disponible pour les moteurs suivants :

- RDS pour Db2
- RDS for Oracle

Rubriques

- RDS Proxy avec RDS for MariaDB
- Proxy RDS avec RDS for MySQL
- Proxy RDS avec RDS for PostgreSQL
- RDS Proxy avec RDS for SQL Server

RDS Proxy avec RDS for MariaDB

Les régions et les versions de moteur suivantes sont disponibles pour RDS Proxy avec RDS for MariaDB.

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Thaïlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MariaDB 11.4	RDS for MariaDB 10.11	RDS for MariaDB 10.6	RDS for MariaDB 10.5	RDS for MariaDB 10.4
Mexique (centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Proxy RDS avec RDS for MySQL

Les régions et les versions de moteur suivantes sont disponibles pour RDS Proxy avec RDS for MySQL.

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS for MySQL 5.7
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS for MySQL 5.7
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Thaïlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS for MySQL 5.7
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour MySQL 8.4	RDS for MySQL 8.0	RDS for MySQL 5.7
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Mexique (centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Proxy RDS avec RDS for PostgreSQL

Les régions et les versions de moteur suivantes sont disponibles pour RDS Proxy avec RDS for PostgreSQL.

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Thaïlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS pour PostgreSQL 17	RDS pour PostgreSQL 16	RDS pour PostgreSQL 15	RDS for PostgreSQL 14	RDS for PostgreSQL 13	RDS for PostgreSQL 12	RDS for PostgreSQL 11	RDS for PostgreSQL 10
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Mexique (centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

RDS Proxy avec RDS for SQL Server

Les régions et les versions de moteur suivantes sont disponibles pour RDS Proxy avec RDS for SQL Server.

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Oregon)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Hyderabad)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Jakarta)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Malaisie)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Melbourne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Mumbai)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Osaka)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapour)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydney)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Thaïlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada-Ouest (Calgary)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Beijing)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Chine (Ningxia)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Francfort)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londres)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	RDS for SQL Server 2019	RDS for SQL Server 2017	RDS for SQL Server 2016
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Espagne)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Stockholm)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Israël (Tel Aviv)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (Bahreïn)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Moyen-Orient (EAU)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Mexique (centre)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Amérique du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (USA Est)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
AWS GovCloud (US-Ouest)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Régions et moteurs de base de données pris en charge pour l'intégration de Secrets Manager à Amazon RDS

Vous pouvez AWS Secrets Manager ainsi remplacer les informations d'identification codées en dur dans votre code, y compris les mots de passe de base de données, par un appel d'API à Secrets Manager pour récupérer le secret par programmation. Pour plus d'informations sur Secrets Manager, consultez le Guide de l'utilisateur AWS Secrets Manager.

Vous pouvez spécifier qu'Amazon RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager pour une instance de base de données Amazon RDS ou un cluster de bases de données multi-AZ. RDS génère le mot de passe, le stocke dans Secrets Manager et effectue régulièrement sa rotation. Pour plus d'informations, consultez Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.

L'intégration de Secrets Manager est disponible partout Régions AWS.

Régions et moteurs de base de données pris en charge pour les intégrations Amazon RDS Zero-ETL

Les intégrations RDS Zero-ETL sont une solution entièrement gérée permettant de rendre les données transactionnelles disponibles dans Amazon Redshift ou Amazon SageMaker après leur écriture sur une instance de base de données Amazon RDS. Pour de plus amples informations, veuillez consulter Intégrations zéro ETL.

Rubriques

- Intégrations sans ETL avec RDS pour MySQL

Intégrations sans ETL avec RDS pour MySQL

Les régions et versions de moteur suivantes sont disponibles pour les intégrations RDS for MySQL Zero-ETL avec Amazon Redshift et. Amazon SageMaker

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
USA Est (Virginie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Est (Ohio)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
USA Ouest (Californie du Nord)	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
USA Ouest (Orego	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Afrique (Le Cap)	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Hong Kong)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
Asie-Pacifique (Hyderad)	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Jakart	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Malais)	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Melbo e)	Non disponible	Non disponible	Non disponible	Non disponible
Asie-Pacifique (Mumb	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
Asie-Pacifique (Osaka	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Asie-Pacifique (Séoul)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Singapr)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Sydne	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Asie-Pacifique (Tokyo	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Canada (Centre	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
Canada Ou est (Calgar	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Beijing	Non disponible	Non disponible	Non disponible	Non disponible
Chine (Ningxi	Non disponible	Non disponible	Non disponible	Non disponible
Europe (Francf t)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Irlande	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Londre	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Milan)	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Europe (Paris)	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
Europe (Espag	Non disponible	Non disponible	Non disponible	Non disponible
Europe (Stockh m)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles
Europe (Zurich	Non disponible	Non disponible	Non disponible	Non disponible
Israël (Tel Aviv)	Non disponible	Non disponible	Non disponible	Non disponible
Moyen-Ori ent (Bahrei	Toutes versions disponibles	Toutes versions disponibles	Non disponible	Non disponible
Moyen-Ori ent (EAU)	Non disponible	Non disponible	Non disponible	Non disponible
Amériq du Sud (São Paulo)	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles	Toutes versions disponibles

Région	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.4	Intégration zéro ETL avec Amazon Redshift pour RDS pour MySQL 8.0	Intégration sans ETL avec Amazon SageMaker for RDS for MySQL 8.4	Intégration sans ETL avec Amazon SageMaker for RDS pour MySQL 8.0
AWS GovClo (USA Est)	Non disponible	Non disponible	Non disponible	Non disponible
AWS GovClo (US-Ouest)	Non disponible	Non disponible	Non disponible	Non disponible

Fonctionnalités natives du moteur dans Amazon RDS

Les moteurs de base de données Amazon RDS prennent également en charge un grand nombre des caractéristiques et fonctions les plus courantes propres aux moteurs. Ces fonctions sont différentes des fonctions natives d'Amazon RDS énumérées sur cette page. Certaines fonctions natives du moteur peuvent avoir une prise en charge limitée ou des privilèges restreints.

Pour obtenir plus d'informations sur les fonctions natives du moteur, consultez :

- Fonctionnalités d'Amazon RDS pour DB2
- Support des fonctionnalités MariaDB sur Amazon RDS
- Fonctionnalités MySQL prises en charge sur Amazon RDS
- Fonctions RDS for Oracle
- Utilisation des SQL fonctionnalités de Postgre prises en charge par Amazon RDS pour Postgre SQL
- Fonctionnalités SQL de Microsoft Server sur Amazon RDS

Facturation d'instance de base de données pour Amazon

Les RDS instances Amazon sont facturées sur la base des composants suivants :

- Heures des instances de base de données (par heure) – En fonction de la classe de l'instance de base de données (par exemple, db.t2.small or db.m4.large). Les prix sont indiqués sur une base horaire, mais les factures sont calculées à la seconde près et les heures sont indiquées sous forme décimale. RDSl'utilisation est facturée par tranches d'une seconde, avec un minimum de 10 minutes. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .
- Stockage (par Gio, tous les mois) – Capacité de stockage provisionnée pour votre instance de base de données. Si vous mettez à l'échelle votre capacité de stockage provisionnée dans le mois, votre facture est calculée au prorata. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS.
- Input/output (I/O) requests (per 1 million requests) – Total number of storage I/Odemandes que vous avez effectuées au cours d'un cycle de facturation, pour uniquement.
- Provisionné IOPS (IOPSpar mois) : IOPS taux provisionné, quelle que soit la IOPS consommation, pour le stockage gp3 Amazon RDS Provisioned IOPS (SSD) et General Purpose (SSD). Le stockage provisionné pour les EBS volumes est facturé par tranches d'une seconde, avec un minimum de 10 minutes.
- Stockage de sauvegarde (par Gio par mois) – Le stockage de sauvegarde est le stockage associé à vos sauvegardes de base de données automatisées et tout instantané de base de données active que vous avez pris. Augmenter votre période de rétention des sauvegardes ou prendre des instantanés de base de données supplémentaires augmente le stockage de sauvegarde consommé par votre base de données. La facturation à la seconde ne s'applique pas au stockage de sauvegarde (mesuré en Go/mois).

  Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.
- Transfert de données (par Go) — Transfert de données vers et depuis votre instance de base de données depuis ou vers Internet et d'autres AWS régions. Pour des exemples utiles, consultez le billet de AWS blog Exploring Data Transfer Coûts for AWS Managed Database.

Amazon RDS propose les options d'achat suivantes pour vous permettre d'optimiser vos coûts en fonction de vos besoins :

- On-Demand instances (Instances à la demande) : paiement à l'heure pour les heures d'instance de base de données que vous utilisez. Les prix sont indiqués sur une base horaire, mais les factures sont calculées à la seconde près et les heures sont indiquées sous forme décimale. RDSl'utilisation est désormais facturée par tranches d'une seconde, avec un minimum de 10 minutes.
- Reserved instances (Instances réservées) : réservez une instance de base de données pour un an ou trois ans, et bénéficiez d'une remise importante par rapport à la tarification des instances de base de données à la demande. Grâce aux instances réservées, vous pouvez lancer, supprimer, démarrer ou arrêter plusieurs instances pendant une heure, puis obtenir l'avantage d'instance réservé pour toutes les instances.

Pour obtenir RDS des informations sur les tarifs d'Amazon, consultez la page de RDS tarification d'Amazon.

Rubriques

- Instances de base de données à la demande pour Amazon RDS
- Instances de base de données réservées pour Amazon RDS

Instances de base de données à la demande pour Amazon RDS

Les instances de base de données à la demande Amazon RDS sont facturées en fonction de la classe de l'instance de base de données (par exemple, db.t3.small ou db.m5.large). Pour plus d'informations sur la tarification de Amazon RDS, consultez la page produit de Amazon RDS.

La facturation commence pour une instance de base de données dès que cette dernière est disponible. La tarification est indiquée selon une base horaire, mais les factures sont calculées à la seconde près et affichent les heures sous une forme décimale. L'utilisation de Amazon RDS est facturée par incréments d'une seconde, avec un minimum de 10 minutes. Dans le cas d'une modification de configuration pouvant être facturée, comme le dimensionnement de la capacité de calcul ou de stockage, un minimum de 10 minutes vous est facturé. La facturation continue jusqu'à la résiliation de l'instance de base de données, qui a lieu lorsque vous supprimez cette dernière ou si elle échoue.

Si vous ne souhaitez plus être facturé pour votre instance de base de données, vous devez l'arrêter ou la supprimer afin d'éviter d'être facturé pour des heures d'instance de base de données supplémentaires. Pour plus d'informations sur les états des instances de base de données pour lesquelles vous êtes facturé, consultez Affichage de l'état de l'instance de base de données dans un cluster Aurora.

Instances de base de données arrêtées

Pendant que votre instance de base de données est arrêtée, le stockage provisionné vous est facturé, y compris les IOPS provisionnés. Le stockage de sauvegarde vous est également facturé, notamment le stockage des instantanés manuels et des sauvegardes automatisées dans la fenêtre de conservation que vous avez spécifiée. Les heures de l'instance de base de données ne vous sont pas facturées.

Instances de base de données multi-AZ

Une configuration multi-AZ améliore la durabilité et la disponibilité des données en provisionnant et en gérant automatiquement une réplique de secours synchrone dans une zone de disponibilité différente. En raison des ressources supplémentaires et de la disponibilité accrue, les déploiements multi-AZ sont plus chers que les déploiements mono-AZ, et peuvent coûter environ deux fois plus cher en raison de l'instance de secours supplémentaire et des ressources associées.

Tenez compte des informations importantes suivantes concernant la tarification multi-AZ :

- Coûts de calcul : facturés par heure d'instance de base de données pour les instances principales et de secours.
- Coûts de stockage : facturés par Go par mois pour le stockage provisionné à la fois pour les instances principales et de secours.
- Coûts de transfert de données : la réplication entre l'instance principale et l'instance de secours est incluse dans le coût, mais d'autres frais de transfert de données peuvent s'appliquer en fonction de votre utilisation.

Pour estimer avec précision vos coûts mensuels en fonction de votre cas d'utilisation spécifique Région AWS, vous pouvez utiliser le Calculateur de tarification AWS. Cet outil vous permet de saisir les détails de votre configuration et fournit une ventilation complète des coûts.

> **Note**
>
> Les prix sont sujets à modification. Consultez la page de tarification d'Amazon RDS pour obtenir le plus up-to-date d'informations.

Instances de base de données réservées pour Amazon RDS

En utilisant des instances de base de données réservées, vous pouvez réserver une instance de base de données pour une durée d'un an ou de trois ans. Ce type d'instance est beaucoup plus économique que les instances de bases de données à la demande. Les instances de bases de données réservées ne sont pas des instances physiques, mais correspondent à une remise sur la facturation appliquée à l'utilisation de certaines instances de bases de données à la demande dans votre compte. Les remises pour instances de base de données réservées sont liées au type d'instance et à la Région AWS.

En règle générale, pour utiliser des instances de bases de données réservées, commencez par recueillir des informations sur les offres disponibles, achetez l'offre qui vous convient, puis consultez le détail des instances de bases de données réservées existantes.

Pour plus d'informations sur l'achat d'instances de base de données réservées et sur la consultation de la facturation des instances de base de données réservées, consultez les sections suivantes.

- Achat d'instances de base de données réservées pour (Amazon RDS)
- Consulter la facturation des instances de base de données réservées pour Amazon RDS

Présentation des instances de base de données réservées

Lorsque vous achetez une instance de base de données réservée dans Amazon RDS, vous achetez la garantie d'obtenir un tarif réduit sur un type d'instance de bases de données spécifique pour la durée de l'instance de base de données réservée. Pour utiliser une instance de base de données réservée Amazon RDS, créez une instance de bases de données, tout comme vous le feriez pour une instance à la demande.

L'instance de base de données que vous créez doit comporter les mêmes spécifications que l'instance de base de données réservée pour les éléments suivants :

- Région AWS
- Moteur de base de données (Il n'est pas nécessaire que le numéro de version du moteur de base de données corresponde.)
- Type d'instance de base de données
- Taille de l'instance de base de données (licence RDS pour Microsoft SQL Server et Amazon RDS for Oracle incluse)

- Édition (RDS pour SQL Server et RDS pour Oracle)
- Type de licence (licence incluse ou) bring-your-own-license

Si les spécifications de la nouvelle instance de bases de données coïncident avec une instance de base de données réservée existante dans votre compte, vous êtes facturé au tarif réduit correspondant à cette dernière. Dans le cas contraire, l'instance de bases de données est facturée selon le tarif à la demande.

Vous pouvez modifier une instance de base de données que vous utilisez en tant qu'instance de base de données réservée. Si la modification est conforme aux spécifications de l'instance de base de données réservée, une partie ou la totalité de la remise s'applique toujours à l'instance de base de données modifiée. Si la modification est en dehors des spécifications, comme la modification de la classe d'instance, la remise ne s'applique plus. Pour plus d'informations, veuillez consulter Instances de base de données réservées de taille flexible.

Rubriques

- Types d'offres
- Instances de base de données réservées de taille flexible
- Exemple de facturation d'une instance de base de données réservée
- Instances de base de données réservées pour un cluster de bases de données multi-AZ
- Suppression d'une instance de base de données réservée

Pour plus d'informations sur les instances de bases de données réservées, ainsi que sur leur tarification, consultez Instances réservées Amazon RDS.

Types d'offres

Trois types d'instances de base de données réservées sont disponibles : sans paiement initial, avec paiement initial partiel et avec paiement initial total. Vous pouvez donc optimiser vos coûts Amazon RDS en vous basant sur votre utilisation prévue.

> **Note**
>
> Toutes les classes d'instances RDS ne prennent pas en charge tous les types d'offres d'instances réservées. Par exemple, certaines classes d'instance peuvent ne pas proposer l'option No Upfront. Pour confirmer la disponibilité, consultez les offres d'instances réservées

dans la commande AWS Management Console ou utilisez la `describe-reserved-db-instances-offerings` AWS CLI commande.

Sans frais initiaux

Cette option vous permet d'accéder à des instances de base de données réservées sans paiement initial. Les instances de bases de données réservées sans frais initiaux n'impliquent aucun paiement initial et sont facturées selon un taux horaire réduit pendant toute la durée de l'engagement, quelle que soit l'utilisation. Cette option est uniquement disponible dans le cadre d'une réservation d'un an.

Frais initiaux partiels

Cette option exige qu'une partie des instances de base de données réservées soit payée d'avance. Les heures restantes pendant la période sont facturées à un taux réduit, quelle que soit l'utilisation. Cette option remplace l'option précédente d'utilisation intensive.

Tous les frais initiaux

Le paiement complet est effectué en totalité au début de la période, sans aucun autre coût pour le reste de la réservation, quel que soit le nombre d'heures utilisé.

Si vous utilisez une facturation consolidée, tous les comptes de l'organisation sont traités comme s'il s'agissait d'un seul compte. Cela signifie que tous les comptes de l'organisation peuvent bénéficier d'un surplus d'heures correspondant aux instances de base de données réservées qui sont achetées par un autre compte. Pour plus d'informations sur la facturation consolidée, consultez Instances de base de données réservées Amazon RDS dans le Guide de l'utilisateur AWS Billing and Cost Management.

Instances de base de données réservées de taille flexible

Lorsque vous achetez une instance de base de données réservée, vous devez spécifier la classe d'instance, par exemple, db.r5.large. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .

Si vous devez augmenter la capacité d'une instance de base de données, l'instance réservée est automatiquement appliquée à l'instance de base de données que vous avez dimensionnée. En d'autres termes, les instances de base de données réservées sont appliquées automatiquement

aux classes d'instances de bases de données, quelle que soit leur taille. Des instances de base de données réservées de taille flexible sont disponibles pour les instances de base de données dotées du même moteur de base Région AWS de données. Des instances de bases de données réservées de taille flexible peuvent uniquement être mises à l'échelle dans leur type de classe d'instance. Par exemple, une instance de base de données réservée pour une classe d'instance db.r5.large peut être appliquée à une classe d'instance db.r5.xlarge, mais pas à db.r6g.large, car db.r5 et db.r6g sont des types de classes d'instance différents.

Les avantages des instances de base de données réservées s'appliquent aux configurations multi-AZ et mono-AZ. Cela signifie que vous pouvez passer librement d'une configuration à l'autre au sein d'un même type de classe d'instance de base de données. Par exemple, vous pouvez passer d'un déploiement mono-AZ exécuté sur une instance de base de données de grande taille (quatre unités normalisées par heure) à un déploiement multi-AZ exécuté sur deux instances de base de données de taille moyenne (2+2 = 4 unités normalisées par heure).

Des instances de base de données réservées de taille flexible sont disponibles pour les moteurs de base de données Amazon RDS suivants :

- RDS for MariaDB
- RDS for MySQL
- RDS pour Oracle, apportez votre propre licence
- RDS for PostgreSQL

La flexibilité de taille ne s'applique pas à la licence RDS pour SQL Server et RDS pour Oracle incluse.

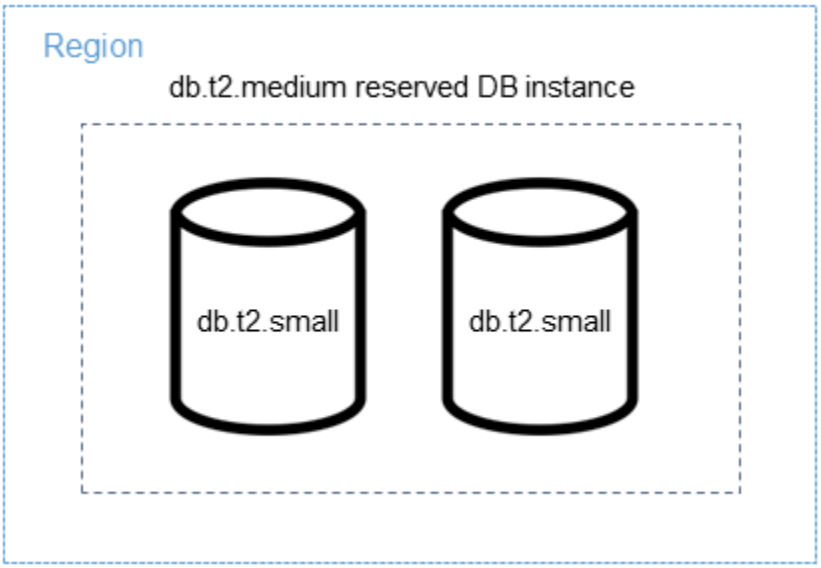
Pour obtenir des détails sur l'utilisation d'instances réservées de taille flexible avec Aurora, consultez Instances de base de données réservées à Aurora.

Les unités normalisées par heure permettent de comparer l'utilisation pour différentes tailles d'instances de base de données réservées. Par exemple, une unité d'utilisation sur deux instances de bases de données db.r3.large équivaut à huit unités normalisées par heure d'utilisation sur une instance db.r3.small. La table suivante indique le nombre d'unités normalisées par heure pour chaque taille d'instance de bases de données.

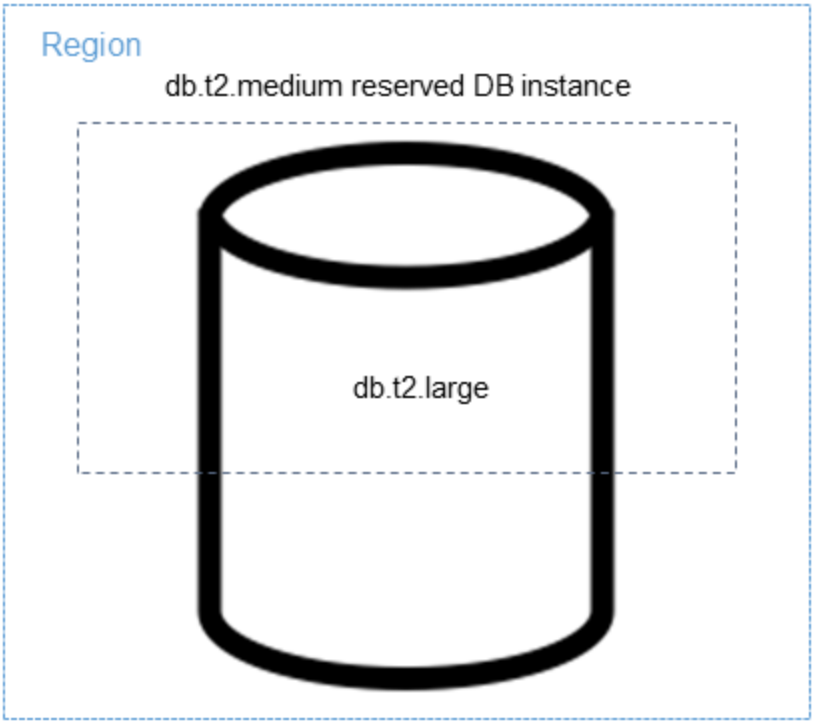
Taille d’instance	Unités normalisées mono-AZ par heure (déploiement avec une instance de base de données)	Unités normalisées d'instance de base de données multi-AZ par heure (déploiem ent avec une instance de base de données et une instance de secours)	Unités normalisées de cluster de bases de données multi-AZ par heure (déploiem ent avec une instance de base de données et deux instances de secours)
micro	0.5	1	1.5
petit	1	2	3
medium	2	4	6
large	4	8	12
xlarge	8	16	24
2xlarge	16	32	48
4xlarge	32	64	96
6xlarge	48	96	144
8xlarge	64	128	192
10xlarge	80	160	240
12xlarge	96	192	288
16xlarge	128	256	384
24xlarge	192	384	576
32xlarge	256	512	768

Par exemple, supposons que vous achetez une instance de base de données réservée `db.t2.medium` et que deux instances de base de données `db.t2.small` sont exécutées dans

votre compte dans la même Région AWS. Dans ce cas, l'avantage de facturation est appliqué entièrement à ces deux instances.



Sinon, si une `db.t2.large` instance est exécutée sur votre compte dans le même compte Région AWS, l'avantage de facturation est appliqué à 50 % de l'utilisation de l'instance de base de données.



Exemple de facturation d'une instance de base de données réservée

Le prix d'une instance de base de données réservée n'offre pas de réduction sur les coûts associés au stockage, aux sauvegardes et aux E/S. Il n'offre une réduction que sur l'utilisation horaire de l'instance à la demande. L'exemple suivant illustre le coût total mensuel pour une instance de base de données réservée :

- Classe d'instances de base de données db.r5.large mono-AZ réservée RDS for MySQL dans la région USA Est (Virginie du Nord) avec l'option No Upfront (aucuns frais initiaux) au coût de 0,12 USD pour l'instance ou de 90 USD par mois
- 400 Gio de stockage Volume à usage général SSD (GP2) au coût de 0,115 par Gio et par mois, soit 45,60 USD par mois
- 600 Gio de stockage de sauvegarde au coût de 0,095 USD, soit 19 USD par mois (400 Gio gratuits)

Ajoutez tous ces frais (90 USD + 45,60 USD + 19 USD) à l'instance de base de données réservée : le coût total mensuel est de 154,60 USD.

Si vous choisissez d'utiliser une instance de base de données à la demande au lieu d'une instance de base de données réservée, une classe d'instances de base de données db.r5.large mono-AZ RDS for MySQL dans la région USA Est (Virginie du Nord) coûte 0,1386 USD par heure ou 101,18 USD par mois. Pour une instance de base de données à la demande, ajoutez toutes ces options (101,18 USD + 45,60 USD + 19 USD) ; le coût total mensuel est de 165,78 USD. L'instance de base de données réservée vous permet d'économiser un peu plus de 11 USD par mois.

> **Note**
>
> Les prix indiqués ici sont des exemples et ne correspondent pas aux prix réels. Pour obtenir des informations sur la tarification d'Amazon RDS, consultez Tarification d'Amazon RDS.

Instances de base de données réservées pour un cluster de bases de données multi-AZ

Pour acheter les instances de base de données réservées pour un cluster de bases de données multi-AZ, voici ce que vous pouvez faire :

- Réservez trois instances de base de données mono-AZ de la même taille que les instances du cluster.

- Réservez une instance de base de données multi-AZ et une instance de base de données mono-AZ, de la même taille que les instances de base de données du cluster.

Par exemple, supposons que vous disposez d'un cluster composé de trois instances de base de données db.m6gd.large. Dans ce cas, vous pouvez soit acheter trois instances de base de données réservées mono-AZ db.m6gd.large, soit une instance de base de données réservée multi-AZ db.m6gd.large et une instance de base de données réservée mono-AZ db.m6gd.large. Chacune de ces options réserve la remise d'instance réservée maximale pour le cluster de bases de données multi-AZ.

Vous pouvez également utiliser des instances de base de données dont la taille est flexible et acheter une instance de base de données plus grande pour couvrir des instances de base de données plus petites dans un ou plusieurs clusters. Par exemple, si vous disposez de deux clusters contenant six instances de base de données db.m6gd.large au total, vous pouvez acheter trois instances de base de données réservées mono-AZ db.m6gd.xl. Cela permet de réserver les six instances de base de données des deux clusters. Pour de plus amples informations, veuillez consulter Instances de base de données réservées de taille flexible.

Vous pouvez réserver des instances de base de données de la même taille que les instances de base de données du cluster, mais réservez moins d'instances de base de données que le nombre total d'instances de base de données du cluster. Toutefois, dans ce cas, le cluster n'est que partiellement réservé. Supposons, par exemple, que vous disposiez d'un cluster avec trois instances de base de données db.m6gd.large et que vous achetiez une instance de base de données réservée multi-AZ db.m6gd.large. Dans ce cas, le cluster n'est que partiellement réservé, car seules deux des trois instances du cluster sont couvertes par des instances de base de données réservées. L'instance de base de données restante est facturée au tarif horaire db.m6gd.large à la demande.

Pour de plus amples informations sur les clusters de base de données multi-AZ, consultez Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.

Suppression d'une instance de base de données réservée

Les conditions d'une instance de base de données réservée impliquent un engagement d'un an ou de trois ans. Il n'est pas possible d'annuler une instance de base de données réservée. Toutefois, vous pouvez supprimer une instance de base de données à laquelle s'applique une remise d'instance de base de données réservée. Le processus de suppression d'une instance de base de données couverte par ce type de remise est le même que pour n'importe quelle autre instance de bases de données.

Vous êtes facturé pour les coûts initiaux, que vous utilisiez ou non les ressources.

Si vous supprimez une instance de base de données à laquelle s'applique une remise d'instance de base de données réservée, vous pouvez lancer toute autre instance de bases de données dont les spécifications sont compatibles. Dans ce cas, vous conservez le tarif réduit jusqu'à la fin de la période de réservation (d'un ou de trois ans).

Achat d'instances de base de données réservées pour (Amazon RDS)

Vous pouvez utiliser l'API AWS Management Console, le AWS CLI, et l'API RDS pour travailler avec des instances de base de données réservées.

console

Vous pouvez utiliser le AWS Management Console pour travailler avec des instances de base de données réservées, comme indiqué dans les procédures suivantes.

Pour obtenir la tarification et les informations relatives aux offres d'instances de bases de données réservées disponibles

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Instances réservées.
3. Choisissez Purchase Reserved DB Instance (Instance de base de données réservée à l'achat).
4. Pour Description du produit, choisissez le moteur de base de données et le type de licence.
5. Pour Classe d'instance de base de données, choisissez la classe d'instance de base de données.
6. Pour Option de déploiement, choisissez si vous souhaitez un déploiement d'instance de base de données mono-AZ ou multi-AZ.

> **Note**
>
> Pour acheter les instances de base de données réservées équivalentes pour un déploiement de cluster de bases de données multi-AZ, achetez trois instances de base de données réservées mono-AZ ou une instance de base de données réservée multi-AZ et une instance de base de données réservée mono-AZ. Pour de plus amples informations, veuillez consulter Instances de base de données réservées pour un cluster de bases de données multi-AZ.

7. Pour Durée, choisissez la durée pendant laquelle vous souhaitez réserver l'instance de base de données.
8. Pour Type d'offre, choisissez le type d'offre.

   Les informations relatives à la tarification s'affichent après la sélection du type d'offre.

   > **Important**
   >
   > Choisissez Annuler pour éviter d'acheter l'instance de base de données réservée et d'avoir à payer des frais.

Une fois que vous disposez des informations requises sur les offres d'instances de bases de données réservées disponibles, vous pouvez utiliser ces informations pour acheter une offre, comme le montre la procédure suivante.

Pour acheter une instance de base de données réservée

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Instances réservées.
3. > **Important**
   >
   > Avant de continuer, vérifiez que vous avez bien saisi la bonne adresse Région AWS. Les instances de base de données réservées sont spécifiques à une région et ne peuvent pas être transférées entre les régions. Vérifiez le sélecteur de région dans le coin supérieur droit de la console pour vous assurer que vous achetez l'instance réservée dans la région souhaitée.
4. Choisissez Purchase reserved DB instance (Acheter une instance de base de données réservée).
5. Pour Description du produit, choisissez le moteur de base de données et le type de licence.
6. Pour Classe d'instance de base de données, choisissez la classe d'instance de base de données.
7. Pour Déploiement multi-AZ, choisissez si vous souhaitez un déploiement d'instance de base de données mono-AZ ou multi-AZ.

> **Note**
>
> Pour acheter les instances de base de données réservées équivalentes pour un déploiement de cluster de bases de données multi-AZ, achetez trois instances de base de données réservées mono-AZ ou une instance de base de données réservée multi-AZ et une instance de base de données réservée mono-AZ. Pour de plus amples informations, veuillez consulter Instances de base de données réservées pour un cluster de bases de données multi-AZ.

8. Pour Durée, choisissez la durée pendant laquelle vous souhaitez que l'instance de base de données soit réservée.
9. Pour Type d'offre, choisissez le type d'offre.

   Les informations relatives à la tarification s'affichent après que vous avez choisi le type d'offre.
10. (Facultatif) Afin de faciliter le suivi des instances de base de données réservées que vous achetez, vous pouvez leur attribuer un identifiant de votre choix. Dans Reserved Id (ID réservé), tapez un identifiant pour l'instance de bases de données réservée.
11. Sélectionnez Envoyer.

    Votre instance de base de données réservée est achetée, puis affichée dans la liste Reserved instances (Instances réservées).

Une fois que vous avez acheté une instance de bases de données réservée, suivez la procédure ci-dessous afin d'en consulter le détail.

Pour obtenir des informations sur les instances de base de données réservées pour votre AWS compte

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet Navigation, choisissez Instances réservées.

   Les instances de bases de données réservées pour votre compte s'affichent. Pour afficher des informations détaillées sur une instance de base de données réservée particulière, choisissez cette instance dans la liste. Vous pouvez alors consulter des informations détaillées sur cette instance dans le volet de détails au bas de la console.

AWS CLI

Vous pouvez utiliser le AWS CLI pour travailler avec des instances de base de données réservées, comme indiqué dans les exemples suivants.

Example d'obtention des offres d'instances de base de données réservées disponibles

Pour obtenir des informations sur les offres d'instances de base de données réservées disponibles, appelez la AWS CLI commande describe-reserved-db-instances-offerings.

```
aws rds describe-reserved-db-instances-offerings
```

Cet appel vous renvoie des informations semblables à ce qui suit :

```
OFFERING  OfferingId                            Class         Multi-AZ  Duration  Fixed
 Price  Usage Price  Description  Offering Type
OFFERING  438012d3-4052-4cc7-b2e3-8d3372e0e706  db.r3.large   y         1y
 1820.00 USD  0.368 USD    mysql        Partial  Upfront
OFFERING  649fd0c8-cf6d-47a0-bfa6-060f8e75e95f  db.r3.small   n         1y
 227.50 USD  0.046 USD    mysql        Partial  Upfront
OFFERING  123456cd-ab1c-47a0-bfa6-12345667232f  db.r3.small   n         1y
 162.00 USD   0.00 USD    mysql        All      Upfront
    Recurring Charges:   Amount  Currency  Frequency
    Recurring Charges:   0.123   USD       Hourly
OFFERING  123456cd-ab1c-37a0-bfa6-12345667232d  db.r3.large   y         1y
 700.00 USD   0.00 USD    mysql        All      Upfront
    Recurring Charges:   Amount  Currency  Frequency
    Recurring Charges:   1.25    USD       Hourly
OFFERING  123456cd-ab1c-17d0-bfa6-12345667234e  db.r3.xlarge  n         1y
 4242.00 USD   2.42 USD    mysql        No       Upfront
```

Une fois que vous disposez des informations requises sur les offres d'instances de base de données réservées disponibles, vous pouvez utiliser ces informations pour acheter une offre, comme le montre l'exemple suivant.

Pour acheter une instance de base de données réservée, utilisez la AWS CLI commande purchase-reserved-db-instances-offeringavec les paramètres suivants :

- `--reserved-db-instances-offering-id` – L'identifiant de l'offre que vous voulez acheter. Reportez-vous à l'exemple précédent pour obtenir l'ID de l'offre.

- `--reserved-db-instance-id` – Vous pouvez attribuer l'identifiant de votre choix aux instances de base de données réservées que vous achetez pour en faciliter le suivi.

Example d'achat d'une instance de base de données réservée

L'exemple suivant achète l'offre d'instance de base de données réservée avec un ID *649fd0c8-cf6d-47a0-bfa6-060f8e75e95f* et attribue l'identifiant de*MyReservation*.

Pour LinuxmacOS, ou Unix :

```
aws rds purchase-reserved-db-instances-offering \
    --reserved-db-instances-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f \
    --reserved-db-instance-id MyReservation
```

Dans Windows :

```
aws rds purchase-reserved-db-instances-offering ^
    --reserved-db-instances-offering-id 649fd0c8-cf6d-47a0-bfa6-060f8e75e95f ^
    --reserved-db-instance-id MyReservation
```

La commande renvoie un résultat semblable à ce qui suit :

```
RESERVATION  ReservationId      Class        Multi-AZ  Start Time
 Duration  Fixed Price  Usage Price  Count  State           Description  Offering Type
RESERVATION  MyReservation      db.r3.small  y         2011-12-19T00:30:23.247Z  1y
    455.00 USD   0.092 USD    1      payment-pending  mysql        Partial  Upfront
```

Après avoir acheté des instances de bases de données réservées, vous pouvez en consulter le détail.

Pour obtenir des informations sur les instances de base de données réservées pour votre AWS compte, appelez la AWS CLI commande `describe-reserved-db-instances`, comme indiqué dans l'exemple suivant.

Example d'obtenir vos instances de bases de données réservées

```
aws rds describe-reserved-db-instances
```

La commande renvoie un résultat semblable à ce qui suit :

```
RESERVATION  ReservationId     Class        Multi-AZ  Start Time
 Duration  Fixed Price  Usage Price  Count  State    Description  Offering Type
RESERVATION  MyReservation     db.r3.small  y         2011-12-09T23:37:44.720Z  1y
   455.00 USD   0.092 USD    1      retired  mysql        Partial  Upfront
```

API RDS

Vous pouvez utiliser l'API RDS pour travailler avec des instances de base de données réservées :

- Pour obtenir des informations sur les offres d'instances de bases de données réservées disponibles, exécutez l'opération de l'API Amazon RDS `DescribeReservedDBInstancesOfferings`.
- Une fois que vous disposez des informations requises sur les offres d'instances de base de données réservées disponibles, vous pouvez utiliser ces informations pour acheter une offre, comme le montre l'exemple suivant. Exécutez l'opération de l'API RDS `PurchaseReservedDBInstancesOffering` avec les paramètres suivants :
  - `--reserved-db-instances-offering-id` – L'identifiant de l'offre que vous voulez acheter.
  - `--reserved-db-instance-id` – Vous pouvez attribuer l'identifiant de votre choix aux instances de base de données réservées que vous achetez pour en faciliter le suivi.
- Après avoir acheté des instances de bases de données réservées, vous pouvez en consulter le détail. Exécutez l'opération de l'API RDS `DescribeReservedDBInstances`.

Consulter la facturation des instances de base de données réservées pour Amazon RDS

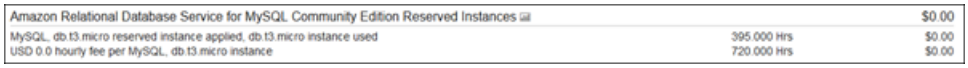
Vous pouvez afficher la facturation de vos instances de base de données réservées dans le tableau de bord de facturation de la AWS Management Console.

Pour afficher la facturation des instances de base de données réservées

1. Connectez-vous au AWS Management Console.
2. De le menu du compte, en haut à droite, choisissez Billing Dashboard (Tableau de bord de facturation).
3. Choisissez Bill Details (Détails de facturation) dans le coin supérieur droit du tableau de bord.
4. Sous AWS Service Charges (Frais de service), développez Relational Database Service (Service de base de données relationnelle).

5. Développez l' Région AWS emplacement de vos instances de base de données réservées, par exemple USA West (Oregon).

   Vos instances de base de données réservées et leurs frais horaires pour le mois en cours sont indiqués sous Amazon Relational Database Service pour ***Database Engine*** Instances réservées.

   L'instance de base de données réservée dans cet exemple a été achetée avec un paiement total anticipé et dès lors, il n'existe pas de frais horaires.

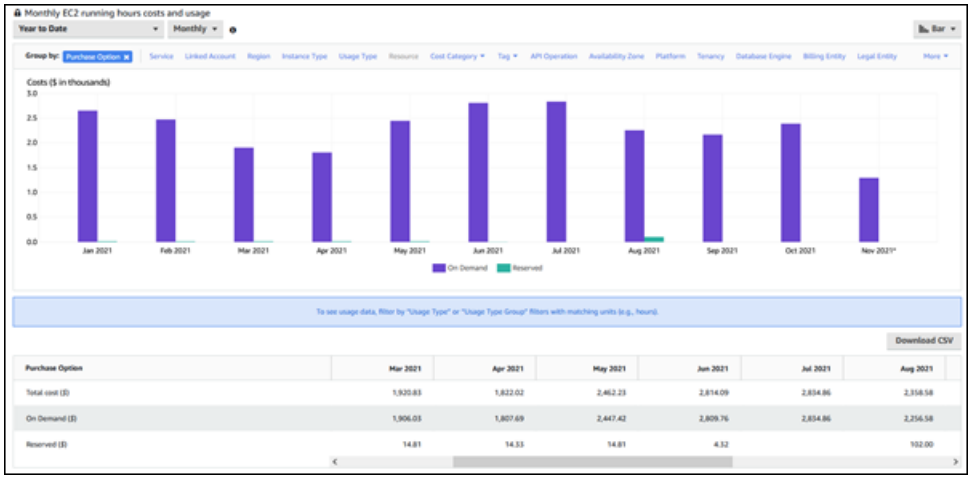
6. Cliquez sur l'icône Cost Explorer (graphique à barres) en regard de l'en-tête Reserved Instances.

   Le Cost Explorer affiche le graphique mensuel des heures de EC2 fonctionnement, des coûts et de l'utilisation.

7. Effacez le filtre Usage Type Group (Groupe de type d'utilisation) situé à droite du graphique.

8. Choisissez la période et l'unité de temps pour lesquelles vous souhaitez examiner les coûts d'utilisation.

   L'exemple suivant illustre les coûts d'utilisation mensuels des instances de base de données à la demande et réservées pour l'année écoulée.

   Les coûts des instances de base de données réservées de janvier à juin 2021 correspondent à des frais mensuels pour une instance avec frais initiaux partiels, tandis que les coûts d'août 2021 correspondent à des frais uniques pour une instance avec tous les frais initiaux.

La remise d'instance réservée pour l'instance avec frais initiaux partiels a expiré en juin 2021, mais l'instance de base de données n'a pas été supprimée. Après la date d'expiration, elle a simplement été facturée au tarif à la demande.

Configuration de votre environnement Amazon RDS

Cette page fournit un guide complet pour configurer Amazon Relational Database Service, y compris la configuration des comptes, la sécurité et la gestion des ressources. Il vous guide à travers les étapes essentielles pour créer, gérer et sécuriser efficacement vos environnements de base de données. Que vous utilisiez Amazon RDS pour la première fois ou que vous vous adaptiez à des exigences spécifiques, ces sections vous aident à garantir que votre configuration est optimisée et conforme aux meilleures pratiques.

Rubriques

- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif
- Octroi d’un accès par programmation
- Déterminer les exigences
- Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC

Si vous en avez déjà un Compte AWS, si vous connaissez vos exigences en matière d'Amazon RDS et que vous préférez utiliser les valeurs par défaut pour les groupes de sécurité IAM et VPC, passez directement à. Mise en route avec Amazon RDS

Inscrivez-vous pour un Compte AWS

Si vous n'en avez pas Compte AWS, procédez comme suit pour en créer un.

Pour vous inscrire à un Compte AWS

1. Ouvrez l'https://portal.aws.amazon.com/billing/inscription.
2. Suivez les instructions en ligne.

   Une partie de la procédure d'inscription consiste à recevoir un appel téléphonique ou un message texte et à saisir un code de vérification sur le clavier du téléphone.

   Lorsque vous vous inscrivez à un Compte AWS, un Utilisateur racine d'un compte AWSest créé. Par défaut, seul l’utilisateur racine a accès à l’ensemble des Services AWS et des ressources

de ce compte. La meilleure pratique de sécurité consiste à attribuer un accès administratif à un utilisateur, et à utiliser uniquement l'utilisateur racine pour effectuer les tâches nécessitant un accès utilisateur racine.

AWS vous envoie un e-mail de confirmation une fois le processus d'inscription terminé. À tout moment, vous pouvez consulter l'activité actuelle de votre compte et gérer votre compte en accédant à https://aws.amazon.com/et en choisissant Mon compte.

Création d'un utilisateur doté d'un accès administratif

Après vous être inscrit à un Compte AWS, sécurisez Utilisateur racine d'un compte AWS AWS IAM Identity Center, activez et créez un utilisateur administratif afin de ne pas utiliser l'utilisateur root pour les tâches quotidiennes.

Sécurisez votre Utilisateur racine d'un compte AWS

1. Connectez-vous en AWS Management Consoletant que propriétaire du compte en choisissant Utilisateur root et en saisissant votre adresse Compte AWS e-mail. Sur la page suivante, saisissez votre mot de passe.

   Pour obtenir de l'aide pour vous connecter en utilisant l'utilisateur racine, consultez Connexion en tant qu'utilisateur racine dans le Guide de l'utilisateur Connexion à AWS .
2. Activez l'authentification multifactorielle (MFA) pour votre utilisateur racine.

   Pour obtenir des instructions, voir Activer un périphérique MFA virtuel pour votre utilisateur Compte AWS root (console) dans le guide de l'utilisateur IAM.

Création d'un utilisateur doté d'un accès administratif

1. Activez IAM Identity Center.

   Pour obtenir des instructions, consultez Activation d' AWS IAM Identity Center dans le Guide de l'utilisateur AWS IAM Identity Center .
2. Dans IAM Identity Center, octroyez un accès administratif à un utilisateur.

   Pour un didacticiel sur l'utilisation du Répertoire IAM Identity Center comme source d'identité, voir Configurer l'accès utilisateur par défaut Répertoire IAM Identity Center dans le Guide de AWS IAM Identity Center l'utilisateur.

Connexion en tant qu'utilisateur doté d'un accès administratif

- Pour vous connecter avec votre utilisateur IAM Identity Center, utilisez l'URL de connexion qui a été envoyée à votre adresse e-mail lorsque vous avez créé l'utilisateur IAM Identity Center.

  Pour obtenir de l'aide pour vous connecter en utilisant un utilisateur d'IAM Identity Center, consultez la section Connexion au portail AWS d'accès dans le guide de l'Connexion à AWS utilisateur.

Attribution d'un accès à d'autres utilisateurs

1. Dans IAM Identity Center, créez un ensemble d'autorisations qui respecte la bonne pratique consistant à appliquer les autorisations de moindre privilège.

   Pour obtenir des instructions, consultez Création d'un ensemble d'autorisations dans le Guide de l'utilisateur AWS IAM Identity Center .
2. Attribuez des utilisateurs à un groupe, puis attribuez un accès par authentification unique au groupe.

   Pour obtenir des instructions, consultez Ajout de groupes dans le Guide de l'utilisateur AWS IAM Identity Center .

Octroi d'un accès par programmation

Les utilisateurs ont besoin d'un accès programmatique s'ils souhaitent interagir avec AWS l'extérieur du AWS Management Console. La manière d'accorder un accès programmatique dépend du type d'utilisateur qui y accède AWS.

Pour accorder aux utilisateurs un accès programmatique, choisissez l'une des options suivantes.

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
Identité de la main-d'œuvre (Utilisateurs gérés dans IAM Identity Center)	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées	Suivez les instructions de l'interface que vous souhaitez utiliser.

Quel utilisateur a besoin d'un accès programmatique ?	Pour	Par
	au AWS CLI AWS SDKs, ou AWS APIs.	• Pour le AWS CLI, voir Configuration du AWS CLI à utiliser AWS IAM Identity Center dans le guide de AWS Command Line Interface l'utilisateur. • Pour AWS SDKs, outils, et AWS APIs, voir Authentification IAM Identity Center dans le guide de référence AWS SDKs et Tools.
IAM	Utilisez des informations d'identification temporaires pour signer les demandes programmatiques adressées au AWS CLI AWS SDKs, ou AWS APIs.	Suivez les instructions de la section Utilisation d'informations d'identification temporaires avec AWS les ressources du Guide de l'utilisateur IAM.

Quel utilisateur a besoin d’un accès programmatique ?	Pour	Par
IAM	(Non recommandé) Utilisez des informations d'identification à long terme pour signer des demandes programmatiques adressées au AWS CLI AWS SDKs, ou AWS APIs.	Suivez les instructions de l’interface que vous souhaitez utiliser. • Pour le AWS CLI, voir Authentification à l'aide des informations d'identification utilisateur IAM dans le Guide de l'AWS Command Line Interface utilisateur. • Pour les outils AWS SDKs et, voir Authentifier à l'aide d'informations d'identif ication à long terme dans le guide de référence des outils AWS SDKs et. • Pour AWS APIs, voir Gestion des clés d'accès pour les utilisateurs IAM dans le Guide de l'utilisateur IAM.

Déterminer les exigences

La fondation de base d'Amazon RDS est l'instance de base de données. Dans une instance de base de données, vous pouvez créer vos bases de données. Une instance de base de données fournit une adresse réseau appelée point de terminaison. Vos applications utilisent ce point de terminaison pour se connecter à votre instance de base de données. Lorsque vous créez une instance de base de données, vous spécifiez des détails tels que le stockage, la mémoire, le moteur et la version de la base de données, la configuration réseau, la sécurité et les périodes de maintenance. Votre contrôlez l'accès réseau à une instance de base de données via un groupe de sécurité.

Avant de créer une instance de base de données et un groupe de sécurité, vous devez connaître les besoins en termes d'instances de base de données et de réseau. Voici quelques éléments importants à prendre en compte :

- Exigences en matière de ressources– Quelles sont les exigences de votre application ou de votre service en termes de mémoire et de processeur ? Vous utilisez ces paramètres pour déterminer plus facilement la classe d'instance de base de données à utiliser. Pour obtenir les caractéristiques des classes des instances de bases de données, consultez Classes d'instances de base de données .
- VPC, sous-réseau et groupe de sécurité– Votre instance de base de données se trouvera le plus probablement dans un Virtual Private Cloud (VPC). Pour vous connecter à votre instance de base de données, vous devez configurer des règles de groupes de sécurité. Ces règles sont configurées différemment selon le type de VPC que vous utilisez et selon la manière dont vous l'utilisez. Par exemple, vous pouvez utiliser : un VPC par défaut ou un VPC défini par l'utilisateur.

  La liste suivante décrit les règles pour chaque option de VPC :

  - VPC par défaut — Si votre AWS compte possède un VPC par défaut dans la région actuelle AWS , ce VPC est configuré pour prendre en charge les instances de base de données. Si vous spécifiez le VPC par défaut lorsque vous créez l'instance de base de données, procédez comme suit :
    - Assurez-vous de créer un groupe de sécurité VPC autorisant les connexions de l'application ou du service à l'instance de base de données Amazon RDS. Utilisez l'option Groupe de sécurité sur la console VPC ou pour créer des groupes AWS CLI de sécurité VPC. Pour plus d'informations, consultez Étape 3 : créer un groupe de sécurité VPC.
    - Spécifiez le groupe de sous-réseaux DB par défaut. S'il s'agit de la première instance de base de données que vous créez dans cette AWS région, Amazon RDS crée le groupe de sous-réseaux de base de données par défaut lors de la création de l'instance de base de données.
  - VPC défini par l'utilisateur– Si vous souhaitez spécifier un VPC défini par l'utilisateur lorsque vous créez une instance de base de données, tenez compte de ce qui suit :
    - Assurez-vous de créer un groupe de sécurité VPC autorisant les connexions de l'application ou du service à l'instance de base de données Amazon RDS. Utilisez l'option Groupe de sécurité sur la console VPC ou pour créer des groupes AWS CLI de sécurité VPC. Pour plus d'informations, consultez Étape 3 : créer un groupe de sécurité VPC.
    - Le VPC doit respecter certaines exigences afin d'héberger des instances de bases de données. Il doit notamment comporter au moins deux sous-réseaux, dans deux zones de disponibilités distinctes. Pour plus d'informations, consultez Amazon VPC et Amazon RDS.

    - Assurez-vous de spécifier un groupe de sous-réseaux DB définissant les sous-réseaux de ce VPC pouvant être utilisés par l'instance de base de données. Pour plus d'informations, consultez la section Groupe de sous-réseau DB de Utilisation d'un(e) instance de base de données dans un VPC.
- Haute disponibilité : avez-vous besoin de la prise en charge du basculement ? Sur Amazon RDS, un déploiement multi-AZ crée une instance de base de données principale et une instance de base de données de secours secondaire dans une autre zone de disponibilité pour la prise en charge du basculement. Nous recommandons les déploiements multi-AZ pour les charges de travail de production afin de maintenir une haute disponibilité. À des fins de développement et de test, vous pouvez utiliser un déploiement qui n'est pas multi-AZ. Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
- Politiques IAM — Votre AWS compte dispose-t-il de politiques qui accordent les autorisations nécessaires pour effectuer des opérations Amazon RDS ? Si vous vous connectez à AWS l'aide d'informations d'identification IAM, votre compte IAM doit disposer de politiques IAM qui accordent les autorisations requises pour effectuer des opérations Amazon RDS. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.
- Ports ouverts : sur quel TCP/IP port votre base de données écoute-t-elle ? Dans certaines entreprises, les pare-feu peuvent bloquer les connexions vers le port par défaut de votre moteur de base de données. Si le pare-feu de votre entreprise bloque le port par défaut, choisissez un autre port pour la nouvelle instance de base de données. Lorsque vous créez une instance de base de données qui écoute sur un port spécifié par vos soins, vous pouvez changer de port en modifiant l'instance de base de données.
- AWS Région — Dans quelle AWS région souhaitez-vous disposer de votre base de données ? La proximité entre votre base de données et votre application ou le service Web service permet de réduire la latence du réseau. Pour de plus amples informations, veuillez consulter Régions, zones de disponibilité et zones locales.
- Sous-système de disque de base de données : quels sont vos besoins en termes de stockage ? Amazon RDS propose trois types de stockages :
    - Usage général (SSD)
    - IOPS provisionnées (PIOPS)
    - Magnétique (également appelé stockage standard)

    Pour plus d'informations sur le stockage Amazon RDS, consultez Stockage d'instance de base de données Amazon RDS.

Lorsque vous disposez de toutes les informations nécessaires pour créer le groupe de sécurité et l'instance de base de données, passez à l'étape suivante.

Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC

Les groupes de sécurité VPC permettent l'accès aux instances de base de données dans un VPC. Ils font office de pare-feu pour l'instance de base de données associée, en contrôlant le trafic entrant et le trafic sortant au niveau de l'instance de base de données. Par défaut, les instances de base de données sont créées avec un pare-feu et un groupe de sécurité par défaut protégeant l'instance de base de données.

Avant de pouvoir vous connecter à votre instance de base de données, vous devez ajouter des règles à un groupe de sécurité qui vous permettent de vous connecter. Utilisez vos informations réseau et de configuration pour créer les règles autorisant l'accès à votre instance de base de données.

Prenons l'exemple d'une application qui a accès à une base de données sur votre instance de base de données dans un VPC. Dans ce cas, vous devez ajouter une règle TCP personnalisée qui spécifie la plage de ports et les adresses IP utilisées par votre application pour accéder à la base de données. Si vous avez une application sur une EC2 instance Amazon, vous pouvez utiliser le groupe de sécurité que vous avez configuré pour l' EC2 instance Amazon.

Vous pouvez configurer la connectivité entre une EC2 instance Amazon et une instance de base de données lorsque vous créez l'instance de base de données. Pour de plus amples informations, veuillez consulter Configuration de la connectivité réseau automatique avec une EC2 instance.

> **Tip**
>
> Vous pouvez configurer automatiquement la connectivité réseau entre une EC2 instance Amazon et une instance de base de données lorsque vous créez l'instance de base de données. Pour de plus amples informations, veuillez consulter Configuration de la connectivité réseau automatique avec une EC2 instance.

Pour plus d'informations sur la façon de connecter les ressources d'Amazon Lightsail à vos instances de base de données, consultez Connect Lightsail resources to using VPC peering. Services AWS

Pour plus d'informations sur les scénarios courants d'accès à une instance de base de données, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.

Pour créer un groupe de sécurité VPC

1. Connectez-vous à la console Amazon VPC AWS Management Console et ouvrez-la à https://console.aws.amazon.com l'adresse /vpc.

   > **Note**
   >
   > Assurez-vous que vous êtes dans la console VPC, et non dans la console RDS.

2. Dans le coin supérieur droit du AWS Management Console, choisissez la AWS région dans laquelle vous souhaitez créer votre groupe de sécurité VPC et votre instance de base de données. Dans la liste des ressources Amazon VPC pour cette région AWS , vous devriez voir au moins un VPC et plusieurs sous-réseaux. Si ce n'est pas le cas, vous n'avez pas de VPC par défaut dans cette AWS région.
3. Dans le panneau de navigation, choisissez Groupes de sécurité.
4. Sélectionnez Create security group (Créer un groupe de sécurité).

   La page Create security group (Créer un groupe de sécurité) s'affiche.
5. DansBasic details (Détails de base), renseignez les champs Security group name (Nom du groupe de sécurité) et Description. Pour VPC, choisissez le VPC dans lequel vous souhaitez créer votre instance de base de données.
6. Dans Inbound rules (Règles entrantes), choisissez Add rule (Ajouter une règle).

   a. Pour Type, choisissez Custom TCP (TCP personnalisé).

   b. Pour Port range (Plage de ports), saisissez le numéro du port à utiliser pour votre instance de base de données.

   c. Pour Source, choisissez un nom de groupe de sécurité ou tapez la plage d'adresses IP (valeur CIDR) à partir de laquelle vous accédez à l'instance de base de données. Si vous choisissez Mon IP, l'accès à l'instance de base de données est autorisé à partir de l'adresse IP détectée dans votre navigateur.
7. Si vous devez ajouter d'autres adresses IP ou des plages de ports différentes, choisissez Add rule (Ajouter une règle) et saisissez les informations relatives à la règle.
8. (Facultatif) Dans Outbound rules (Règles sortantes), ajoutez des règles pour le trafic sortant. Par défaut, tous les trafics sortant sont autorisés.

9. Sélectionnez Créer un groupe de sécurité.

Vous pouvez utiliser le groupe de sécurité VPC que vous venez de créer comme groupe de sécurité pour votre instance de base de données lors de sa création.

> **Note**
>
> Si vous utilisez un VPC par défaut, un groupe de sous-réseaux par défaut couvrant l'ensemble des sous-réseaux du VPC a déjà été créé pour vous. Lorsque vous créez une instance de base de données, vous pouvez sélectionner le VPC par défaut et utiliser par défaut en regard de DB Subnet Group (Groupe de sous-réseaux de base de données).

Une fois que vous avez terminé les exigences de configuration, vous pouvez créer une instance de base de données en utilisant votre configuration et votre groupe de sécurité. Pour ce faire, suivez les instructions dans Création d'une instance de base de données Amazon RDS. Pour plus d'informations sur le démarrage en créant une instance de base de données qui utilise un moteur de base de données spécifique, reportez-vous à la documentation pertinente dans le tableau suivant.

Moteur de base de données	Documentation
MariaDB	Création d'une instance de base de données MariaDB et connexion à cette instance
Microsoft SQL Server	Création et connexion à une instance de base de données Microsoft SQL Server
MySQL	Création d'une instance de base de données MySQL et connexion à cette instance
Oracle	Création et connexion à une instance de base de données Oracle
PostgreSQL	Création et connexion à une instance de base de données PostgreSQL

> **Note**
>
> Si vous ne parvenez pas à vous connecter à une instance de base de données après l'avoir créée, veuillez consulter les informations de dépannage dans Impossible de se connecter à l'instance de base de données Amazon RDS.

Mise en route avec Amazon RDS

Les exemples ci-dessous décrivent comment créer une instance de base de données et comment vous y connecter à l'aide d'Amazon Relational Database Service (Amazon RDS). Vous pouvez créer une instance de base de données qui utilise Db2, MariaDB, MySQL, Microsoft SQL Server, Oracle ou PostgreSQL.

> **Important**
>
> Vous devez réaliser les tâches de la section Configuration de votre environnement Amazon RDS avant de créer une instance de base de données ou de vous y connecter.

La création d'une instance de base de données et la connexion à une base de données d'une instance est légèrement différente pour chacun des moteurs de bases de données. Choisissez l'un des moteurs de base de données suivants que vous souhaitez utiliser pour en savoir plus sur la création et la connexion à l'instance de base de données. Une fois que vous avez assuré la création de votre instance de base de données, et la connexion à cette dernière, des instructions sont fournies pour vous aider à supprimer l'instance de base de données.

Rubriques

- Création d'une instance de base de données MariaDB et connexion à cette instance
- Création et connexion à une instance de base de données Microsoft SQL Server
- Création d'une instance de base de données MySQL et connexion à cette instance
- Création et connexion à une instance de base de données Oracle
- Création et connexion à une instance de base de données PostgreSQL
- Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS
- Tutoriel : Utilisation d'une fonction Lambda pour accéder à une base de données Amazon RDS

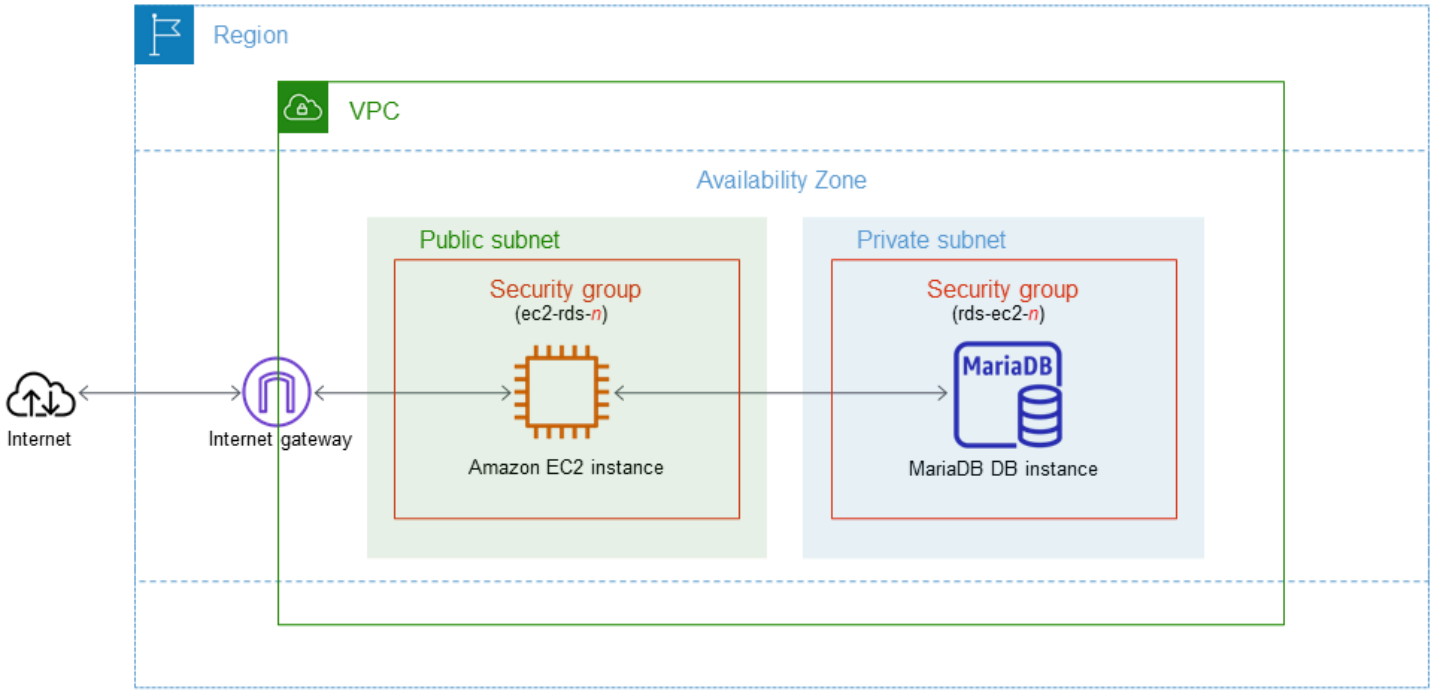
Création d'une instance de base de données MariaDB et connexion à cette instance

Ce didacticiel crée une EC2 instance et une instance de base de données RDS pour MariaDB. Le didacticiel explique comment accéder à l'instance de base de données depuis l' EC2 instance à l'aide d'un client MySQL standard. En tant que bonne pratique, ce didacticiel crée une instance de base de données privée dans un cloud privé virtuel (VPC). Dans la plupart des cas, les autres ressources du même VPC, telles que les EC2 instances, peuvent accéder à l'instance de base de données, mais les ressources extérieures au VPC ne peuvent pas y accéder.

Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Dans une zone de disponibilité, l' EC2 instance se trouve dans le sous-réseau public et l'instance de base de données dans le sous-réseau privé.

> **Important**
>
> La création d'un Compte AWS. Toutefois, au cours de ce didacticiel, des coûts peuvent être générés par l'utilisation des ressources . Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Ce didacticiel vous permet de créer vos ressources en utilisant l'une des méthodes suivantes :

1. Utilisez le AWS Management Console - Création d'une EC2 instance et Création d'une instance de base de données MariaDB
2. AWS CloudFormation À utiliser pour créer l'instance de base de données et l' EC2 instance - (Facultatif) Créez un VPC, une instance et une EC2 instance MariaDB en utilisant AWS CloudFormation

La première méthode utilise Easy create pour créer une instance de base de données MariaDB privée avec le. AWS Management Console Ici, vous spécifiez uniquement le type de moteur de base de données, la taille de l'instance de base de données et l'identifiant de l'instance de base de données. L'option Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration.

Lorsque vous utilisez plutôt Standard Create, vous pouvez spécifier d'autres options de configuration lorsque vous créez une instance de base de données. Ces options incluent les paramètres de disponibilité, de sécurité, de sauvegarde et de maintenance. Pour créer une instance de base de données publique, vous devez utiliser Création standard. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Rubriques

- Prérequis
- Création d'une EC2 instance
- Création d'une instance de base de données MariaDB
- (Facultatif) Créez un VPC, une instance et une EC2 instance MariaDB en utilisant AWS CloudFormation
- Se connecter à une instance de base de données MariaDB
- Supprimer l' EC2 instance et l'instance de base de données
- (Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation
- (Facultatif) Connecter votre instance de base de données à une fonction Lambda

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

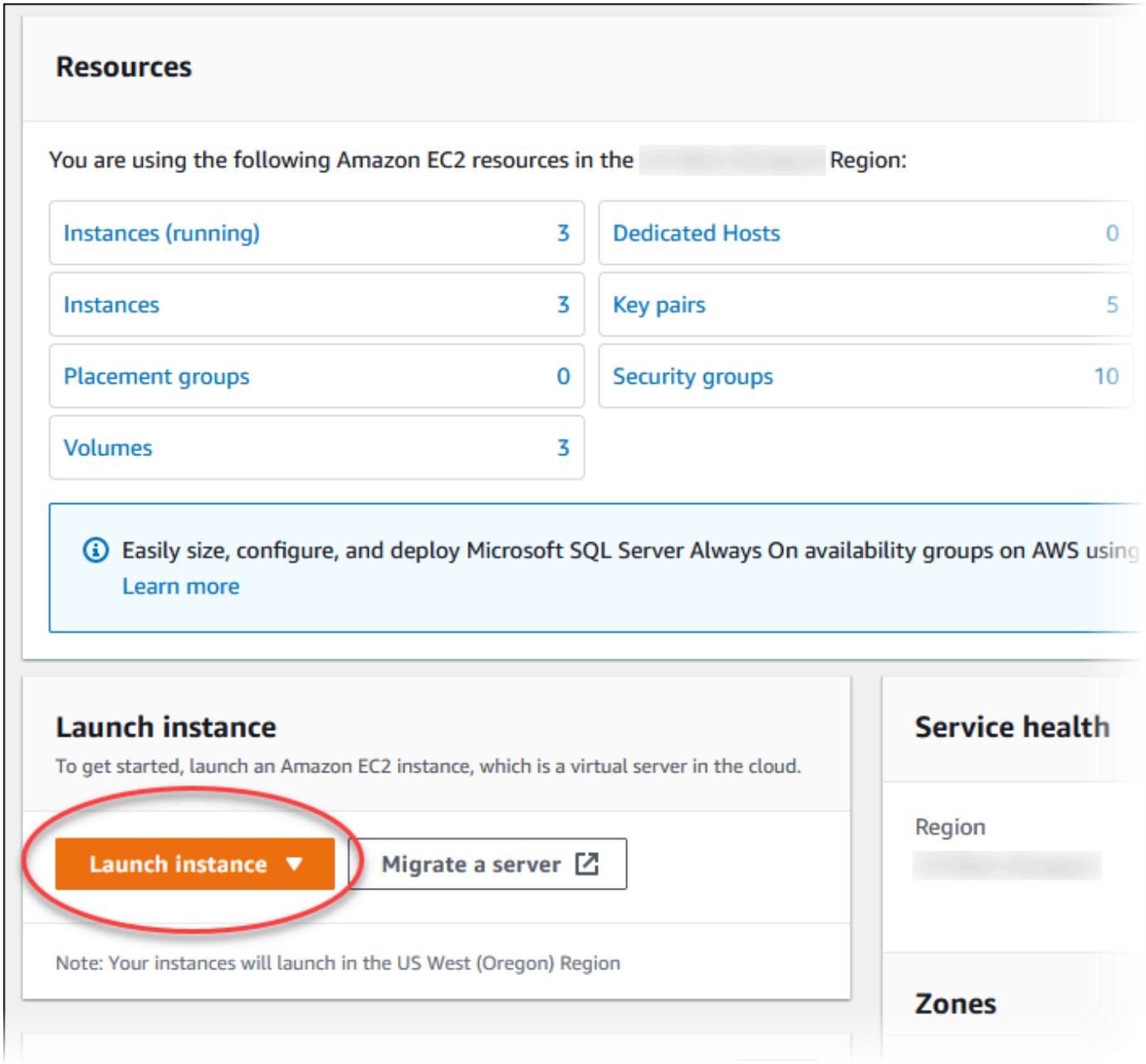
- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Création d'une EC2 instance

Créez une EC2 instance Amazon que vous utiliserez pour vous connecter à votre base de données.

Pour créer une EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit de AWS Management Console, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l' EC2 instance.
3. Choisissez EC2 Dashboard, puis Launch instance, comme illustré dans l'image suivante.

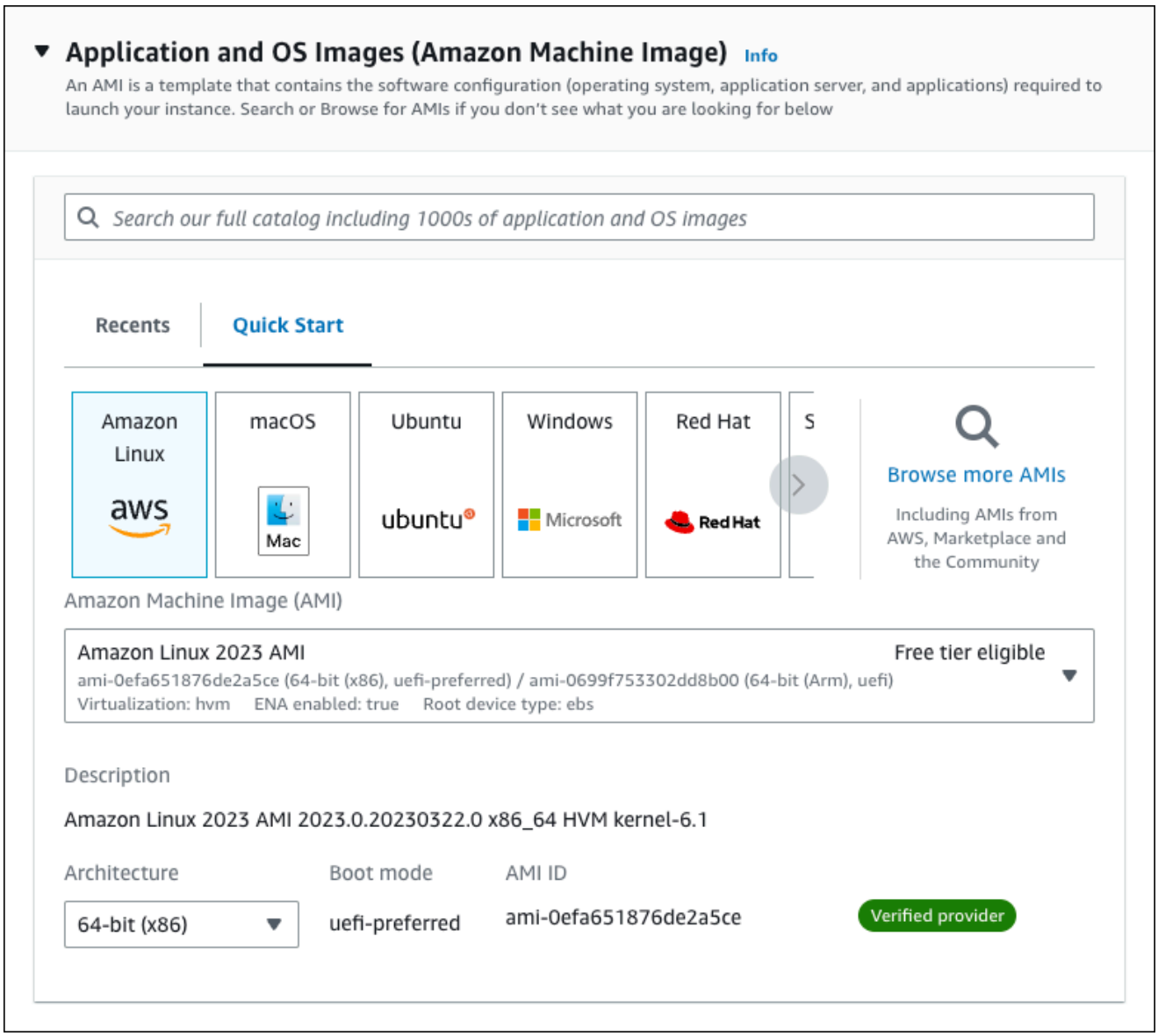


La page Lancer une instance s'ouvre.

4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`ec2-database-connect`**.

   b. Sous Application et images OS (Amazon Machine Image), choisissez Amazon Linux, puis Amazon Linux 2023 AMI. Conservez les sélections par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l' EC2 instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le guide de EC2 l'utilisateur Amazon.

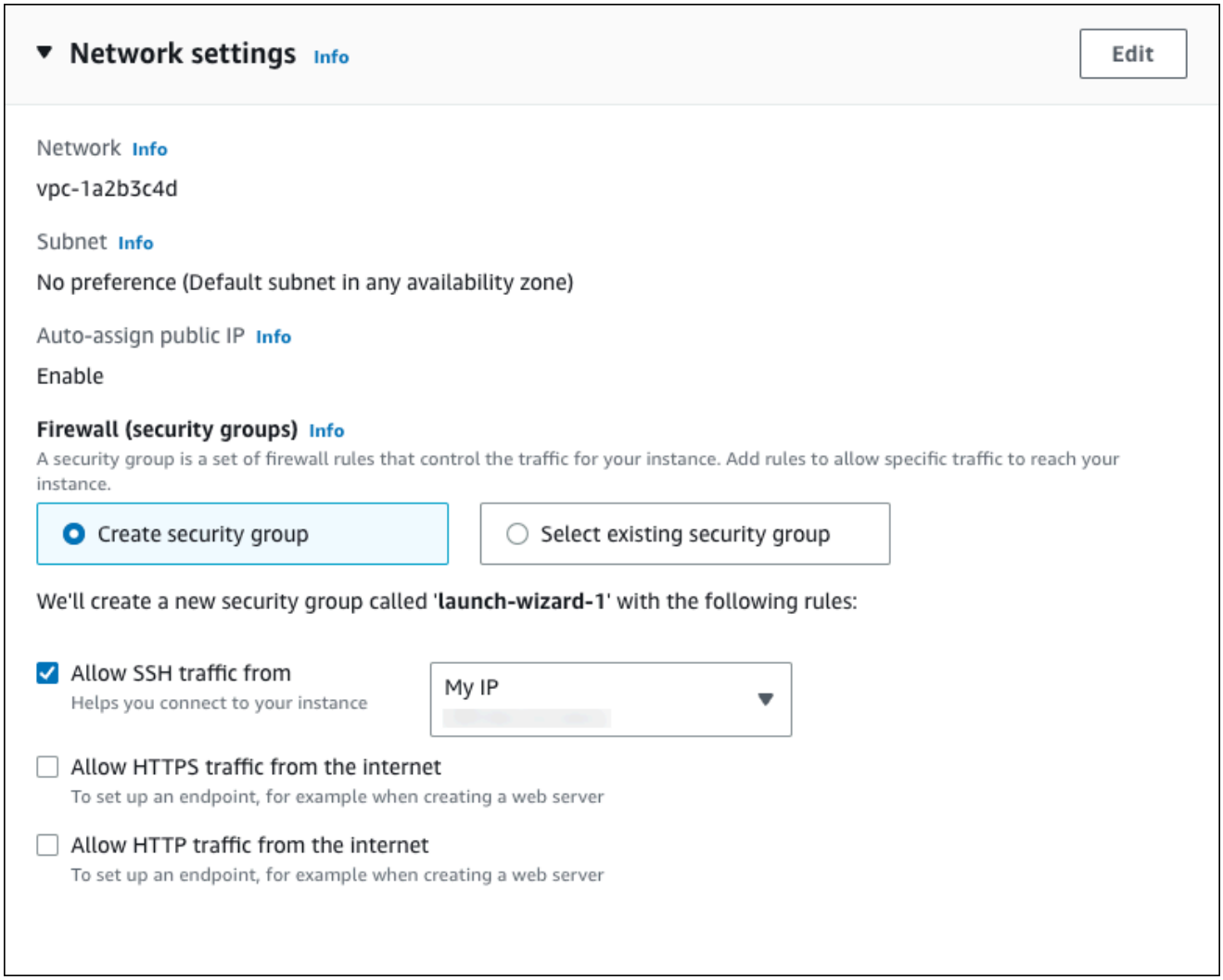
e. Pour Autoriser le trafic SSH dans les paramètres réseau, choisissez la source des connexions SSH à l' EC2instance.

Vous pouvez choisir My IP (Mon IP) si l'adresse IP affichée est correcte pour les connexions SSH. Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : 192.0.2.1/32.

Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

> **Warning**
>
> Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

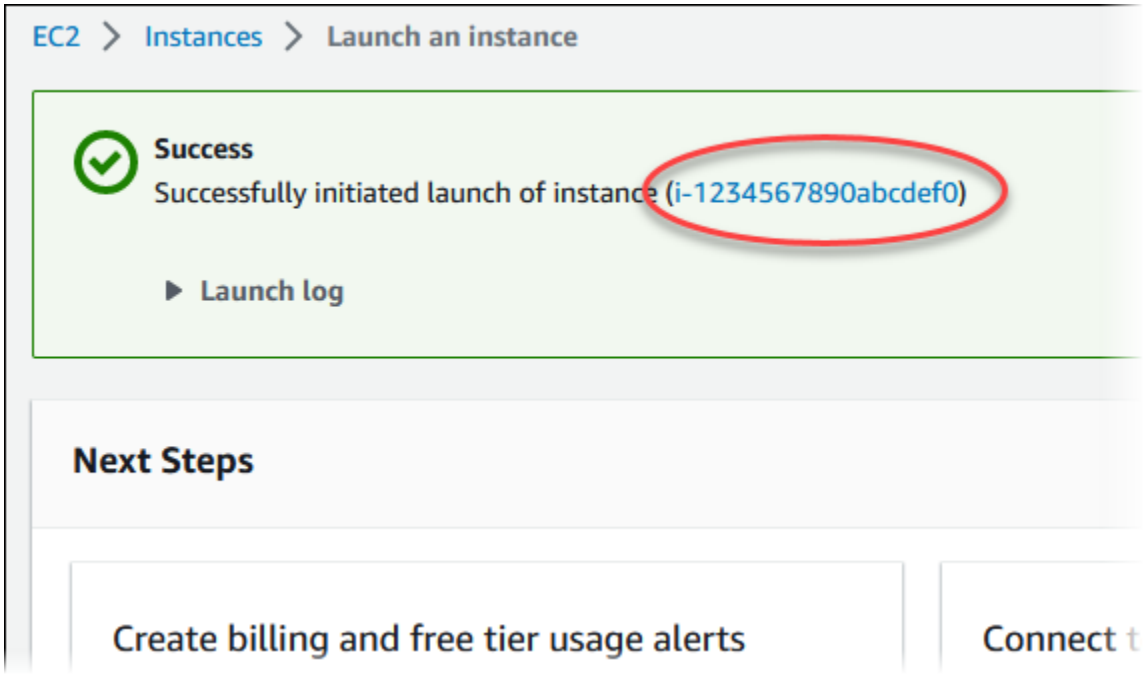
L'image suivante présente un exemple de la section Paramètres réseau.



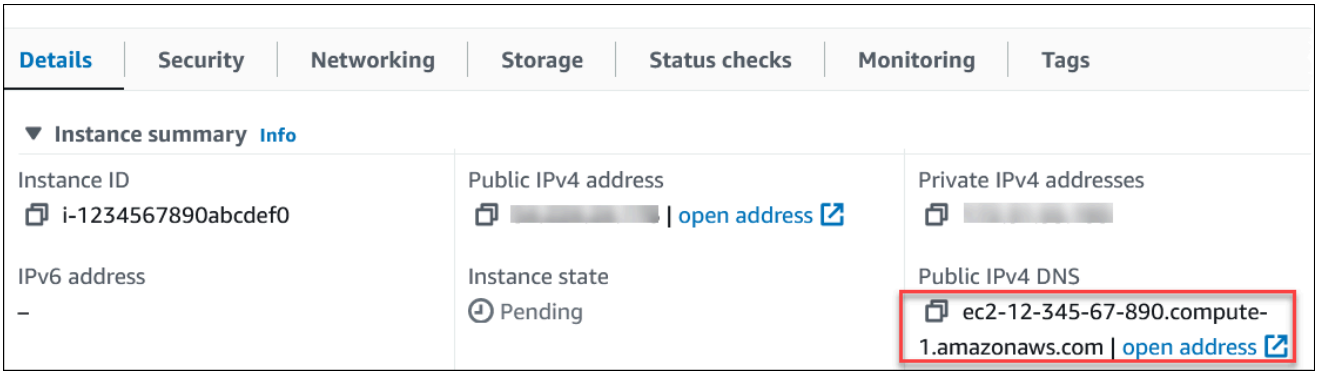
f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre EC2 instance dans le panneau Résumé et, lorsque vous êtes prêt, choisissez Launch instance.

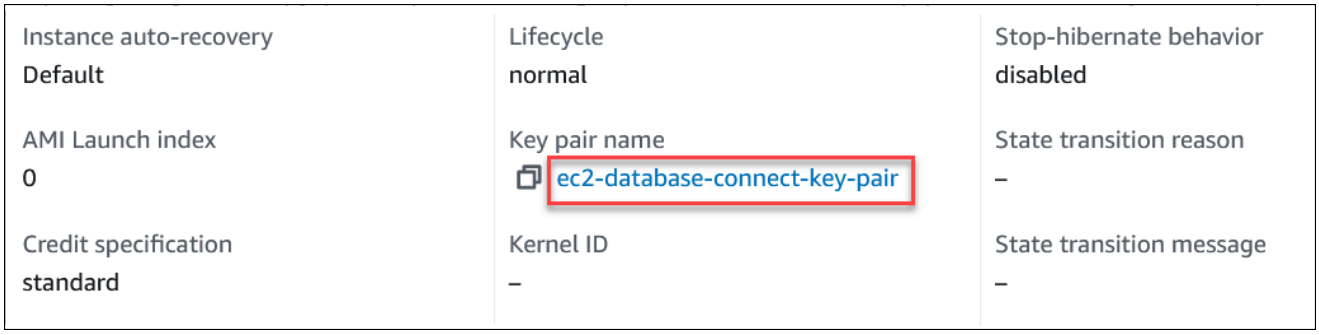
5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant d' EC2 instance pour ouvrir la liste des EC2 instances, puis sélectionnez votre EC2 instance.
7. Dans l'onglet Détails, notez les valeurs suivantes. Vous en aurez besoin lorsque vous vous connecterez via SSH :
   a. Dans Résumé de l'instance, notez la valeur du IPv4 DNS public.



   b. Dans Détails de l'instance, notez la valeur pour Nom de la paire de clés.



8. Attendez que l'état Instance de votre EC2 instance soit défini sur En cours d'exécution avant de continuer.

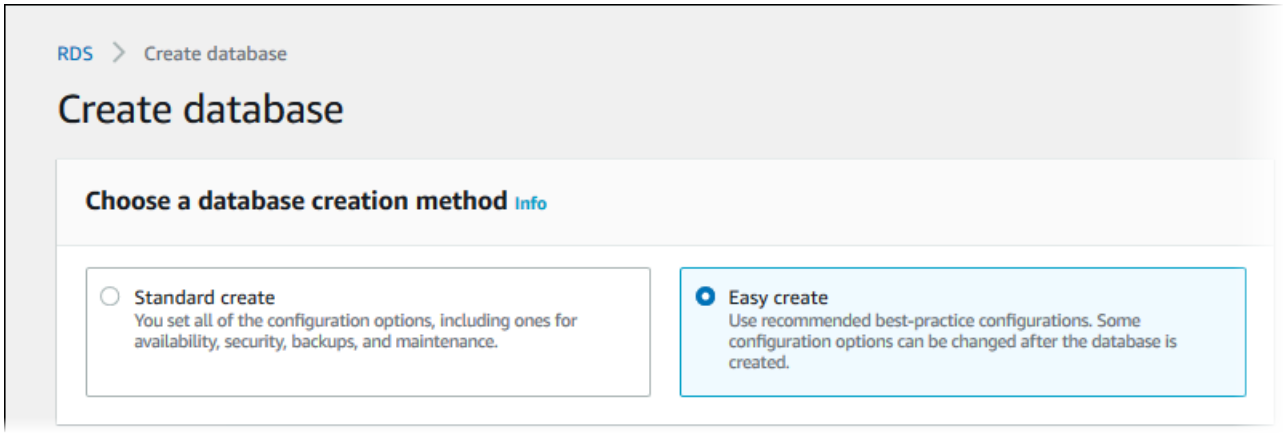
Création d'une instance de base de données MariaDB

La fondation de base d'Amazon RDS est l'instance de base de données. Il s'agit de l'environnement dans lequel vous exécutez vos bases de données MariaDB.

Dans cet exemple, vous utilisez Easy create pour créer une instance de base de données exécutant le moteur de base de données MariaDB avec une classe d'instance de base de données db.t4g.micro.

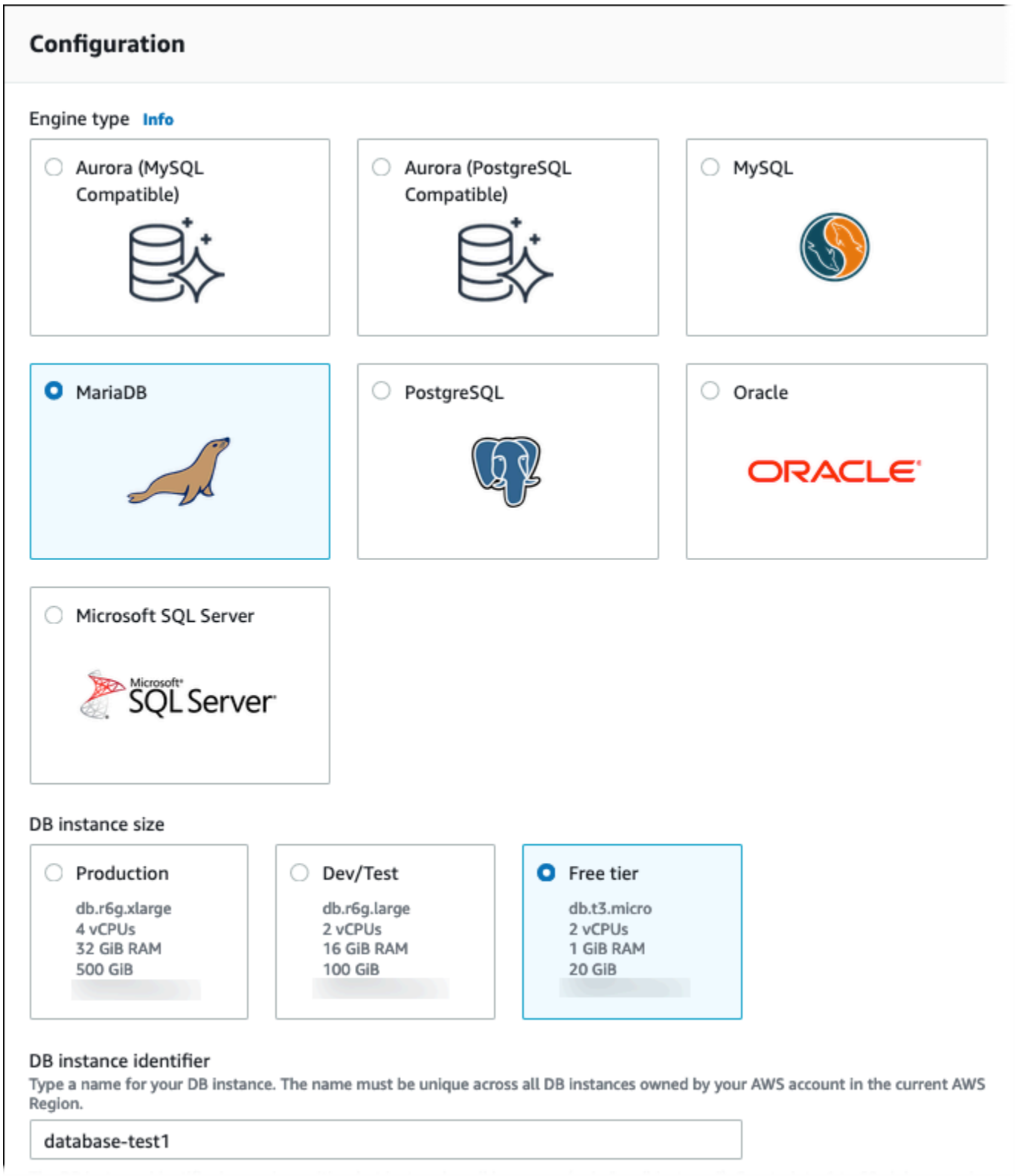
Pour créer une instance de base de données MariaDB avec l'option Easy create (Création facile)

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l'instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Créer une base de données et sélectionnez Création facile.



5. Dans Configuration, choisissez MariaDB.
6. Pour la taille de l'instance de base de données, choisissez Free tier ou Sandbox. Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.
7. Pour l'identifiant de l'instance DB, saisissez **`database-test1`**.
8. Pour Nom d'utilisateur principal, saisissez un nom pour l'utilisateur principal ou conservez le nom par défaut.

   La page Create database (Créer une base de données) doit ressembler à l'image suivante. Pour les comptes de forfait gratuit, le niveau gratuit apparaît. Pour les comptes de forfaits payants, Sandbox apparaît.

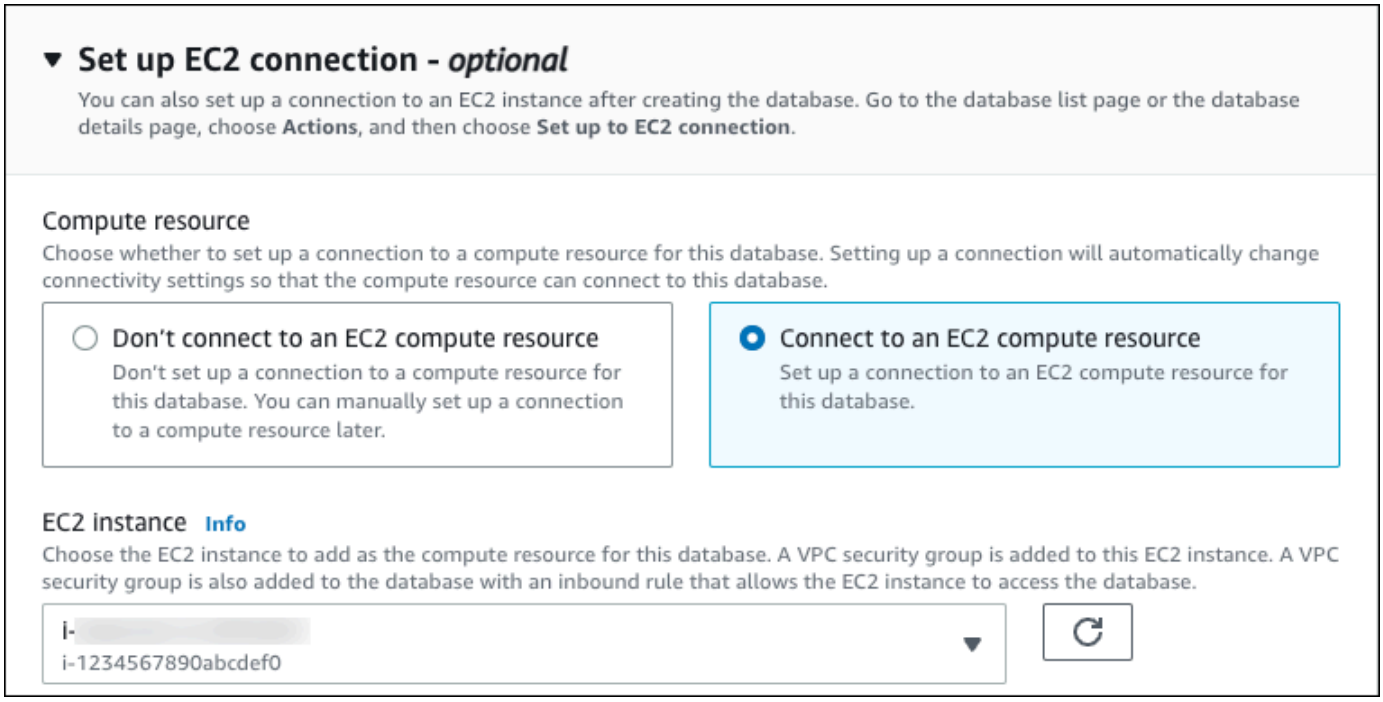


9. Pour utiliser un mot de passe principal généré automatiquement pour l'instance de base de données, sélectionnez Générer automatiquement un mot de passe.

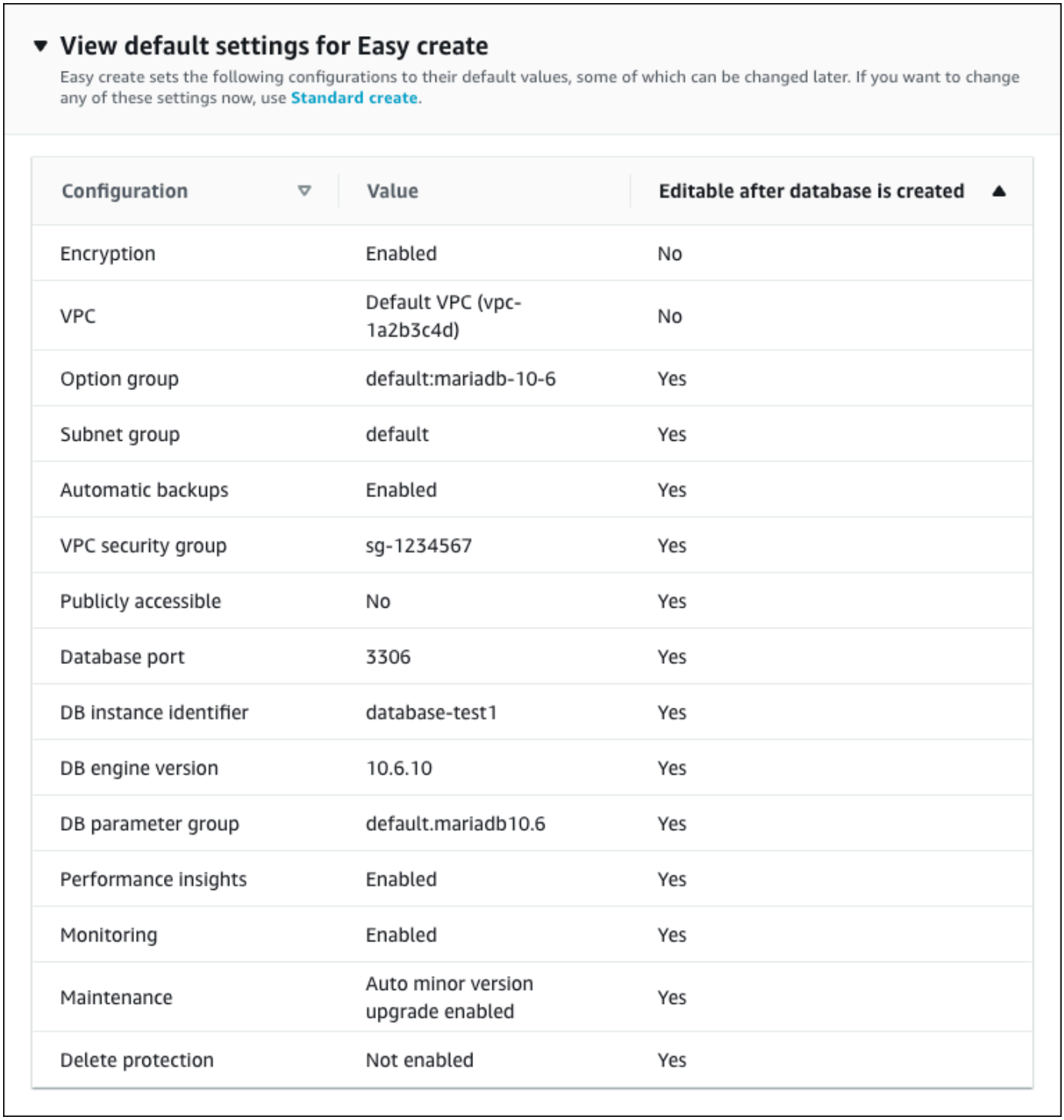
Pour saisir votre mot de passe principal, décochez la case Générer automatiquement un mot de passe, puis entrez le même mot de passe dans Mot de passe principal et Confirmer le mot de passe principal.

10. Pour configurer une connexion avec l' EC2 instance que vous avez créée précédemment, développez Configurer la EC2 connexion - facultatif.

    Sélectionnez Se connecter à une ressource EC2 de calcul. Choisissez l' EC2instance que vous avez créée précédemment.



11. Développez les paramètres par défaut de View pour créer facilement.



View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

Configuration	Value	Editable after database is created
Encryption	Enabled	No
VPC	Default VPC (vpc-1a2b3c4d)	No
Option group	default:mariadb-10-6	Yes
Subnet group	default	Yes
Automatic backups	Enabled	Yes
VPC security group	sg-1234567	Yes
Publicly accessible	No	Yes
Database port	3306	Yes
DB instance identifier	database-test1	Yes
DB engine version	10.6.10	Yes
DB parameter group	default.mariadb10.6	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

Vous pouvez examiner les paramètres par défaut utilisés quand l'option Easy create (Création facile) est activée. La colonne Modifiable après la création de la base de données indique les options que vous pouvez modifier après avoir créé la base de données.

- Si un réglage contient Non dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données.
- Si un réglage contient Oui dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données ou vous pouvez modifier l'instance de base de données après l'avoir créée pour modifier le réglage.

12. Choisissez Créer une base de données.

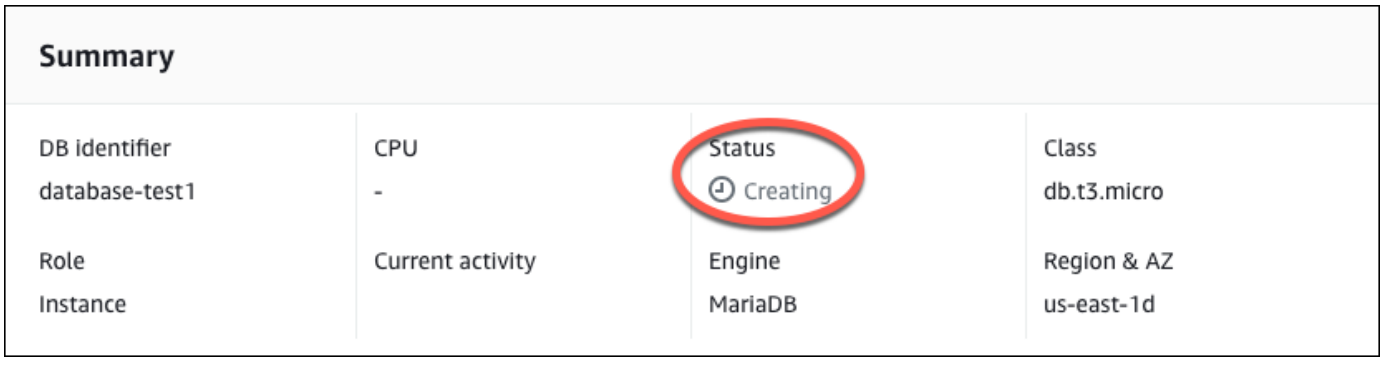
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Vous pouvez utiliser l'identifiant et le mot de passe affichés pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier.
    > Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, vous pouvez le faire en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

13. Dans la liste Bases de données, choisissez le nom de la nouvelle instance de base de données MariaDB pour afficher ses détails.

    L'instance de base de données a le statut Création en cours jusqu'à ce qu'elle soit prête à l'emploi.



Lorsque l’état passe à Available (Disponible), vous pouvez vous connecter à l’instance de base de données. En fonction de la quantité de stockage et de la classe d’instance de base de données, la mise à disposition de la nouvelle instance peut prendre jusqu’à 20 minutes.

(Facultatif) Créez un VPC, une instance et une EC2 instance MariaDB en utilisant AWS CloudFormation

Au lieu d'utiliser la console pour créer votre VPC, votre instance et votre EC2 instance MariaDB, vous pouvez les utiliser AWS CloudFormation pour provisionner des AWS ressources en traitant l'infrastructure comme du code. Pour vous aider à organiser vos AWS ressources en unités plus petites et plus faciles à gérer, vous pouvez utiliser la fonctionnalité de pile AWS CloudFormation imbriquée. Pour plus d'informations, consultez les sections Création d'une pile sur la AWS CloudFormation console et Utilisation de piles imbriquées.

> **Important**
>
> AWS CloudFormation est gratuit, mais les ressources qui en CloudFormation découlent sont vivantes. Vous devez payer les frais d'utilisation standard pour ces ressources jusqu'à ce que vous y mettiez fin. Pour plus d'informations, consultez la section Tarification de RDS pour MariaDB.

- Téléchargez le CloudFormation modèle
- Configurez vos ressources à l'aide de CloudFormation

Téléchargez le CloudFormation modèle

Un CloudFormation modèle est un fichier texte JSON ou YAML qui contient les informations de configuration relatives aux ressources que vous souhaitez créer dans la pile. Ce modèle crée également un VPC et un hôte bastion pour vous, ainsi que l'instance RDS.

Pour télécharger le fichier modèle, ouvrez le lien suivant, modèle CloudFormation MariaDB.

Sur la page Github, cliquez sur le bouton Télécharger le fichier brut pour enregistrer le modèle de fichier YAML.

Configurez vos ressources à l'aide de CloudFormation

> **Note**
>
> Avant de commencer ce processus, assurez-vous que vous disposez d'une paire de clés pour une EC2 instance dans votre Compte AWS. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Lorsque vous utilisez le AWS CloudFormation modèle, vous devez sélectionner les paramètres appropriés pour vous assurer que vos ressources sont créées correctement. Procédez de la façon suivante :

1. Connectez-vous à la AWS CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile.
3. Dans la section Spécifier le modèle, sélectionnez Télécharger un fichier modèle depuis votre ordinateur, puis cliquez sur Suivant.
4. Dans la page Spécifier les détails de la pile, définissez les paramètres suivants :
   a. Définissez le nom de la pile sur Maria DBTest Stack.
   b. Sous Paramètres, définissez les zones de disponibilité en sélectionnant trois zones de disponibilité.
   c. Dans Configuration de l'hôte Linux Bastion, pour Nom de la clé, sélectionnez une paire de clés pour vous connecter à votre EC2 instance.
   d. Dans les paramètres de configuration de l'hôte Linux Bastion, définissez la plage d'adresses IP autorisées sur votre adresse IP. Pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH), déterminez votre adresse IP publique à l'aide du service à l'adresse. https://checkip.amazonaws.com Exemple d'adresse IP : 192.0.2.1/32.

      > **Warning**
      >
      > Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les

environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

e. Dans Configuration générale de la base de données, définissez la classe d'instance de base de données sur db.t3.micro.

f. Définissez le nom de base de données sur**`database-test1`**.

g. Dans Nom d'utilisateur principal de base de données, entrez le nom de l'utilisateur principal.

h. Définissez le mot de passe utilisateur principal de Manage DB avec Secrets Manager sur `false` pour ce didacticiel.

i. Pour le mot de passe de la base de données, définissez le mot de passe de votre choix. N'oubliez pas ce mot de passe pour suivre les étapes suivantes du didacticiel.

j. Sous Configuration du stockage de base de données, définissez le type de stockage de base de données sur gp2.

k. Dans Configuration de la surveillance des bases de données, définissez Enable RDS Performance Insights sur false.

l. Conservez tous les autres paramètres comme valeurs par défaut. Cliquez sur Suivant pour continuer.

5. Sur la page Review stack, sélectionnez Soumettre après avoir vérifié les options de la base de données et de l'hôte Linux Bastion.

Une fois le processus de création des piles terminé, visualisez les piles avec leurs noms BastionStacket leurs RDSNS pour noter les informations dont vous avez besoin pour vous connecter à la base de données. Pour plus d'informations, consultez la section Affichage des données et des ressources de la AWS CloudFormation pile sur le AWS Management Console.

Se connecter à une instance de base de données MariaDB

Vous pouvez utiliser n'importe quelle application client SQL standard pour vous connecter à l'instance de base de données. Dans cet exemple, vous vous connectez à une instance de base de données MariaDB en utilisant le client de ligne de commande mysql.

Pour vous connecter à une instance de base de données MariaDB

1. Trouvez le point de terminaison (nom DNS) et le numéro de port pour votre instance de base de données.

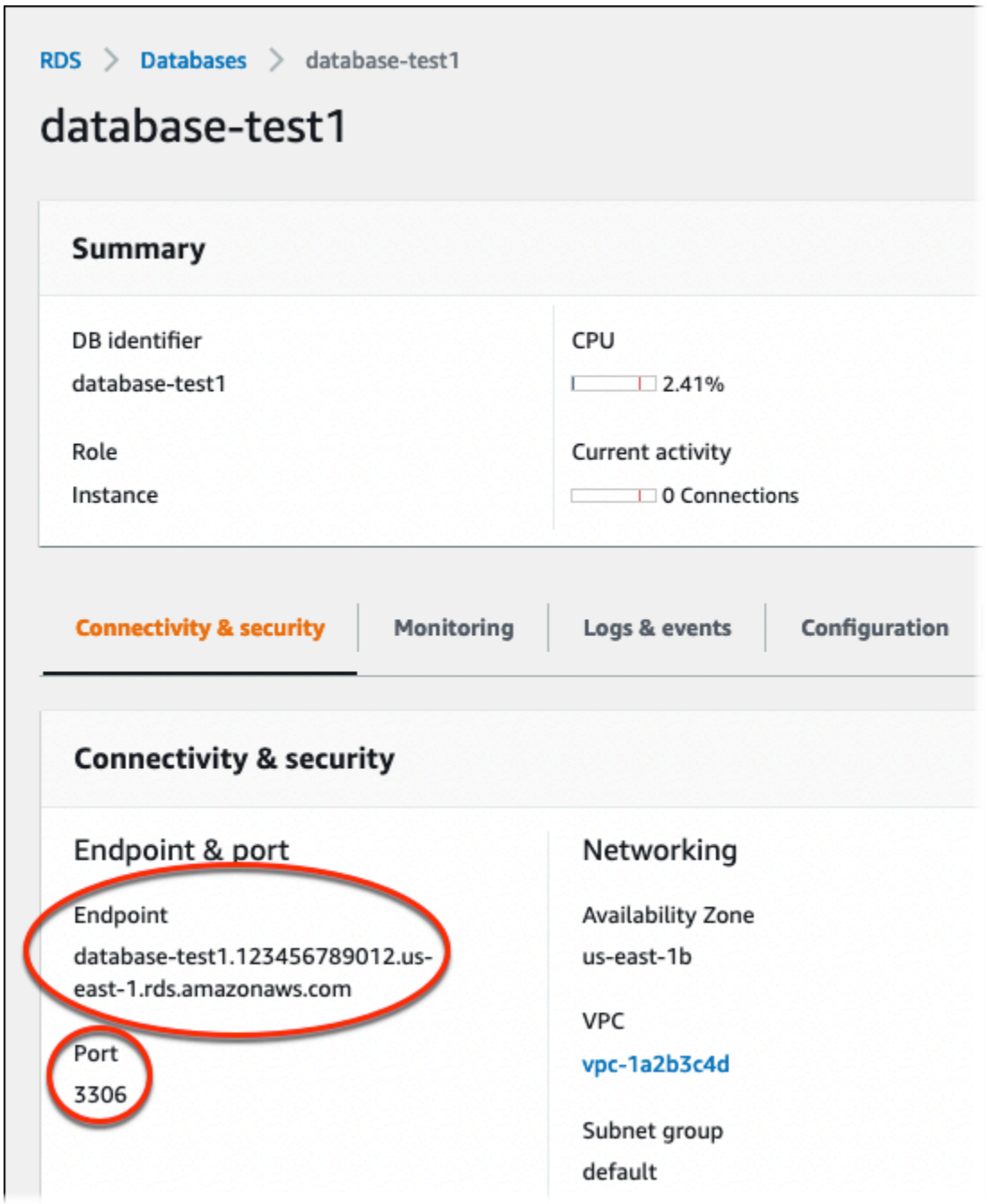
a. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

b. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance de base Région AWS de données.

c. Dans le panneau de navigation, choisissez Databases (Bases de données).

d. Choisissez le nom de l'instance de base de données MariaDB pour afficher ses détails.

e. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



2. Connectez-vous à l' EC2 instance que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Linux dans le guide de EC2 l'utilisateur Amazon.

   Nous vous recommandons de vous connecter à votre EC2 instance via SSH. Si l'utilitaire client SSH est installé sur Windows, Linux ou Mac, vous pouvez vous connecter à l'instance à l'aide du format de commande suivant :

```
ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
```

Supposons, par exemple, qu'`ec2-database-connect-key-pair.pem`il soit stocké `/dir1` sous Linux et que le IPv4 DNS public de votre EC2 instance l'est`ec2-12-345-678-90.compute-1.amazonaws.com`. Votre commande SSH se présenterait comme suit :

```
ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-
user@ec2-12-345-678-90.compute-1.amazonaws.com
```

3. Obtenez les dernières corrections de bogues et mises à jour de sécurité en mettant à jour le logiciel de votre EC2 instance. Pour ce faire, exécutez la commande suivante.

> **Note**
>
> L'option `-y` installe les mises à jour sans demander de confirmation. Pour examiner les mises à jour avant de les installer, omettez cette option.

```
sudo dnf update -y
```

4. Installez le client de ligne de commande mysql depuis MariaDB.

   Pour installer le client de ligne de commande MariaDB sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install mariadb105
```

5. Connectez-vous à l'instance de base de données MariaDB. Par exemple, saisissez la commande suivante. Cette action vous permet de vous connecter à l'instance de base de données MariaDB à l'aide du client MySQL.

   Remplacez le point de terminaison de votre instance de base de données (nom DNS) pour *`endpoint`* et remplacez le nom d'utilisateur principal que vous avez utilisé pour *`admin`*. Indiquez le mot de passe principal que vous avez utilisé lorsque vous êtes invité à entrer un mot de passe.

```
mysql -h endpoint -P 3306 -u admin -p
```

Après avoir entré le mot de passe pour l'utilisateur, le résultat suivant devrait normalement s'afficher.

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 156
Server version: 10.6.10-MariaDB-log managed by https://aws.amazon.com/rds/

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Pour de plus amples informations sur la connexion à une instance de base de données MariaDB, veuillez consulter Connexion à votre instance de base de données MariaDB. Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Pour des raisons de sécurité, une bonne pratique consiste à recommander d'utiliser des connexions chiffrées. Utilisez une connexion MariaDB non chiffrée uniquement quand le client et le serveur sont dans le même VPC et que le réseau est approuvé. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Connexion à votre instance de base de données MariaDB sur RDS Amazon TLS avec/ depuis le client SSL de ligne de commande SQL My (crypté).

6. Exécutez des commandes SQL.

   Par exemple, la commande SQL suivante indique la date et l'heure actuelles :

   ```
   SELECT CURRENT_TIMESTAMP;
   ```

Supprimer l' EC2 instance et l'instance de base de données

Après vous être connecté et exploré l'exemple d' EC2 instance et l'instance de base de données que vous avez créées, supprimez-les afin qu'ils ne vous soient plus facturés.

Si vous aviez AWS CloudFormation l'habitude de créer des ressources, ignorez cette étape et passez à l'étape suivante.

Pour supprimer l' EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l' EC2 instance, puis choisissez État de l'instance, Terminer l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une EC2 instance, consultez Résilier votre instance dans le guide de EC2 l'utilisateur Amazon.

Pour supprimer l'instance de base de données sans instantané de base de données final

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous voulez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Décochez Créer un instantané final et Conserver les sauvegardes automatiques.
6. Terminez la confirmation et choisissez Supprimer.

(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation

Si vous aviez l'habitude AWS CloudFormation de créer des ressources, supprimez la CloudFormation pile une fois que vous vous êtes connecté à l'instance d'exemple et à l' EC2 instance de base de données et que vous les avez explorées, afin de ne plus être facturées.

Pour supprimer les CloudFormation ressources

1. Ouvrez la AWS CloudFormation console.
2. Sur la page Stacks de la CloudFormation console, sélectionnez la pile racine (la pile sans nom VPCStack, BastionStack ou RDSNS).
3. Sélectionnez Delete (Supprimer).
4. Sélectionnez Supprimer la pile lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une pile dans CloudFormation, voir Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Connecter votre instance de base de données à une fonction Lambda

Vous pouvez également connecter votre instance de base de données RDS for MariaDB à une ressource de calcul sans serveur Lambda. Les fonctions Lambda vous permettent d'exécuter du code sans provisionner ni gérer l'infrastructure. Une fonction Lambda vous permet également de répondre automatiquement aux demandes d'exécution de code à n'importe quelle échelle, d'une douzaine d'événements par jour à des centaines par seconde. Pour de plus amples informations, veuillez consulter Connexion automatique d'une fonction Lambda et d'une instance de base de données.

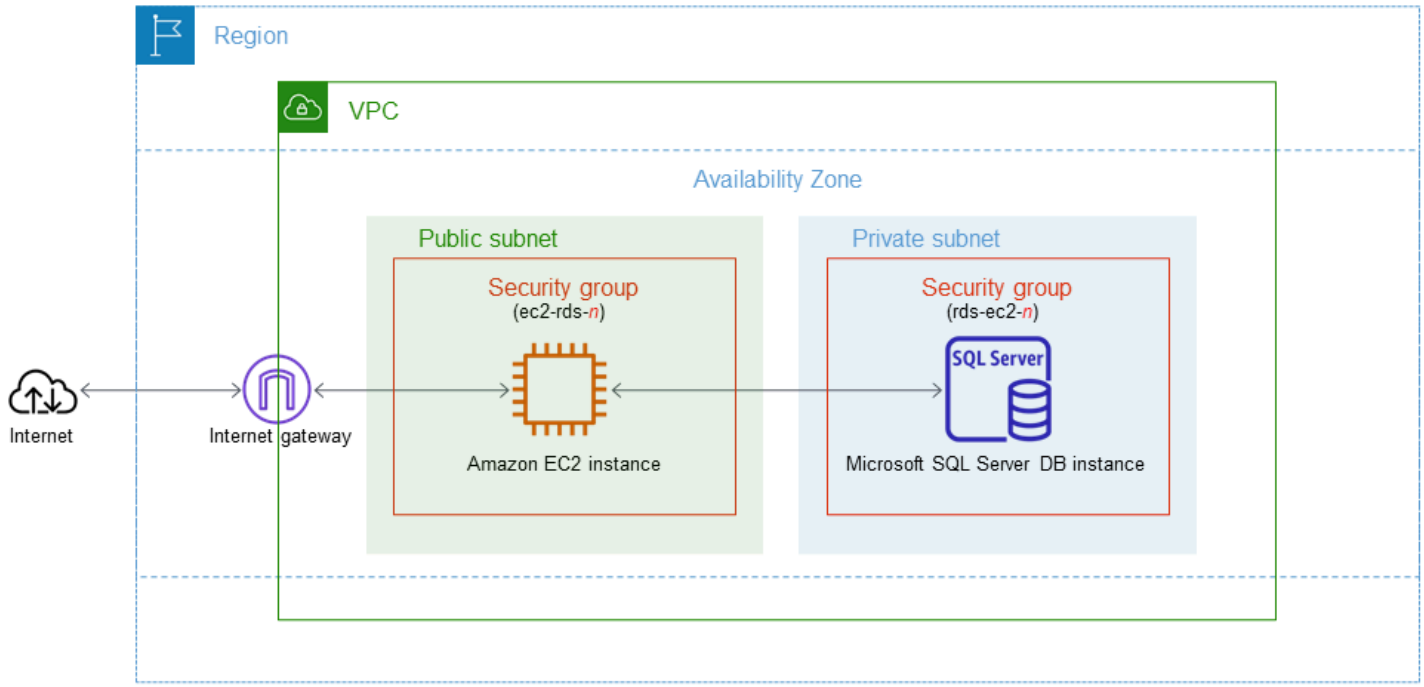
Création et connexion à une instance de base de données Microsoft SQL Server

Ce didacticiel crée une EC2 instance et une instance de base de données RDS pour Microsoft SQL Server. Le didacticiel explique comment accéder à l'instance de base de données à partir de l' EC2 instance à l'aide du client Microsoft SQL Server Management Studio. En tant que bonne pratique, ce didacticiel crée une instance de base de données privée dans un cloud privé virtuel (VPC). Dans la plupart des cas, les autres ressources du même VPC, telles que les EC2 instances, peuvent accéder à l'instance de base de données, mais les ressources extérieures au VPC ne peuvent pas y accéder.

Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Dans une zone de disponibilité, l' EC2 instance se trouve dans le sous-réseau public et l'instance de base de données dans le sous-réseau privé.

> **Important**
>
> La création d'un AWS compte est gratuite. Cependant, en suivant ce didacticiel, les AWS ressources que vous utilisez peuvent vous coûter cher. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Ce didacticiel vous permet de créer vos ressources en utilisant l'une des méthodes suivantes :

1. Utilisez le AWS Management Console - Création d'une instance de base de données SQL Server et Création d'une instance EC2
2. AWS CloudFormation À utiliser pour créer l'instance de base de données et l' EC2 instance - (Facultatif) Créez un VPC, une instance et une EC2 instance SQL Server à l'aide de AWS CloudFormation

La première méthode utilise Easy create pour créer une instance de base de données SQL Server privée avec le AWS Management Console. Ici, vous spécifiez uniquement le type de moteur de base de données, la taille de l'instance de base de données et l'identifiant de l'instance de base de données. L'option Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration.

Lorsque vous utilisez plutôt Standard Create, vous pouvez spécifier d'autres options de configuration lorsque vous créez une instance de base de données. Ces options incluent les paramètres de disponibilité, de sécurité, de sauvegarde et de maintenance. Pour créer une instance de base de données publique, vous devez utiliser Création standard. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Rubriques

- Prérequis
- Création d'une instance EC2
- Création d'une instance de base de données SQL Server
- (Facultatif) Créez un VPC, une instance et une EC2 instance SQL Server à l'aide de AWS CloudFormation
- Connectez-vous à votre instance de base de données SQL Server
- Explorez votre exemple d'instance de base de données SQL Server
- Supprimer l' EC2instance et l'instance de base de données
- (Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation
- (Facultatif) Connecter votre instance de base de données à une fonction Lambda

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

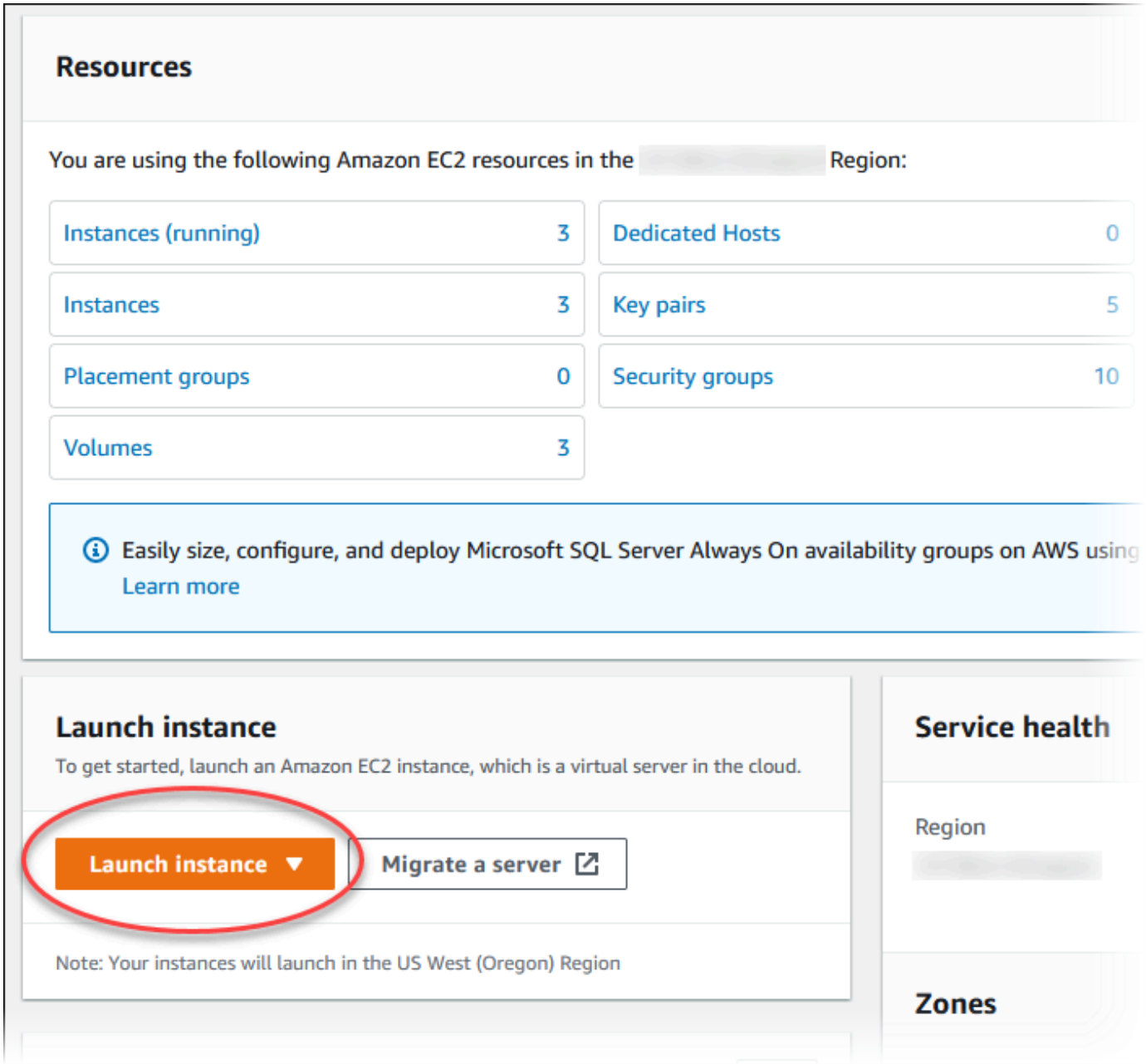
- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Création d'une instance EC2

Créez une EC2 instance Amazon que vous utiliserez pour vous connecter à votre base de données.

Pour créer une EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit du AWS Management Console, choisissez celui que Région AWS vous avez utilisé pour la base de données précédemment.
3. Choisissez EC2 Dashboard, puis Launch instance, comme illustré dans l'image suivante.

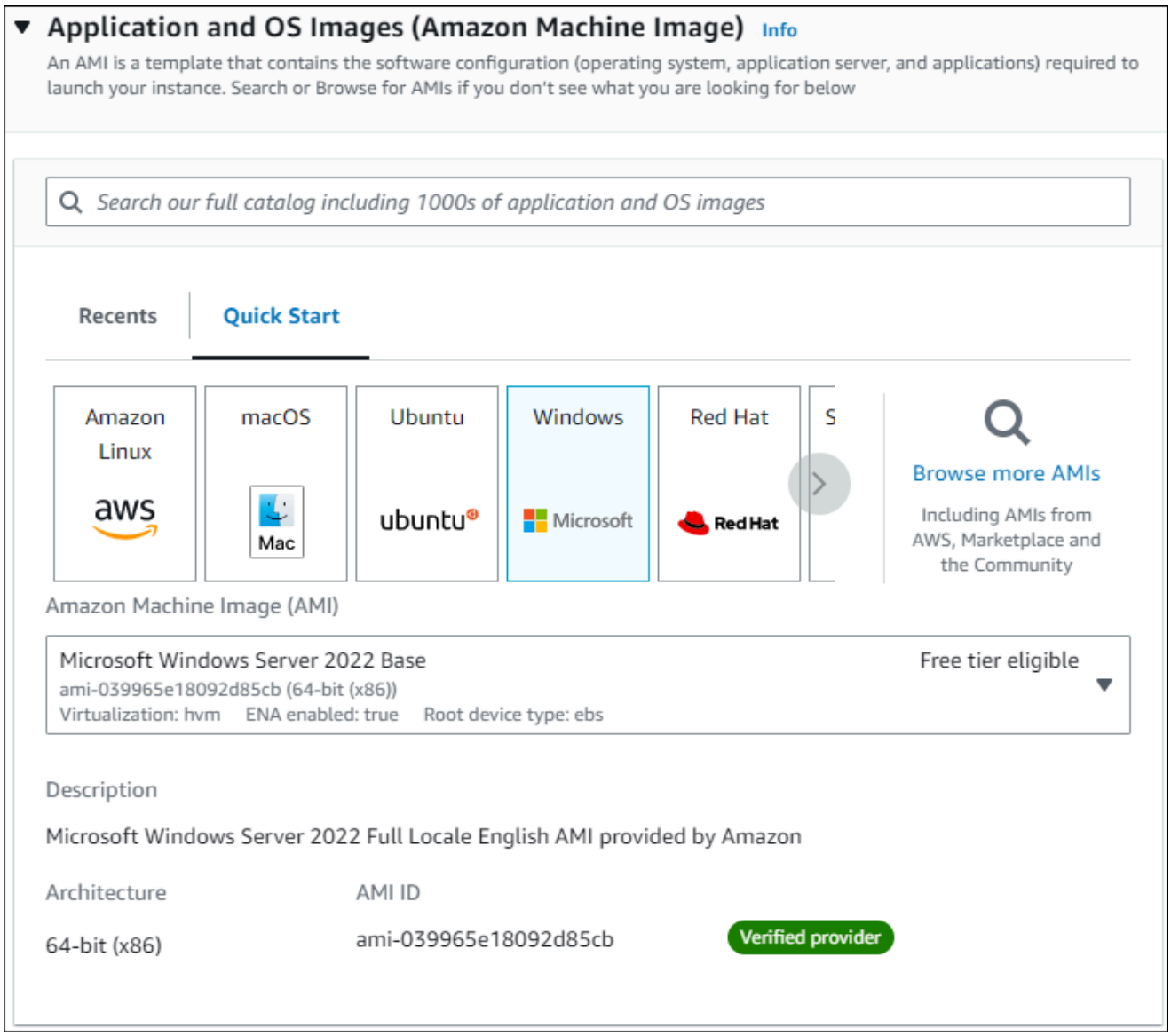


La page Lancer une instance s'ouvre.

4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`ec2-database-connect`**.

   b. Sous Images d'applications et de systèmes d'exploitation (Amazon Machine Image), choisissez Windows, puis Microsoft Windows Server 2022 Base. Conservez les sélections par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l' EC2 instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le Guide de EC2 l'utilisateur Amazon pour les instances Windows.

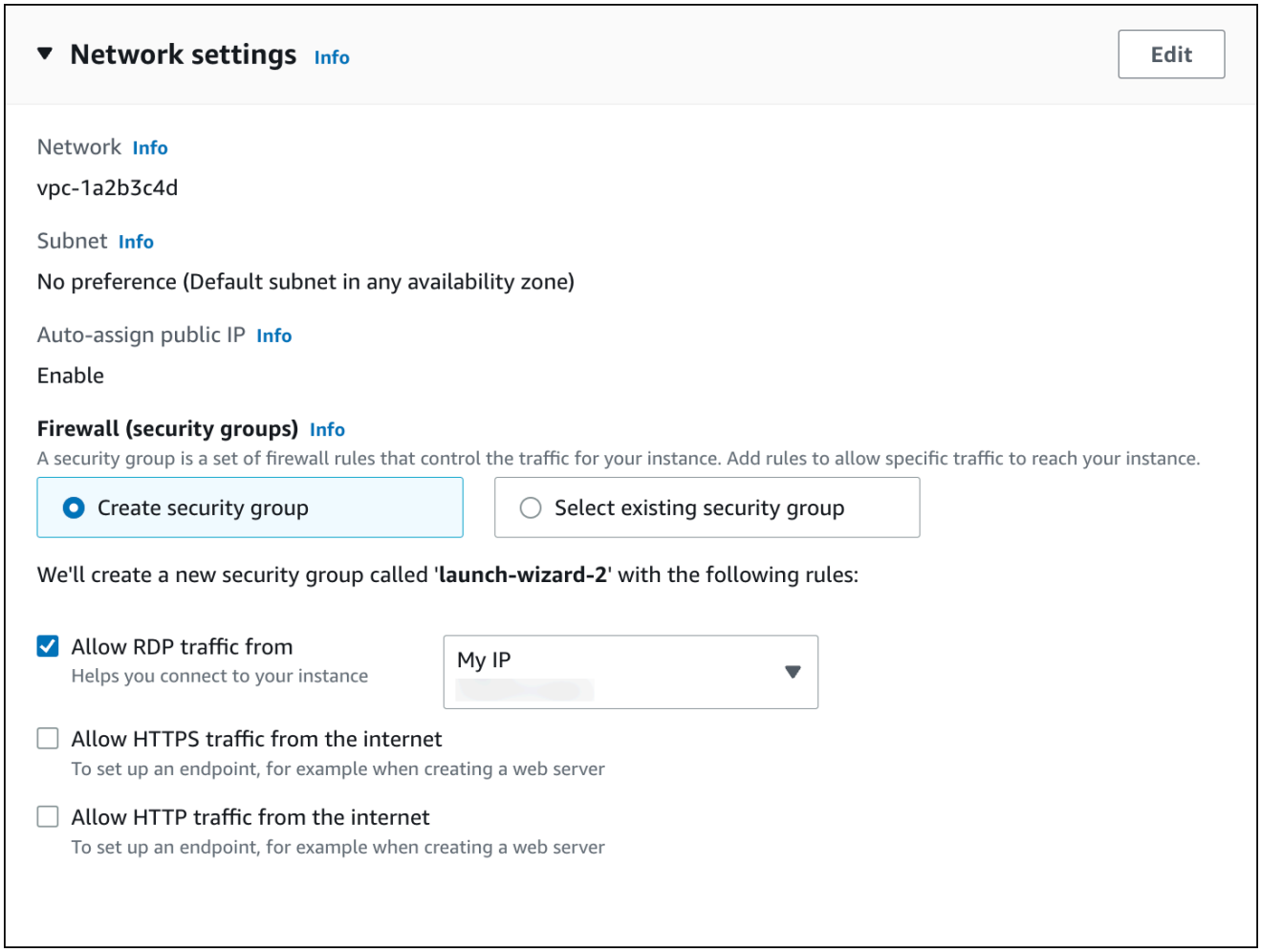
e. Pour Pare-feu (groupes de sécurité) dans les paramètres réseau, choisissez Autoriser le trafic RDP depuis pour vous connecter à l' EC2 instance.

Vous pouvez choisir Mon IP si l'adresse IP affichée est correcte pour les connexions RDP. Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide du protocole RDP. Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : 192.0.2.1/32.

Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

> **Warning**
>
> Si vous utilisez l'`0.0.0.0/0`accès RDP, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques à l'aide du protocole RDP. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances à l'aide du protocole RDP.

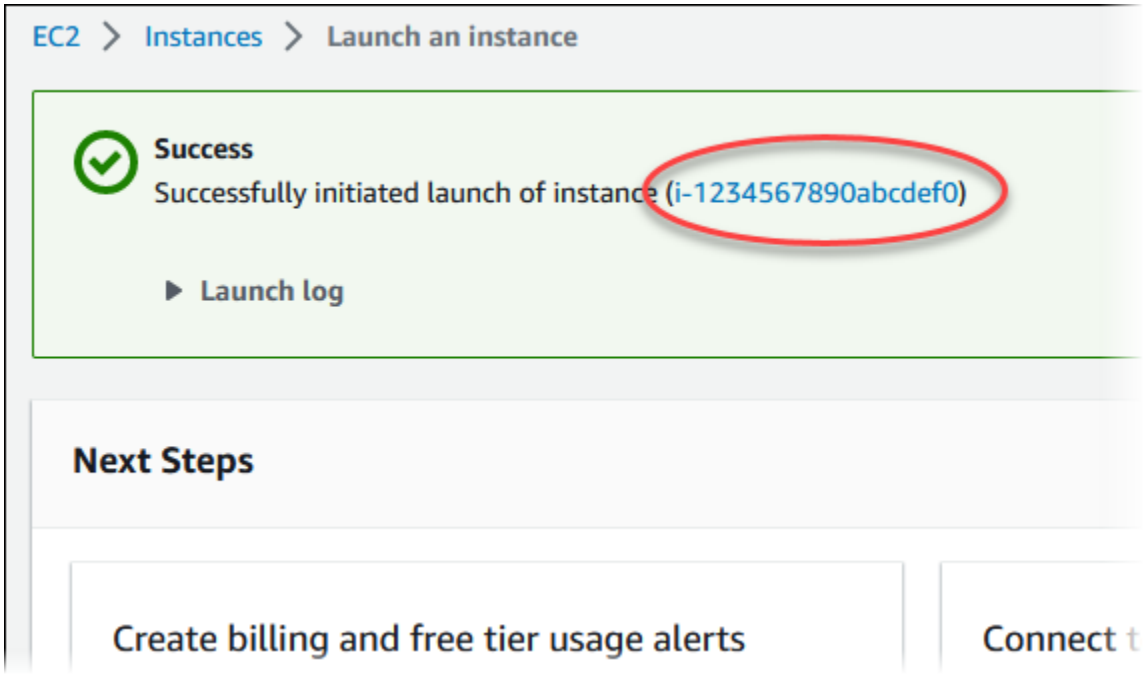
L'image suivante présente un exemple de la section Paramètres réseau.



f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre EC2 instance dans le panneau Résumé et, lorsque vous êtes prêt, choisissez Launch instance.

5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant de l' EC2 instance pour ouvrir la liste des EC2 instances.
7. Attendez que l'état Instance de votre EC2 instance soit défini sur En cours d'exécution avant de continuer.

Création d'une instance de base de données SQL Server

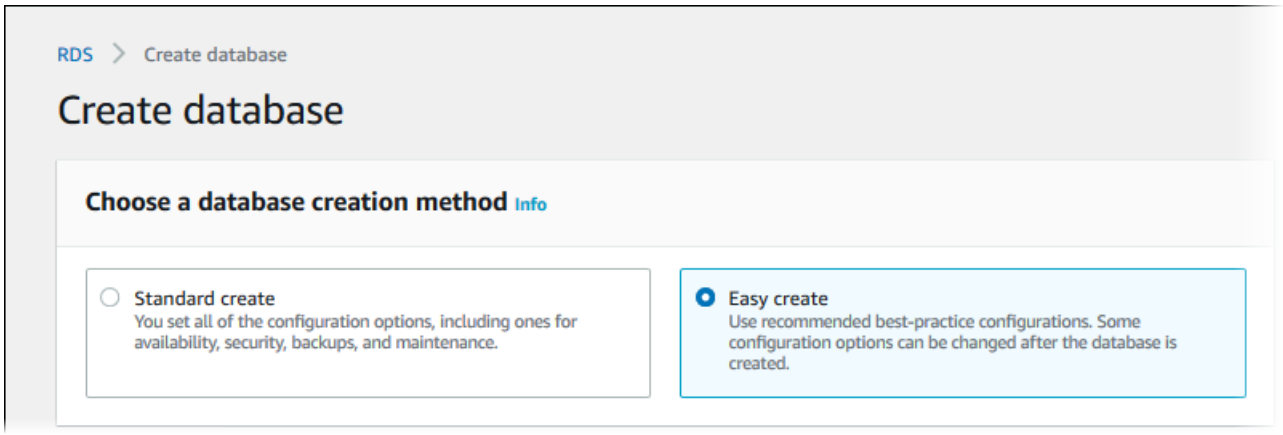
La fondation de base d'Amazon RDS est l'instance de base de données. Il s'agit de l'environnement dans lequel vous exécutez vos bases de données SQL Server.

Dans cet exemple, vous utilisez Création facile pour créer une instance de base de données exécutant le moteur de base de données SQL Server avec une classe d'instance de base de données db.t2.micro.

Pour créer une instance de base de données Microsoft SQL Server avec l'option Easy create (Création facile)

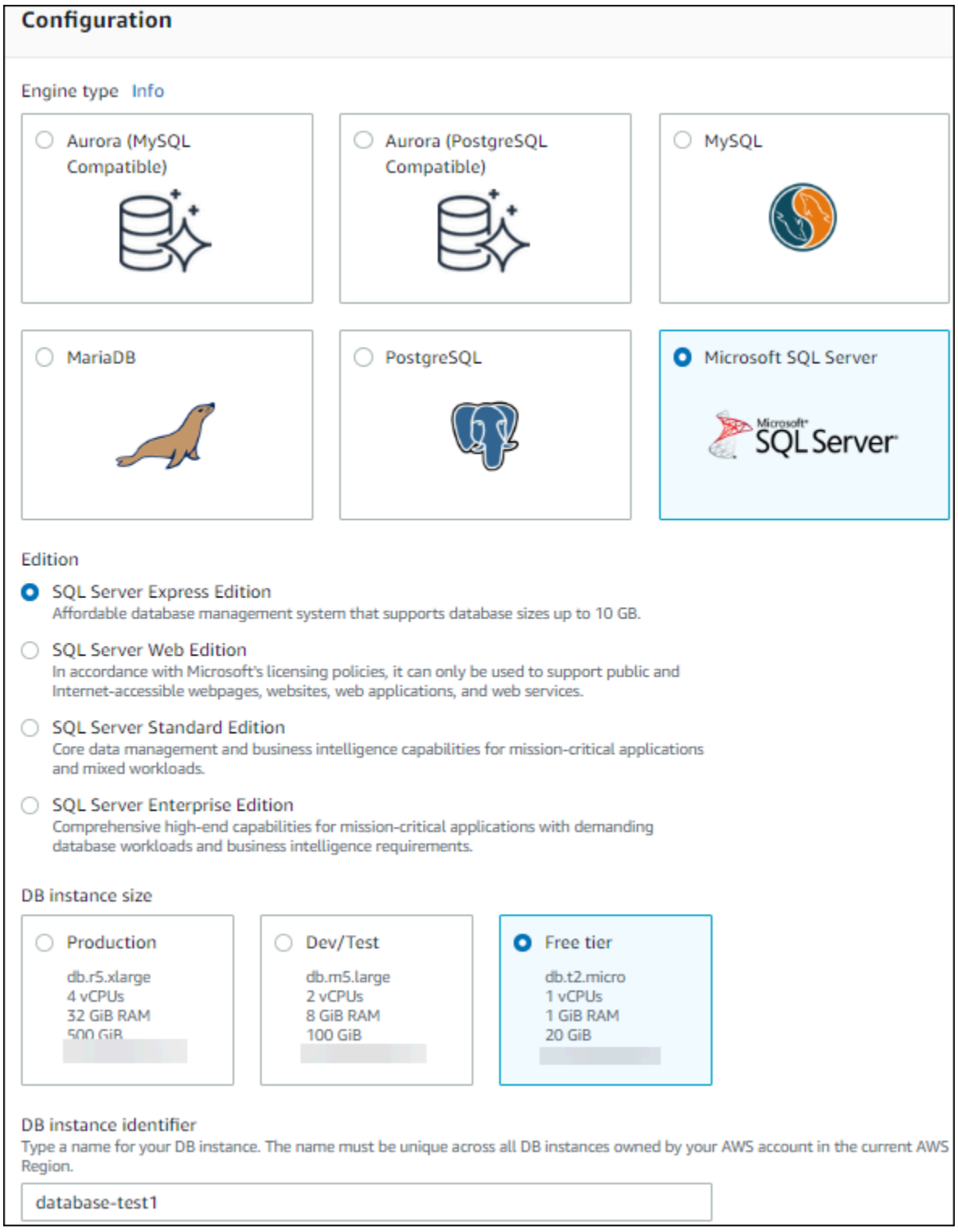
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l'instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).

4. Choisissez Create database (Créer une base de données) et veillez à choisir Easy create (Création facile).



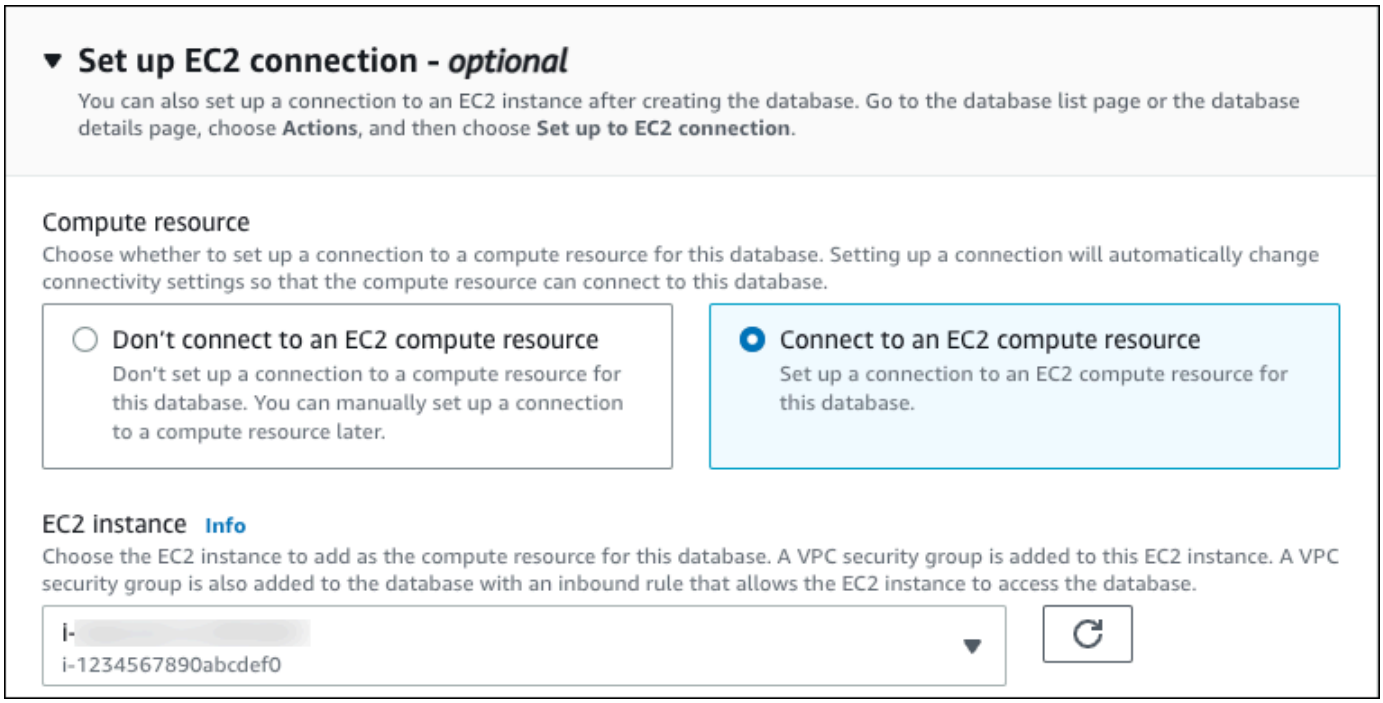
5. Dans Configuration, choisissez Microsoft SQL Server.
6. Pour Édition, choisissez SQL Server Express Edition.
7. Pour DB instance size (Taille de l'instance de base de données), choisissez Free tier (Offre gratuite). Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.
8. Pour l'identifiant de l'instance DB, saisissez **`database-test1`**.

   La page Create database (Créer une base de données) doit ressembler à l'image suivante. Pour les comptes de forfait gratuit, le niveau gratuit apparaît. Pour les comptes payants, Sandbox apparaît.



9. Pour Nom d'utilisateur principal, saisissez un nom pour l'utilisateur principal ou conservez le nom par défaut.
10. Pour configurer une connexion avec l' EC2 instance que vous avez créée précédemment, ouvrez Configurer la EC2 connexion - facultatif.

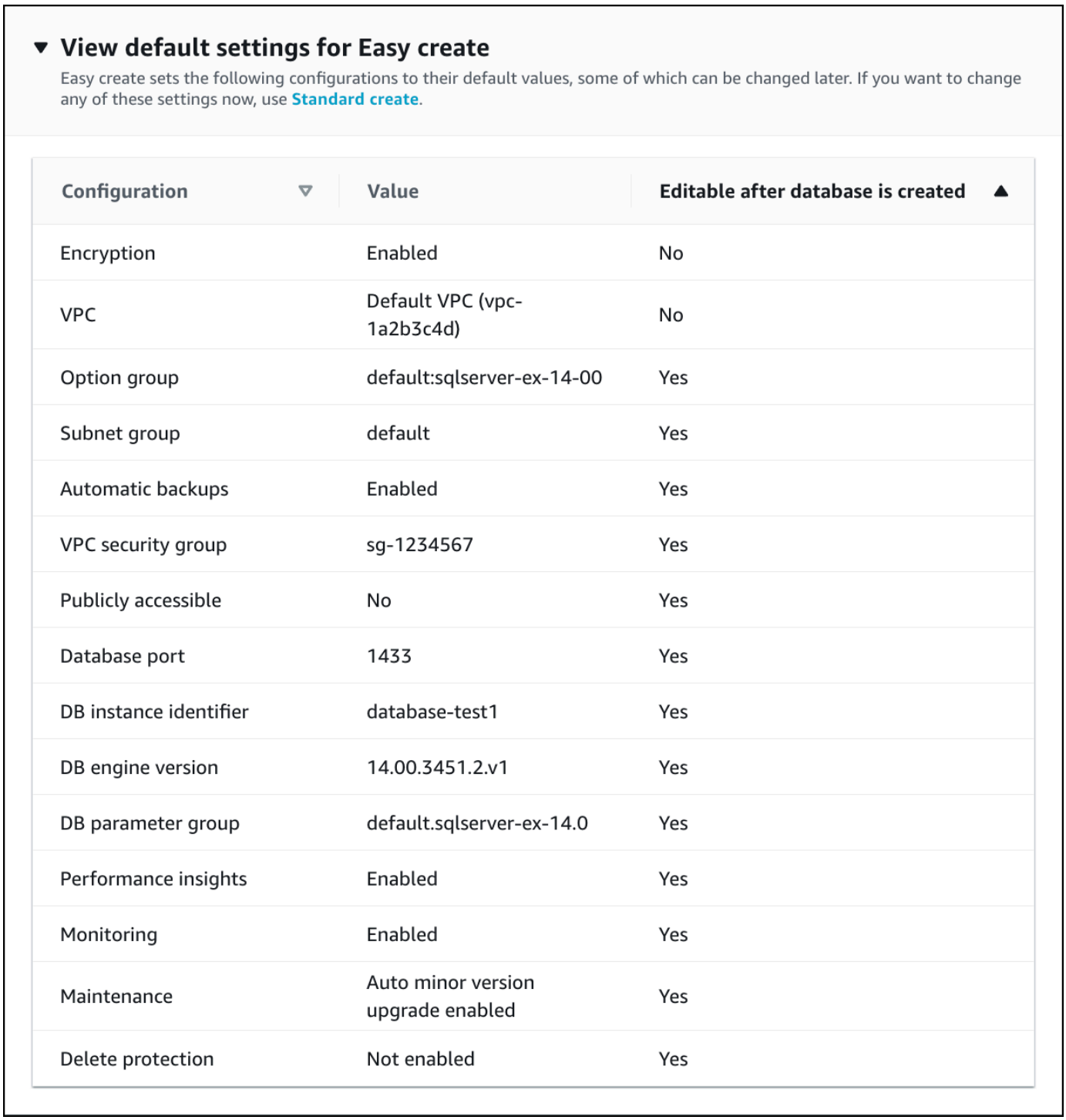
Sélectionnez Se connecter à une ressource EC2 de calcul. Choisissez l' EC2instance que vous avez créée précédemment.



11. Pour utiliser un mot de passe principal généré automatiquement pour l'instance de base de données, sélectionnez la case Auto generate a password (Générer un mot de passe automatiquement).

    Pour entrer votre mot de passe principal, décochez la case Auto generate a password (Générer un mot de passe automatiquement), puis entrez le même mot de passe dans les champs Master password (Mot de passe principal) et Confirm password (Confirmer le mot de passe).

12. Ouvrez Afficher les paramètres par défaut pour Création facile.



View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

Configuration	Value	Editable after database is created
Encryption	Enabled	No
VPC	Default VPC (vpc-1a2b3c4d)	No
Option group	default:sqlserver-ex-14-00	Yes
Subnet group	default	Yes
Automatic backups	Enabled	Yes
VPC security group	sg-1234567	Yes
Publicly accessible	No	Yes
Database port	1433	Yes
DB instance identifier	database-test1	Yes
DB engine version	14.00.3451.2.v1	Yes
DB parameter group	default.sqlserver-ex-14.0	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

Vous pouvez examiner les paramètres par défaut utilisés quand l'option Easy create (Création facile) est activée. La colonne Modifiable après la création de la base de données indique les options que vous pouvez modifier après avoir créé la base de données.

- Si un réglage contient Non dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données.

- Si un réglage contient Oui dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données ou vous pouvez modifier l'instance de base de données après l'avoir créée pour modifier le réglage.

13. Choisissez Créer une base de données.

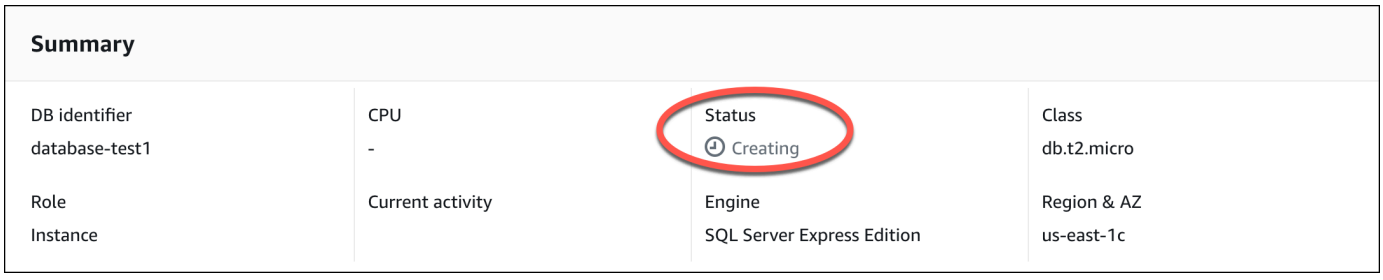
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Vous pouvez utiliser l'identifiant et le mot de passe affichés pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier.
    > Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, vous pouvez le faire en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

14. Dans la liste Bases de données, choisissez le nom de la nouvelle instance de base de données SQL Server pour afficher ses détails.

    L'instance de base de données a le statut Création en cours jusqu'à ce qu'elle soit prête à l'emploi.



    Lorsque l'état passe à Available (Disponible), vous pouvez vous connecter à l'instance de base de données. En fonction de la quantité de stockage et de la classe d'instance de base de données, la mise à disposition de la nouvelle instance peut prendre jusqu'à 20 minutes.

(Facultatif) Créez un VPC, une instance et une EC2 instance SQL Server à l'aide de AWS CloudFormation

Au lieu d'utiliser la console pour créer votre VPC, votre instance et votre EC2 instance SQL Server, vous pouvez utiliser la console AWS CloudFormation pour provisionner AWS des ressources en traitant l'infrastructure comme du code. Pour vous aider à organiser vos AWS ressources en unités plus petites et plus faciles à gérer, vous pouvez utiliser la fonctionnalité de pile AWS CloudFormation imbriquée. Pour plus d'informations, consultez les sections Création d'une pile sur la AWS CloudFormation console et Utilisation de piles imbriquées.

> **Important**
>
> AWS CloudFormation est gratuit, mais les ressources qui en CloudFormation découlent sont vivantes. Vous devez payer les frais d'utilisation standard pour ces ressources jusqu'à ce que vous y mettiez fin. Pour plus d'informations, consultez la section Tarification de RDS pour SQL Server.

Pour créer vos ressources à l'aide de la AWS CloudFormation console, procédez comme suit :

- Téléchargez le CloudFormation modèle
- Configurez vos ressources à l'aide de CloudFormation

Téléchargez le CloudFormation modèle

Un CloudFormation modèle est un fichier texte JSON ou YAML qui contient les informations de configuration relatives aux ressources que vous souhaitez créer dans la pile. Ce modèle crée également un VPC et un hôte bastion pour vous, ainsi que l'instance RDS.

Pour télécharger le fichier modèle, ouvrez le lien suivant, CloudFormation Modèle SQL Server.

Sur la page Github, cliquez sur le bouton Télécharger le fichier brut pour enregistrer le modèle de fichier YAML.

Configurez vos ressources à l'aide de CloudFormation

> **Note**
>
> Avant de commencer ce processus, assurez-vous que vous disposez d'une paire de clés pour une EC2 instance dans votre Compte AWS. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Lorsque vous utilisez le AWS CloudFormation modèle, vous devez sélectionner les paramètres appropriés pour vous assurer que vos ressources sont créées correctement. Procédez de la façon suivante :

1. Connectez-vous à la AWS CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile.
3. Dans la section Spécifier le modèle, sélectionnez Télécharger un fichier modèle depuis votre ordinateur, puis cliquez sur Suivant.
4. Dans la page Spécifier les détails de la pile, définissez les paramètres suivants :
   a. Définissez le nom de la pile sur SQLServerTestStack.
   b. Sous Paramètres, définissez les zones de disponibilité en sélectionnant trois zones de disponibilité.
   c. Dans Configuration de l'hôte Linux Bastion, pour Nom de la clé, sélectionnez une paire de clés pour vous connecter à votre EC2 instance.
   d. Dans les paramètres de configuration de l'hôte Linux Bastion, définissez la plage d'adresses IP autorisées sur votre adresse IP. Pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH), déterminez votre adresse IP publique à l'aide du service à l'adresse. https://checkip.amazonaws.com Exemple d'adresse IP : 192.0.2.1/32.

      > **Warning**
      >
      > Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les

environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

e. Dans Configuration générale de la base de données, définissez la classe d'instance de base de données sur db.t3.micro.

f. Définissez le nom de base de données sur**`database-test1`**.

g. Dans Nom d'utilisateur principal de base de données, entrez le nom de l'utilisateur principal.

h. Définissez le mot de passe utilisateur principal de Manage DB avec Secrets Manager sur `false` pour ce didacticiel.

i. Pour le mot de passe de la base de données, définissez le mot de passe de votre choix. N'oubliez pas ce mot de passe pour les étapes suivantes du didacticiel.

j. Sous Configuration du stockage de base de données, définissez le type de stockage de base de données sur gp2.

k. Dans Configuration de la surveillance des bases de données, définissez Enable RDS Performance Insights sur false.

l. Conservez tous les autres paramètres comme valeurs par défaut. Cliquez sur Suivant pour continuer.

5. Sur la page Configurer les options de pile, conservez toutes les options par défaut. Cliquez sur Suivant pour continuer.
6. Sur la page Review stack, sélectionnez Soumettre après avoir vérifié les options de la base de données et de l'hôte Linux Bastion.

Une fois le processus de création des piles terminé, visualisez les piles avec leurs noms BastionStacket leurs RDSNS pour noter les informations dont vous avez besoin pour vous connecter à la base de données. Pour plus d'informations, consultez la section Affichage des données et des ressources de la AWS CloudFormation pile sur le AWS Management Console.

Connectez-vous à votre instance de base de données SQL Server

Dans la procédure suivante, vous vous connectez à votre instance de base de données à l'aide de Microsoft SQL Server Management Studio (SSMS).

Pour se connecter à votre instance de base de données RDS for SQL Server à l'aide de SSMS

1. Trouvez le point de terminaison (nom DNS) et le numéro de port pour votre instance de base de données.

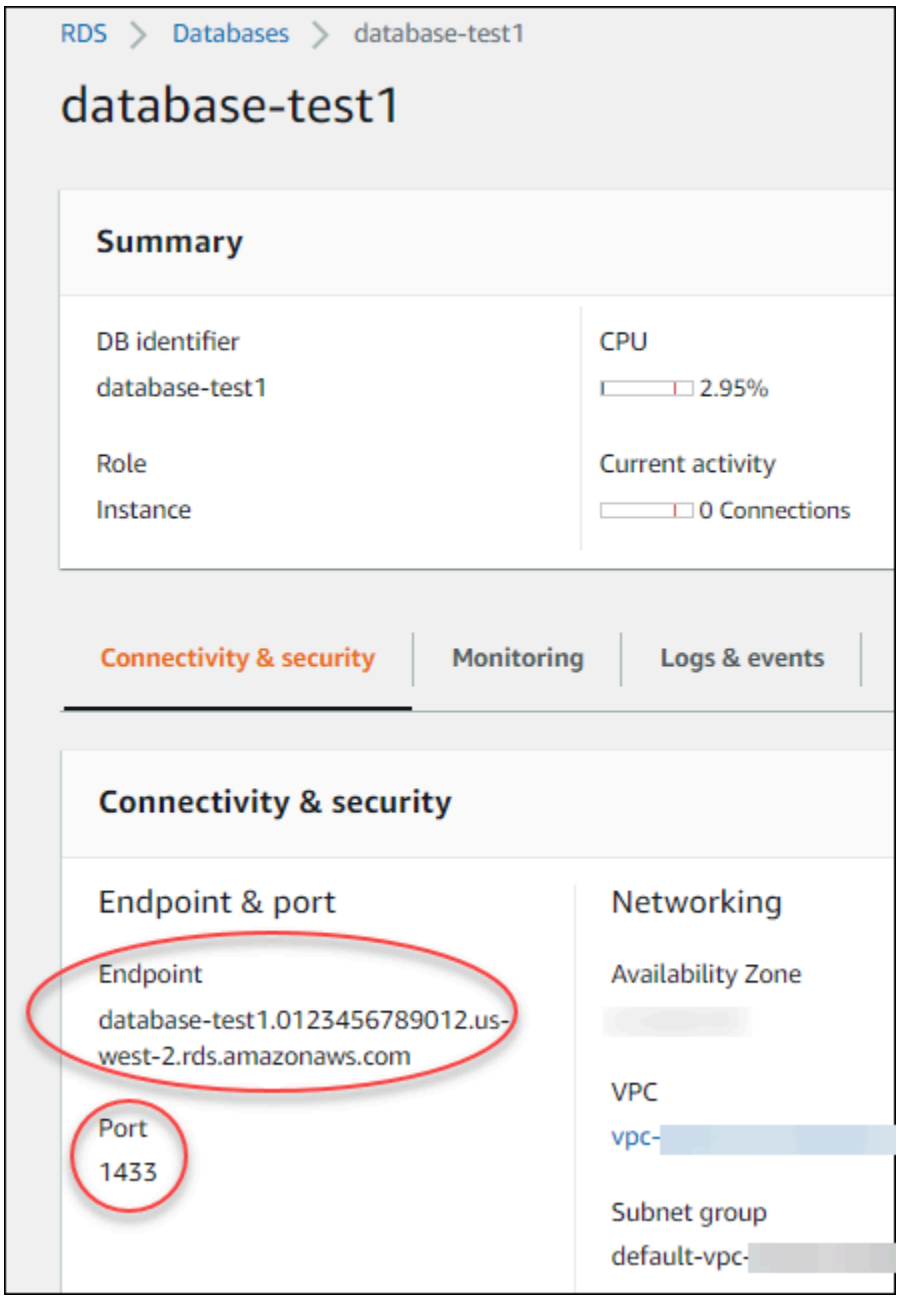
a. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

b. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance de base Région AWS de données.

c. Dans le panneau de navigation, choisissez Databases (Bases de données).

d. Choisissez le nom de l'instance de base de données SQL Server pour afficher ses détails.

e. Dans l'onglet Connectivité, copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



2. Connectez-vous à l' EC2 instance que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Microsoft Windows dans le Guide de EC2 l'utilisateur Amazon pour les instances Windows.
3. Installez le client SQL Server Management Studio (SSMS) de Microsoft.

   Pour télécharger une version autonome de SSMS sur votre EC2 instance, consultez la section Télécharger SQL Server Management Studio (SSMS) dans la documentation Microsoft.

   a. Utilisez le menu Démarrer pour ouvrir Internet Explorer.

b. Utilisez Internet Explorer pour télécharger et installer une version autonome de SSMS. Si vous êtes invité à indiquer que le site n'est pas fiable, ajoutez-le à la liste des sites de confiance.

4. Démarrez SQL Server Management Studio (SSMS).

   La boîte de dialogue Connect to Server (Se connecter à un serveur) s'affiche.

5. Fournissez les informations suivantes relatives à votre exemple d'instance de base de données :

   a. Pour Server type (Type de serveur), choisissez Database Engine (Moteur de base de données).

   b. Pour Server name (Nom du serveur), entrez le nom DNS, suivi d'une virgule et du numéro de port (le port par défaut est 1433). Par exemple, le nom du serveur doit se présenter comme suit :

   ```
   database-test1.0123456789012.us-west-2.rds.amazonaws.com,1433
   ```

   c. Pour Authentication, choisissez Authentication SQL Server.

   d. Pour Connexion, saisissez le nom d'utilisateur que vous avez choisi d'utiliser pour votre exemple d'instance de base de données. Ce nom est également appelé « identifiant principal ».

   e. Pour Password (Mot de passe), saisissez le mot de passe que vous avez choisi auparavant pour votre exemple d'instance de base de données. Ce nom est également appelé « mot de passe d'utilisateur principal ».

6. Choisissez Connexion.

   Après quelques instants, SSMS se connecte à votre instance de base de données. Pour des raisons de sécurité, une bonne pratique consiste à recommander d'utiliser des connexions chiffrées. N'utilisez une connexion SQL Server non chiffrée que quand le client et le serveur sont dans le même VPC et que le réseau est approuvé. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Utilisation SSL avec une instance de base de données Microsoft SQL Server.

Pour plus d'informations sur la connexion à une instance de base de données Microsoft SQL Server, consultez Connexion à votre instance de base de données Microsoft SQL Server.

Pour des informations sur les problèmes de connexion, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Explorez votre exemple d'instance de base de données SQL Server

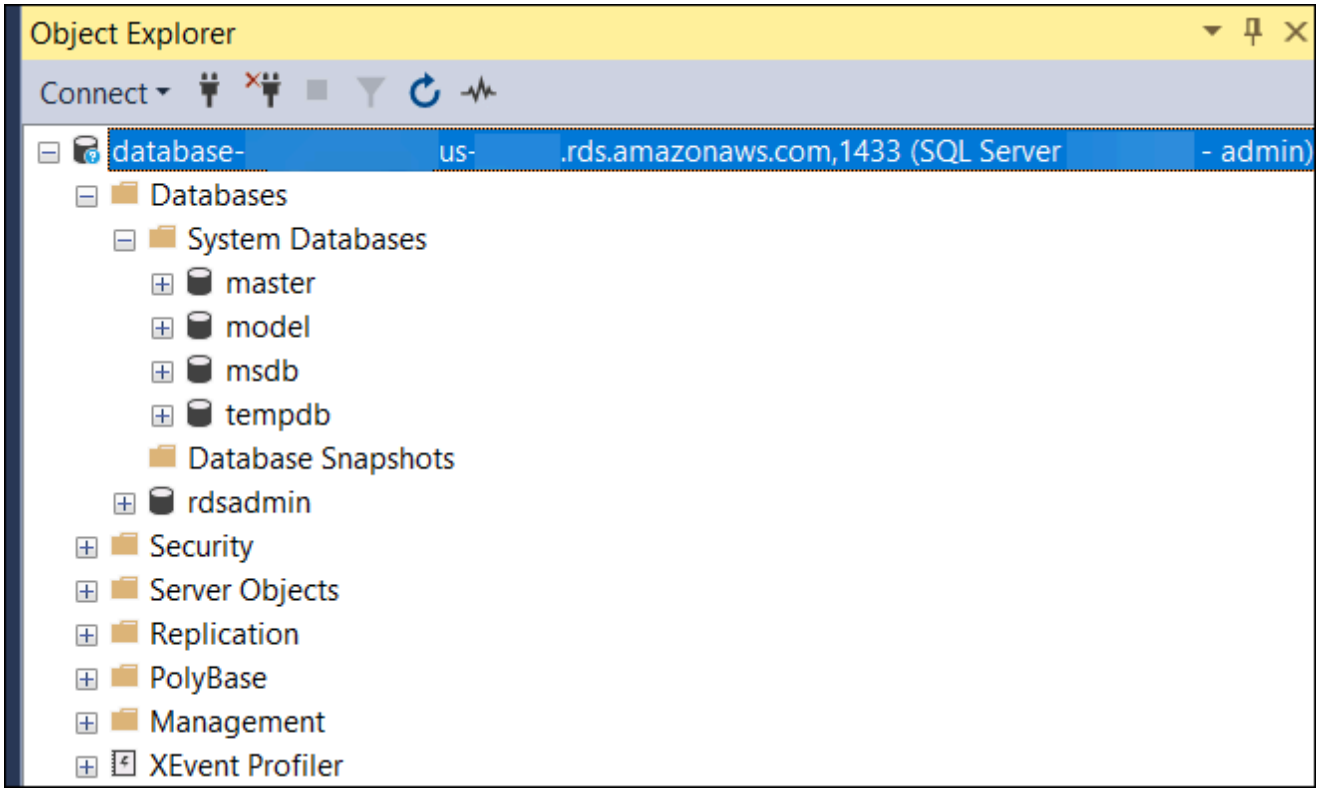
Vous pouvez explorer votre exemple d'instance base de données à l'aide de Microsoft SQL Server Management Studio (SSMS).

Pour explorer une instance de base de données à l'aide de SSMS

1. Votre instance de base de données SQL Server est fournie avec les bases de données système intégrées standard de SQL Server (master, model, msdb et tempdb). Pour explorer les bases de données système, effectuez les opérations suivantes :

   a. Dans SSMS, dans le menu View (Afficher), choisissez Object Explorer (Navigateur d'objet).

   b. Développez votre instance de base de données, développez Databases (Bases de données), puis System Databases (Bases de données système) comme indiqué.



   Votre instance de base de données SQL Server est également accompagnée d'une base de données nommée `rdsadmin`. Amazon RDS utilise cette base de données pour stocker les objets dont il se sert pour gérer votre base de données. La base de données `rdsadmin` inclut également des procédures stockées que vous pouvez exécuter pour effectuer des tâches avancées.

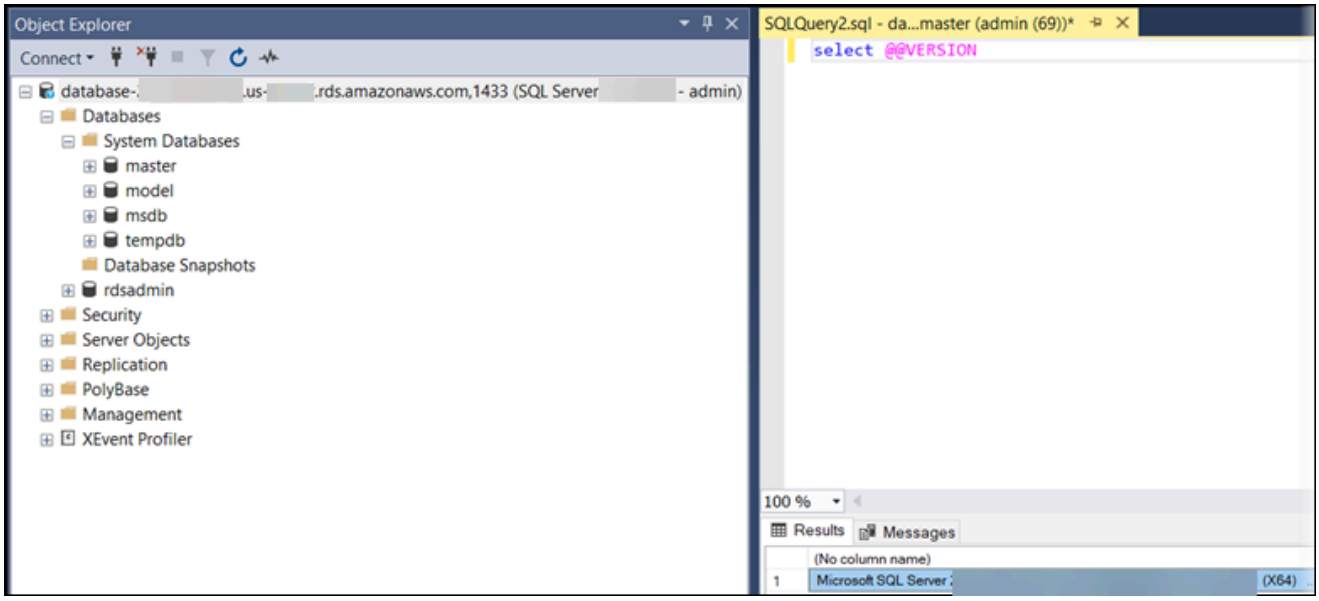
2. Commencez à créer vos propres bases de données et à exécuter des requêtes sur votre instance de base de données et vos bases de données comme d'habitude. Pour exécuter une requête de test sur votre exemple d'instance de base de données, procédez comme suit :

   a. Dans SSMS, dans le menu Fichier, pointez sur Nouveau, puis choisissez Requête avec la connexion actuelle.

   b. Saisissez la requête SQL suivante :

   ```
   select @@VERSION
   ```

   c. Exécutez la requête. SSMS renvoie la version SQL Server de votre instance de base de données Amazon RDS.



Supprimer l' EC2instance et l'instance de base de données

Après vous être connecté et exploré l'exemple d' EC2 instance et l'instance de base de données que vous avez créées, supprimez-les afin qu'ils ne vous soient plus facturés.

Si vous aviez AWS CloudFormation l'habitude de créer des ressources, ignorez cette étape et passez à l'étape suivante.

Pour supprimer l' EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.

2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l' EC2 instance, puis choisissez État de l'instance, Terminer l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une EC2 instance, voir Résilier votre instance dans le Guide de l'utilisateur pour les instances Windows.

Pour supprimer l'instance de base de données sans instantané de base de données final

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Décochez Créer un instantané final et Conserver les sauvegardes automatiques.
6. Terminez la confirmation et choisissez Supprimer.

(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation

Si vous aviez l'habitude AWS CloudFormation de créer des ressources, supprimez la CloudFormation pile une fois que vous vous êtes connecté à l'instance d'exemple et à l' EC2 instance de base de données et que vous les avez explorées, afin de ne plus être facturées.

Pour supprimer les CloudFormation ressources

1. Ouvrez la AWS CloudFormation console.
2. Sur la page Stacks du CloudFormationconsole, sélectionnez la pile racine (la pile sans nom VPCStack, BastionStack ou RDSNS).
3. Sélectionnez Delete (Supprimer).
4. Sélectionnez Supprimer la pile lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une pile dans CloudFormation, voir Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Connecter votre instance de base de données à une fonction Lambda

Vous pouvez également connecter votre instance de base de données RDS for SQL Server à une ressource de calcul sans serveur Lambda. Les fonctions Lambda vous permettent d'exécuter du code sans provisionner ni gérer l'infrastructure. Une fonction Lambda vous permet également de répondre automatiquement aux demandes d'exécution de code à n'importe quelle échelle, d'une douzaine d'événements par jour à des centaines par seconde. Pour de plus amples informations, veuillez consulter Connexion automatique d'une fonction Lambda et d'une instance de base de données.

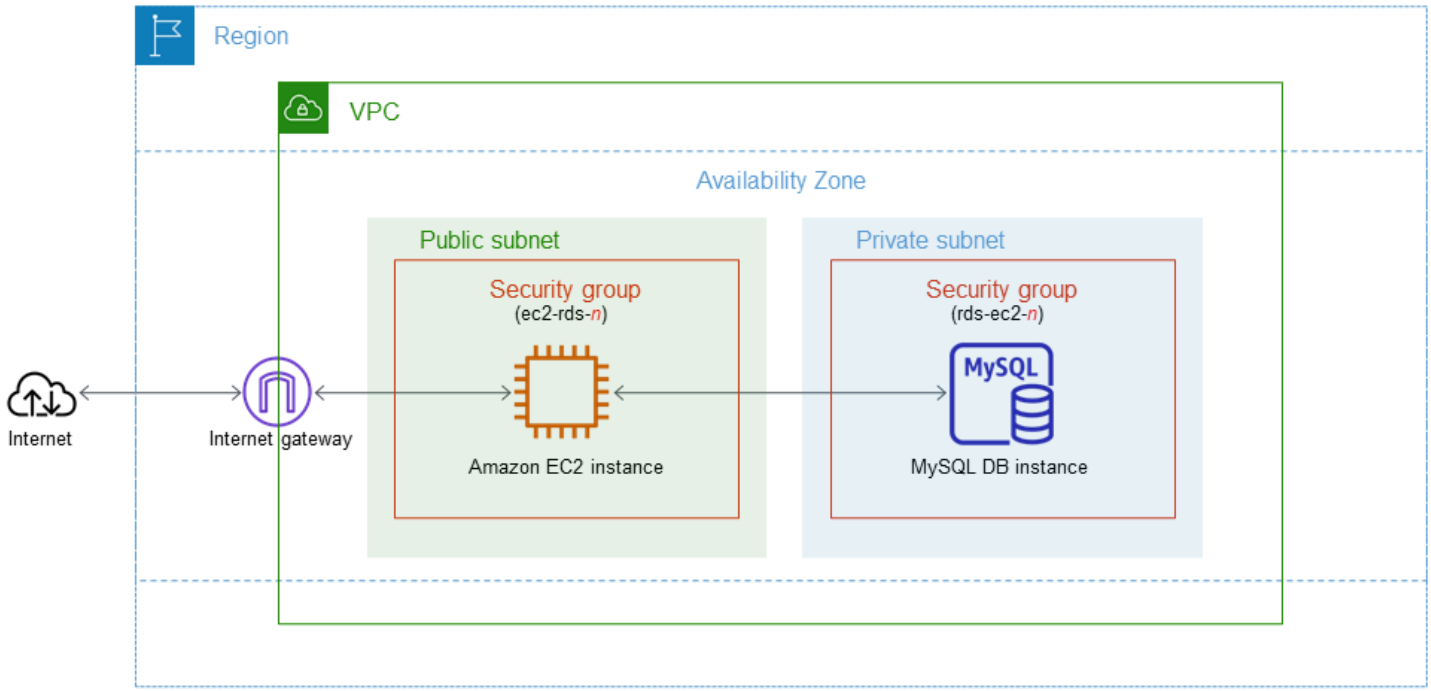
Création d'une instance de base de données MySQL et connexion à cette instance

Ce didacticiel crée une EC2 instance et une instance de base de données RDS pour MySQL. Le didacticiel explique comment accéder à l'instance de base de données à partir de l' EC2 instance à l'aide d'un client MySQL standard. En tant que bonne pratique, ce didacticiel crée une instance de base de données privée dans un cloud privé virtuel (VPC). Dans la plupart des cas, les autres ressources du même VPC, telles que les EC2 instances, peuvent accéder à l'instance de base de données, mais les ressources extérieures au VPC ne peuvent pas y accéder.

Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Dans une zone de disponibilité, l' EC2 instance se trouve dans le sous-réseau public et l'instance de base de données dans le sous-réseau privé.

> **Important**
>
> La création d'un AWS compte est gratuite. Cependant, en suivant ce didacticiel, les AWS ressources que vous utilisez peuvent vous coûter cher. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Ce didacticiel vous permet de créer vos ressources en utilisant l'une des méthodes suivantes :

1. Utilisez le AWS Management Console - Création d'une instance de base de données MySQL et Création d'une instance EC2
2. AWS CloudFormation À utiliser pour créer l'instance de base de données et l' EC2 instance - (Facultatif) Créez un VPC, une instance et une EC2 instance MySQL en utilisant AWS CloudFormation

La première méthode utilise Easy create pour créer une instance de base de données MySQL privée avec le AWS Management Console. Ici, vous spécifiez uniquement le type de moteur de base de données, la taille de l'instance de base de données et l'identifiant de l'instance de base de données. L'option Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration.

Lorsque vous utilisez plutôt Standard Create, vous pouvez spécifier d'autres options de configuration lorsque vous créez une instance de base de données. Ces options incluent les paramètres de disponibilité, de sécurité, de sauvegarde et de maintenance. Pour créer une instance de base de données publique, vous devez utiliser Création standard. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Rubriques

- Prérequis
- Création d'une instance EC2
- Création d'une instance de base de données MySQL
- (Facultatif) Créez un VPC, une instance et une EC2 instance MySQL en utilisant AWS CloudFormation
- Se connecter à une instance de base de données MySQL
- Supprimer l' EC2 instance et l'instance de base de données
- (Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation
- (Facultatif) Connecter votre instance de base de données à une fonction Lambda

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

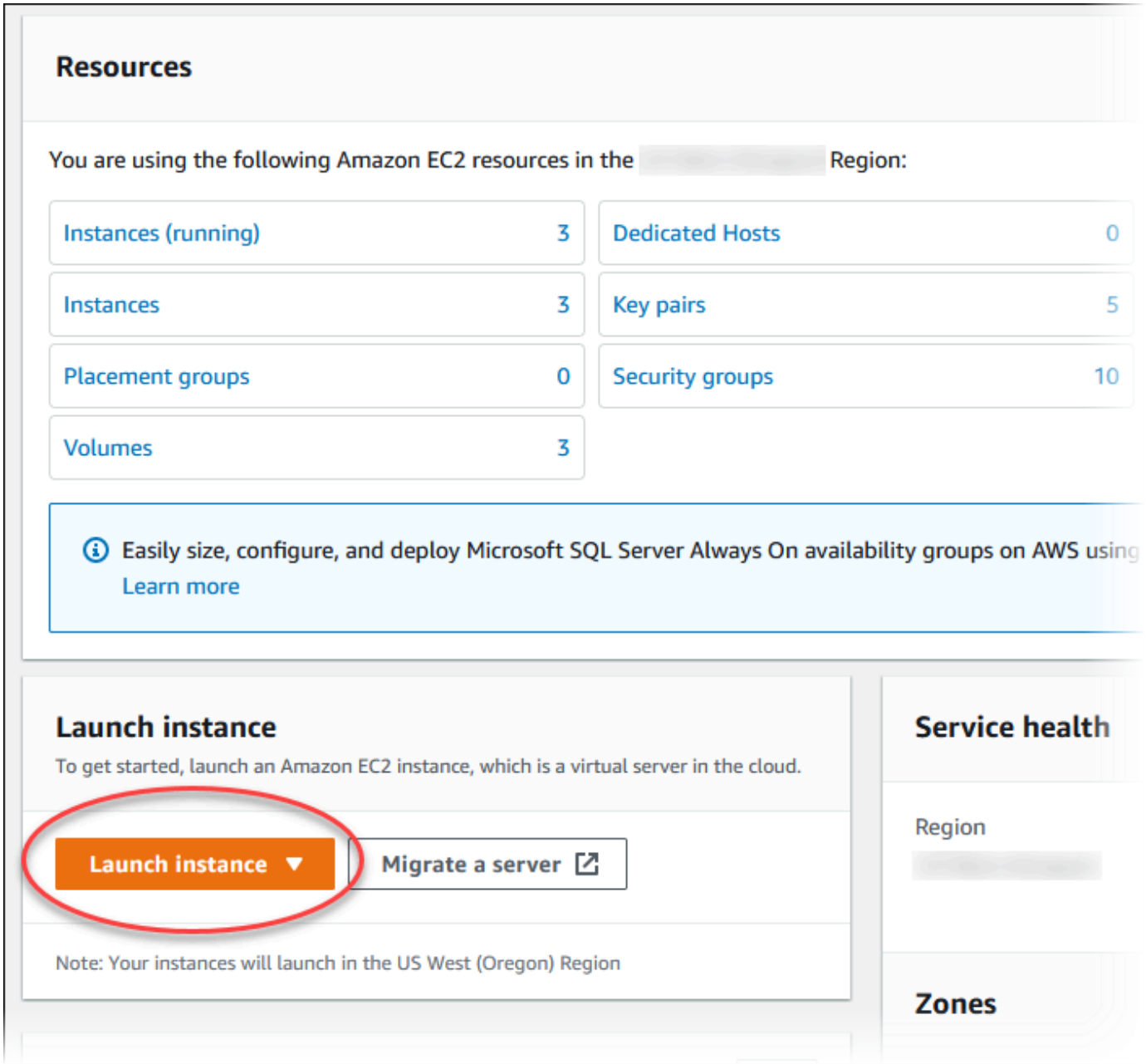
- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Création d'une instance EC2

Créez une EC2 instance Amazon que vous utiliserez pour vous connecter à votre base de données.

Pour créer une EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit de AWS Management Console, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l' EC2 instance.
3. Choisissez EC2 Dashboard, puis Launch instance, comme illustré dans l'image suivante.

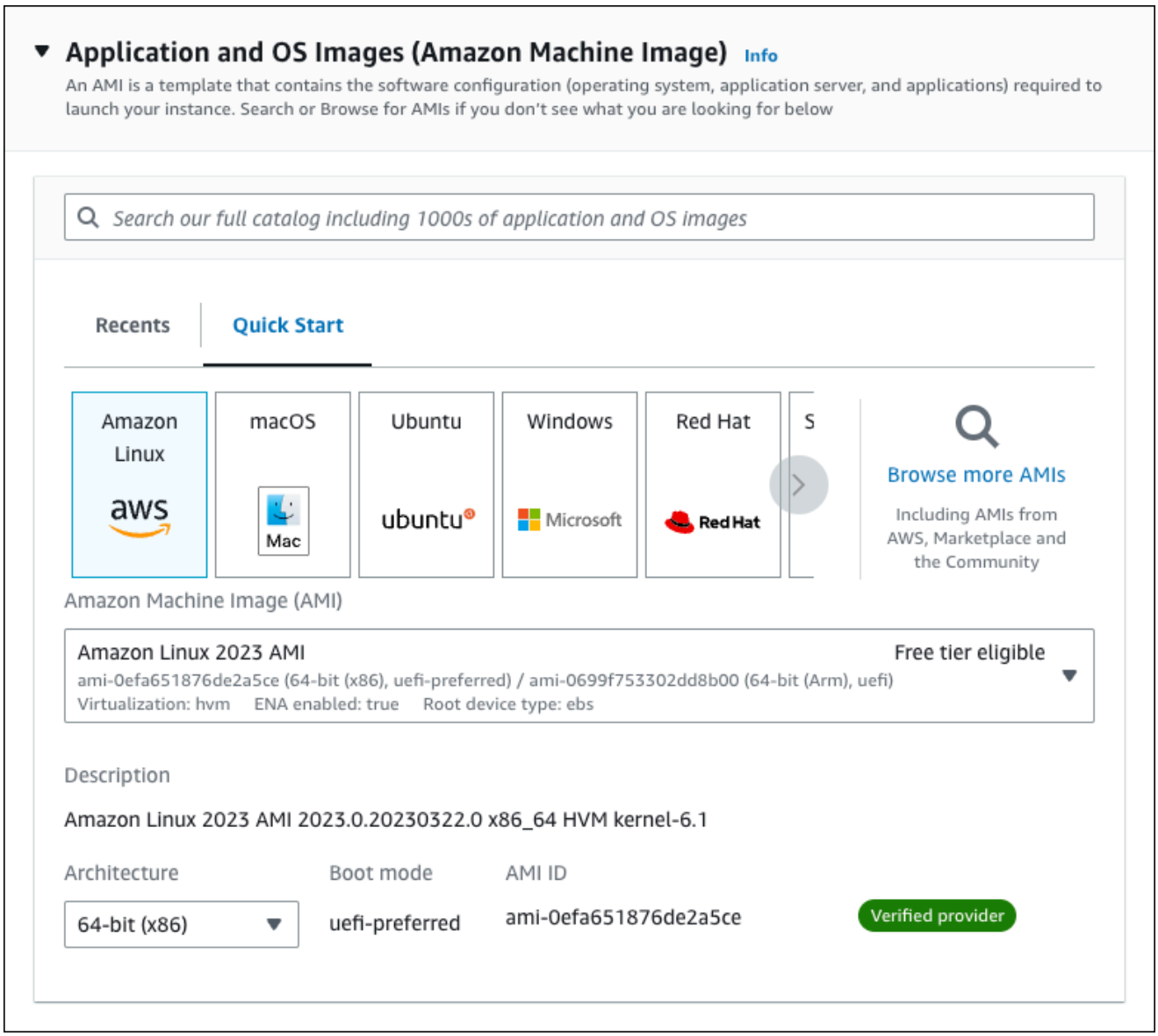


La page Lancer une instance s'ouvre.

4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`ec2-database-connect`**.

   b. Sous Application et images OS (Amazon Machine Image), choisissez Amazon Linux, puis Amazon Linux 2023 AMI. Conservez les sélections par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l' EC2 instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le guide de EC2 l'utilisateur Amazon.

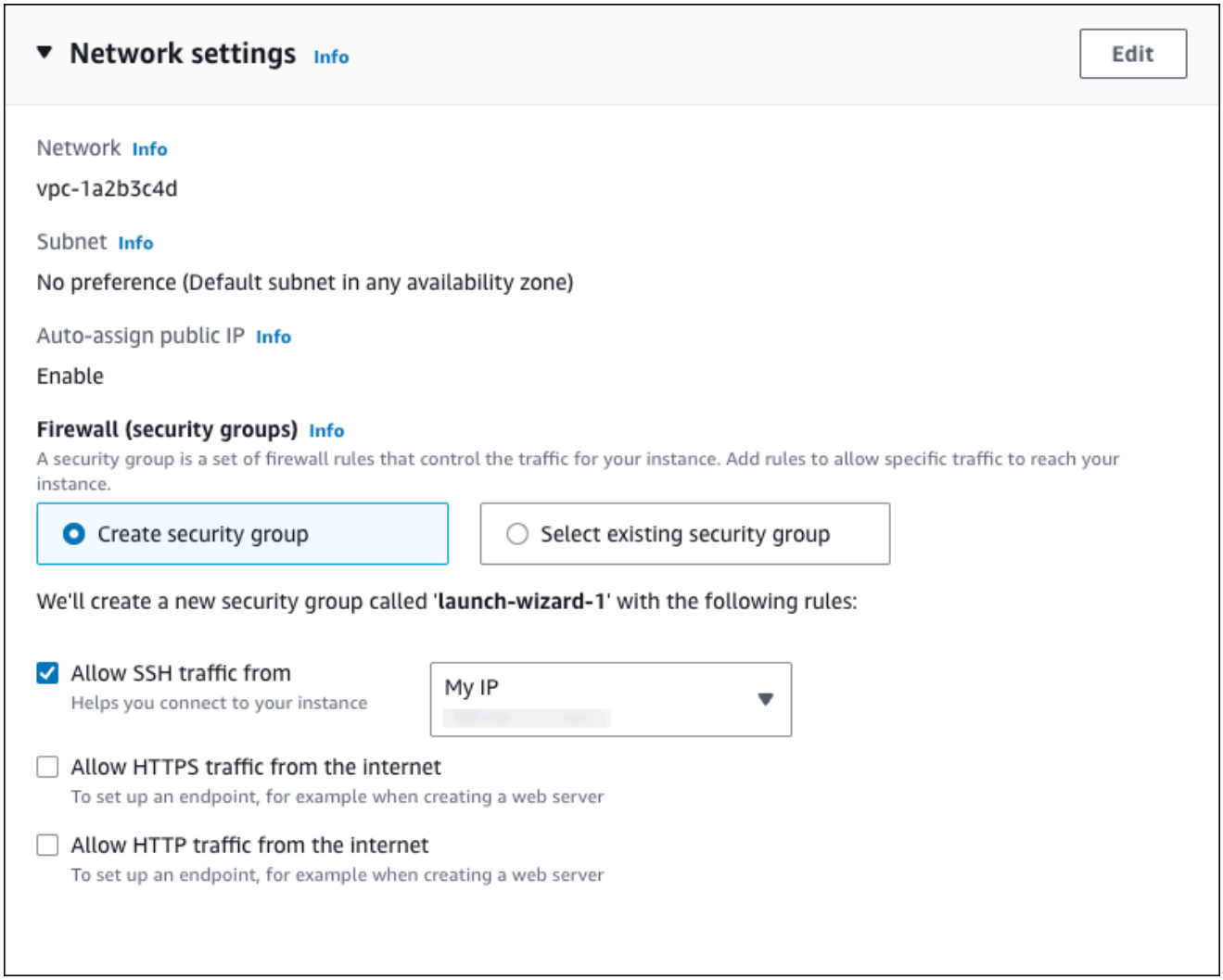
e. Pour Autoriser le trafic SSH dans les paramètres réseau, choisissez la source des connexions SSH à l' EC2instance.

Vous pouvez choisir My IP (Mon IP) si l'adresse IP affichée est correcte pour les connexions SSH. Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : 192.0.2.1/32.

Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

> **Warning**
>
> Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

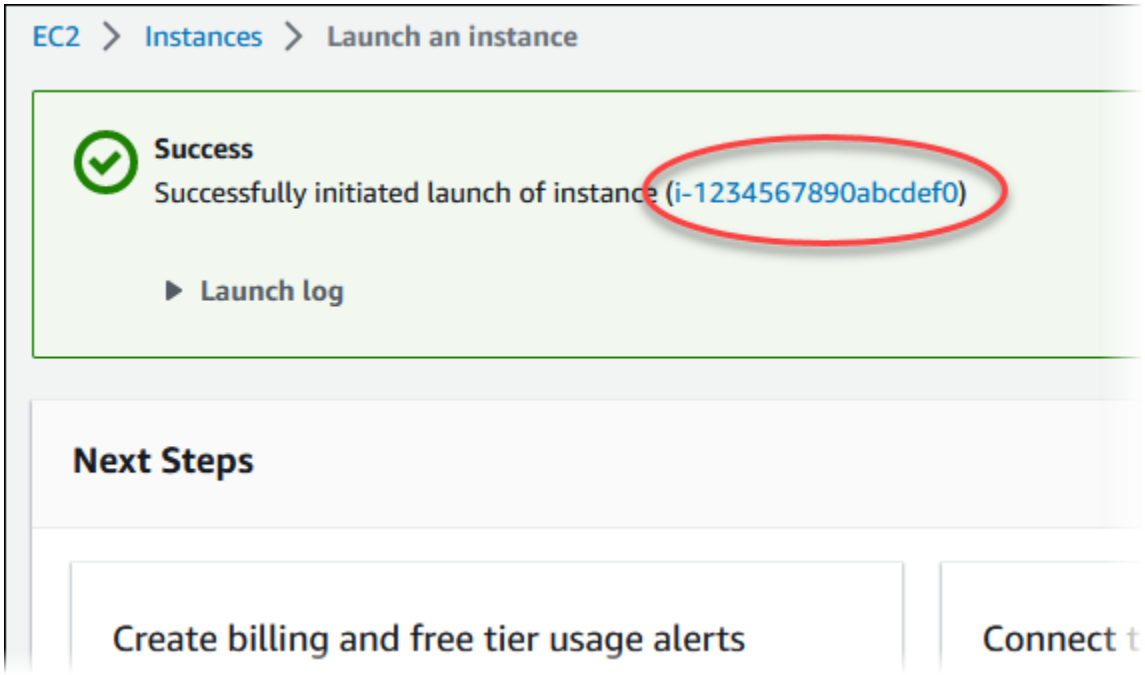
L'image suivante présente un exemple de la section Paramètres réseau.



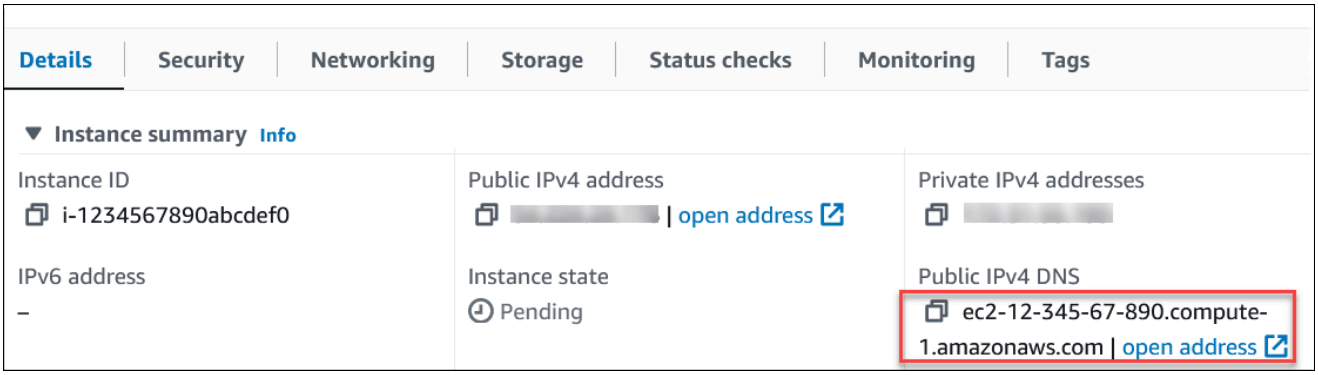
f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre EC2 instance dans le panneau Résumé et, lorsque vous êtes prêt, choisissez Launch instance.

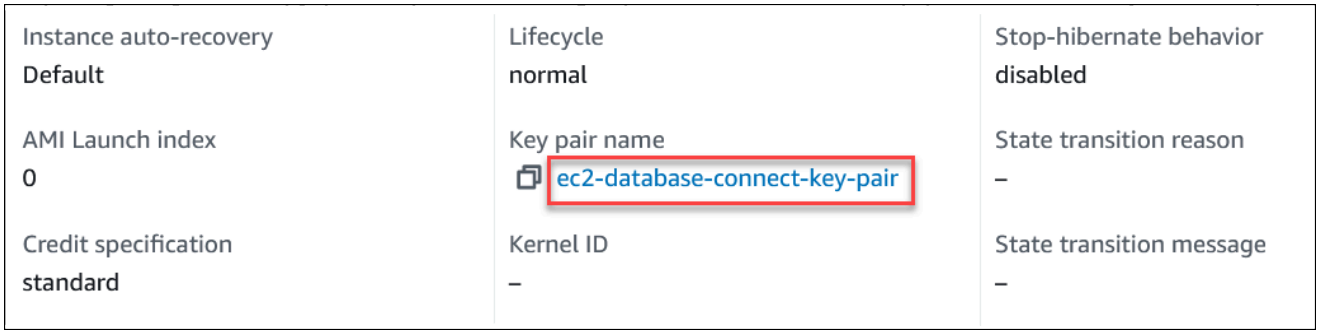
5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant d' EC2 instance pour ouvrir la liste des EC2 instances, puis sélectionnez votre EC2 instance.
7. Dans l'onglet Détails, notez les valeurs suivantes. Vous en aurez besoin lorsque vous vous connecterez via SSH :
   a. Dans Résumé de l'instance, notez la valeur du IPv4 DNS public.



   b. Dans Détails de l'instance, notez la valeur pour Nom de la paire de clés.



8. Attendez que l'état Instance de votre EC2 instance soit défini sur En cours d'exécution avant de continuer.

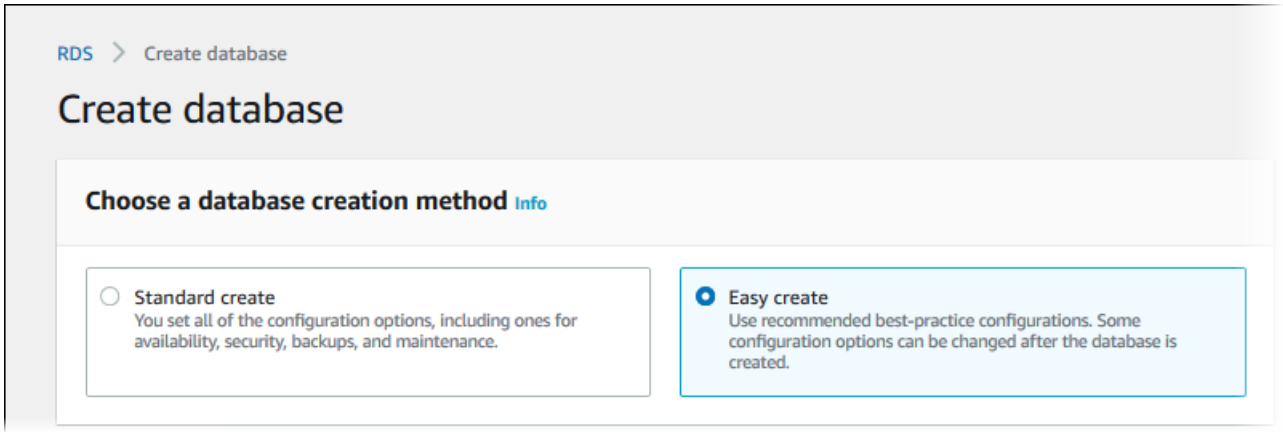
Création d'une instance de base de données MySQL

La fondation de base d'Amazon RDS est l'instance de base de données. Il s'agit de l'environnement dans lequel vous exécutez vos bases de données MySQL.

Dans cet exemple, vous utilisez l'option Création facile pour créer une instance de base de données exécutant le moteur de base de données MySQL avec une classe d'instance de base de données db.t3.micro.

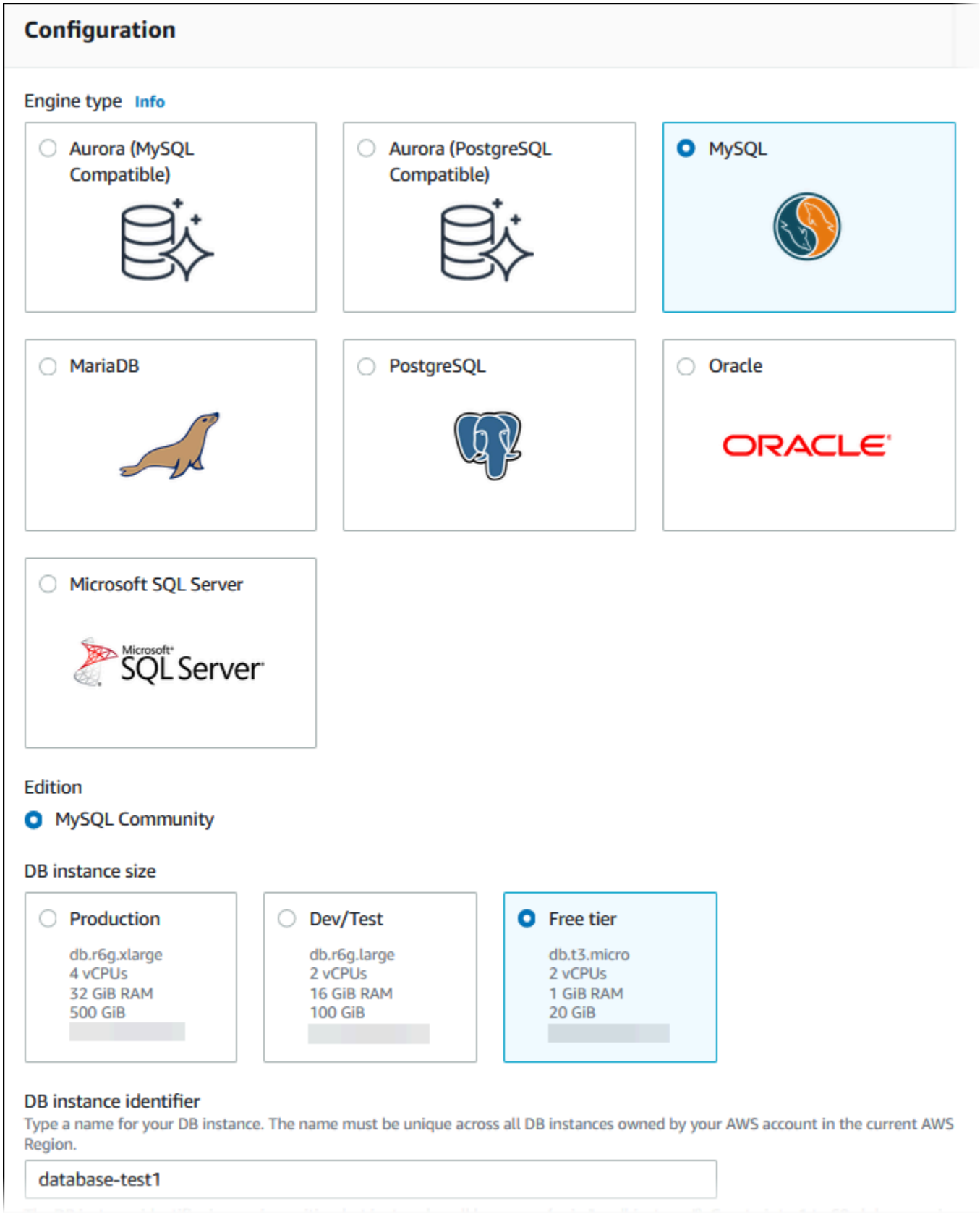
Pour créer une instance de base de données MySQL avec l'option Easy create (Création facile)

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez celui Région AWS que vous avez utilisé pour l' EC2instance précédemment.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données) et veillez à choisir Easy create (Création facile).



5. Dans Configuration, choisissez MySQL.
6. Pour DB instance size (Taille de l'instance de base de données), choisissez Free tier (Offre gratuite). Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.
7. Pour l'identifiant de l'instance DB, saisissez **`database-test1`**.
8. Pour Nom d'utilisateur principal, saisissez un nom pour l'utilisateur principal ou conservez le nom par défaut.

   La page Create database (Créer une base de données) doit ressembler à l'image suivante. Pour les comptes de forfait gratuit, le niveau gratuit apparaît. Pour les comptes payants, Sandbox apparaît.

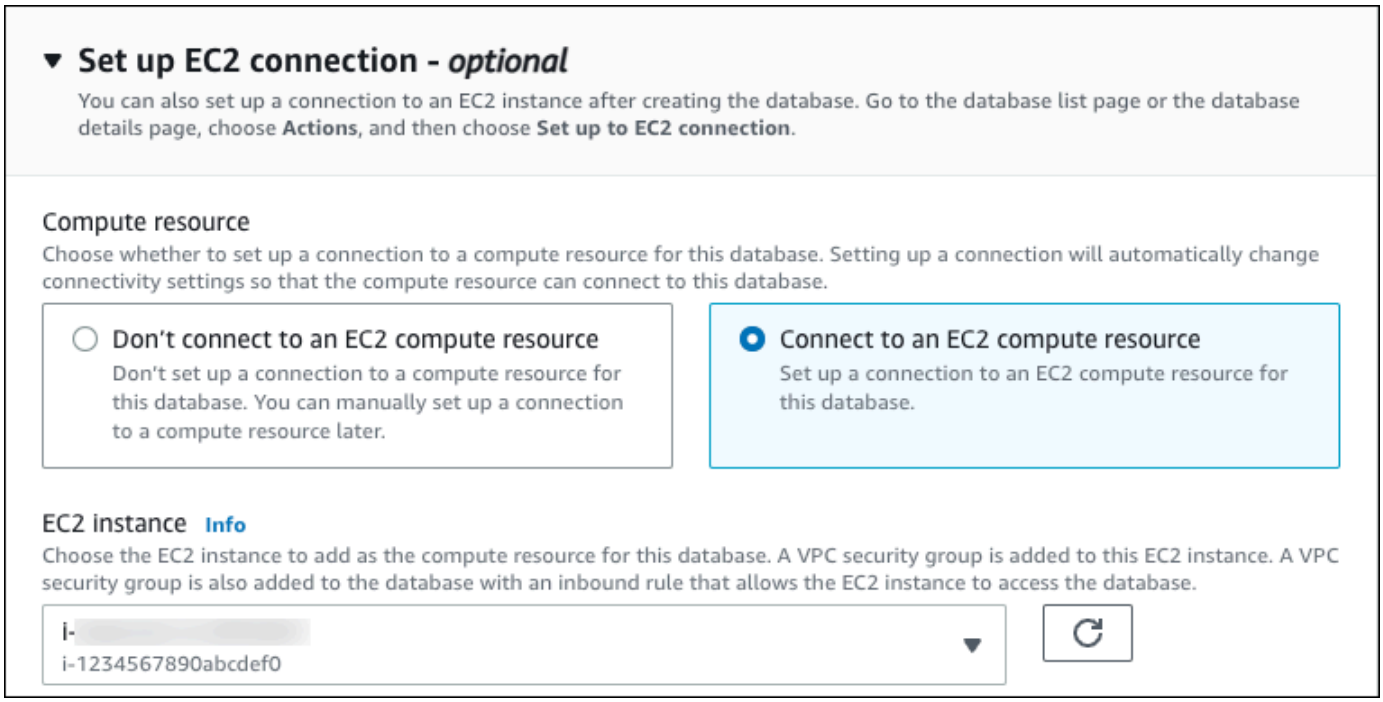

Configuration
Engine type Info
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
MariaDB
PostgreSQL
Oracle
ORACLE
Microsoft SQL Server
Microsoft SQL Server
Edition
MySQL Community
DB instance size
Production
db.r6g.xlarge
4 vCPUs
32 GiB RAM
500 GiB
Dev/Test
db.r6g.large
2 vCPUs
16 GiB RAM
100 GiB
Free tier
db.t3.micro
2 vCPUs
1 GiB RAM
20 GiB
DB instance identifier
Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.
database-test1

9. Pour utiliser un mot de passe principal généré automatiquement pour l'instance de base de données, sélectionnez Générer automatiquement un mot de passe.

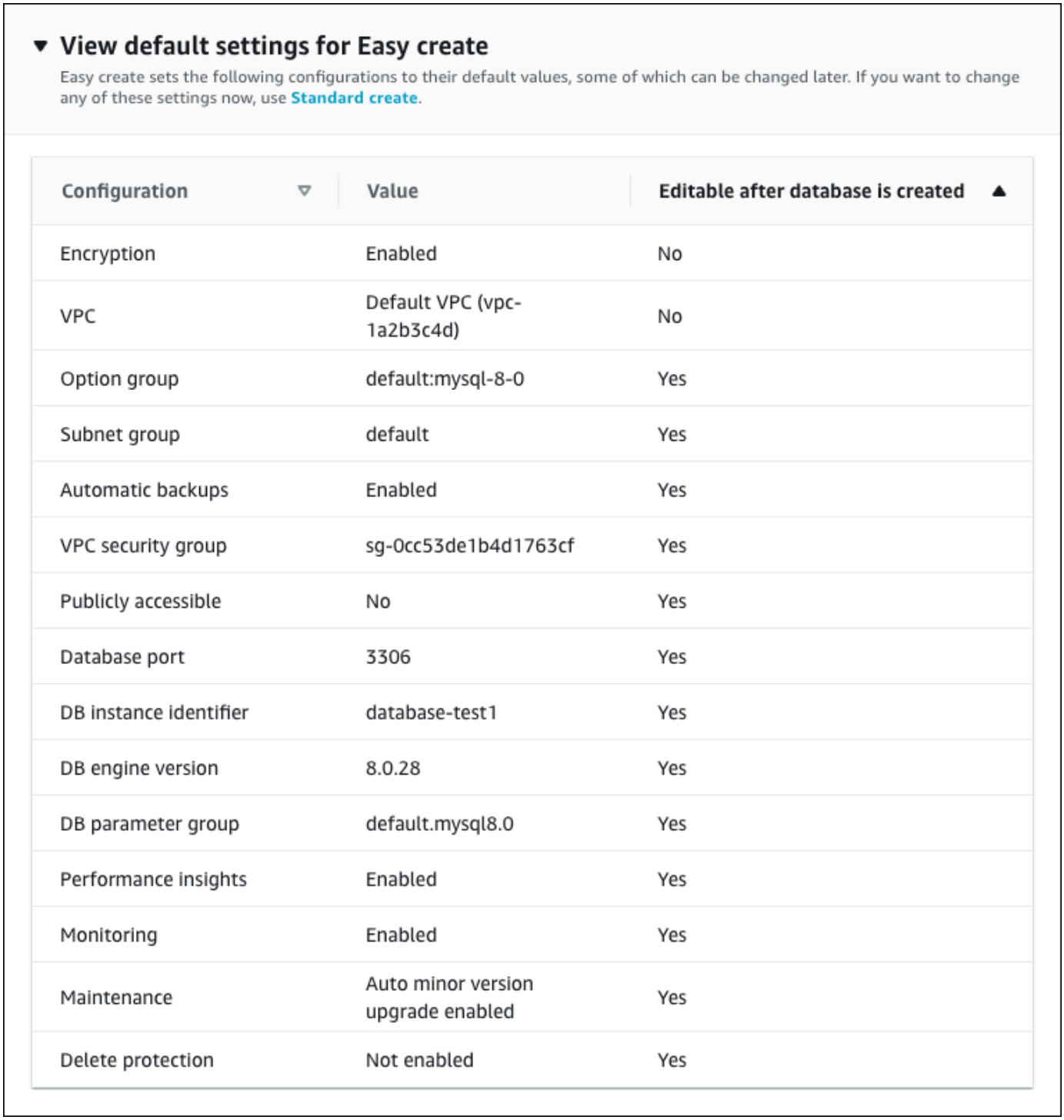
   Pour entrer votre mot de passe principal, veillez à ce que la case Générer automatiquement un mot de passe soit décochée, puis saisissez le même mot de passe dans Mot de passe principal et Confirmer le mot de passe.

10. Pour configurer une connexion avec l' EC2 instance que vous avez créée précédemment, ouvrez Configurer la EC2 connexion - facultatif.

    Sélectionnez Se connecter à une ressource EC2 de calcul. Choisissez l' EC2 instance que vous avez créée précédemment.



11. (Facultatif) Ouvrez View default settings for Easy Create (Afficher les paramètres par défaut pour Création facile).



View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

Configuration	Value	Editable after database is created
Encryption	Enabled	No
VPC	Default VPC (vpc-1a2b3c4d)	No
Option group	default:mysql-8-0	Yes
Subnet group	default	Yes
Automatic backups	Enabled	Yes
VPC security group	sg-0cc53de1b4d1763cf	Yes
Publicly accessible	No	Yes
Database port	3306	Yes
DB instance identifier	database-test1	Yes
DB engine version	8.0.28	Yes
DB parameter group	default.mysql8.0	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

Vous pouvez examiner les paramètres par défaut utilisés quand l'option Easy create (Création facile) est activée. La colonne Modifiable après la création de la base de données indique les options que vous pouvez modifier après avoir créé la base de données.

- Si un réglage contient Non dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données.
- Si un réglage contient Oui dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données ou vous pouvez modifier l'instance de base de données après l'avoir créée pour modifier le réglage.

12. Choisissez Créer une base de données.

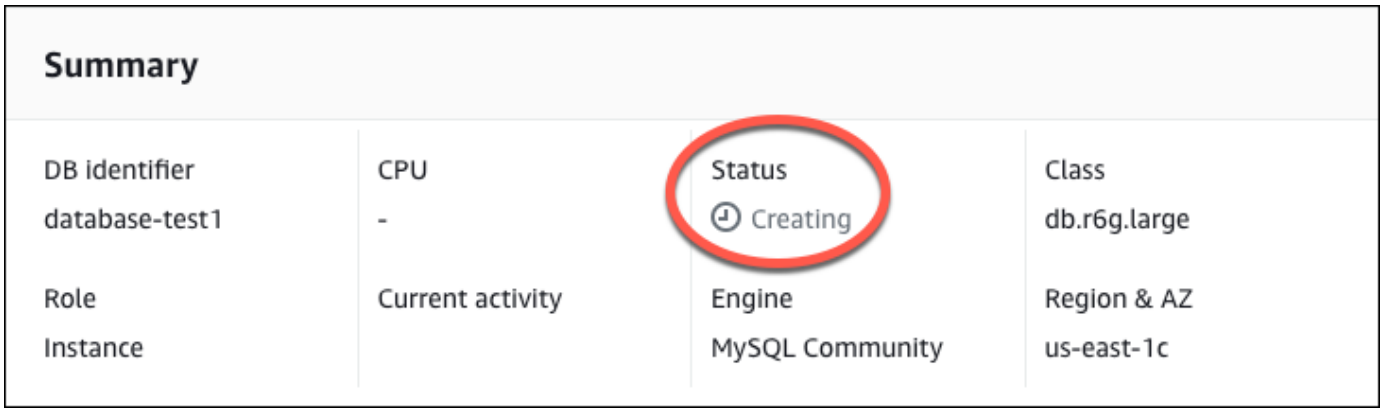
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Vous pouvez utiliser l'identifiant et le mot de passe affichés pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier.
    > Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, vous pouvez le faire en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

13. Dans la liste Bases de données, choisissez le nom de la nouvelle instance de base de données MySQL pour afficher ses détails.

    L'instance de base de données a le statut Création en cours jusqu'à ce qu'elle soit prête à l'emploi.



Lorsque l'état passe à Available (Disponible), vous pouvez vous connecter à l'instance de base de données. En fonction de la quantité de stockage et de la classe d'instance de base de données, la mise à disposition de la nouvelle instance peut prendre jusqu'à 20 minutes.

(Facultatif) Créez un VPC, une instance et une EC2 instance MySQL en utilisant AWS CloudFormation

Au lieu d'utiliser la console pour créer votre VPC, votre instance et votre EC2 instance MySQL, vous pouvez utiliser la console AWS CloudFormation pour provisionner AWS des ressources en traitant l'infrastructure comme du code. Pour vous aider à organiser vos AWS ressources en unités plus petites et plus faciles à gérer, vous pouvez utiliser la fonctionnalité de pile AWS CloudFormation imbriquée. Pour plus d'informations, consultez les sections Création d'une pile sur la AWS CloudFormation console et Utilisation de piles imbriquées.

> **Important**
>
> AWS CloudFormation est gratuit, mais les ressources qui en CloudFormation découlent sont vivantes. Vous devez payer les frais d'utilisation standard pour ces ressources jusqu'à ce que vous y mettiez fin. Pour plus d'informations, consultez la section Tarification de RDS for MySQL.

Pour créer vos ressources à l'aide de la AWS CloudFormation console, procédez comme suit :

- Téléchargez le CloudFormation modèle
- Configurez vos ressources à l'aide de CloudFormation

Téléchargez le CloudFormation modèle

Un CloudFormation modèle est un fichier texte JSON ou YAML qui contient les informations de configuration relatives aux ressources que vous souhaitez créer dans la pile. Ce modèle crée également un VPC et un hôte bastion pour vous, ainsi que l'instance RDS.

Pour télécharger le fichier modèle, ouvrez le lien suivant, CloudFormation modèle MySQL.

Sur la page Github, cliquez sur le bouton Télécharger le fichier brut pour enregistrer le modèle de fichier YAML.

Configurez vos ressources à l'aide de CloudFormation

> **Note**
>
> Avant de commencer ce processus, assurez-vous que vous disposez d'une paire de clés pour une EC2 instance dans votre Compte AWS. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Lorsque vous utilisez le AWS CloudFormation modèle, vous devez sélectionner les paramètres appropriés pour vous assurer que vos ressources sont créées correctement. Procédez de la façon suivante :

1. Connectez-vous à la AWS CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile.
3. Dans la section Spécifier le modèle, sélectionnez Télécharger un fichier modèle depuis votre ordinateur, puis cliquez sur Suivant.
4. Dans la page Spécifier les détails de la pile, définissez les paramètres suivants :
   a. Définissez le nom de la pile sur My SQLTest Stack.
   b. Sous Paramètres, définissez les zones de disponibilité en sélectionnant trois zones de disponibilité.
   c. Dans Configuration de l'hôte Linux Bastion, pour Nom de la clé, sélectionnez une paire de clés pour vous connecter à votre EC2 instance.
   d. Dans les paramètres de configuration de l'hôte Linux Bastion, définissez la plage d'adresses IP autorisées sur votre adresse IP. Pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH), déterminez votre adresse IP publique à l'aide du service à l'adresse. https://checkip.amazonaws.com Exemple d'adresse IP : 192.0.2.1/32.

      > **Warning**
      >
      > Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les

environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

e. Dans Configuration générale de la base de données, définissez la classe d'instance de base de données sur db.t3.micro.

f. Définissez le nom de base de données sur**`database-test1`**.

g. Dans Nom d'utilisateur principal de base de données, entrez le nom de l'utilisateur principal.

h. Définissez le mot de passe utilisateur principal de Manage DB avec Secrets Manager sur `false` pour ce didacticiel.

i. Pour le mot de passe de la base de données, définissez le mot de passe de votre choix. N'oubliez pas ce mot de passe pour suivre les étapes suivantes du didacticiel.

j. Sous Configuration du stockage de base de données, définissez le type de stockage de base de données sur gp2.

k. Dans Configuration de la surveillance des bases de données, définissez Enable RDS Performance Insights sur false.

l. Conservez tous les autres paramètres comme valeurs par défaut. Cliquez sur Suivant pour continuer.

5. Sur la page Configurer les options de pile, conservez toutes les options par défaut. Cliquez sur Suivant pour continuer.
6. Sur la page Review Stack, sélectionnez Soumettre après avoir vérifié les options de la base de données et de l'hôte Linux Bastion.

Une fois le processus de création des piles terminé, visualisez les piles avec leurs noms BastionStacket leurs RDSNS pour noter les informations dont vous avez besoin pour vous connecter à la base de données. Pour plus d'informations, consultez la section Affichage des données et des ressources de la AWS CloudFormation pile sur le AWS Management Console.

Se connecter à une instance de base de données MySQL

Vous pouvez utiliser n'importe quelle application client SQL standard pour vous connecter à l'instance de base de données. Dans cet exemple, vous vous connectez à une instance de base de données MySQL en utilisant le client de ligne de commande mysql.

Pour se connecter à une instance de base de données MySQL

1. Trouvez le point de terminaison (nom DNS) et le numéro de port pour votre instance de base de données.

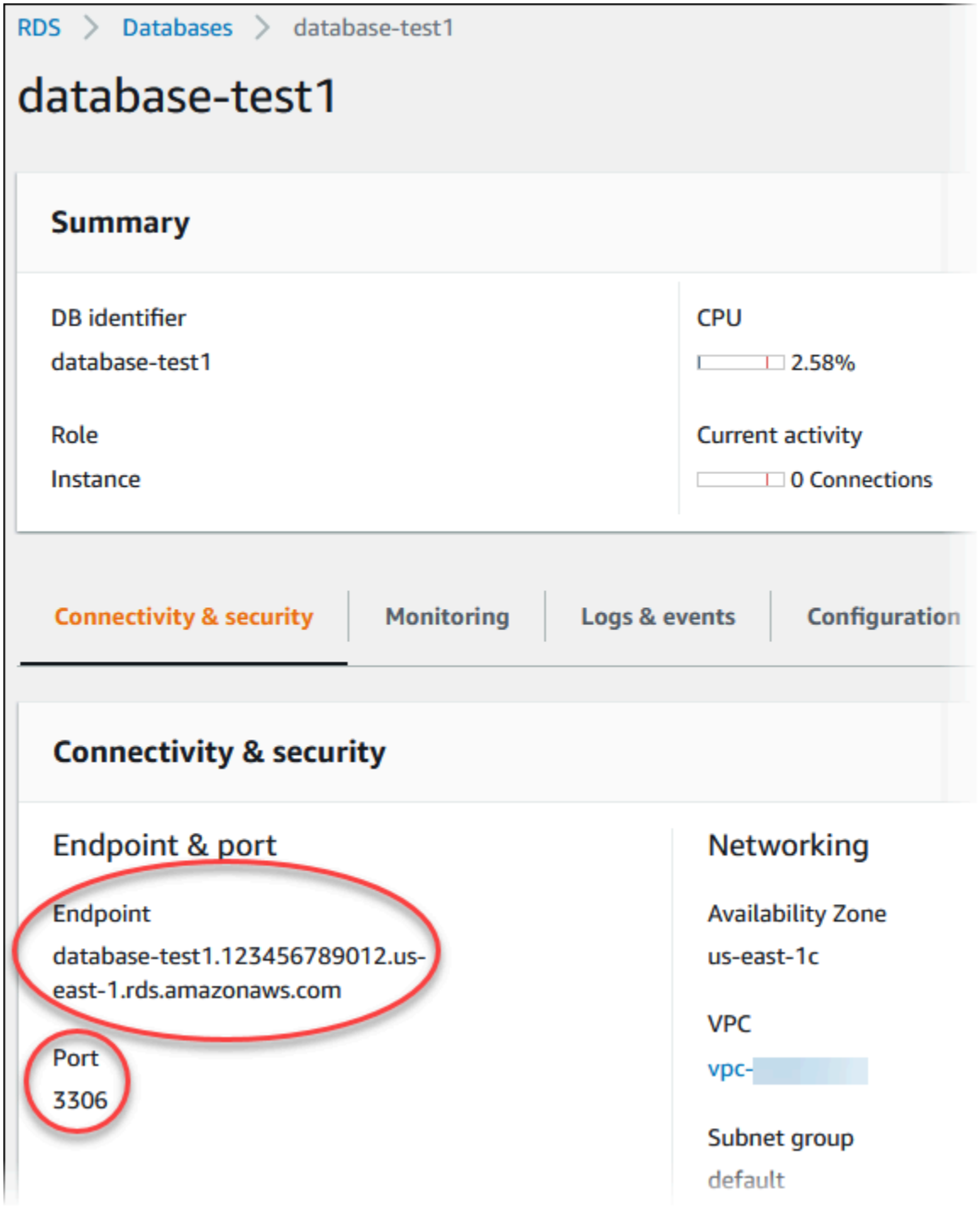
   a. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

   b. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance de base Région AWS de données.

   c. Dans la panneau de navigation, choisissez Databases (Bases de données).

   d. Choisissez le nom de l'instance de base de données MySQL pour afficher ses détails.

   e. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



2. Connectez-vous à l' EC2 instance que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Linux dans le guide de EC2 l'utilisateur Amazon.

   Nous vous recommandons de vous connecter à votre EC2 instance via SSH. Si l'utilitaire client SSH est installé sur Windows, Linux ou Mac, vous pouvez vous connecter à l'instance à l'aide du format de commande suivant :

```
ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
```

Supposons, par exemple, qu'`ec2-database-connect-key-pair.pem`il soit stocké `/dir1` sous Linux et que le IPv4 DNS public de votre EC2 instance l'est`ec2-12-345-678-90.compute-1.amazonaws.com`. Votre commande SSH se présenterait comme suit :

```
ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-
user@ec2-12-345-678-90.compute-1.amazonaws.com
```

3. Obtenez les dernières corrections de bogues et mises à jour de sécurité en mettant à jour le logiciel de votre EC2 instance. Pour ce faire, exécutez la commande suivante.

> **Note**
>
> L'option `-y` installe les mises à jour sans demander de confirmation. Pour examiner les mises à jour avant de les installer, omettez cette option.

```
sudo dnf update -y
```

4. Pour installer le client de ligne de commande mysql depuis MariaDB sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install mariadb105
```

5. Connectez-vous à l'instance de base de données MySQL. Par exemple, saisissez la commande suivante. Cette action vous permet de vous connecter à l'instance de base de données MySQL à l'aide du client MySQL.

   Remplacez le point de terminaison de votre instance de base de données (nom DNS) pour *`endpoint`* et remplacez le nom d'utilisateur principal que vous avez utilisé pour *`admin`*. Indiquez le mot de passe principal que vous avez utilisé lorsque vous êtes invité à entrer un mot de passe.

```
mysql -h endpoint -P 3306 -u admin -p
```

Après avoir entré le mot de passe pour l'utilisateur, le résultat suivant devrait normalement s'afficher.

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 3082
Server version: 8.0.28 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]>
```

Pour plus d'informations sur la connexion à votre instance de base de données MySQL, consultez Connexion à votre instance de base de données MySQL. Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Pour des raisons de sécurité, une bonne pratique consiste à recommander d'utiliser des connexions chiffrées. N'utilisez une connexion MySQL non chiffrée que quand le client et le serveur sont dans le même VPC et que le réseau est approuvé. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Connexion à votre instance My SQL DB sur Amazon RDS SSL avec/ TLS depuis le client de SQL ligne de commande My (crypté).

6. Exécutez des commandes SQL.

   Par exemple, la commande SQL suivante indique la date et l'heure actuelles :

   ```
   SELECT CURRENT_TIMESTAMP;
   ```

Supprimer l' EC2 instance et l'instance de base de données

Après vous être connecté et exploré l'exemple d' EC2 instance et l'instance de base de données que vous avez créées, supprimez-les afin qu'ils ne vous soient plus facturés.

Si vous aviez AWS CloudFormation l'habitude de créer des ressources, ignorez cette étape et passez à l'étape suivante.

Pour supprimer l' EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l' EC2 instance, puis choisissez État de l'instance, Terminer l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une EC2 instance, consultez Résilier votre instance dans le guide de EC2 l'utilisateur Amazon.

Pour supprimer l'instance de base de données sans instantané de base de données final

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Décochez Créer un instantané final et Conserver les sauvegardes automatiques.
6. Terminez la confirmation et choisissez Supprimer.

(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation

Si vous aviez l'habitude AWS CloudFormation de créer des ressources, supprimez la CloudFormation pile une fois que vous vous êtes connecté à l'instance d'exemple et à l' EC2 instance de base de données et que vous les avez explorées, afin qu'elles ne vous soient plus facturées.

Pour supprimer les CloudFormation ressources

1. Ouvrez la AWS CloudFormation console.
2. Sur la page Stacks du CloudFormationconsole, sélectionnez la pile racine (la pile sans nom VPCStack, BastionStack ou RDSNS).
3. Sélectionnez Delete (Supprimer).
4. Sélectionnez Supprimer la pile lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une pile dans CloudFormation, consultez la section Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Connecter votre instance de base de données à une fonction Lambda

Vous pouvez également connecter votre instance de base de données RDS for MySQL à une ressource de calcul sans serveur Lambda. Les fonctions Lambda vous permettent d'exécuter du code sans provisionner ni gérer l'infrastructure. Une fonction Lambda vous permet également de répondre automatiquement aux demandes d'exécution de code à n'importe quelle échelle, d'une douzaine d'événements par jour à des centaines par seconde. Pour de plus amples informations, veuillez consulter Connexion automatique d'une fonction Lambda et d'une instance de base de données.

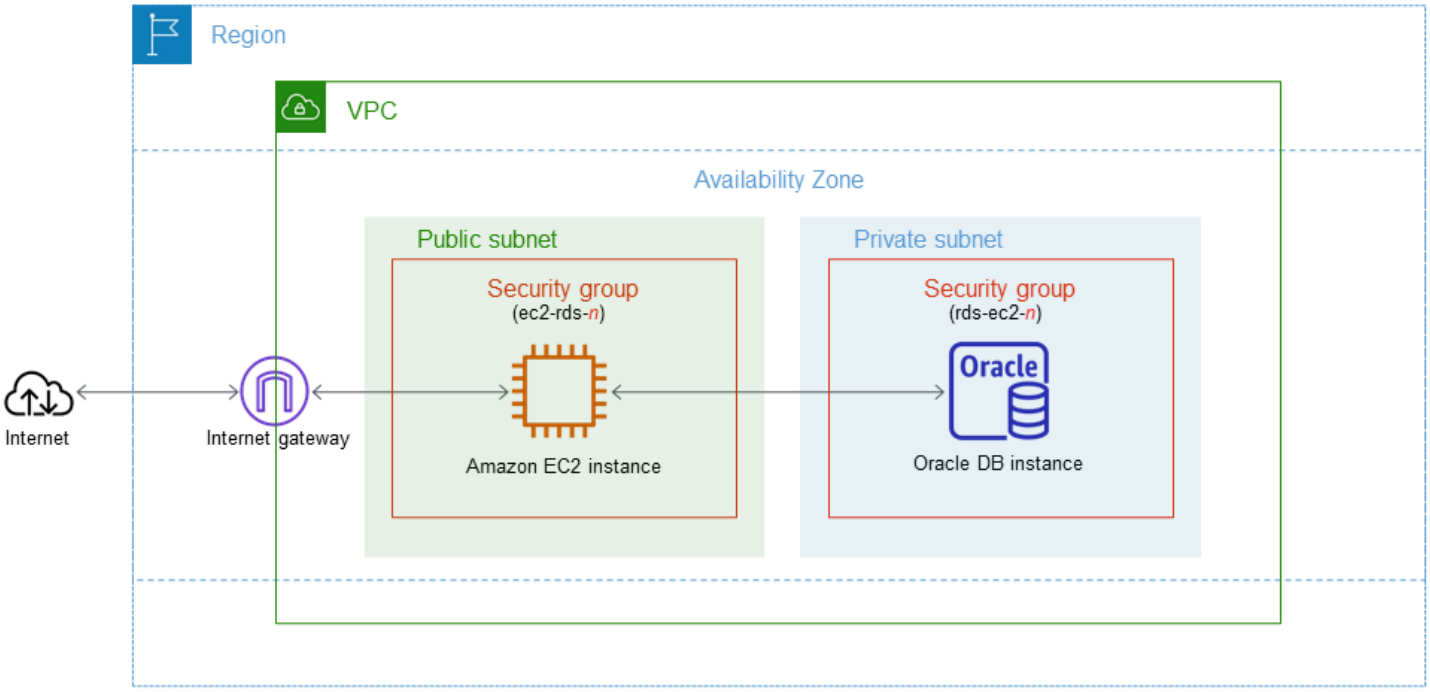
Création et connexion à une instance de base de données Oracle

Ce didacticiel crée une EC2 instance et une instance de base de données RDS pour Oracle. Le didacticiel explique comment accéder à l'instance de base de données à partir de l' EC2 instance à l'aide d'un client Oracle standard. En tant que bonne pratique, ce didacticiel crée une instance de base de données privée dans un cloud privé virtuel (VPC). Dans la plupart des cas, les autres ressources du même VPC, telles que les EC2 instances, peuvent accéder à l'instance de base de données, mais les ressources extérieures au VPC ne peuvent pas y accéder.

Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Dans une zone de disponibilité, l' EC2 instance se trouve dans le sous-réseau public et l'instance de base de données dans le sous-réseau privé.

> **Important**
>
> La création d'un AWS compte est gratuite. Cependant, en suivant ce didacticiel, les AWS ressources que vous utilisez peuvent vous coûter cher. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Ce didacticiel vous permet de créer vos ressources en utilisant l'une des méthodes suivantes :

1. Utilisez le AWS Management Console - Étape 2 : Créer une instance de base de données Oracle et Étape 1 : créer une EC2 instance
2. AWS CloudFormation À utiliser pour créer l'instance de base de données et l' EC2 instance - (Facultatif) Créez un VPC, une instance et une EC2 instance de base de données Oracle à l'aide de AWS CloudFormation

La première méthode utilise Easy create pour créer une instance de base de données Oracle privée avec le AWS Management Console. Ici, vous spécifiez uniquement le type de moteur de base de données, la taille de l'instance de base de données et l'identifiant de l'instance de base de données. L'option Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration.

Lorsque vous utilisez plutôt Standard Create, vous pouvez spécifier d'autres options de configuration lorsque vous créez une instance de base de données. Ces options incluent les paramètres de disponibilité, de sécurité, de sauvegarde et de maintenance. Pour créer une instance de base de données publique, vous devez utiliser Création standard. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Rubriques

- Prérequis
- Étape 1 : créer une EC2 instance
- Étape 2 : Créer une instance de base de données Oracle
- (Facultatif) Créez un VPC, une instance et une EC2 instance de base de données Oracle à l'aide de AWS CloudFormation
- Étape 3 : Connecter votre client SQL à une instance de base de données Oracle
- Étape 4 : Supprimer l' EC2 instance et l'instance de base de données
- (Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation
- (Facultatif) Connecter votre instance de base de données à une fonction Lambda

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

- Inscrivez-vous pour un Compte AWS

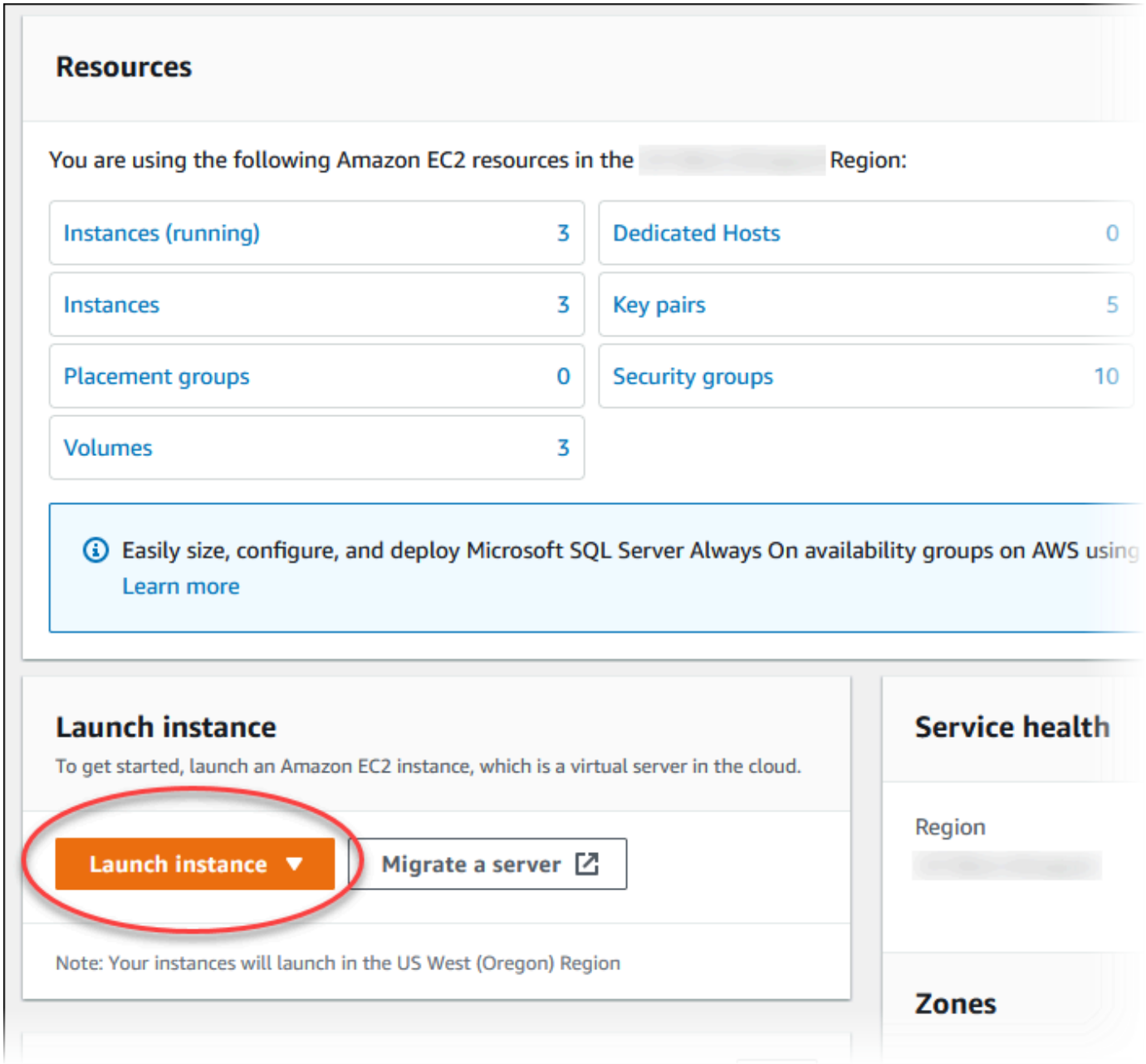
- Création d’un utilisateur doté d’un accès administratif

Étape 1 : créer une EC2 instance

Créez une EC2 instance Amazon que vous utiliserez pour vous connecter à votre base de données.

Pour créer une EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit de AWS Management Console, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l' EC2 instance.
3. Choisissez EC2 Dashboard, puis Launch instance, comme illustré dans l'image suivante.

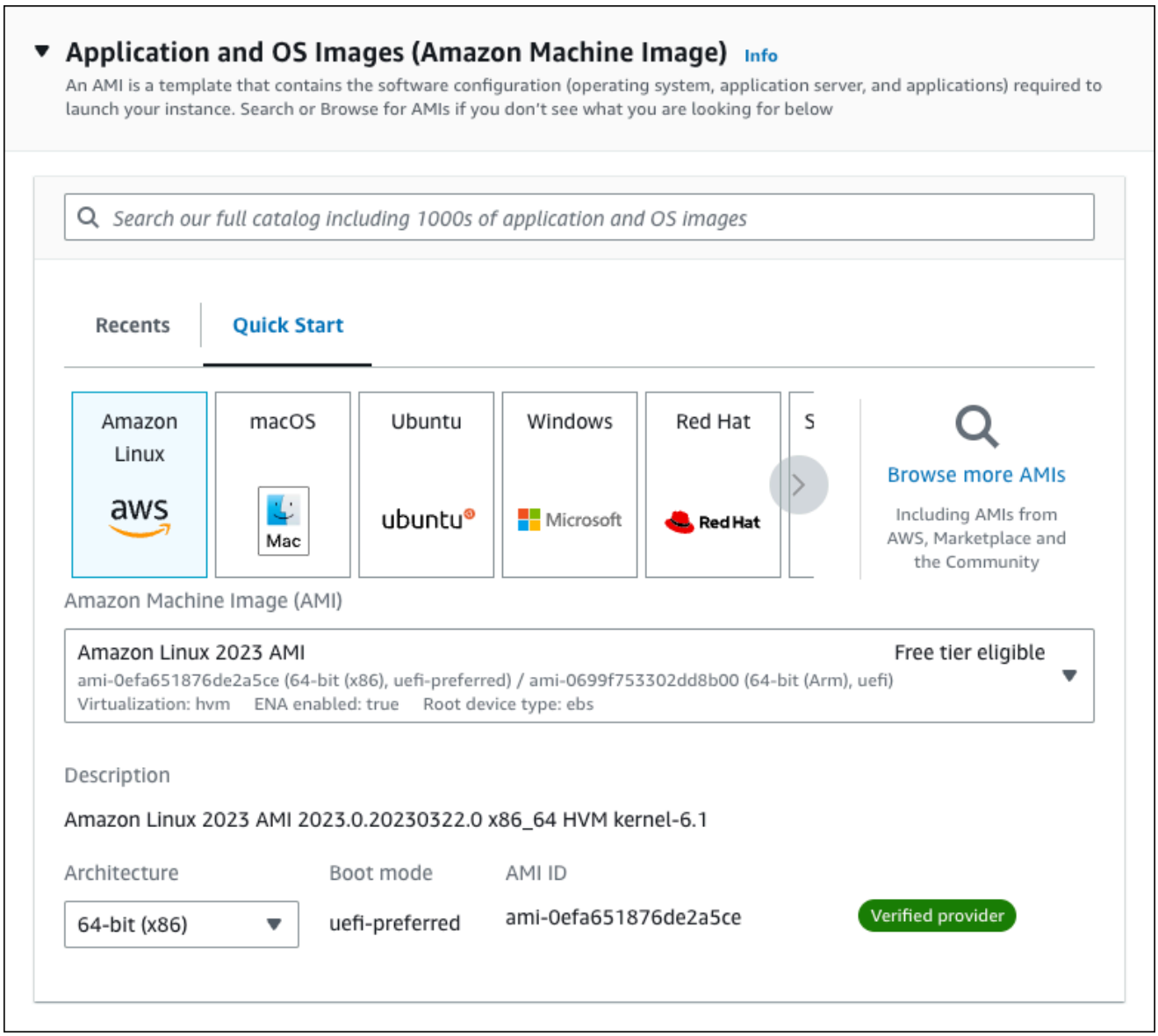


La page Lancer une instance s'ouvre.

4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`ec2-database-connect`**.

   b. Sous Application et images OS (Amazon Machine Image), choisissez Amazon Linux, puis Amazon Linux 2023 AMI. Conservez les sélections par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l' EC2 instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le guide de EC2 l'utilisateur Amazon.

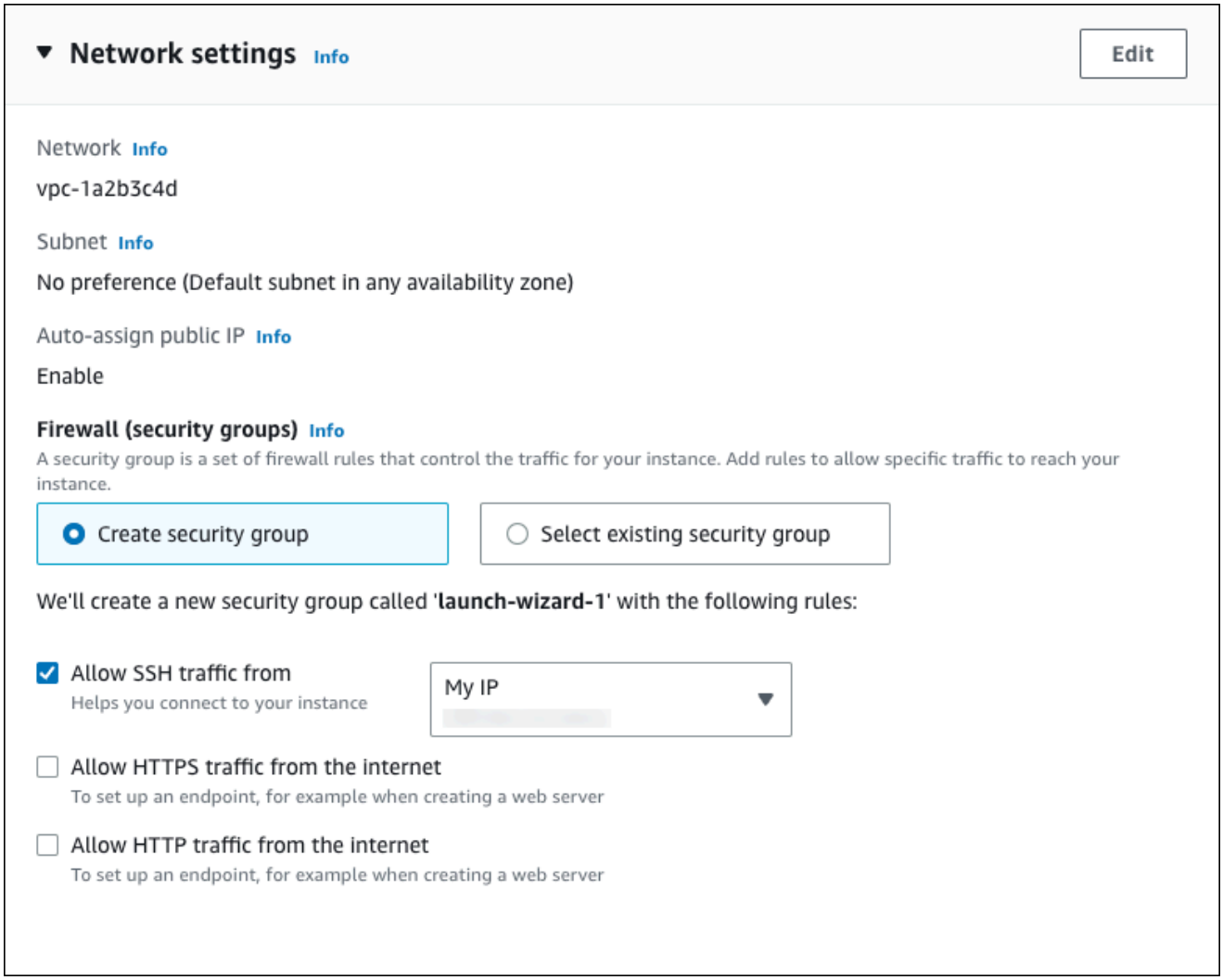
e. Pour Autoriser le trafic SSH dans les paramètres réseau, choisissez la source des connexions SSH à l' EC2instance.

Vous pouvez choisir My IP (Mon IP) si l'adresse IP affichée est correcte pour les connexions SSH. Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : 192.0.2.1/32.

Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

> **Warning**
>
> Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

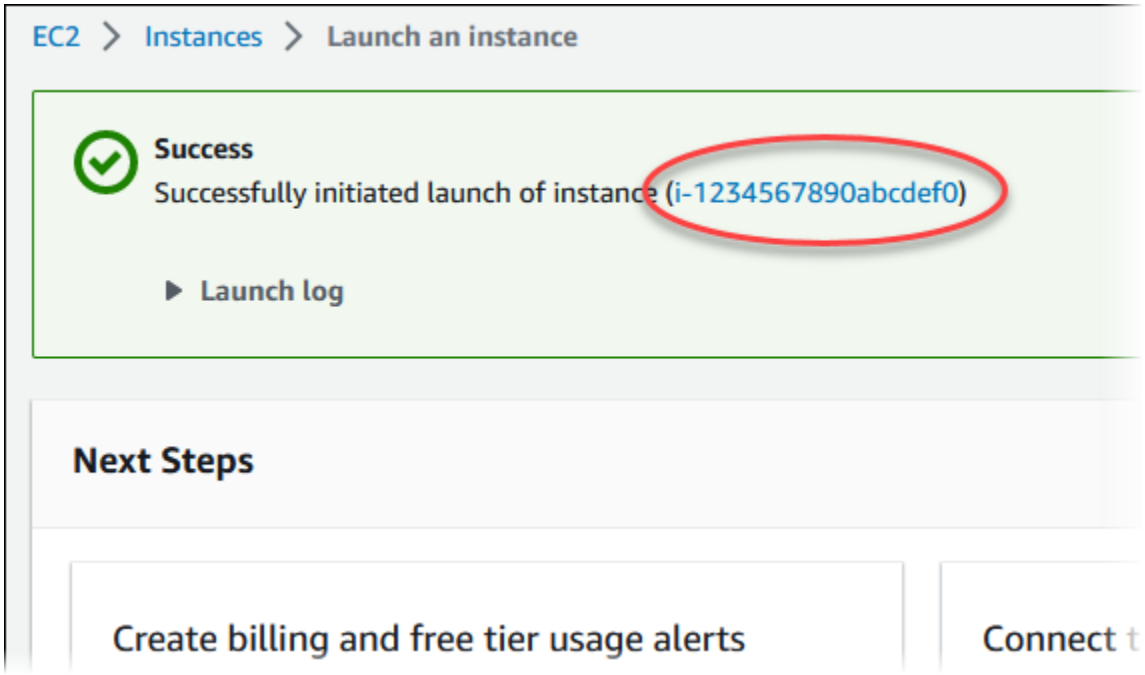
L'image suivante présente un exemple de la section Paramètres réseau.



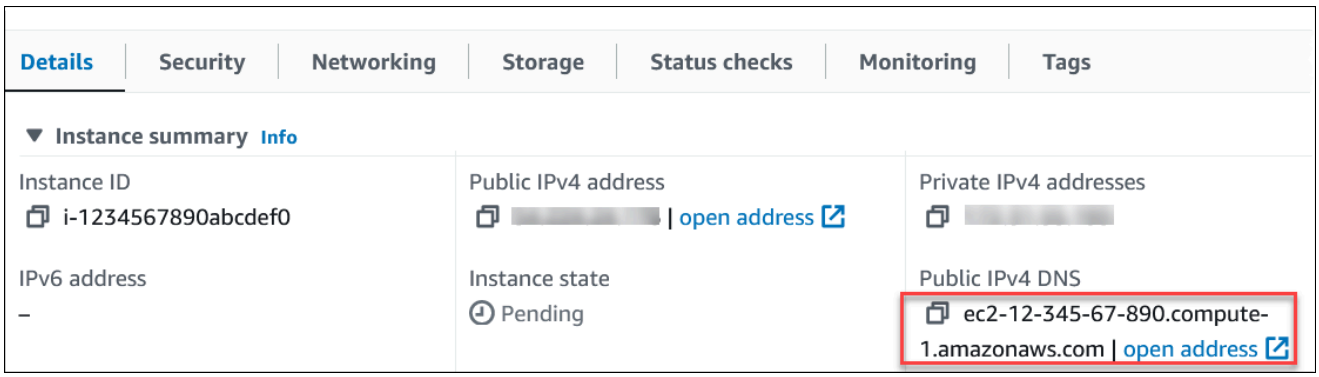
f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre EC2 instance dans le panneau Résumé et, lorsque vous êtes prêt, choisissez Launch instance.

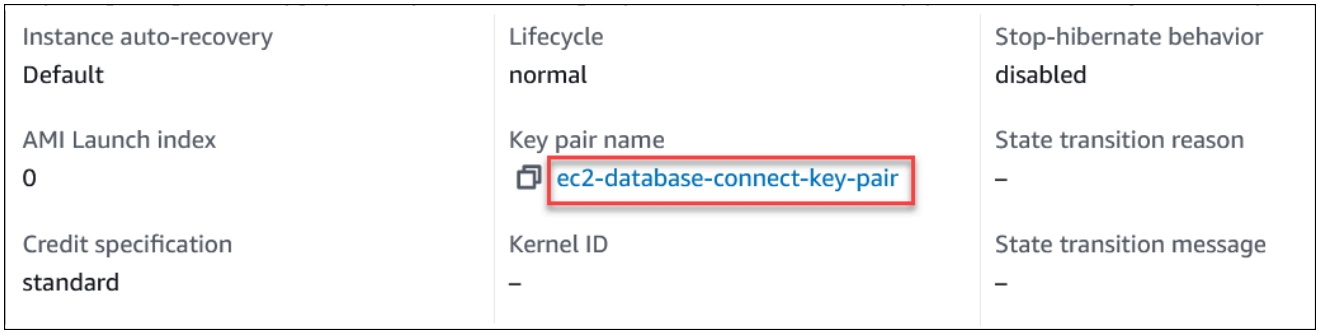
5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant d' EC2 instance pour ouvrir la liste des EC2 instances, puis sélectionnez votre EC2 instance.
7. Dans l'onglet Détails, notez les valeurs suivantes. Vous en aurez besoin lorsque vous vous connecterez via SSH :
   a. Dans Résumé de l'instance, notez la valeur du IPv4 DNS public.



   b. Dans Détails de l'instance, notez la valeur pour Nom de la paire de clés.



8. Attendez que l'état Instance de votre EC2 instance soit défini sur En cours d'exécution avant de continuer.

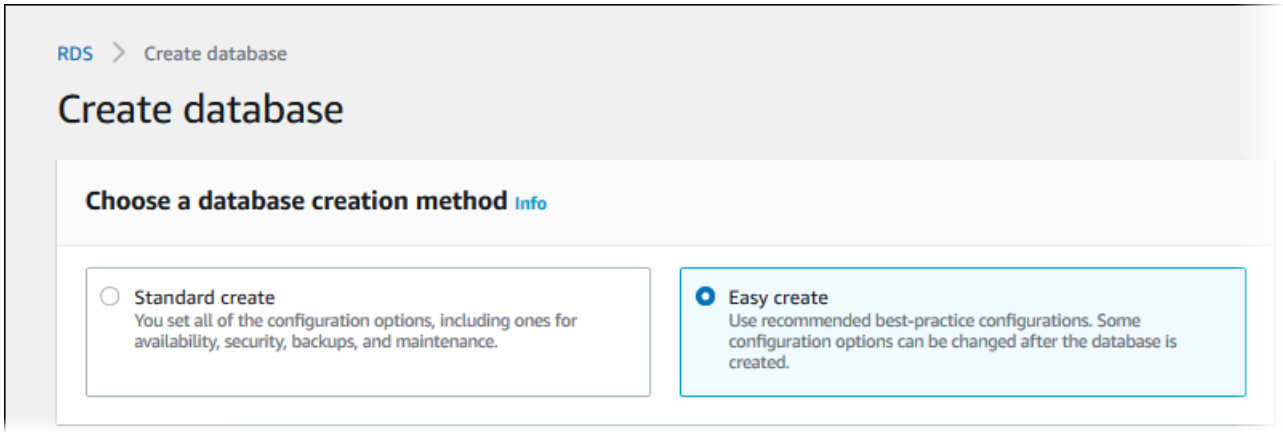
Étape 2 : Créer une instance de base de données Oracle

La fondation de base d'Amazon RDS est l'instance de base de données. Il s'agit de l'environnement dans lequel vous exécutez vos bases de données Oracle.

Dans cet exemple, vous utilisez Création facile pour créer une instance de base de données exécutant le moteur de base de données Oracle avec une classe d'instance de base de données db.m5.large.

Pour créer une instance de base de données Oracle avec l'option Création facile

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l'instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données) et veillez à choisir Easy create (Création facile).



5. Dans Configuration, choisissez Oracle.
6. Pour DB instance size (Taille de l'instance de base de données), choisissez Dev/Test.
7. Pour l'identifiant de l'instance DB, saisissez **`database-test1`**.
8. Pour Nom d'utilisateur principal, saisissez un nom pour l'utilisateur principal ou conservez le nom par défaut.

   La page Create database (Créer une base de données) doit ressembler à l'image suivante.

Configuration

Engine type Info

- Aurora (MySQL Compatible)
- Aurora (PostgreSQL Compatible)
- MySQL
- MariaDB
- PostgreSQL
- Oracle
- Microsoft SQL Server

Edition

- Oracle Enterprise Edition
  Affordable and full-featured database management system supporting up to 16 vCPUs.
- Oracle Standard Edition Two
  Affordable and full-featured database management system supporting up to 16 vCPUs. Oracle Database Standard Edition Two is a replacement for Standard Edition and Standard Edition One.

DB instance size

- Production
  db.r5.large
  2 vCPUs
  16 GiB RAM
  500 GiB
- Dev/Test
  db.m5.large
  2 vCPUs
  8 GiB RAM
  100 GiB

DB instance identifier

Type a name for your DB instance. The name must be unique across all DB instances owned by your AWS account in the current AWS Region.

database-test1

The DB instance identifier is case-insensitive, but is stored as all lowercase (as in "mydbinstance"). Constraints: 1 to 60 alphanumeric characters or hyphens. First character must be a letter. Can't contain two consecutive hyphens. Can't end with a hyphen.

Master username Info

Type a login ID for the master user of your DB instance.

admin

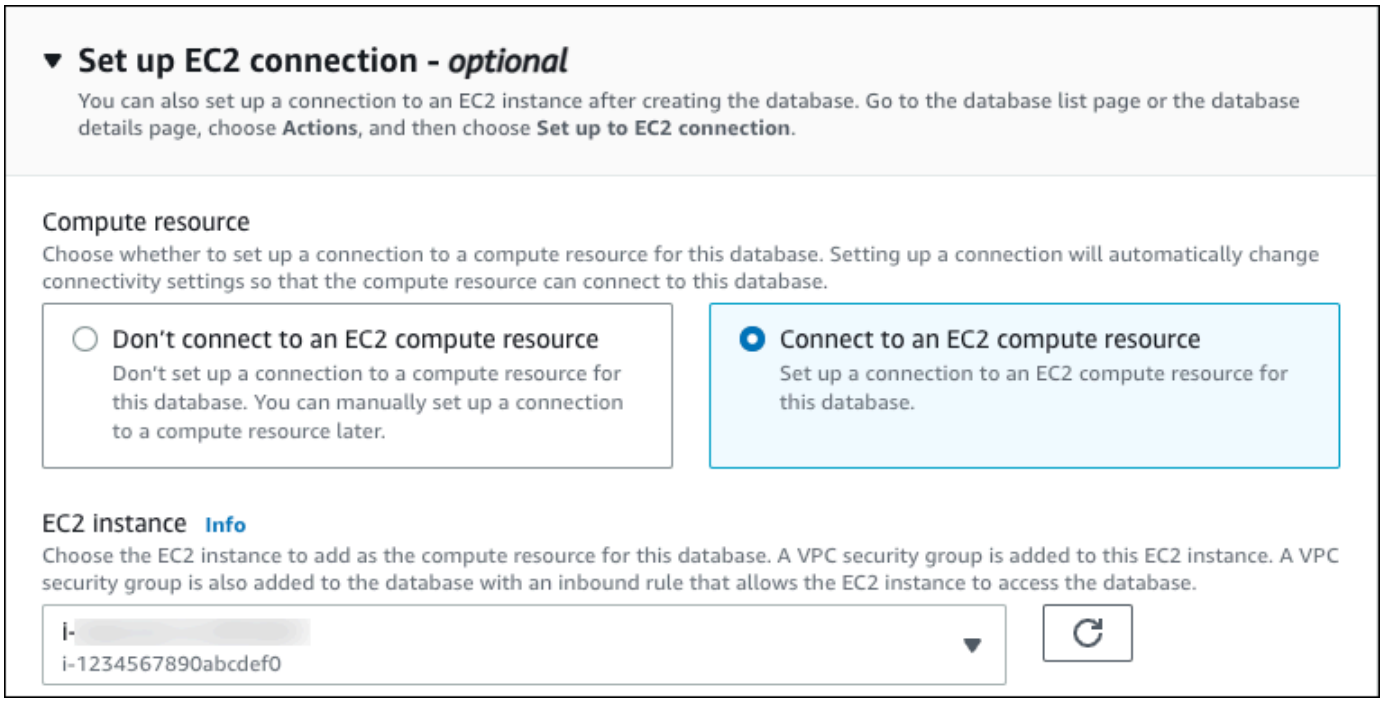
1 to 16 alphanumeric characters. First character must be a letter.

9. Pour utiliser un mot de passe principal généré automatiquement pour l'instance de base de données, sélectionnez Générer automatiquement un mot de passe.

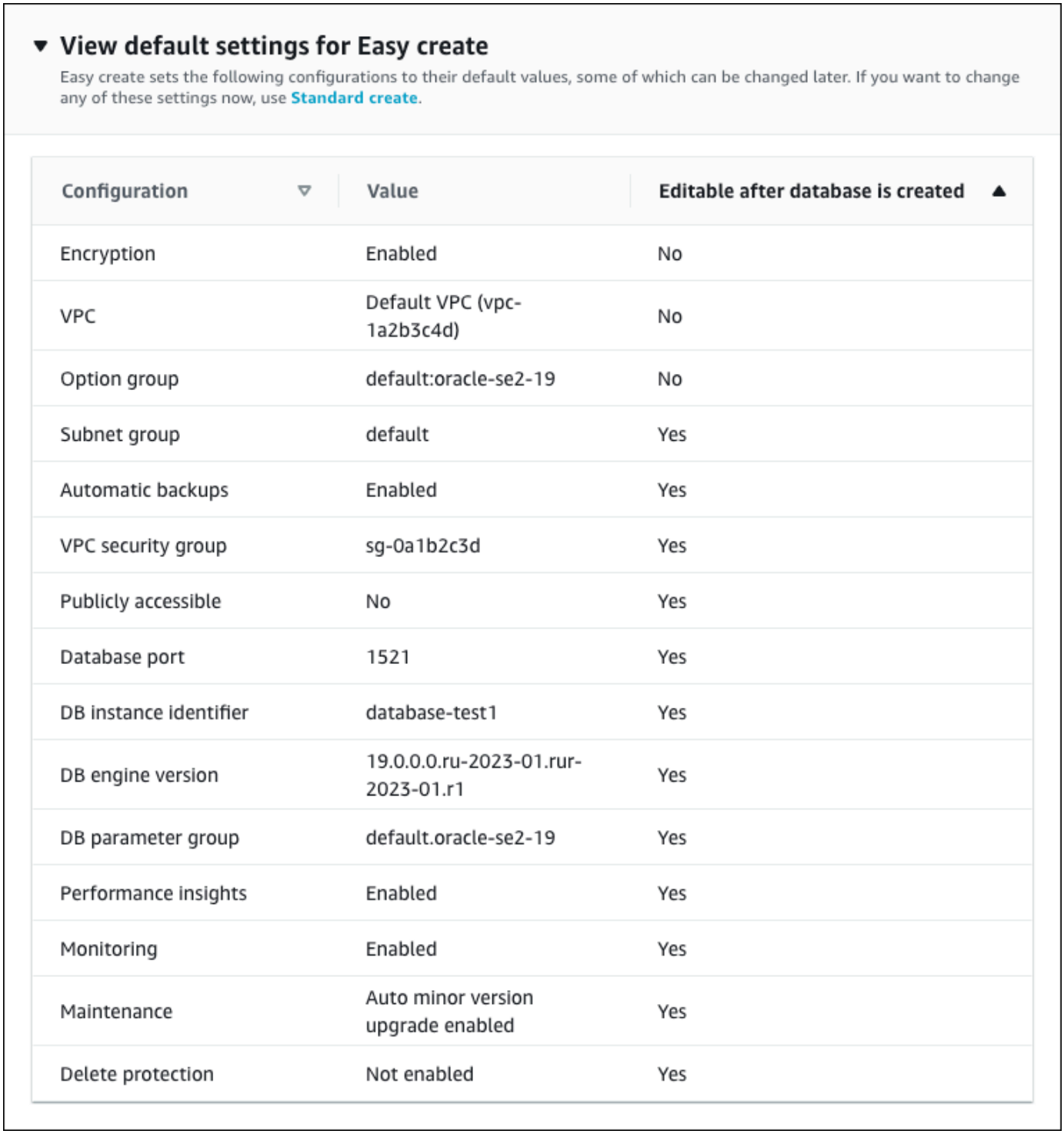
   Pour entrer votre mot de passe principal, veillez à ce que la case Générer automatiquement un mot de passe soit décochée, puis saisissez le même mot de passe dans Mot de passe principal et Confirmer le mot de passe.

10. Pour configurer une connexion avec l' EC2 instance que vous avez créée précédemment, ouvrez Configurer la EC2 connexion - facultatif.

    Sélectionnez Se connecter à une ressource EC2 de calcul. Choisissez l' EC2instance que vous avez créée précédemment.



11. Ouvrez Afficher les paramètres par défaut pour Création facile.



View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

Configuration	Value	Editable after database is created
Encryption	Enabled	No
VPC	Default VPC (vpc-1a2b3c4d)	No
Option group	default:oracle-se2-19	No
Subnet group	default	Yes
Automatic backups	Enabled	Yes
VPC security group	sg-0a1b2c3d	Yes
Publicly accessible	No	Yes
Database port	1521	Yes
DB instance identifier	database-test1	Yes
DB engine version	19.0.0.0.ru-2023-01.rur-2023-01.r1	Yes
DB parameter group	default.oracle-se2-19	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

Vous pouvez examiner les paramètres par défaut utilisés quand l'option Easy create (Création facile) est activée. La colonne Modifiable après la création de la base de données indique les options que vous pouvez modifier après avoir créé la base de données.

- Si un réglage contient Non dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données.
- Si un réglage contient Oui dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données ou vous pouvez modifier l'instance de base de données après l'avoir créée pour modifier le réglage.

12. Choisissez Créer une base de données.

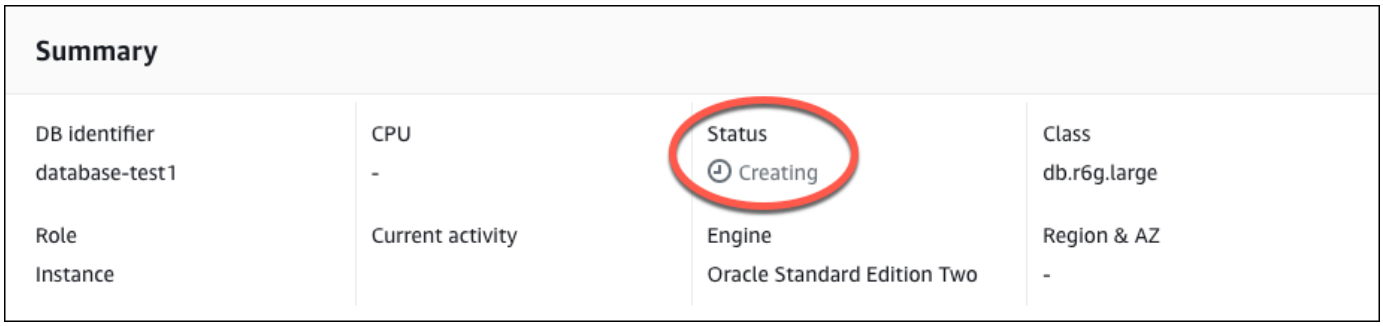
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Vous pouvez utiliser l'identifiant et le mot de passe affichés pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier.
    > Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, vous pouvez le faire en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

13. Dans la liste Bases de données, choisissez le nom de la nouvelle instance de base de données Oracle pour afficher ses détails.

    L'instance de base de données a le statut Création en cours jusqu'à ce qu'elle soit prête à l'emploi.



Lorsque l'état passe à Available (Disponible), vous pouvez vous connecter à l'instance de base de données. En fonction de la quantité de stockage et de la classe d'instance de base de données, la mise à disposition de la nouvelle instance peut prendre jusqu'à 20 minutes. Pendant la création de l'instance de base de données, vous pouvez passer à l'étape suivante et créer une EC2 instance.

(Facultatif) Créez un VPC, une instance et une EC2 instance de base de données Oracle à l'aide de AWS CloudFormation

Au lieu d'utiliser la console pour créer votre VPC, votre instance et votre EC2 instance de base de données Oracle, vous pouvez les utiliser AWS CloudFormation pour provisionner AWS des ressources en traitant l'infrastructure comme du code. Pour vous aider à organiser vos AWS ressources en unités plus petites et plus faciles à gérer, vous pouvez utiliser la fonctionnalité de pile AWS CloudFormation imbriquée. Pour plus d'informations, consultez les sections Création d'une pile sur la AWS CloudFormation console et Utilisation de piles imbriquées.

> **Important**
>
> AWS CloudFormation est gratuit, mais les ressources qui en CloudFormation découlent sont vivantes. Vous devez payer les frais d'utilisation standard pour ces ressources jusqu'à ce que vous y mettiez fin. Pour plus d'informations, consultez la section Tarification de RDS pour Oracle.

Pour créer vos ressources à l'aide de la AWS CloudFormation console, procédez comme suit :

- Étape 1 : Téléchargez le CloudFormation modèle
- Étape 2 : configurez vos ressources à l'aide de CloudFormation

Téléchargez le CloudFormation modèle

Un CloudFormation modèle est un fichier texte JSON ou YAML qui contient les informations de configuration relatives aux ressources que vous souhaitez créer dans la pile. Ce modèle crée également un VPC et un hôte bastion pour vous, ainsi que l'instance RDS.

Pour télécharger le fichier modèle, ouvrez le lien suivant, CloudFormation Modèle Oracle.

Sur la page Github, cliquez sur le bouton Télécharger le fichier brut pour enregistrer le modèle de fichier YAML.

Configurez vos ressources à l'aide de CloudFormation

> **Note**
>
> Avant de commencer ce processus, assurez-vous que vous disposez d'une paire de clés pour une EC2 instance dans votre Compte AWS. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Lorsque vous utilisez le AWS CloudFormation modèle, vous devez sélectionner les paramètres appropriés pour vous assurer que vos ressources sont créées correctement. Procédez de la façon suivante :

1. Connectez-vous à la AWS CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile.
3. Dans la section Spécifier le modèle, sélectionnez Télécharger un fichier modèle depuis votre ordinateur, puis cliquez sur Suivant.
4. Dans la page Spécifier les détails de la pile, définissez les paramètres suivants :
   a. Définissez le nom de la pile sur OracleTestStack.
   b. Sous Paramètres, définissez les zones de disponibilité en sélectionnant trois zones de disponibilité.
   c. Dans Configuration de l'hôte Linux Bastion, pour Nom de la clé, sélectionnez une paire de clés pour vous connecter à votre EC2 instance.
   d. Dans les paramètres de configuration de l'hôte Linux Bastion, définissez la plage d'adresses IP autorisées sur votre adresse IP. Pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH), déterminez votre adresse IP publique à l'aide du service à l'adresse. https://checkip.amazonaws.com Exemple d'adresse IP : 192.0.2.1/32.

      > **Warning**
      >
      > Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour

une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

e. Dans Configuration générale de la base de données, définissez la classe d'instance de base de données sur db.t3.micro.

f. Définissez le nom de base de données sur**`database-test1`**.

g. Dans Nom d'utilisateur principal de base de données, entrez le nom de l'utilisateur principal.

h. Définissez le mot de passe utilisateur principal de Manage DB avec Secrets Manager sur `false` pour ce didacticiel.

i. Pour le mot de passe de la base de données, définissez le mot de passe de votre choix. N'oubliez pas ce mot de passe pour suivre les étapes suivantes du didacticiel.

j. Sous Configuration du stockage de base de données, définissez le type de stockage de base de données sur gp2.

k. Dans Configuration de la surveillance des bases de données, définissez Enable RDS Performance Insights sur false.

l. Conservez tous les autres paramètres comme valeurs par défaut. Cliquez sur Suivant pour continuer.

5. Sur la page Configurer les options de pile, conservez toutes les options par défaut. Cliquez sur Suivant pour continuer.
6. Sur la page Review Stack, sélectionnez Soumettre après avoir vérifié les options de la base de données et de l'hôte Linux Bastion.

Une fois le processus de création des piles terminé, visualisez les piles avec leurs noms BastionStacket leurs RDSNS pour noter les informations dont vous avez besoin pour vous connecter à la base de données. Pour plus d'informations, consultez la section Affichage des données et des ressources de la AWS CloudFormation pile sur le AWS Management Console.

Étape 3 : Connecter votre client SQL à une instance de base de données Oracle

Vous pouvez utiliser n'importe quelle application client SQL standard pour vous connecter à votre instance de base de données. Dans cet exemple, vous vous connectez à une instance de base de données Oracle en utilisant le client de ligne de commande Oracle.

Pour se connecter à une instance de base de données Oracle

1. Trouvez le point de terminaison (nom DNS) et le numéro de port pour votre instance de base de données.

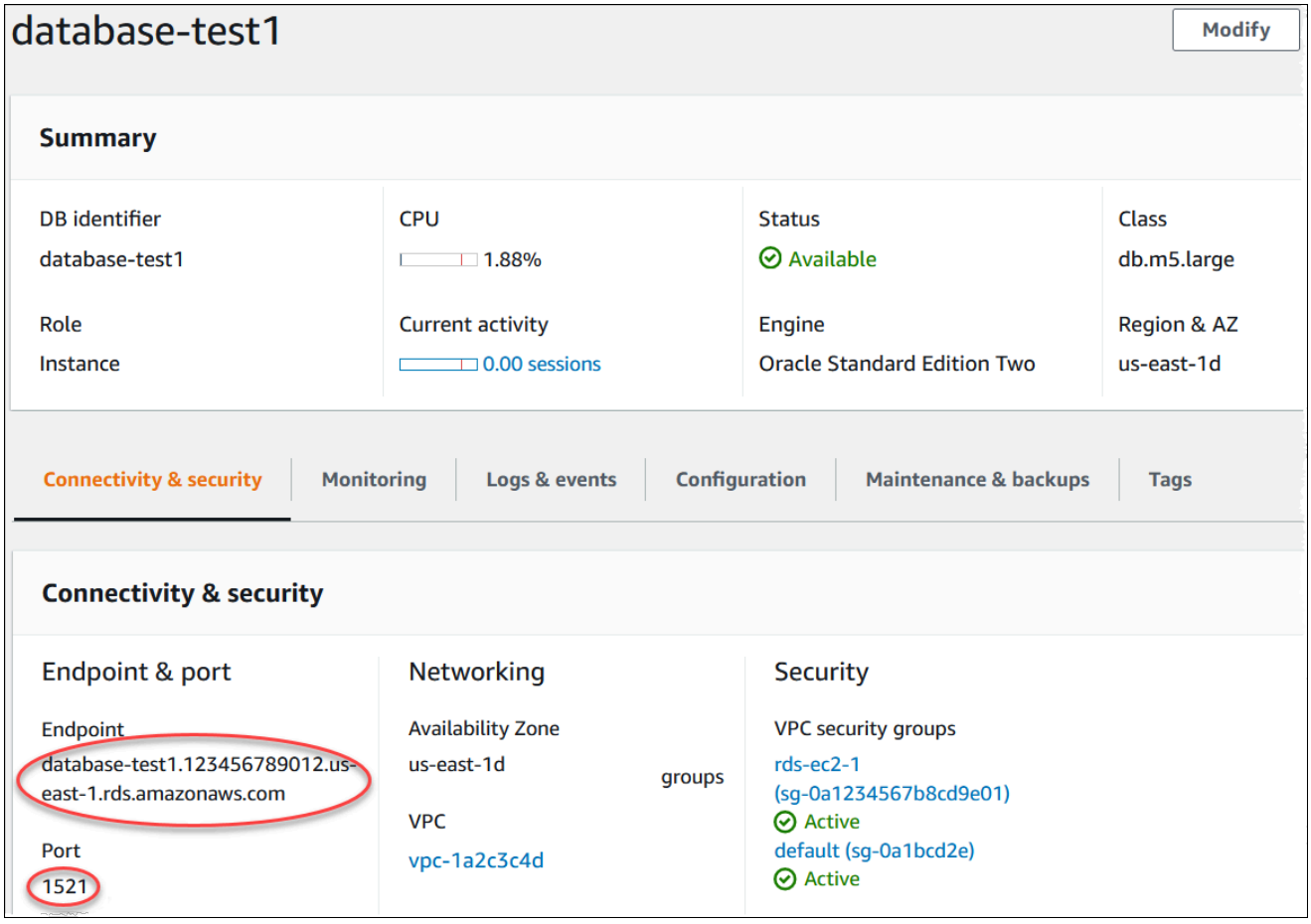
   a. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

   b. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS pour l'instance de base de données.

   c. Dans le panneau de navigation, choisissez Databases (Bases de données).

   d. Choisissez le nom de l'instance de base de données Oracle pour afficher ses détails.

   e. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



2. Connectez-vous à l' EC2 instance que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Linux dans le guide de EC2 l'utilisateur Amazon.

   Nous vous recommandons de vous connecter à votre EC2 instance via SSH. Si l'utilitaire client SSH est installé sur Windows, Linux ou Mac, vous pouvez vous connecter à l'instance à l'aide du format de commande suivant :

   ```
   ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
   ```

   Supposons, par exemple, qu'`ec2-database-connect-key-pair.pem`il soit stocké `/dir1` sous Linux et que le IPv4 DNS public de votre EC2 instance l'est`ec2-12-345-678-90.compute-1.amazonaws.com`. Votre commande SSH se présenterait comme suit :

   ```
   ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-
   user@ec2-12-345-678-90.compute-1.amazonaws.com
   ```

3. Obtenez les dernières corrections de bogues et mises à jour de sécurité en mettant à jour le logiciel de votre EC2 instance. Pour cela, utilisez la commande suivante.

   > **Note**
   >
   > L'option `-y` installe les mises à jour sans demander de confirmation. Pour examiner les mises à jour avant de les installer, omettez cette option.

   ```
   sudo dnf update -y
   ```

4. Dans un navigateur Web, accédez au https://www.oracle.com/database/technologies/instant-client/linuxfichier -x86-64-downloads.html.

5. Pour que la dernière version de la base de données s'affiche sur la page Web, copiez les liens .rpm (et non les liens .zip) pour le package Instant Client Basic et le package SQL*Plus. Par exemple, les liens suivants concernent la version 21.9 de la base de données Oracle :

   - https://download.oracle.com/otn_software/linux/instantclient/219000/oracle-instantclient-basic-21.9.0.0.0-1.el8.x86_64 .rpm

- https://download.oracle.com/otn_software/linux/instantclient/219000/oracle-instantclient-sqlplus-21.9.0.0.0-1.el8.x86_64 .rpm

6. Dans votre session SSH, exécutez la commande `wget` pour télécharger les fichiers .rpm à partir des liens que vous avez obtenus à l'étape précédente. L'exemple suivant télécharge les fichiers .rpm pour la version 21.9 de la base de données Oracle :

```
wget https://download.oracle.com/otn_software/linux/instantclient/219000/oracle-instantclient-basic-21.9.0.0.0-1.el8.x86_64.rpm
wget https://download.oracle.com/otn_software/linux/instantclient/219000/oracle-instantclient-sqlplus-21.9.0.0.0-1.el8.x86_64.rpm
```

7. Installez les packages en exécutant la commande `dnf` suivante :

```
sudo dnf install oracle-instantclient-*.rpm
```

8. Lancez SQL*Plus et connectez-vous à l'instance de base de données Oracle. Par exemple, saisissez la commande suivante.

   Remplacez le point de terminaison de votre instance de base de données (nom DNS) pour *`oracle-db-instance-endpoint`* et remplacez le nom d'utilisateur principal que vous avez utilisé pour *`admin`*. Lorsque vous utilisez Création facile pour Oracle, le nom de la base de données est `DATABASE`. Indiquez le mot de passe principal que vous avez utilisé lorsque vous êtes invité à entrer un mot de passe.

```
sqlplus admin@oracle-db-instance-endpoint:1521/DATABASE
```

   Après avoir entré le mot de passe pour l'utilisateur, le résultat suivant devrait normalement s'afficher.

```
SQL*Plus: Release 21.0.0.0.0 - Production on Wed Mar 1 16:41:28 2023
Version 21.9.0.0.0

Copyright (c) 1982, 2022, Oracle.  All rights reserved.

Enter password:
Last Successful login time: Wed Mar 01 2023 16:30:52 +00:00

Connected to:
Oracle Database 19c Standard Edition 2 Release 19.0.0.0.0 - Production
Version 19.18.0.0.0
```

```
SQL>
```

Pour plus d'informations sur la connexion à une instance de base de données RDS for Oracle, consultez Connexion à votre instance de base de données Oracle. Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Pour des raisons de sécurité, une bonne pratique consiste à recommander d'utiliser des connexions chiffrées. N'utilisez une connexion Oracle non chiffrée que quand le client et le serveur sont dans le même VPC et que le réseau est approuvé. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Sécurisation des connexions d'instance de base de données Oracle.

9. Exécutez des commandes SQL.

   Par exemple, la commande SQL suivante indique la date actuelle :

```
SELECT SYSDATE FROM DUAL;
```

Étape 4 : Supprimer l' EC2 instance et l'instance de base de données

Après vous être connecté et exploré l'exemple d' EC2 instance et l'instance de base de données que vous avez créées, supprimez-les afin qu'ils ne vous soient plus facturés.

Si vous aviez AWS CloudFormation l'habitude de créer des ressources, ignorez cette étape et passez à l'étape suivante.

Pour supprimer l' EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l' EC2 instance, puis choisissez État de l'instance, Terminer l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une EC2 instance, consultez Résilier votre instance dans le guide de EC2 l'utilisateur Amazon.

Pour supprimer l'instance de base de données sans instantané de base de données final

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Décochez Créer un instantané final et Conserver les sauvegardes automatiques.
6. Terminez la confirmation et choisissez Supprimer.

(Facultatif) Supprimez l' EC2 instance et l'instance de base de données créées avec CloudFormation

Si vous aviez l'habitude AWS CloudFormation de créer des ressources, supprimez la CloudFormation pile une fois que vous vous êtes connecté à l'instance d'exemple et à l' EC2 instance de base de données et que vous les avez explorées, afin qu'elles ne vous soient plus facturées.

Pour supprimer les CloudFormation ressources

1. Ouvrez la AWS CloudFormation console.
2. Sur la page Stacks du CloudFormationconsole, sélectionnez la pile racine (la pile sans nom VPCStack, BastionStack ou RDSNS).
3. Sélectionnez Delete (Supprimer).
4. Sélectionnez Supprimer la pile lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une pile dans CloudFormation, consultez la section Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Connecter votre instance de base de données à une fonction Lambda

Vous pouvez également connecter votre instance de base de données RDS for Oracle à une ressource de calcul sans serveur Lambda. Les fonctions Lambda vous permettent d'exécuter du code sans provisionner ni gérer l'infrastructure. Une fonction Lambda vous permet également de répondre

automatiquement aux demandes d'exécution de code à n'importe quelle échelle, d'une douzaine d'événements par jour à des centaines par seconde. Pour de plus amples informations, veuillez consulter Connexion automatique d'une fonction Lambda et d'une instance de base de données.

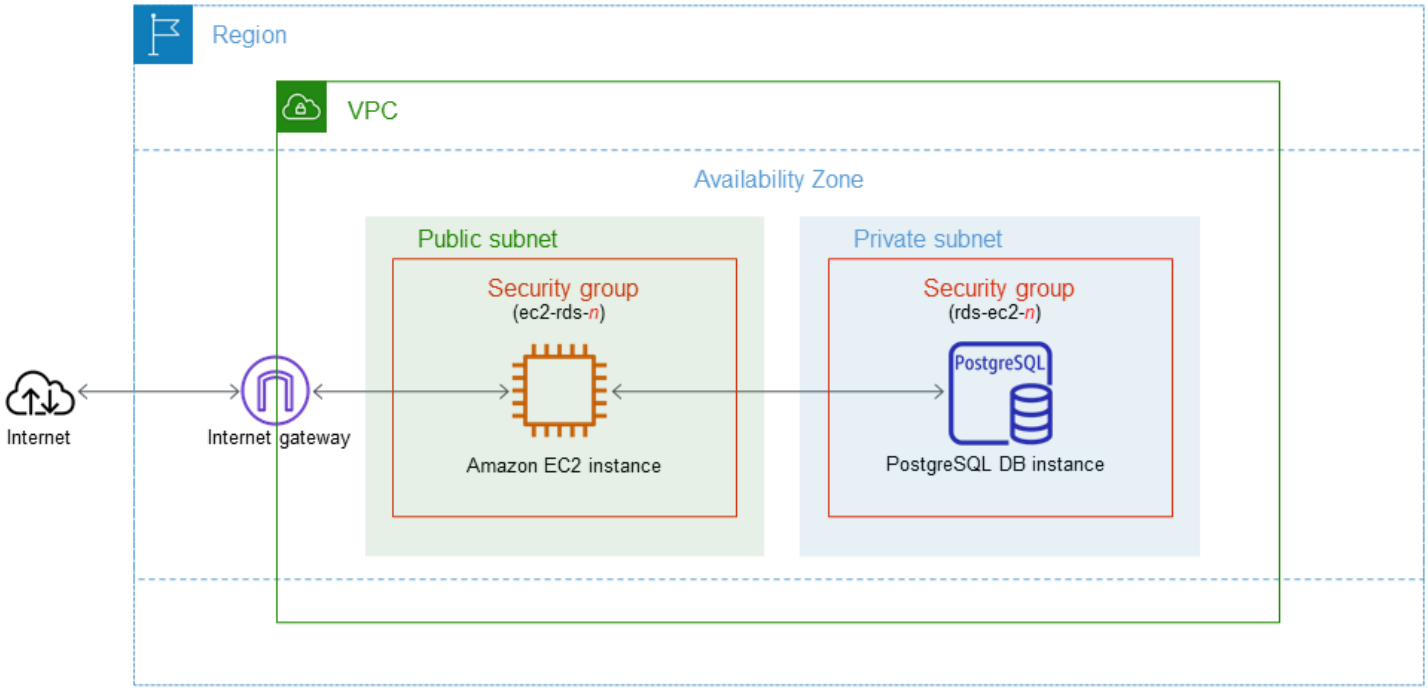
Création et connexion à une instance de base de données PostgreSQL

Ce didacticiel crée une EC2 instance et une instance de base de données RDS pour PostgreSQL. Le didacticiel explique comment accéder à l'instance de base de données depuis l' EC2 instance à l'aide d'un client PostgreSQL standard. En tant que bonne pratique, ce didacticiel crée une instance de base de données privée dans un cloud privé virtuel (VPC). Dans la plupart des cas, les autres ressources du même VPC, telles que les EC2 instances, peuvent accéder à l'instance de base de données, mais les ressources extérieures au VPC ne peuvent pas y accéder.

Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Dans une zone de disponibilité, l' EC2 instance se trouve dans le sous-réseau public et l'instance de base de données dans le sous-réseau privé.

> **Important**
>
> La création d'un AWS compte est gratuite. Cependant, en suivant ce didacticiel, les AWS ressources que vous utilisez peuvent vous coûter cher. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Ce didacticiel vous permet de créer vos ressources en utilisant l'une des méthodes suivantes :

1. Utilisez le AWS Management Console - Création d'une EC2 instance et Création d'une instance de base de données PostgreSQL
2. AWS CloudFormation À utiliser pour créer l'instance de base de données et l' EC2 instance - (Facultatif) Créez un VPC, une instance et une EC2 instance PostgreSQL à l'aide de AWS CloudFormation

La première méthode utilise Easy create pour créer une instance de base de données PostgreSQL privée avec le. AWS Management Console Ici, vous spécifiez uniquement le type de moteur de base de données, la taille de l'instance de base de données et l'identifiant de l'instance de base de données. L'option Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration.

Lorsque vous utilisez plutôt Standard Create, vous pouvez spécifier d'autres options de configuration lorsque vous créez une instance de base de données. Ces options incluent les paramètres de disponibilité, de sécurité, de sauvegarde et de maintenance. Pour créer une instance de base de données publique, vous devez utiliser Création standard. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Rubriques

- Prérequis
- Création d'une EC2 instance
- Création d'une instance de base de données PostgreSQL
- (Facultatif) Créez un VPC, une instance et une EC2 instance PostgreSQL à l'aide de AWS CloudFormation
- Se connecter à une instance DB PostgreSQL
- Supprimer l' EC2instance et l'instance de base de données
- (Facultatif) Supprimez l'EC2 instance et l'instance de base de données créées avec CloudFormation
- (Facultatif) Connecter votre instance de base de données à une fonction Lambda

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

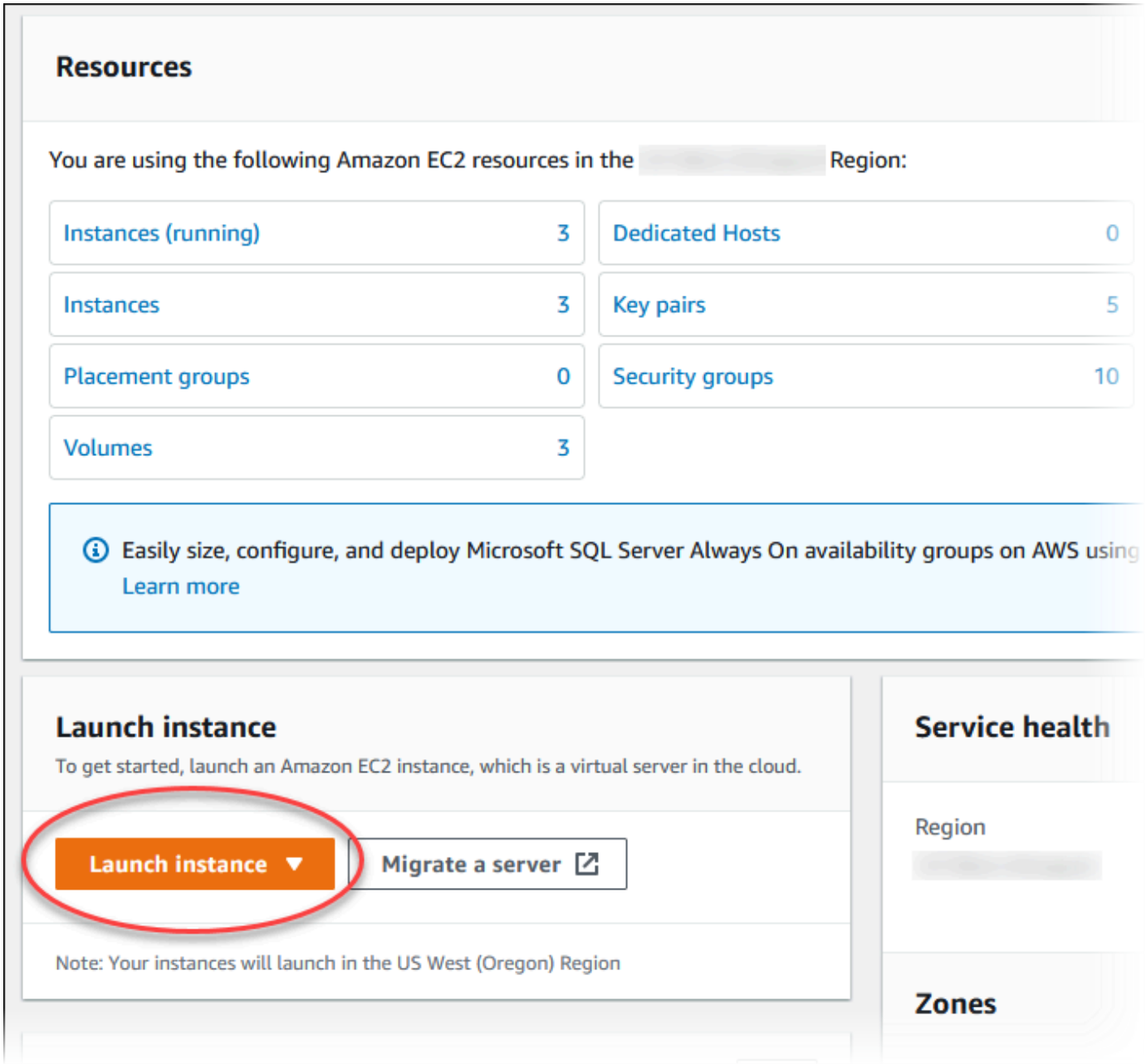
- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Création d'une EC2 instance

Créez une EC2 instance Amazon que vous utiliserez pour vous connecter à votre base de données.

Pour créer une EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit de AWS Management Console, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l' EC2 instance.
3. Choisissez EC2 Dashboard, puis Launch instance, comme illustré dans l'image suivante.

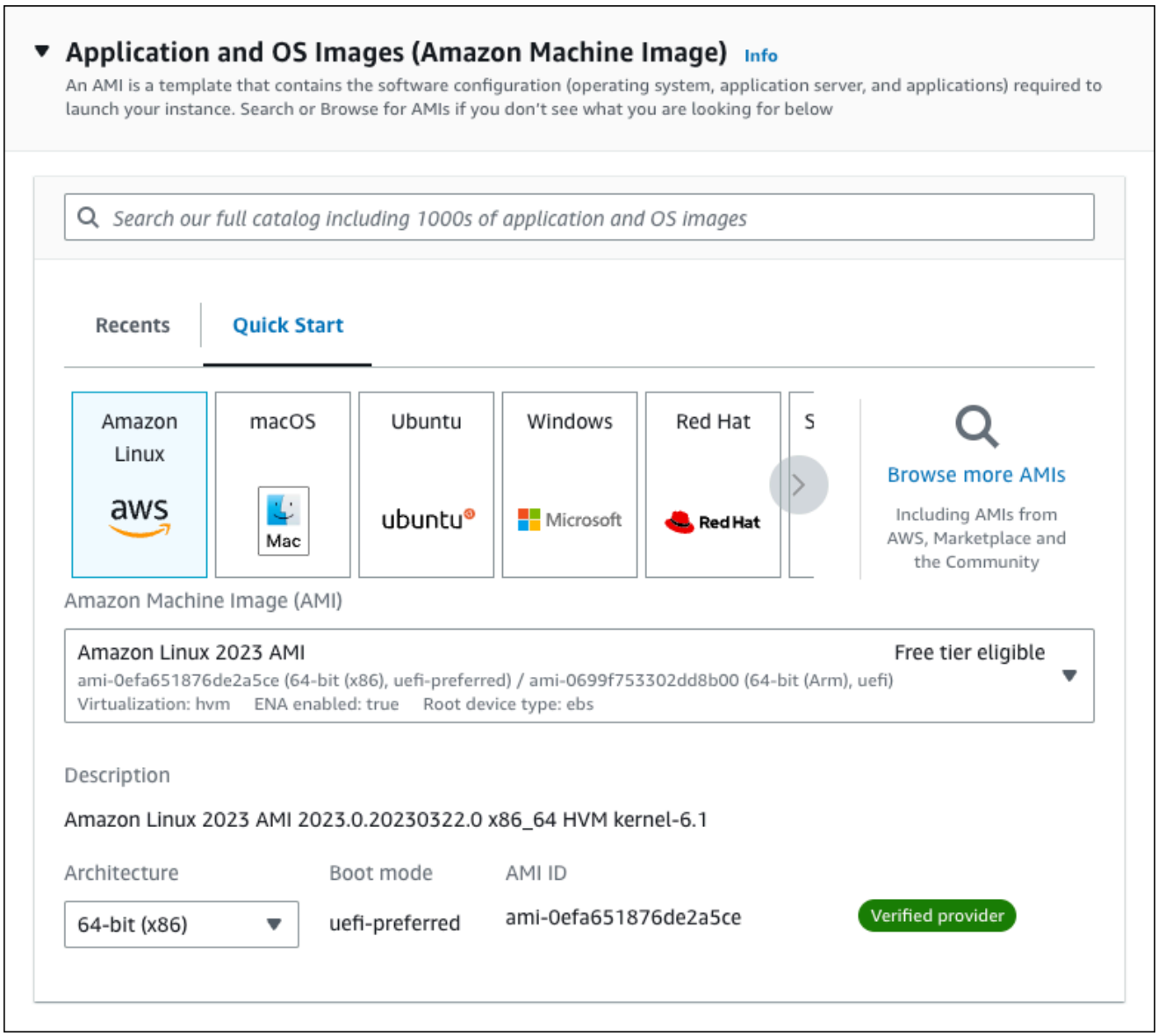


La page Lancer une instance s'ouvre.

4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`ec2-database-connect`**.

   b. Sous Application et images OS (Amazon Machine Image), choisissez Amazon Linux, puis Amazon Linux 2023 AMI. Conservez les sélections par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l' EC2 instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le guide de EC2 l'utilisateur Amazon.

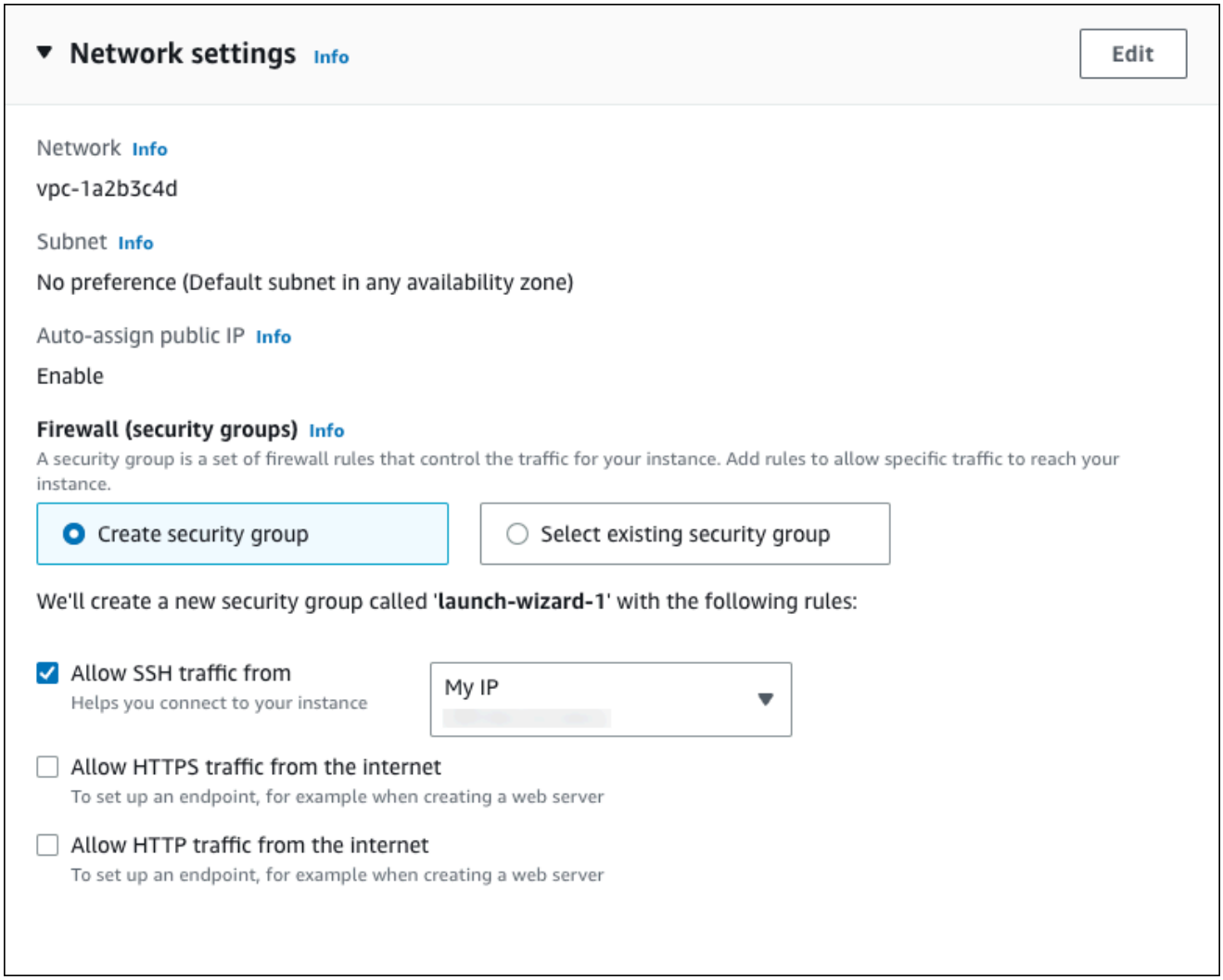
e. Pour Autoriser le trafic SSH dans les paramètres réseau, choisissez la source des connexions SSH à l' EC2instance.

Vous pouvez choisir My IP (Mon IP) si l'adresse IP affichée est correcte pour les connexions SSH. Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : 192.0.2.1/32.

Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

> **Warning**
>
> Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

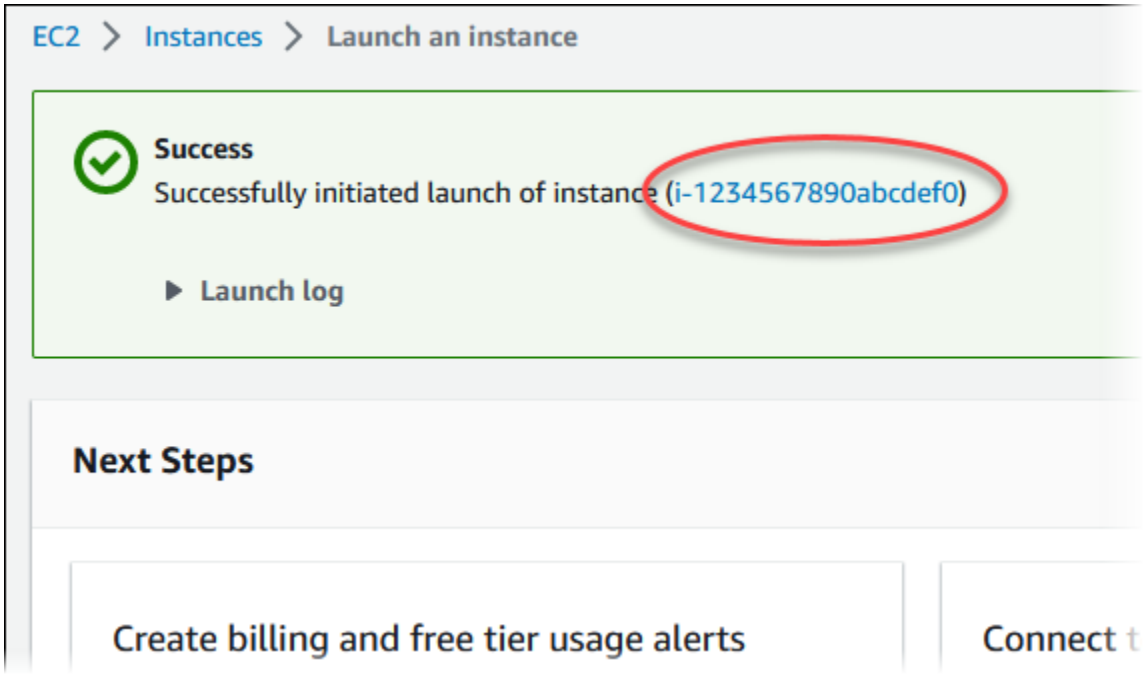
L'image suivante présente un exemple de la section Paramètres réseau.



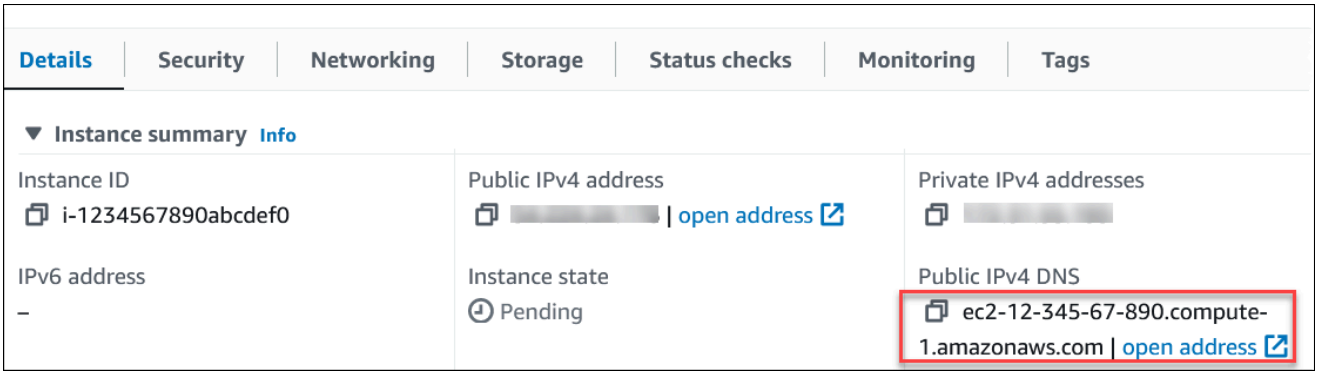
f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre EC2 instance dans le panneau Résumé et, lorsque vous êtes prêt, choisissez Launch instance.

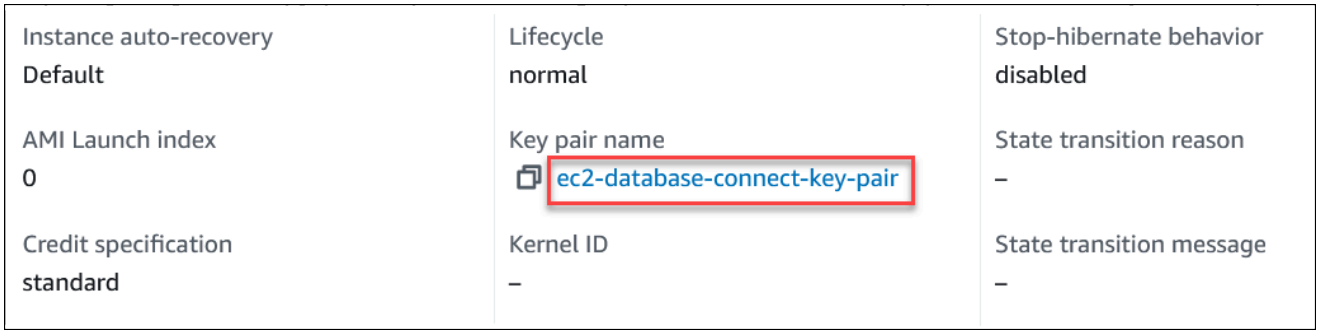
5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant d' EC2 instance pour ouvrir la liste des EC2 instances, puis sélectionnez votre EC2 instance.
7. Dans l'onglet Détails, notez les valeurs suivantes. Vous en aurez besoin lorsque vous vous connecterez via SSH :
   a. Dans Résumé de l'instance, notez la valeur du IPv4 DNS public.



   b. Dans Détails de l'instance, notez la valeur pour Nom de la paire de clés.



8. Attendez que l'état Instance de votre EC2 instance soit défini sur En cours d'exécution avant de continuer.

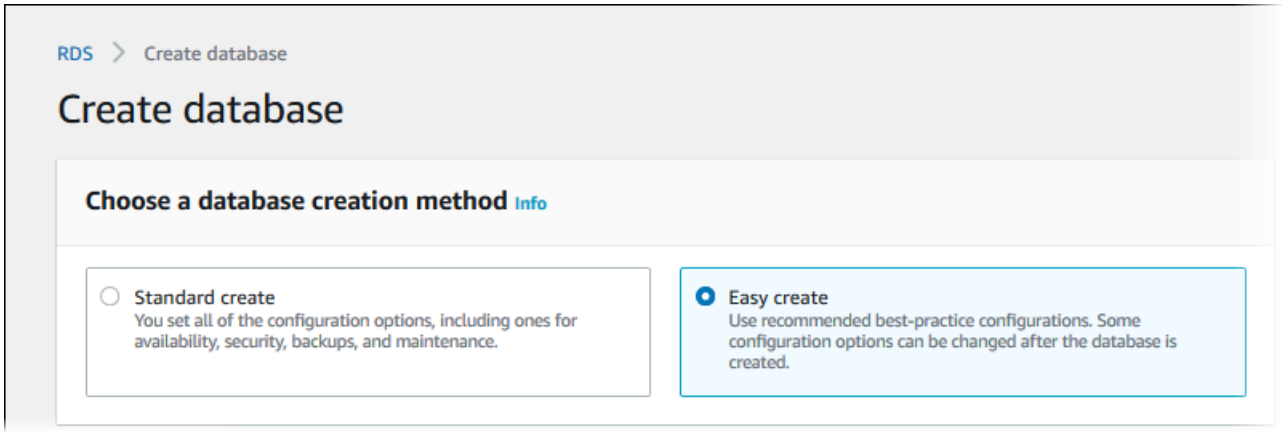
Création d'une instance de base de données PostgreSQL

La fondation de base d'Amazon RDS est l'instance de base de données. Il s'agit de l'environnement dans lequel vous exécutez vos bases de données PostgreSQL.

Dans cet exemple, vous utilisez Création facile pour créer une instance de base de données exécutant le moteur de base de données PostgreSQL avec une classe d'instance de base de données db.t3.micro.

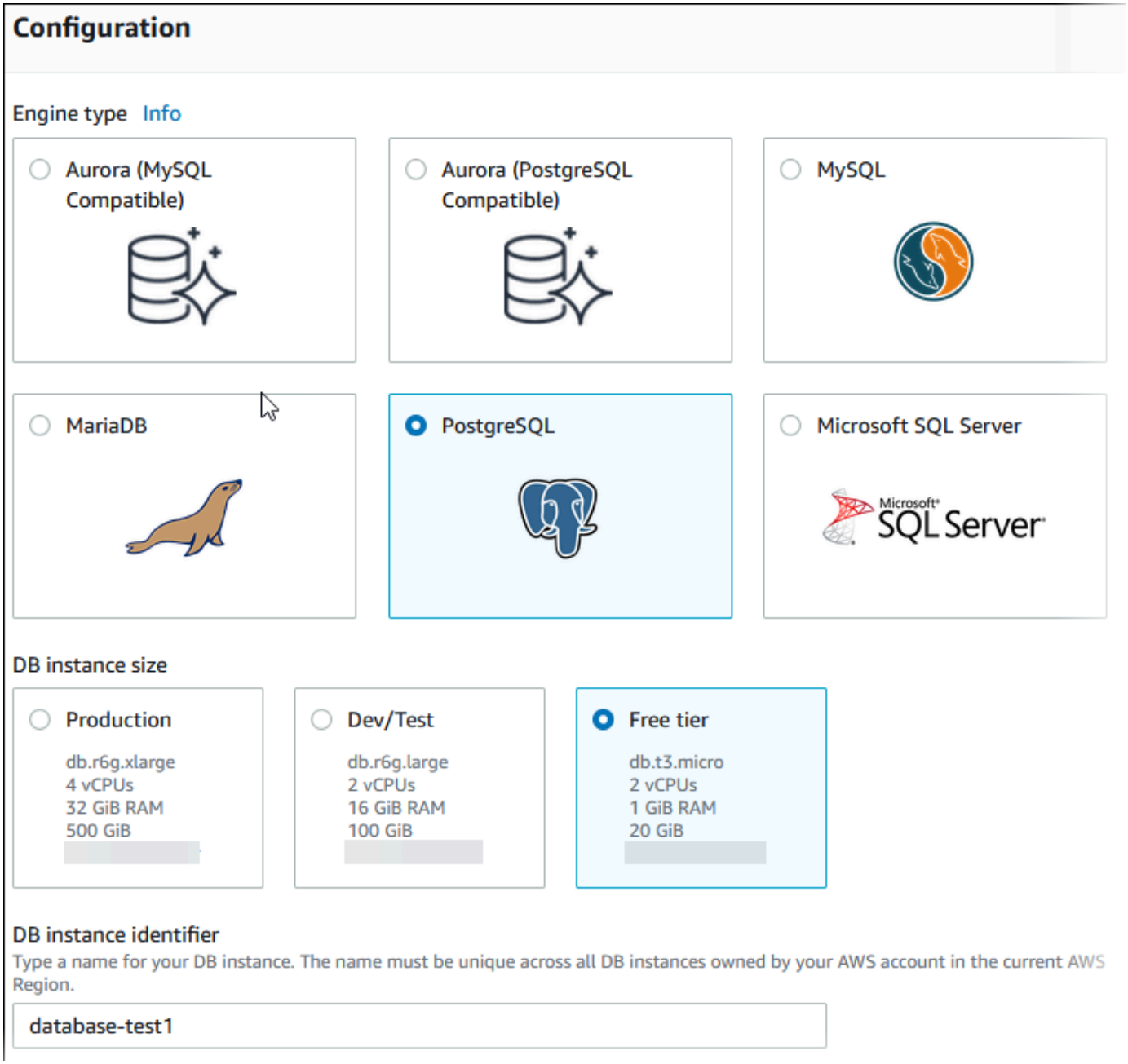
Pour créer une instance de base de données PostgreSQL avec l'option Easy create (Création facile)

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la AWS région dans laquelle vous souhaitez créer l'instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données) et veillez à choisir Easy create (Création facile).



5. Dans Configuration, choisissez PostgreSQL.
6. Pour DB instance size (Taille de l'instance de base de données), choisissez Free tier (Offre gratuite). Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.
7. Pour l'identifiant de l'instance DB, saisissez **`database-test1`**.
8. Pour Identifiant principal, saisissez un nom pour l'utilisateur principal ou conservez le nom par défaut (**`postgres`**).

   La page Create database (Créer une base de données) doit ressembler à l'image suivante. Pour les comptes de forfait gratuit, le niveau gratuit apparaît. Pour les comptes payants, Sandbox apparaît.

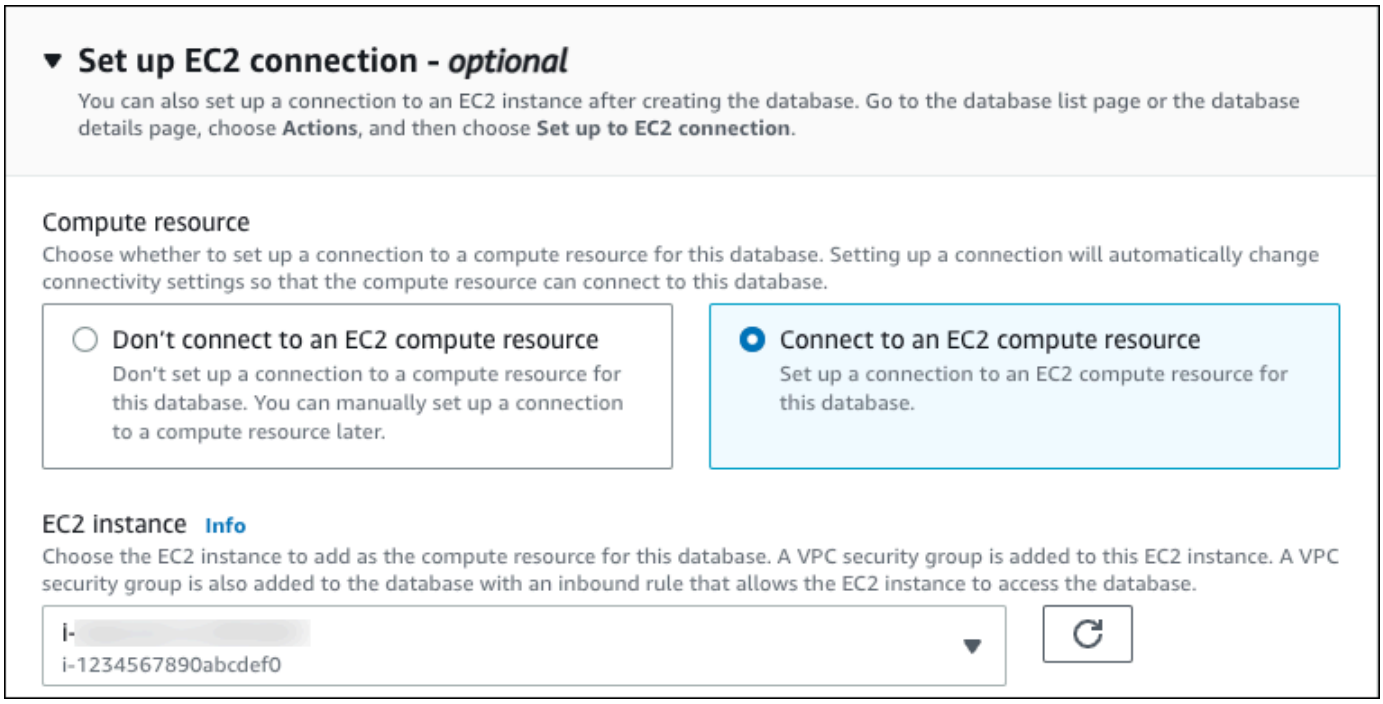


9. Pour utiliser un mot de passe principal généré automatiquement pour l'instance de base de données, sélectionnez Générer automatiquement un mot de passe.

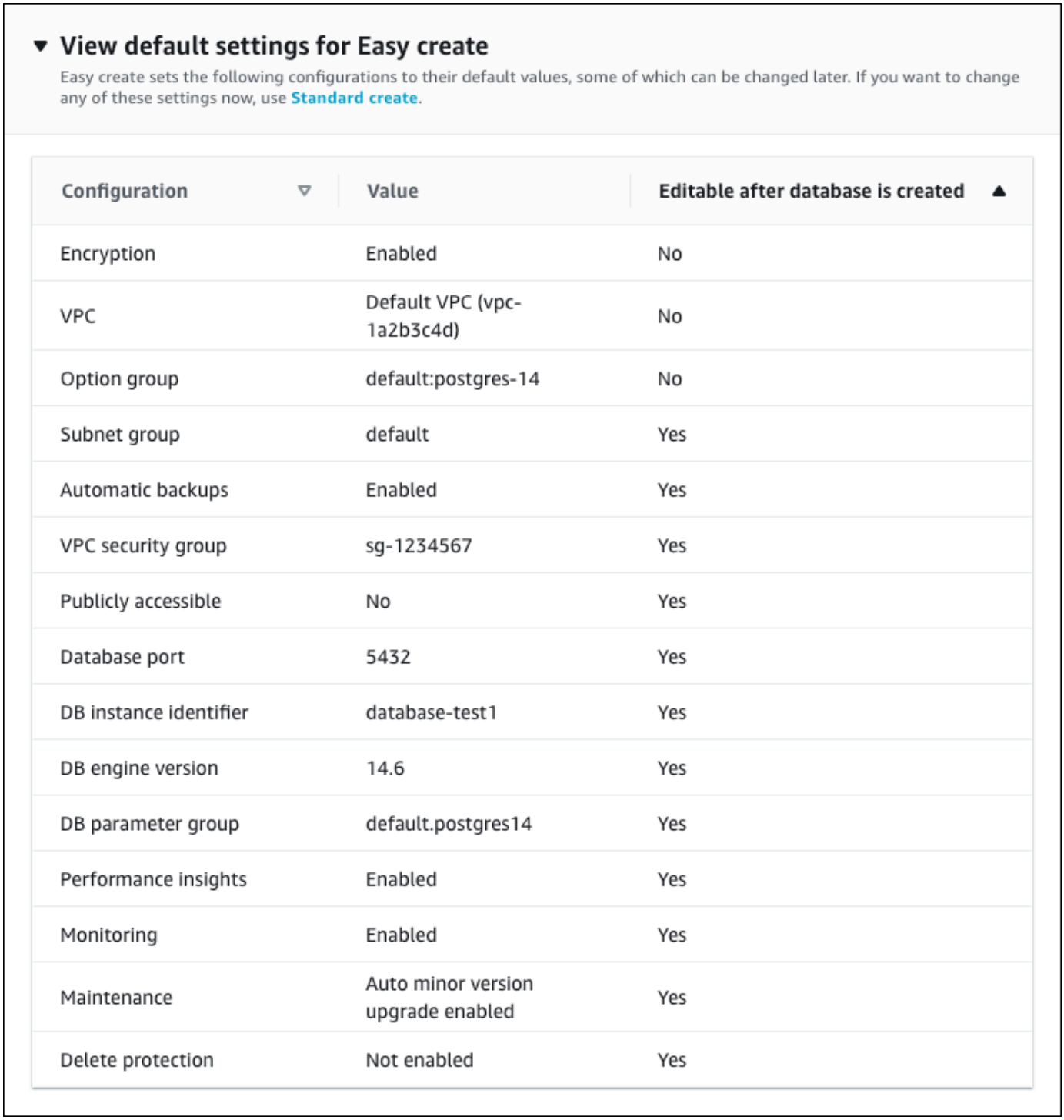
   Pour entrer votre mot de passe principal, veillez à ce que la case Générer automatiquement un mot de passe soit décochée, puis saisissez le même mot de passe dans Mot de passe principal et Confirmer le mot de passe.

10. Pour configurer une connexion avec l' EC2 instance que vous avez créée précédemment, ouvrez Configurer la EC2 connexion - facultatif.

Sélectionnez Se connecter à une ressource EC2 de calcul. Choisissez l' EC2instance que vous avez créée précédemment.



11. Ouvrez Afficher les paramètres par défaut pour Création facile.



View default settings for Easy create

Easy create sets the following configurations to their default values, some of which can be changed later. If you want to change any of these settings now, use **Standard create**.

Configuration	Value	Editable after database is created
Encryption	Enabled	No
VPC	Default VPC (vpc-1a2b3c4d)	No
Option group	default:postgres-14	No
Subnet group	default	Yes
Automatic backups	Enabled	Yes
VPC security group	sg-1234567	Yes
Publicly accessible	No	Yes
Database port	5432	Yes
DB instance identifier	database-test1	Yes
DB engine version	14.6	Yes
DB parameter group	default.postgres14	Yes
Performance insights	Enabled	Yes
Monitoring	Enabled	Yes
Maintenance	Auto minor version upgrade enabled	Yes
Delete protection	Not enabled	Yes

Vous pouvez examiner les paramètres par défaut utilisés quand l'option Easy create (Création facile) est activée. La colonne Modifiable après la création de la base de données indique les options que vous pouvez modifier après avoir créé la base de données.

- Si un réglage contient Non dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données.
- Si un réglage contient Oui dans cette colonne et que vous souhaitez un réglage différent, vous pouvez utiliser Création standard pour créer l'instance de base de données ou vous pouvez modifier l'instance de base de données après l'avoir créée pour modifier le réglage.

12. Choisissez Créer une base de données.

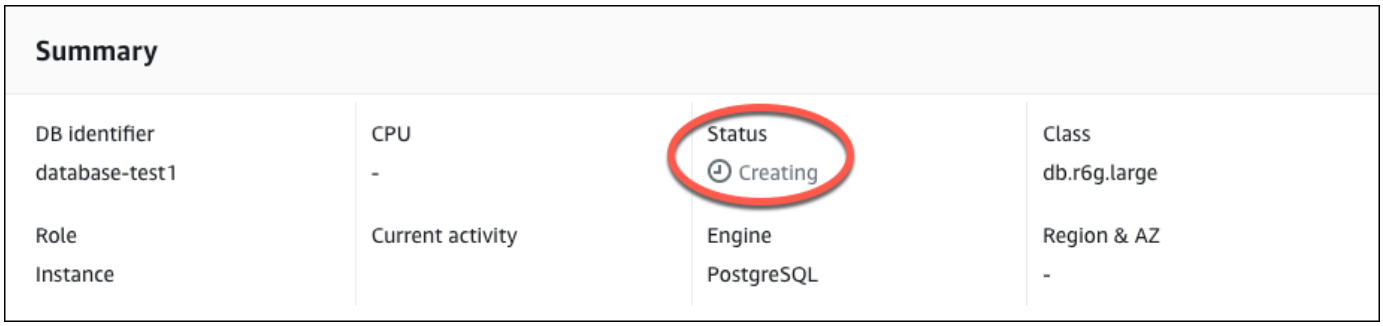
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Vous pouvez utiliser l'identifiant et le mot de passe affichés pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier.
    > Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, vous pouvez le faire en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

13. Dans la liste Bases de données, choisissez le nom de la nouvelle instance de base de données PostgreSQL pour afficher ses détails.

    L'instance de base de données a le statut Création en cours jusqu'à ce qu'elle soit prête à l'emploi.



Lorsque l’état passe à Available (Disponible), vous pouvez vous connecter à l’instance de base de données. En fonction de la quantité de stockage et de la classe d’instance de base de données, la mise à disposition de la nouvelle instance peut prendre jusqu’à 20 minutes.

(Facultatif) Créez un VPC, une instance et une EC2 instance PostgreSQL à l'aide de AWS CloudFormation

Au lieu d'utiliser la console pour créer votre VPC, votre instance et votre EC2 instance PostgreSQL, vous pouvez les utiliser AWS CloudFormation pour provisionner des AWS ressources en traitant l'infrastructure comme du code. Pour vous aider à organiser vos AWS ressources en unités plus petites et plus faciles à gérer, vous pouvez utiliser la fonctionnalité de pile AWS CloudFormation imbriquée. Pour plus d'informations, consultez les sections Création d'une pile sur la AWS CloudFormation console et Utilisation de piles imbriquées.

> **Important**
>
> AWS CloudFormation est gratuit, mais les ressources qui en CloudFormation découlent sont vivantes. Vous devez payer les frais d'utilisation standard pour ces ressources jusqu'à ce que vous y mettiez fin. Pour plus d'informations, consultez la section Tarification de RDS pour PostgreSQL.

Pour créer vos ressources à l'aide de la AWS CloudFormation console, procédez comme suit :

- Téléchargez le CloudFormation modèle
- Configurez vos ressources à l'aide de CloudFormation

Téléchargez le CloudFormation modèle

Un CloudFormation modèle est un fichier texte JSON ou YAML qui contient les informations de configuration relatives aux ressources que vous souhaitez créer dans la pile. Ce modèle crée également un VPC et un hôte bastion pour vous, ainsi que l'instance RDS.

Pour télécharger le fichier modèle, ouvrez le lien suivant, Modèle CloudFormation PostgreSQL.

Sur la page Github, cliquez sur le bouton Télécharger le fichier brut pour enregistrer le modèle de fichier YAML.

Configurez vos ressources à l'aide de CloudFormation

> **Note**
>
> Avant de commencer ce processus, assurez-vous que vous disposez d'une paire de clés pour une EC2 instance dans votre Compte AWS. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Lorsque vous utilisez le AWS CloudFormation modèle, vous devez sélectionner les paramètres appropriés pour vous assurer que vos ressources sont créées correctement. Procédez de la façon suivante :

1. Connectez-vous à la AWS CloudFormation console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudformation.
2. Sélectionnez Créer une pile.
3. Dans la section Spécifier le modèle, sélectionnez Télécharger un fichier modèle depuis votre ordinateur, puis cliquez sur Suivant.
4. Dans la page Spécifier les détails de la pile, définissez les paramètres suivants :
   a. Définissez le nom de la pile sur Postgre SQLTest Stack.
   b. Sous Paramètres, définissez les zones de disponibilité en sélectionnant trois zones de disponibilité.
   c. Dans Configuration de l'hôte Linux Bastion, pour Nom de la clé, sélectionnez une paire de clés pour vous connecter à votre EC2 instance.
   d. Dans les paramètres de configuration de l'hôte Linux Bastion, définissez la plage d'adresses IP autorisées sur votre adresse IP. Pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH), déterminez votre adresse IP publique à l'aide du service à l'adresse. https://checkip.amazonaws.com Exemple d'adresse IP : 192.0.2.1/32.

      > **Warning**
      >
      > Si vous utilisez `0.0.0.0/0` l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos EC2 instances publiques via SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les

environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques à accéder à vos EC2 instances via SSH.

e. Dans Configuration générale de la base de données, définissez la classe d'instance de base de données sur db.t3.micro.

f. Définissez le nom de base de données sur**`database-test1`**.

g. Dans Nom d'utilisateur principal de base de données, entrez le nom de l'utilisateur principal.

h. Définissez le mot de passe utilisateur principal de Manage DB avec Secrets Manager sur `false` pour ce didacticiel.

i. Pour le mot de passe de la base de données, définissez le mot de passe de votre choix. N'oubliez pas ce mot de passe pour suivre les étapes suivantes du didacticiel.

j. Sous Configuration du stockage de base de données, définissez le type de stockage de base de données sur gp2.

k. Dans Configuration de la surveillance des bases de données, définissez Enable RDS Performance Insights sur false.

l. Conservez tous les autres paramètres comme valeurs par défaut. Cliquez sur Suivant pour continuer.

5. Sur la page Configurer les options de pile, conservez toutes les options par défaut. Cliquez sur Suivant pour continuer.
6. Sur la page Review stack, sélectionnez Soumettre après avoir vérifié les options de la base de données et de l'hôte Linux Bastion.

Une fois le processus de création des piles terminé, visualisez les piles avec leurs noms BastionStacket leurs RDSNS pour noter les informations dont vous avez besoin pour vous connecter à la base de données. Pour plus d'informations, consultez la section Affichage des données et des ressources de la AWS CloudFormation pile sur le AWS Management Console.

Se connecter à une instance DB PostgreSQL

Vous pouvez vous connecter à l'instance de base de données en utilisant pgadmin ou psql. Cet exemple explique comment se connecter à une instance de base de données PostgreSQL à l'aide du client de ligne de commande psql.

Pour se connecter à une instance de base de données PostgreSQL avec psql

1. Trouvez le point de terminaison (nom DNS) et le numéro de port pour votre instance de base de données.

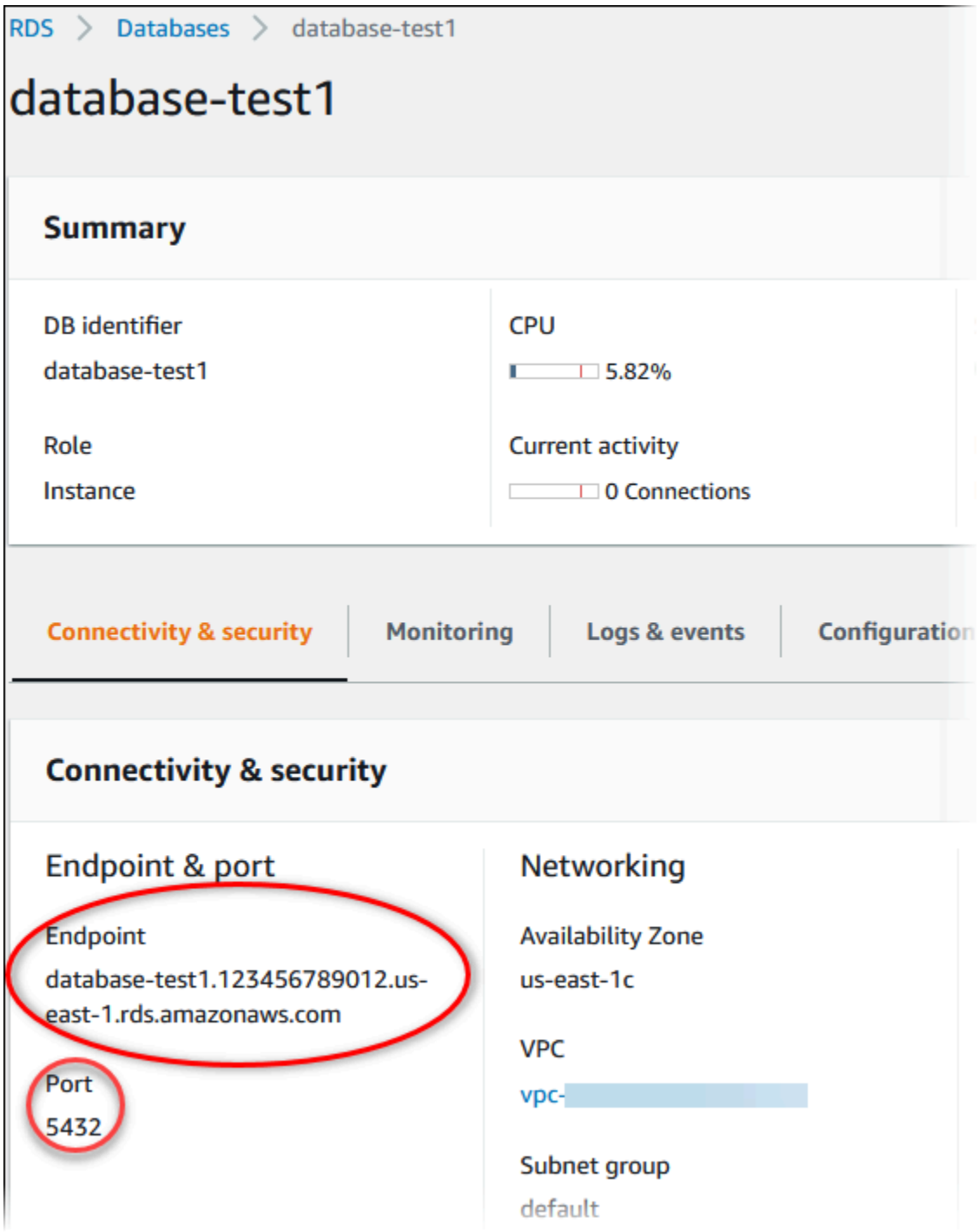
   a. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

   b. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS pour l'instance de base de données.

   c. Dans le panneau de navigation, choisissez Databases (Bases de données).

   d. Choisissez le nom de l'instance de base de données PostgreSQL pour afficher ses détails.

   e. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



2. Connectez-vous à l' EC2 instance que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Linux dans le guide de EC2 l'utilisateur Amazon.

   Nous vous recommandons de vous connecter à votre EC2 instance via SSH. Si l'utilitaire client SSH est installé sur Windows, Linux ou Mac, vous pouvez vous connecter à l'instance à l'aide du format de commande suivant :

```
ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
```

Supposons, par exemple, qu'`ec2-database-connect-key-pair.pem`il soit stocké `/dir1` sous Linux et que le IPv4 DNS public de votre EC2 instance l'est`ec2-12-345-678-90.compute-1.amazonaws.com`. Votre commande SSH se présenterait comme suit :

```
ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-
user@ec2-12-345-678-90.compute-1.amazonaws.com
```

3. Obtenez les dernières corrections de bogues et mises à jour de sécurité en mettant à jour le logiciel de votre EC2 instance. Pour ce faire, exécutez la commande suivante.

> **Note**
>
> L'option `-y` installe les mises à jour sans demander de confirmation. Pour examiner les mises à jour avant de les installer, omettez cette option.

```
sudo dnf update -y
```

4. Pour installer le client de ligne de commande psql depuis PostgreSQL sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install postgresql15
```

5. Connectez-vous à une instance de base de données PostgreSQL. Par exemple, saisissez la commande suivante dans une invite de commande sur un ordinateur client. Cette action vous permet de vous connecter à l'instance de base de données PostgreSQL à l'aide du client psql.

   Remplacez le point de terminaison de l'instance de base de données (nom DNS) pour *`endpoint`*, remplacez le nom `--dbname` de la base de données à laquelle vous voulez vous connecter pour *`postgres`*, et remplacez le nom d'utilisateur principal que vous avez utilisé pour *`postgres`*. Indiquez le mot de passe principal que vous avez utilisé lorsque vous êtes invité à entrer un mot de passe.

```
psql --host=endpoint --port=5432 --dbname=postgres --username=postgres
```

Après avoir saisi le mot de passe pour l'utilisateur, le résultat suivant devrait normalement s'afficher :

```
psql (14.3, server 14.6)
SSL connection (protocol: TLSv1.2, cipher: ECDHE-RSA-AES256-GCM-SHA384, bits: 256,
 compression: off)
Type "help" for help.

postgres=>
```

Pour plus d'informations sur la connexion à une instance de base de données PostgreSQL, consultez Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL. Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Résolution des problèmes de connexion à votre instance RDS for PostgreSQL.

Pour des raisons de sécurité, une bonne pratique consiste à recommander d'utiliser des connexions chiffrées. N'utilisez une connexion PostgreSQL non chiffrée que quand le client et le serveur sont dans le même VPC et que le réseau est approuvé. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Connexion à une instance de base de données PostgreSQL via SSL.

6. Exécutez des commandes SQL.

   Par exemple, la commande SQL suivante indique la date et l'heure actuelles :

   ```
   SELECT CURRENT_TIMESTAMP;
   ```

Supprimer l' EC2instance et l'instance de base de données

Après vous être connecté et exploré l'exemple d' EC2 instance et l'instance de base de données que vous avez créées, supprimez-les afin qu'ils ne vous soient plus facturés.

Si vous aviez AWS CloudFormation l'habitude de créer des ressources, ignorez cette étape et passez à l'étape suivante.

Pour supprimer l' EC2 instance

1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le panneau de navigation, choisissez Instances.
3. Sélectionnez l' EC2 instance, puis choisissez État de l'instance, Terminer l'instance.
4. Choisissez Résilier lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une EC2 instance, consultez Résilier votre instance dans le guide de EC2 l'utilisateur Amazon.

Pour supprimer une instance de base de données sans instantané de base de données final

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Décochez Créer un instantané final et Conserver les sauvegardes automatiques.
6. Terminez la confirmation et choisissez Supprimer.

(Facultatif) Supprimez l'EC2 instance et l'instance de base de données créées avec CloudFormation

Si vous aviez l'habitude AWS CloudFormation de créer des ressources, supprimez la CloudFormation pile une fois que vous vous êtes connecté à l'instance d'exemple et à l' EC2 instance de base de données et que vous les avez explorées, afin de ne plus être facturées.

Pour supprimer les CloudFormation ressources

1. Ouvrez la AWS CloudFormation console.
2. Sur la page Stacks du CloudFormationconsole, sélectionnez la pile racine (la pile sans nom VPCStack, BastionStack ou RDSNS).
3. Sélectionnez Delete (Supprimer).
4. Sélectionnez Supprimer la pile lorsque vous êtes invité à confirmer.

Pour plus d'informations sur la suppression d'une pile dans CloudFormation, voir Supprimer une pile sur la AWS CloudFormation console dans le Guide de AWS CloudFormation l'utilisateur.

(Facultatif) Connecter votre instance de base de données à une fonction Lambda

Vous pouvez également connecter votre instance de base de données RDS for PostgreSQL à une ressource de calcul sans serveur Lambda. Les fonctions Lambda vous permettent d'exécuter du code sans provisionner ni gérer l'infrastructure. Une fonction Lambda vous permet également de répondre automatiquement aux demandes d'exécution de code à n'importe quelle échelle, d'une douzaine d'événements par jour à des centaines par seconde. Pour de plus amples informations, veuillez consulter Connexion automatique d'une fonction Lambda et d'une instance de base de données.

Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS

Ce didacticiel vous montre comment installer un serveur web Apache avec PHP et créer une base de données MariaDB, MySQL ou PostgreSQL. Le serveur web s'exécute sur une instance Amazon EC2 utilisant Amazon Linux 2023 et vous pouvez choisir entre une instance de base de données MySQL ou PostgreSQL. L'instance Amazon EC2 et l'instance de base de données s'exécutent tous deux dans un Virtual Private Cloud (VPC) basé sur le service Amazon VPC.

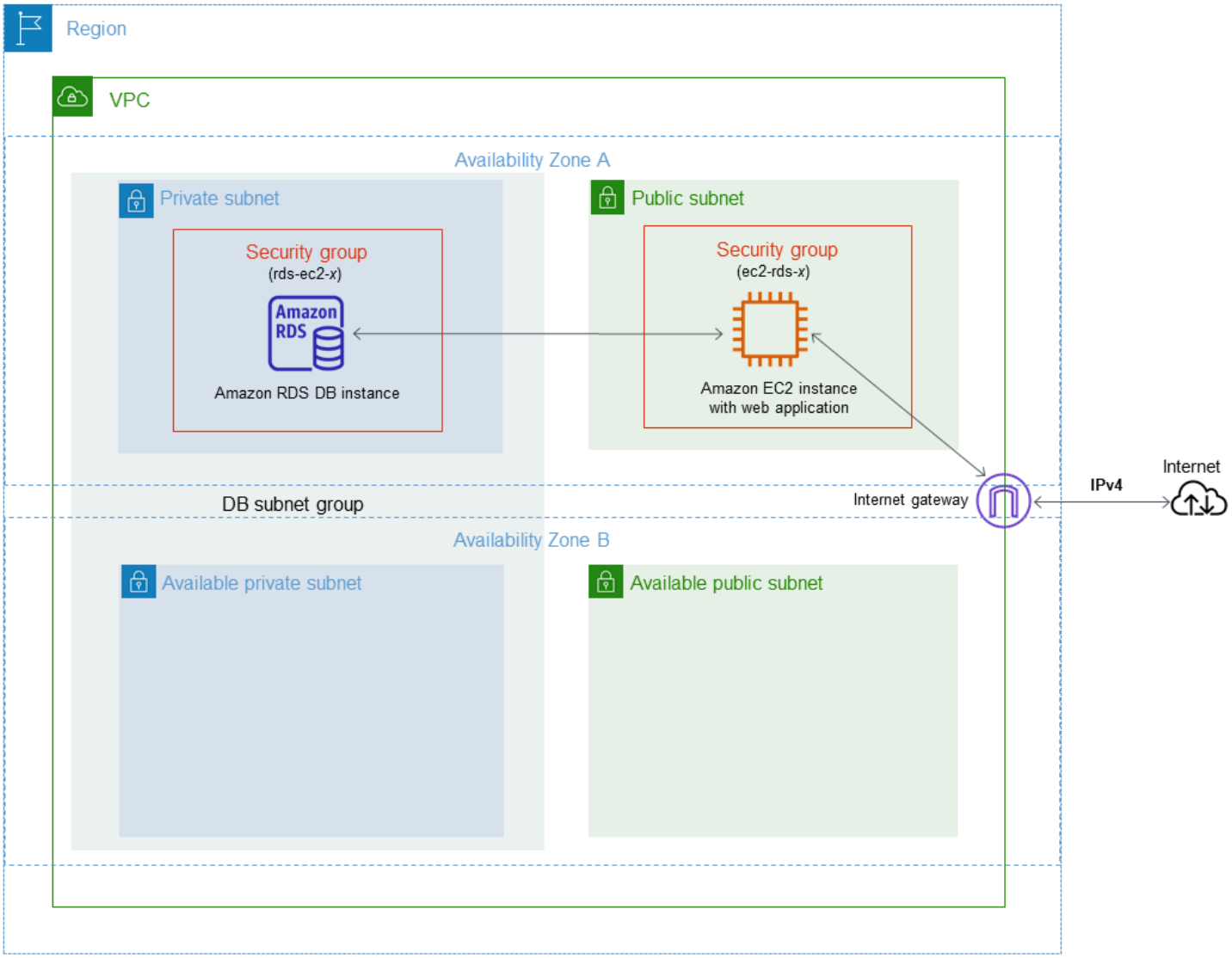
> **Important**
>
> Il n'y a pas de frais pour la création d'un compte AWS. Toutefois, au cours de ce didacticiel, des coûts peuvent être générés par l'utilisation des ressources AWS. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

> **Note**
>
> Ce didacticiel s'applique à Amazon Linux 2023 et peut ne pas fonctionner pour d'autres versions de Linux.

Dans le tutoriel qui suit, vous créez une instance EC2 qui utilise le VPC, les sous-réseaux et le groupe de sécurité par défaut pour votre Compte AWS. Ce tutoriel vous montre comment créer l'instance de base de données et configurer automatiquement la connectivité avec l'instance EC2 que vous avez créée. Le tutoriel vous montre ensuite comment installer le serveur web sur l'instance EC2. Vous connectez votre serveur Web à votre instance de base de données dans le VPC en utilisant le point de terminaison de l'instance de la base de données.

1. Lancez une EC2 instance pour vous connecter à votre instance de base de données de de bases de données
2. Créer une instance de base de données Amazon RDS
3. Installer un serveur web sur votre instance EC2

Le diagramme suivant affiche la configuration obtenue au terme de ce didacticiel.



Note

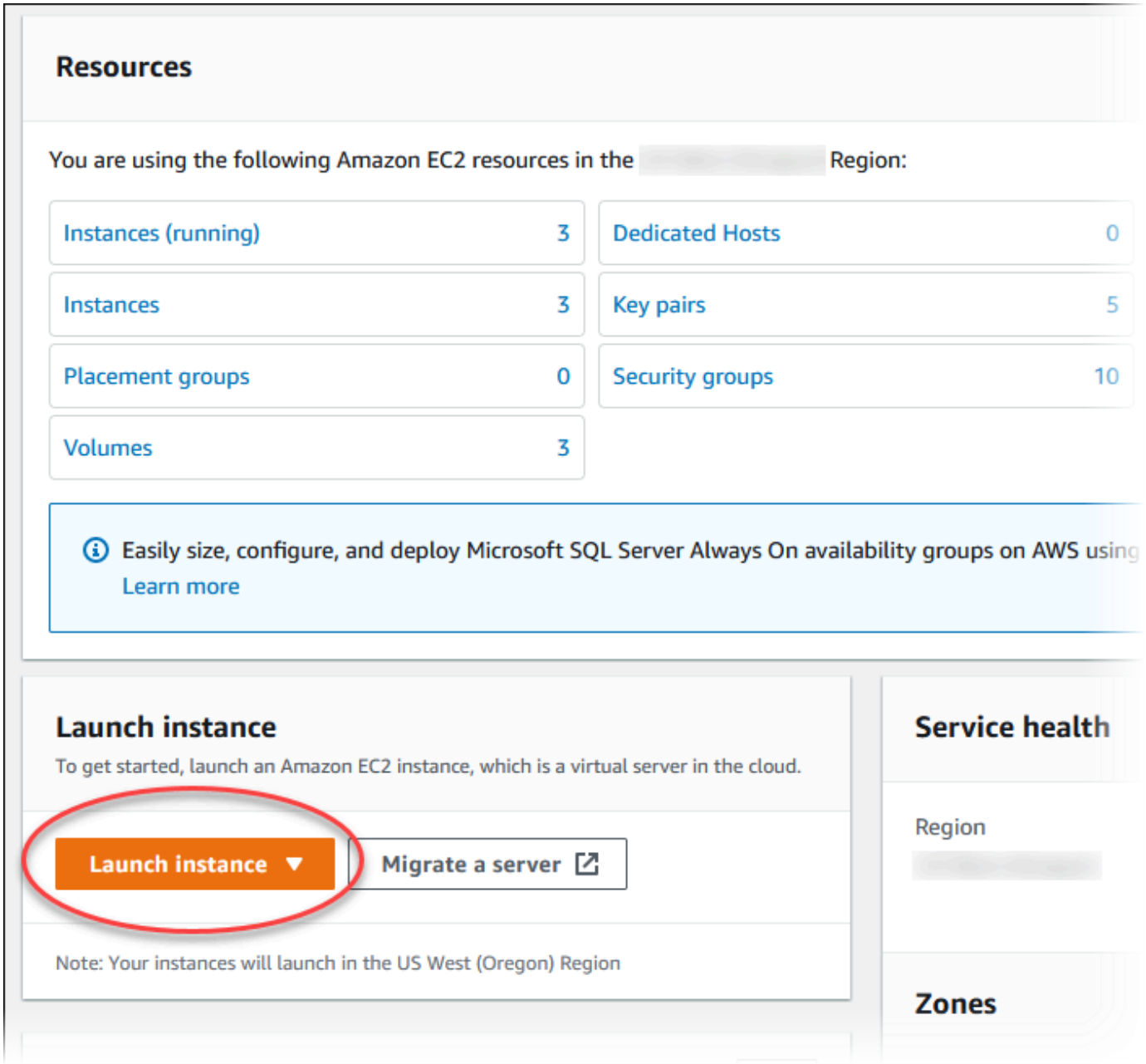
Une fois le tutoriel terminé, chaque zone de disponibilité de votre VPC comporte un sous-réseau public et un sous-réseau privé. Ce tutoriel utilise le VPC par défaut pour votre Compte AWS et configure automatiquement la connectivité entre votre instance EC2 et l'instance de base de données. Si vous préférez plutôt configurer un nouveau VPC pour ce scénario, suivez les étapes décrites dans Tutoriel : Création d'un VPC à utiliser avec un) IPv4 .

Lancez une EC2 instance pour vous connecter à votre instance de base de données de de bases de données

Créez une EC2 instance Amazon dans le sous-réseau public de votreVPC.

Pour lancer une instance EC2

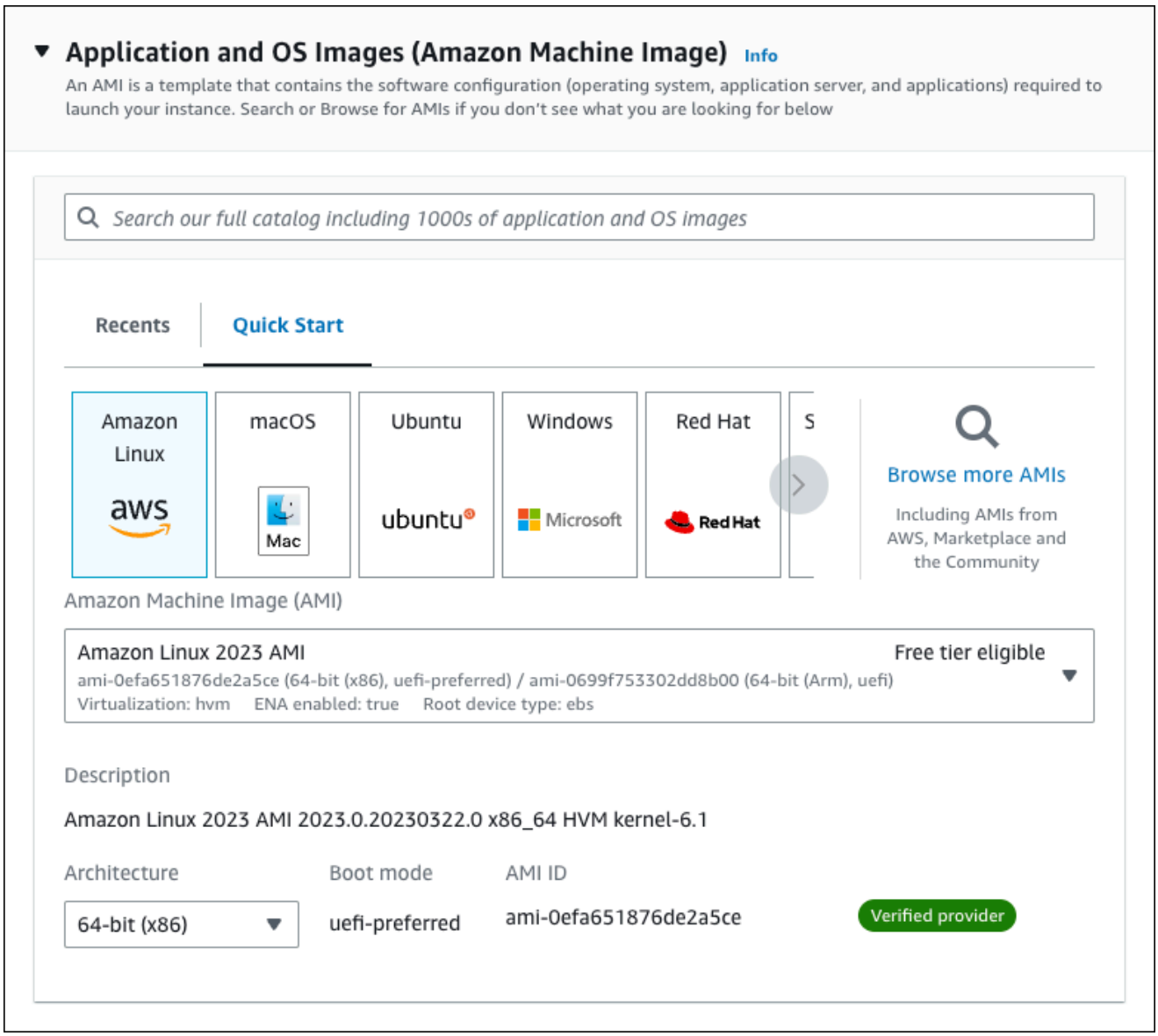
1. Connectez-vous à la EC2 console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/ec2/.
2. Dans le coin supérieur droit du AWS Management Console, choisissez l' Région AWS endroit où vous souhaitez créer l'EC2instance.
3. Choisissez EC2Dashboard, puis Launch instance, comme illustré ci-dessous.



4. Choisissez les paramètres suivants sur la page Lancer une instance.

   a. Sous Name and tags (Nom et identifications), pour Name (Nom), saisissez **`tutorial-ec2-instance-web-server`**.

   b. Sous Images d'applications et de systèmes d'exploitation (Amazon Machine Image), choisissez Amazon Linux, puis Amazon Linux 2023 AMI. Conservez les valeurs par défaut pour les autres choix.



c. Sous Instance type (Type d'instance), choisissez t2.micro.

d. Sous Key pair (login) [Paire de clés (connexion)], choisissez une valeur Key pair name (Nom de paire de clés) pour utiliser une paire de clés existante. Pour créer une nouvelle paire de clés pour l'EC2instance Amazon, choisissez Create new key pair, puis utilisez la fenêtre Create key Pair pour la créer.

   Pour plus d'informations sur la création d'une nouvelle paire de clés, consultez la section Créer une paire de clés dans le guide de EC2 l'utilisateur Amazon.

e. Sous Network settings (Paramètres réseau), définissez ces valeurs et conservez les autres valeurs par défaut :

- Pour Autoriser SSH le trafic depuis, choisissez la source des SSH connexions à l'EC2instance.

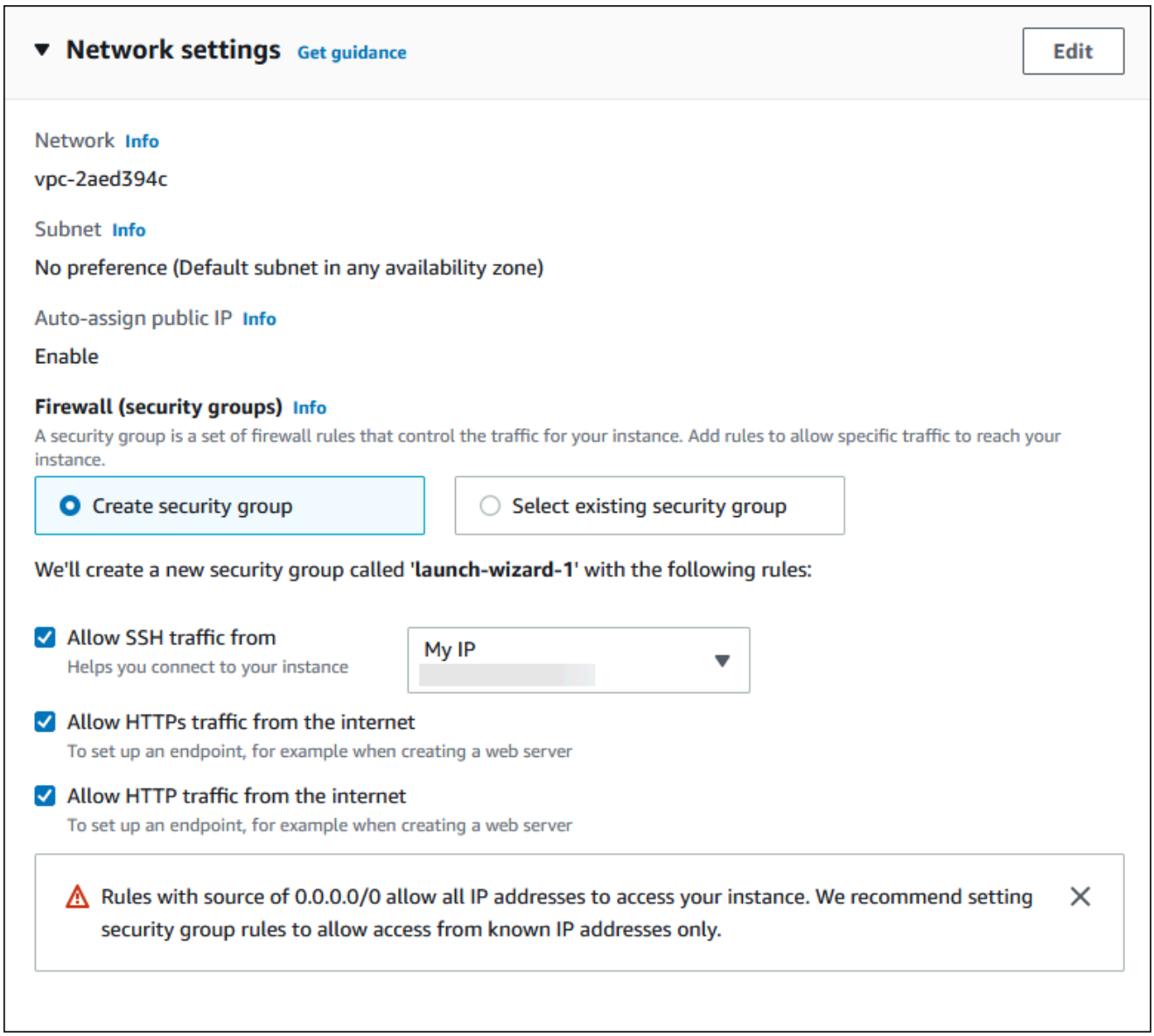
  Vous pouvez choisir Mon adresse IP si l'adresse IP affichée est correcte pour les SSH connexions.

  Sinon, vous pouvez déterminer l'adresse IP à utiliser pour vous connecter à EC2 des instances à VPC l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une fenêtre ou un onglet de navigateur différent, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : `203.0.113.25/32`.

  Dans de nombreux cas, vous pouvez vous connecter via un fournisseur de services Internet (ISP) ou derrière votre pare-feu sans adresse IP statique. Si tel est le cas, assurez-vous de déterminer la plage d'adresses IP utilisées par les ordinateurs clients.

  > **Warning**
  >
  > Si vous utilisez `0.0.0.0/0` for SSH access, vous permettez à toutes les adresses IP d'accéder à vos instances publiques en utilisantSSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. En production, autorisez uniquement une adresse IP ou une plage d'adresses spécifiques pour accéder à vos instances à l'aide deSSH.

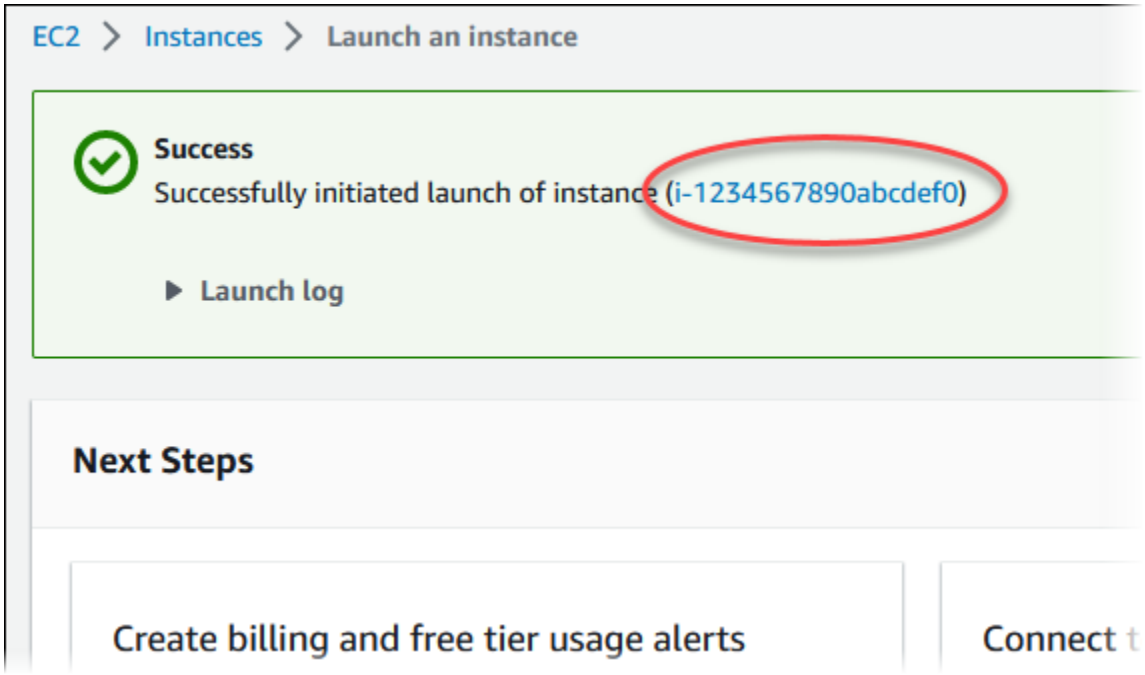
- Activez Autoriser HTTPs le trafic en provenance d'Internet.
- Activez Autoriser HTTP le trafic en provenance d'Internet.



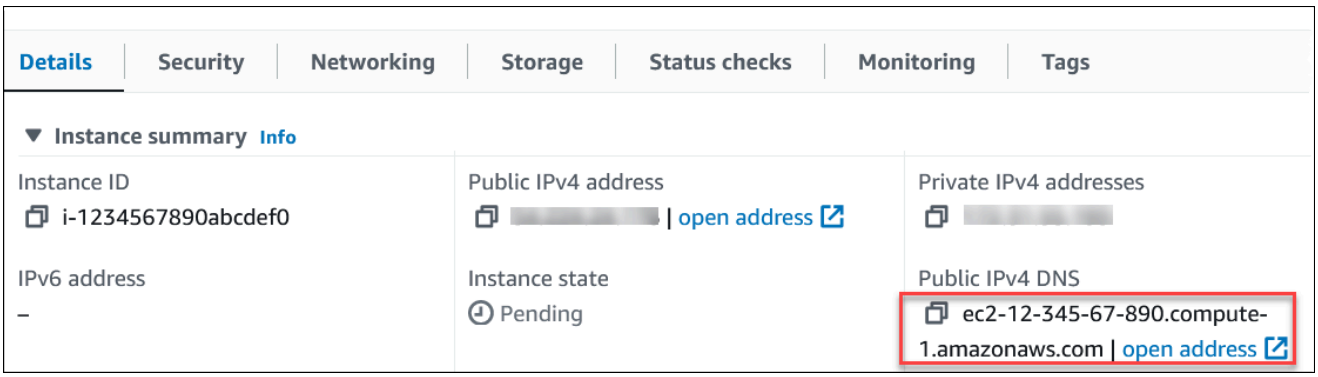
f. Laissez les valeurs par défaut pour les autres sections.

g. Consultez un résumé de la configuration de votre instance dans le panneau Summary (Récapitulatif) et, lorsque vous êtes prêt, choisissez Launch instance (Lancer l'instance).

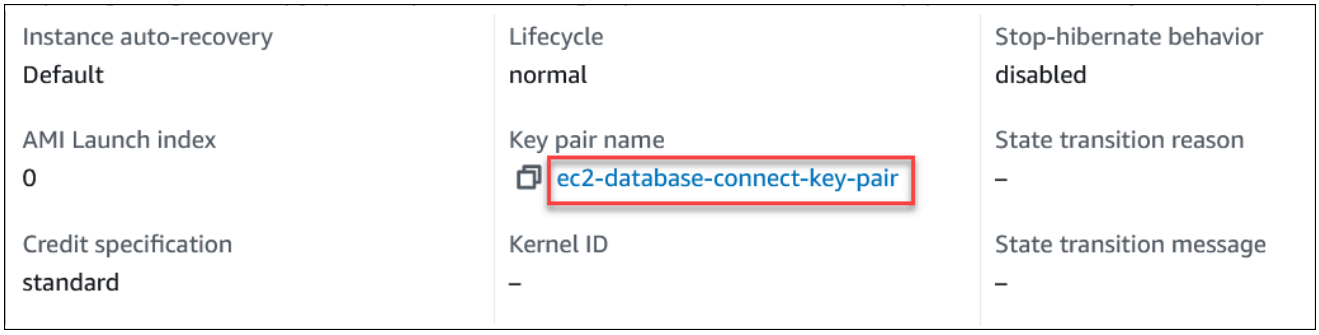
5. Sur la page État du lancement, notez l'identifiant de votre nouvelle EC2 instance, par exemple :`i-1234567890abcdef0`.



6. Choisissez l'identifiant d'EC2instance pour ouvrir la liste des EC2 instances, puis sélectionnez votre EC2 instance.
7. Dans l'onglet Détails, notez les valeurs suivantes, dont vous avez besoin lorsque vous vous connectez en utilisant SSH :
   a. Dans Résumé de l'instance, notez la valeur de Public IPv4 DNS.



   b. Dans Détails de l'instance, notez la valeur pour Nom de la paire de clés.



8. Attendez que la valeur Instance state (État de votre instance) soit Running (En cours d'exécution) avant de continuer.
9. Termin Créer une instance de base de données Amazon RDS.

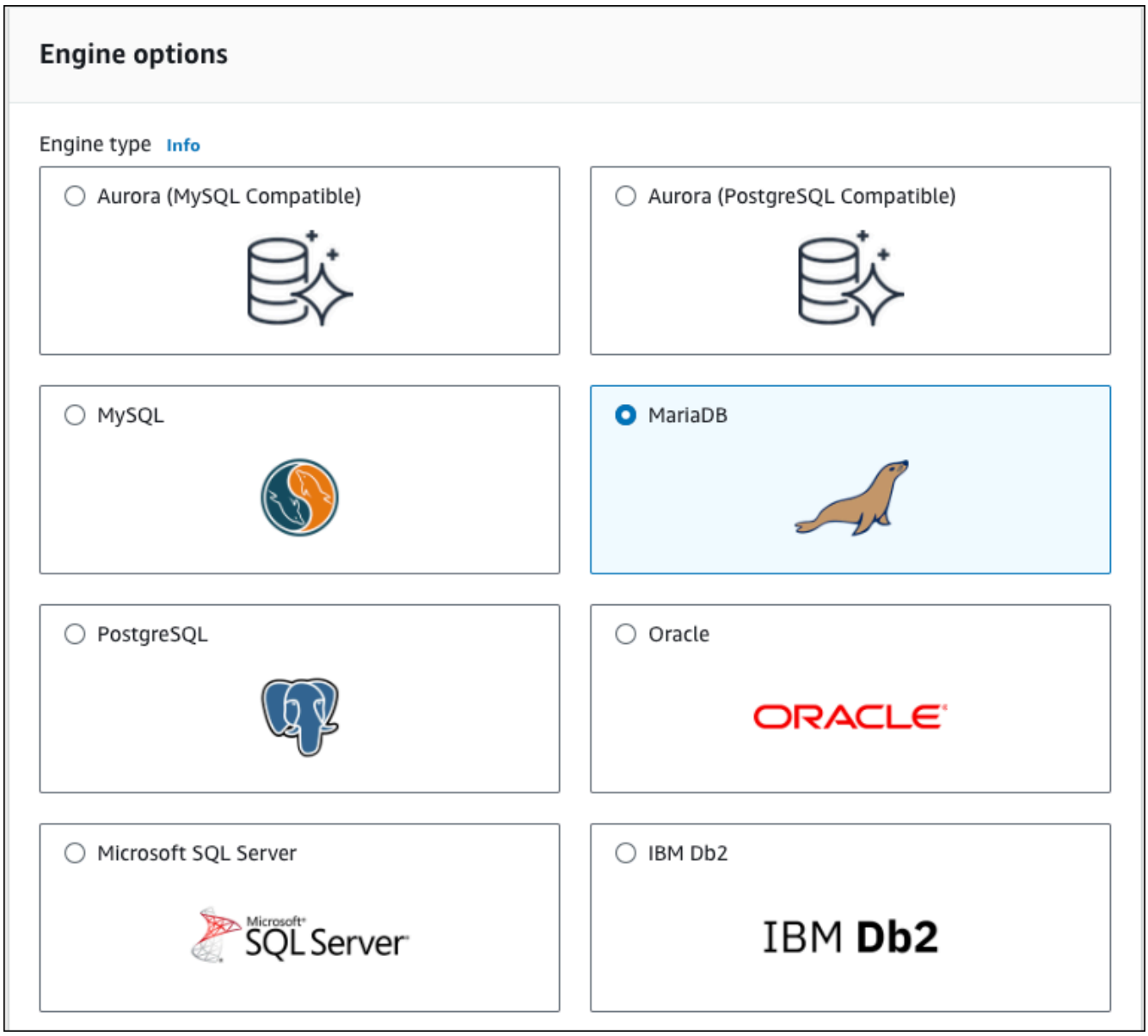
Créer une instance de base de données Amazon RDS

Créez une instance de base de données RDS for MariaDB, RDS for MySQL ou RDS for PostgreSQL qui conserve les données utilisées par une application web.

RDS for MariaDB

Pour créer une instance MariaDB

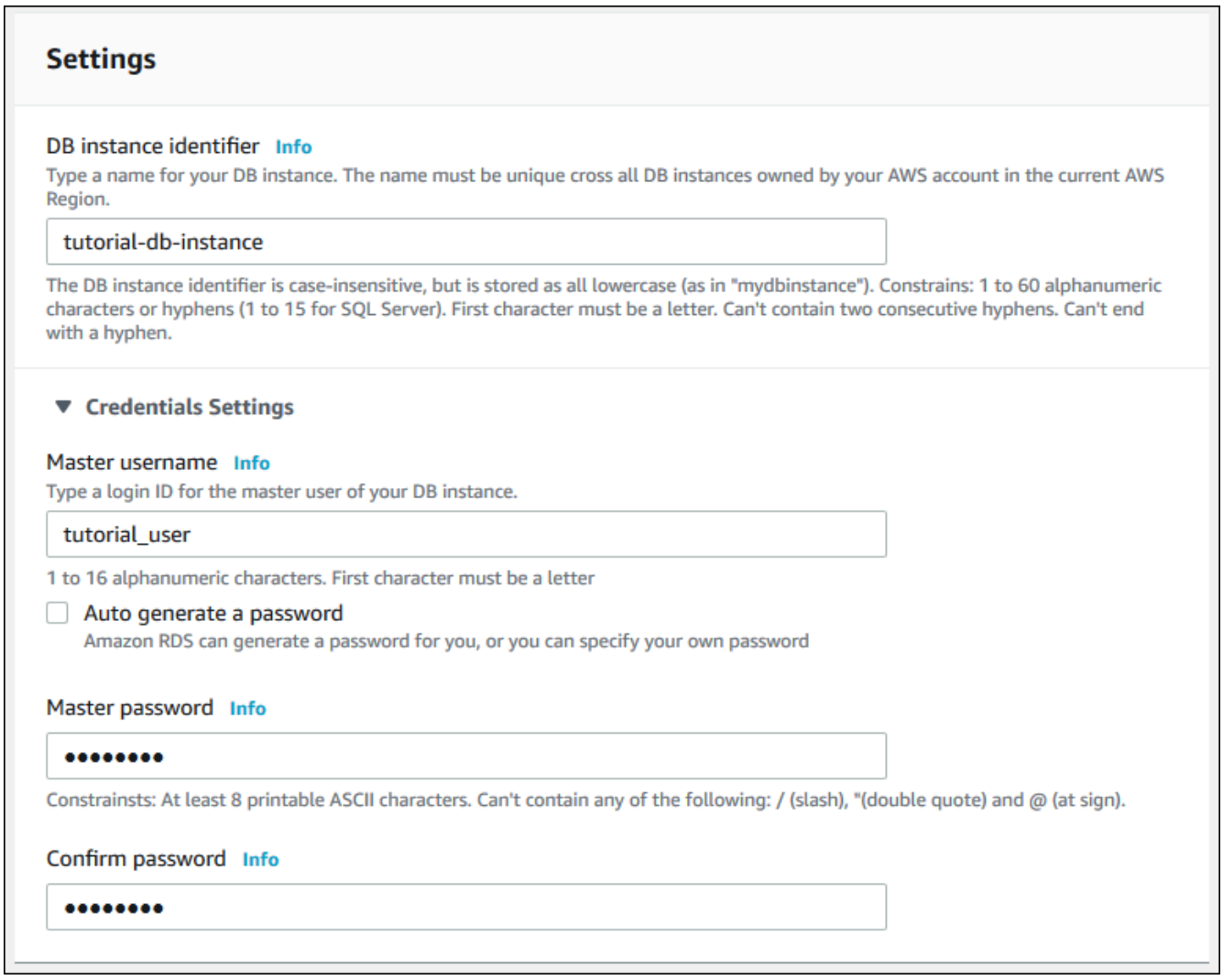
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit du AWS Management Console, cochez la case. Région AWS Il doit être le même que celui dans lequel vous avez créé votre EC2 instance.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).
5. Sur la page Créer une base de données, choisissez Création standard.
6. Dans Options de moteur, choisissez MariaDB.



7. Pour les modèles, choisissez le niveau gratuit ou Sandbox. Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.

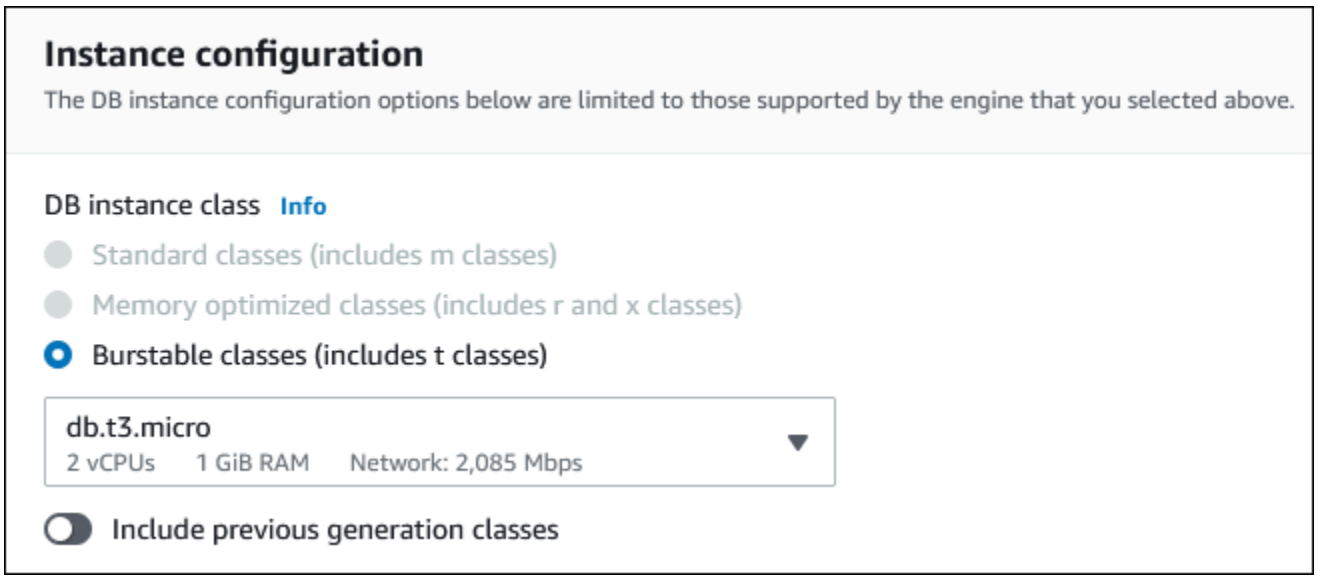


8. Dans la section Availability and durability (Disponibilité et durabilité), conservez les valeurs par défaut.
9. Dans la section Settings (Paramètres), définissez les valeurs suivantes :
   - DB Instance Identifier (Identifiant d'instance de base de données) : saisissez **`tutorial-db-instance`**.
   - Master username (Identifiant principal) : saisissez **`tutorial_user`**.
   - Auto generate a password (Génération automatique d'un mot de passe) : laissez cette option désactivée.
   - Master password (Mot de passe principal) : saisissez un mot de passe.
   - Confirm password (Confirmer le mot de passe) – Saisissez à nouveau le mot de passe.



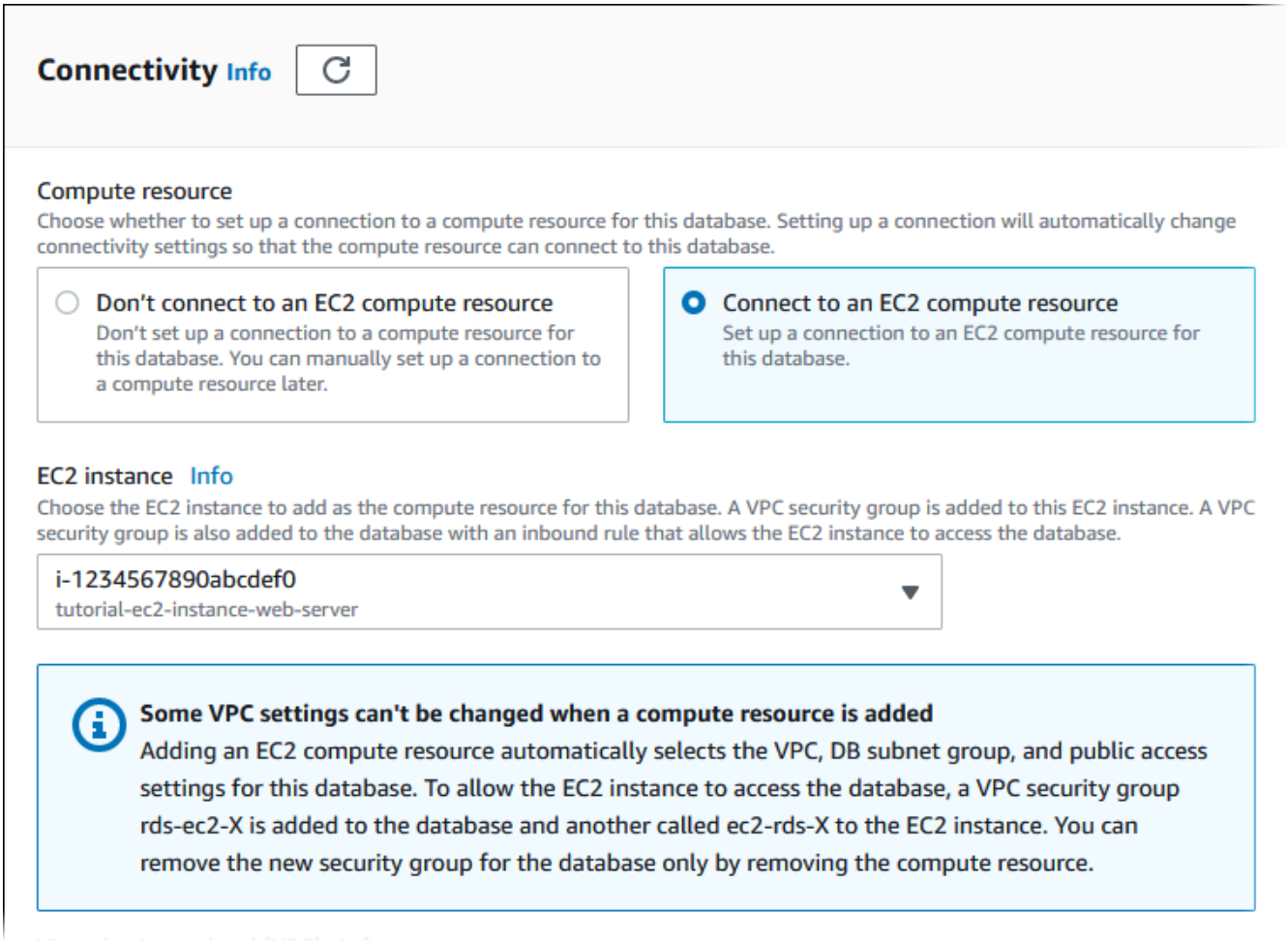
10. Dans la section Instance configuration (Configuration de l'instance), définissez les valeurs suivantes :

    - Classe à capacité extensible (inclut les classes t)
    - db.t3.micro



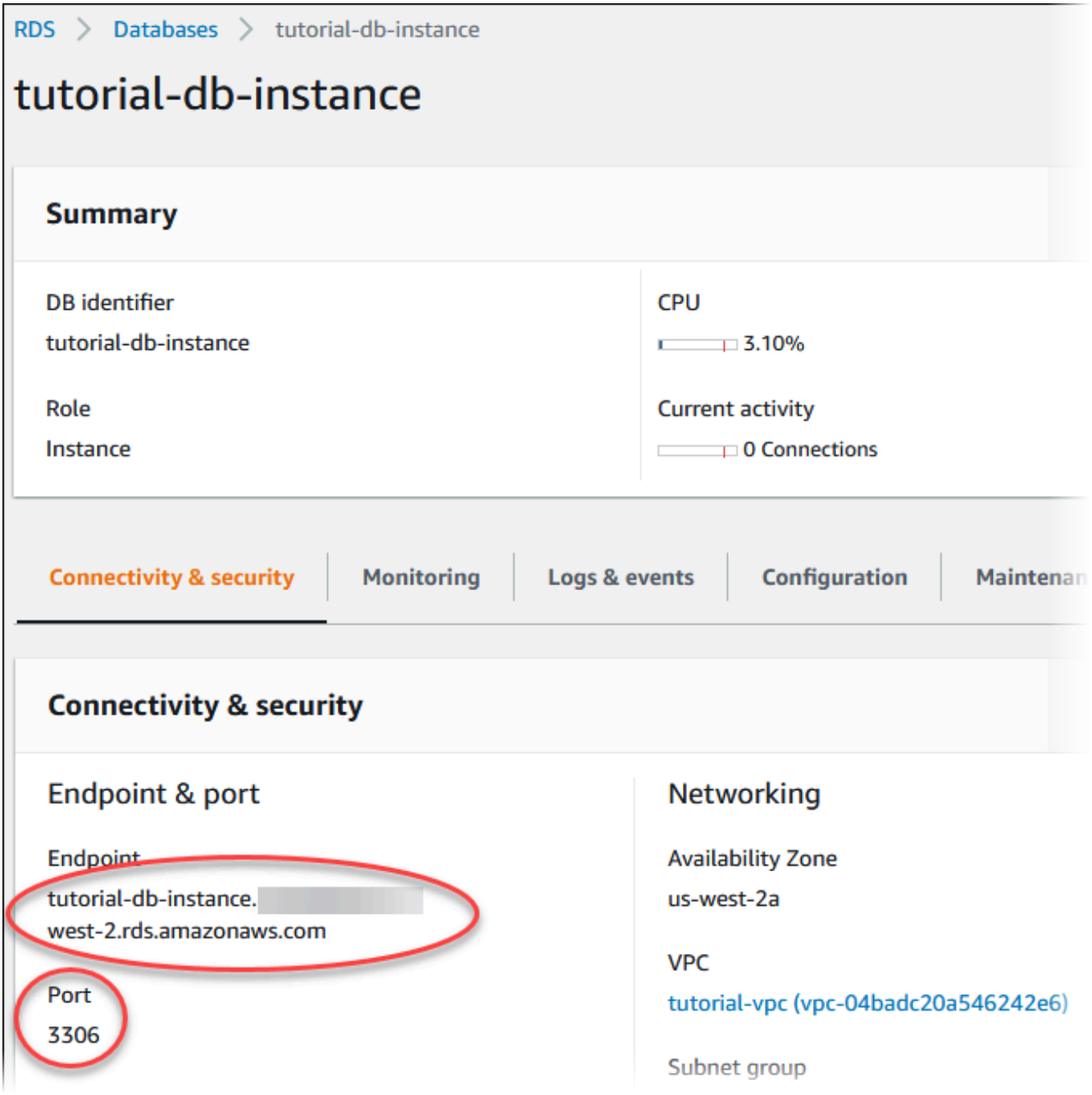
11. Dans la section Storage (Stockage), conservez les valeurs par défaut.
12. Dans la section Connectivity (Connectivité), définissez ces valeurs et conservez les autres valeurs par défaut :

    - Dans le champ Ressource de calcul, choisissez Connect to an EC2 Compute Resource.
    - Par EC2 exemple, choisissez l' EC2 instance que vous avez créée précédemment, telle que tutorial-ec2 -. instance-web-server



13. Dans la section Database authentication (Authentification de base de données, assurez-vous que l'option Password authentication(Authentification par mot de passe) est sélectionnée.
14. Ouvrez la section Additional configuration (Configuration supplémentaire), puis entrez **`sample`** pour Initial database name (Nom de la base de données initiale). Conservez les paramètres par défaut pour les autres options.
15. Pour créer votre instance MariaDB, choisissez Créer une base de données.

    Votre nouvelle instance de base de données apparaît dans la liste Databases (Bases de données) avec l'état Creating (Création en cours).
16. Attendez que le Status (État) de votre instance de base de données affiche Available (Disponible). Sélectionnez ensuite le nom de l'instance de base de données pour afficher les détails.
17. Dans la section Connectivity & security (Connectivité et sécurité), affichez le Endpoint (Point de terminaison) et le Port de l'instance de base de données.



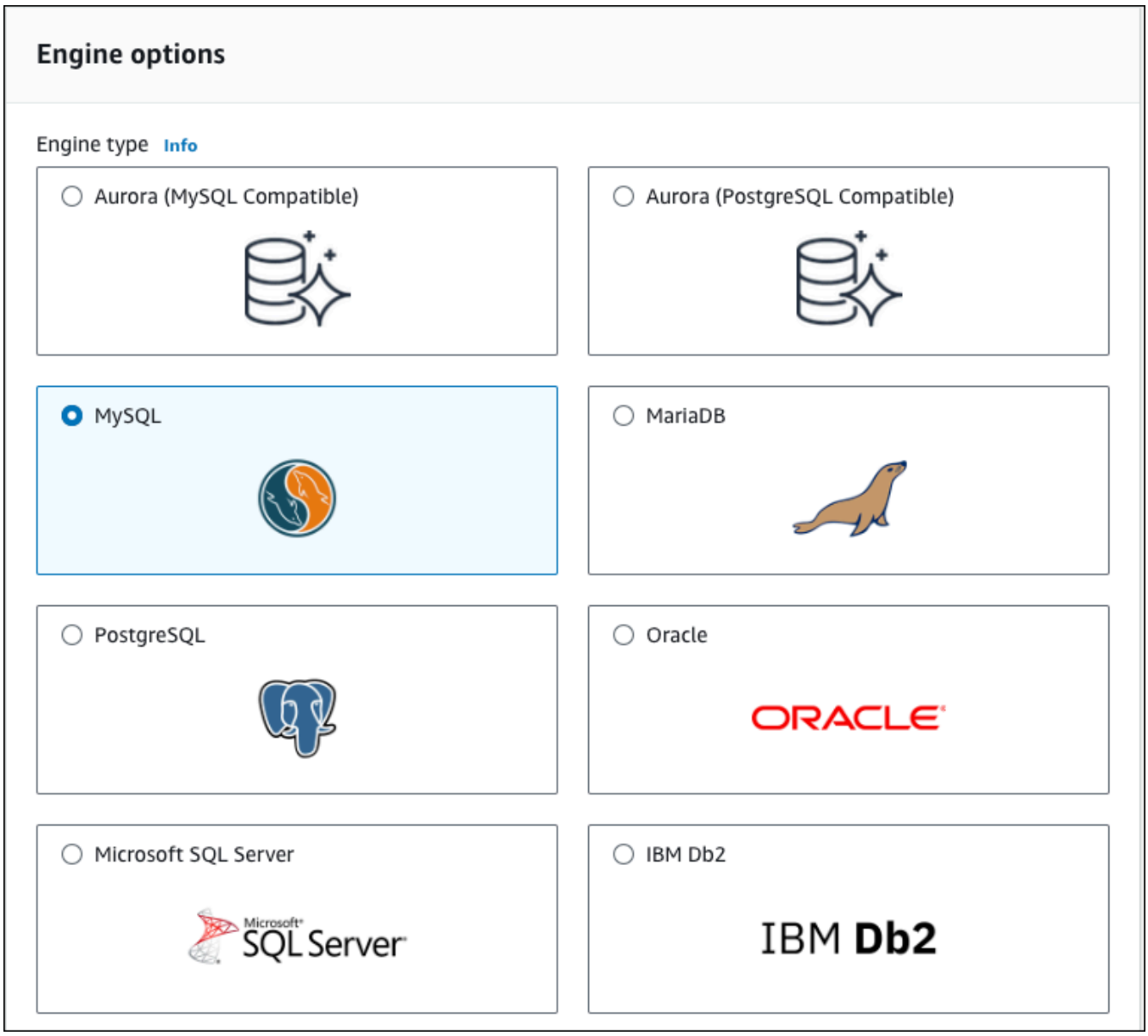
Notez le point de terminaison et le port de votre instance de base de données. Vous utilisez ces informations pour connecter votre serveur web à votre instance de base de données.

18. Termin Installer un serveur web sur votre instance EC2.

RDS for MySQL

Pour créer une instance de base de données MySQL

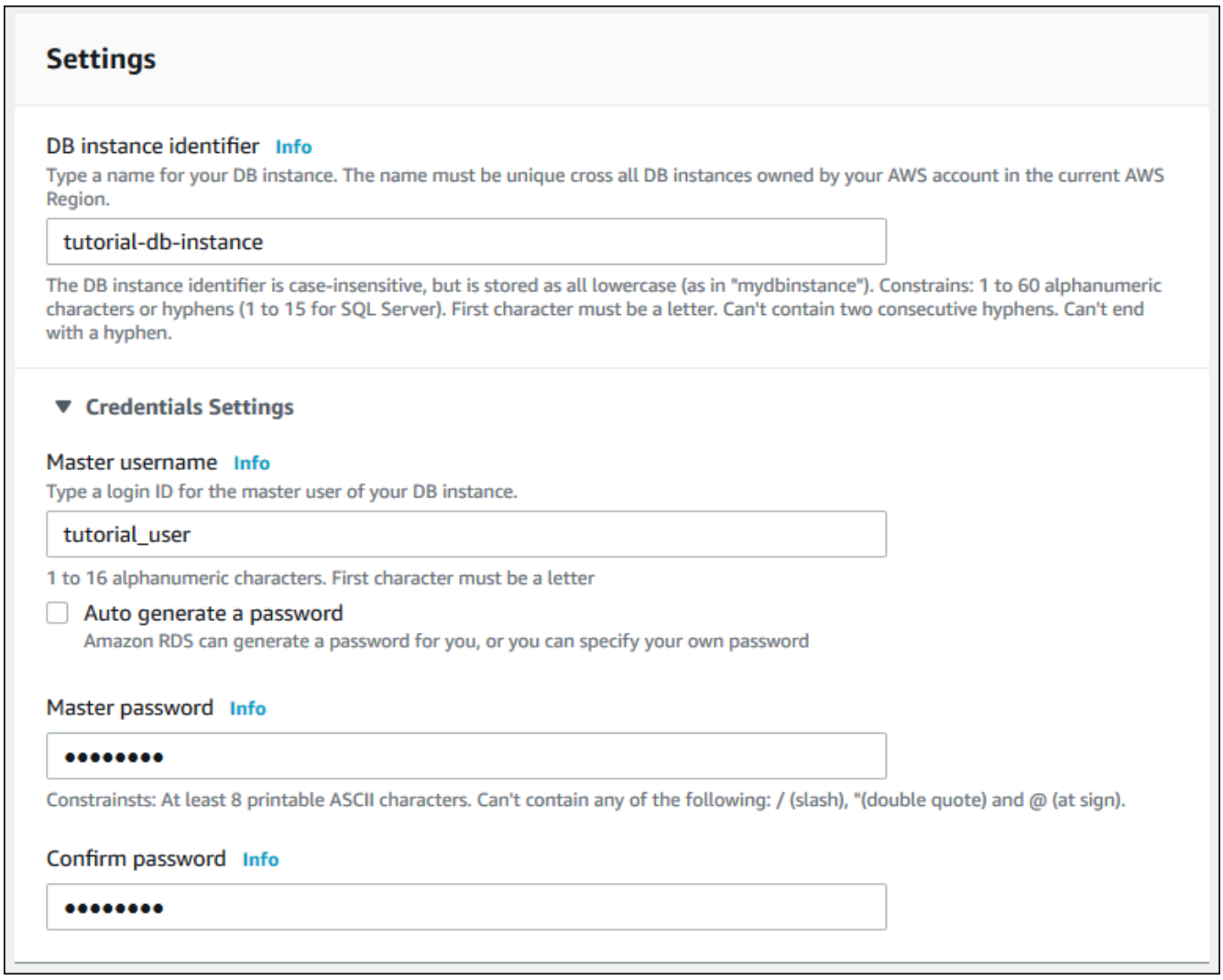
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit du AWS Management Console, cochez la case. Région AWS Il doit être le même que celui dans lequel vous avez créé votre EC2 instance.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).
5. Sur la page Créer une base de données, choisissez Création standard.
6. Pour Options de moteur, choisissez MySQL.



7. Pour les modèles, choisissez le niveau gratuit ou Sandbox. Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.

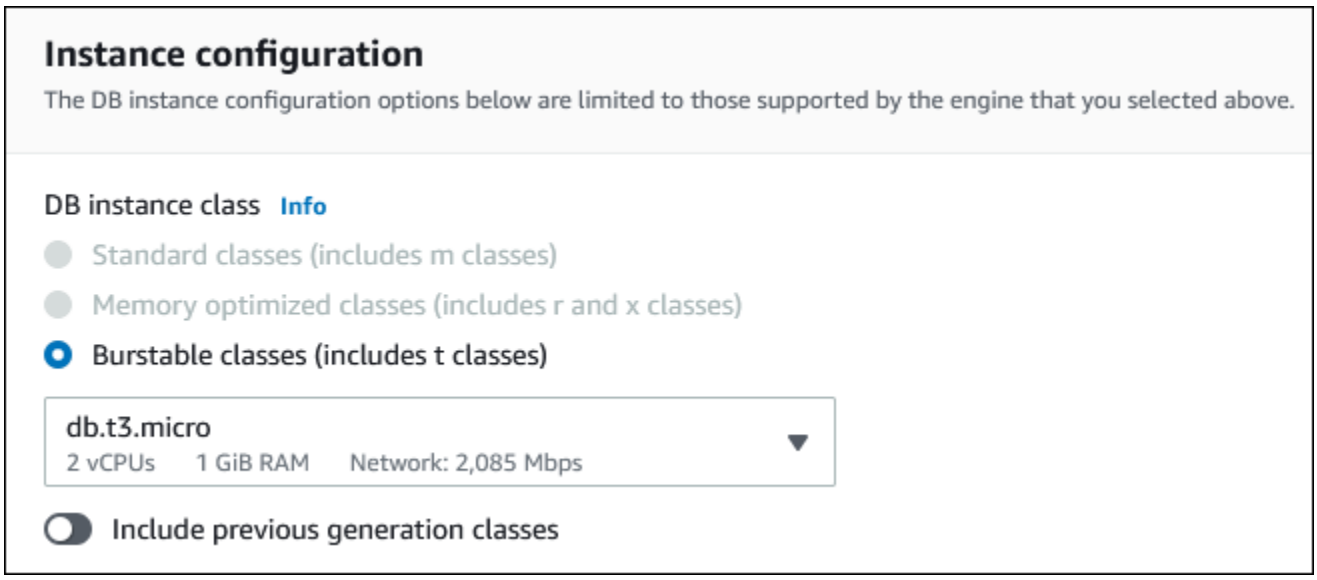


8. Dans la section Availability and durability (Disponibilité et durabilité), conservez les valeurs par défaut.
9. Dans la section Settings (Paramètres), définissez les valeurs suivantes :
   - DB Instance Identifier (Identifiant d'instance de base de données) : saisissez **`tutorial-db-instance`**.
   - Master username (Identifiant principal) : saisissez **`tutorial_user`**.
   - Auto generate a password (Génération automatique d'un mot de passe) : laissez cette option désactivée.
   - Master password (Mot de passe principal) : saisissez un mot de passe.
   - Confirm password (Confirmer le mot de passe) – Saisissez à nouveau le mot de passe.



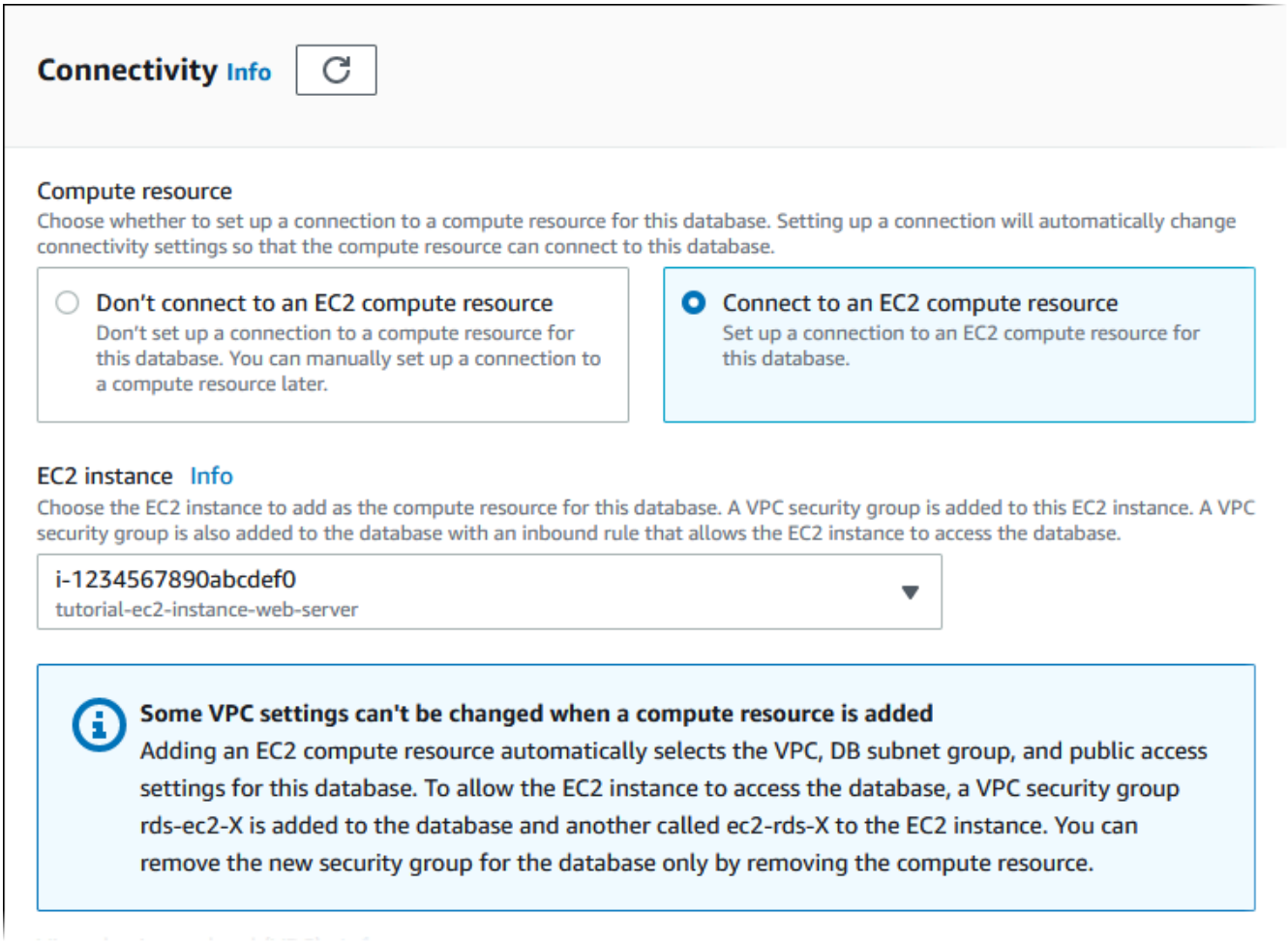
10. Dans la section Instance configuration (Configuration de l'instance), définissez les valeurs suivantes :

    - Classe à capacité extensible (inclut les classes t)
    - db.t3.micro



11. Dans la section Storage (Stockage), conservez les valeurs par défaut.
12. Dans la section Connectivity (Connectivité), définissez ces valeurs et conservez les autres valeurs par défaut :

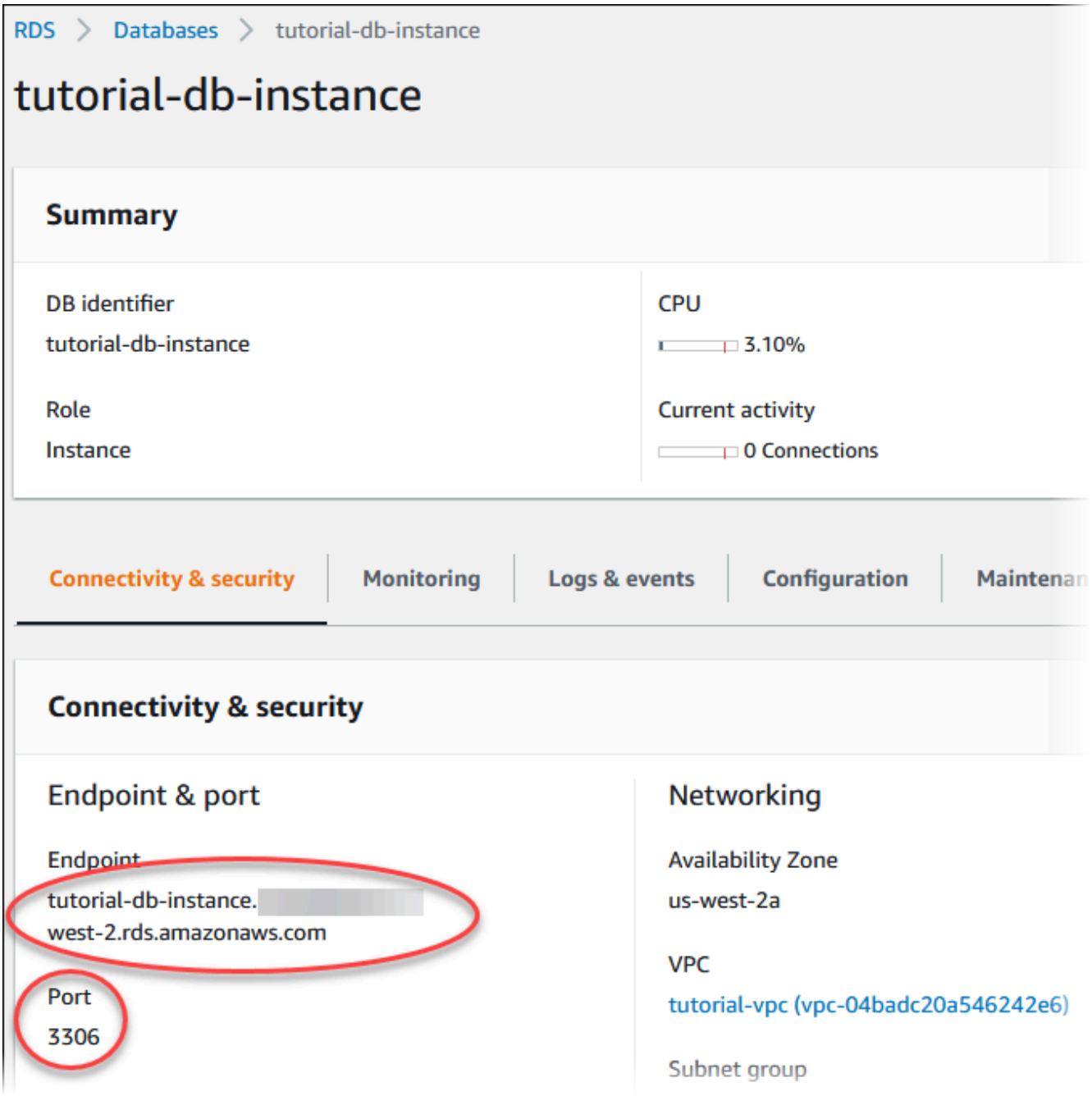
    - Dans le champ Ressource de calcul, choisissez Connect to an EC2 Compute Resource.
    - Par EC2 exemple, choisissez l' EC2 instance que vous avez créée précédemment, telle que tutorial-ec2 -. instance-web-server



13. Dans la section Database authentication (Authentification de base de données, assurez-vous que l'option Password authentication(Authentification par mot de passe) est sélectionnée.
14. Ouvrez la section Additional configuration (Configuration supplémentaire), puis entrez **`sample`** pour Initial database name (Nom de la base de données initiale). Conservez les paramètres par défaut pour les autres options.
15. Pour créer votre instance de base de données MySQL, choisissez Create database (Créer une base de données).

    Votre nouvelle instance de base de données apparaît dans la liste Databases (Bases de données) avec l'état Creating (Création en cours).
16. Attendez que le Status (État) de votre instance de base de données affiche Available (Disponible). Sélectionnez ensuite le nom de l'instance de base de données pour afficher les détails.

17. Dans la section Connectivity & security (Connectivité et sécurité), affichez le Endpoint (Point de terminaison) et le Port de l'instance de base de données.



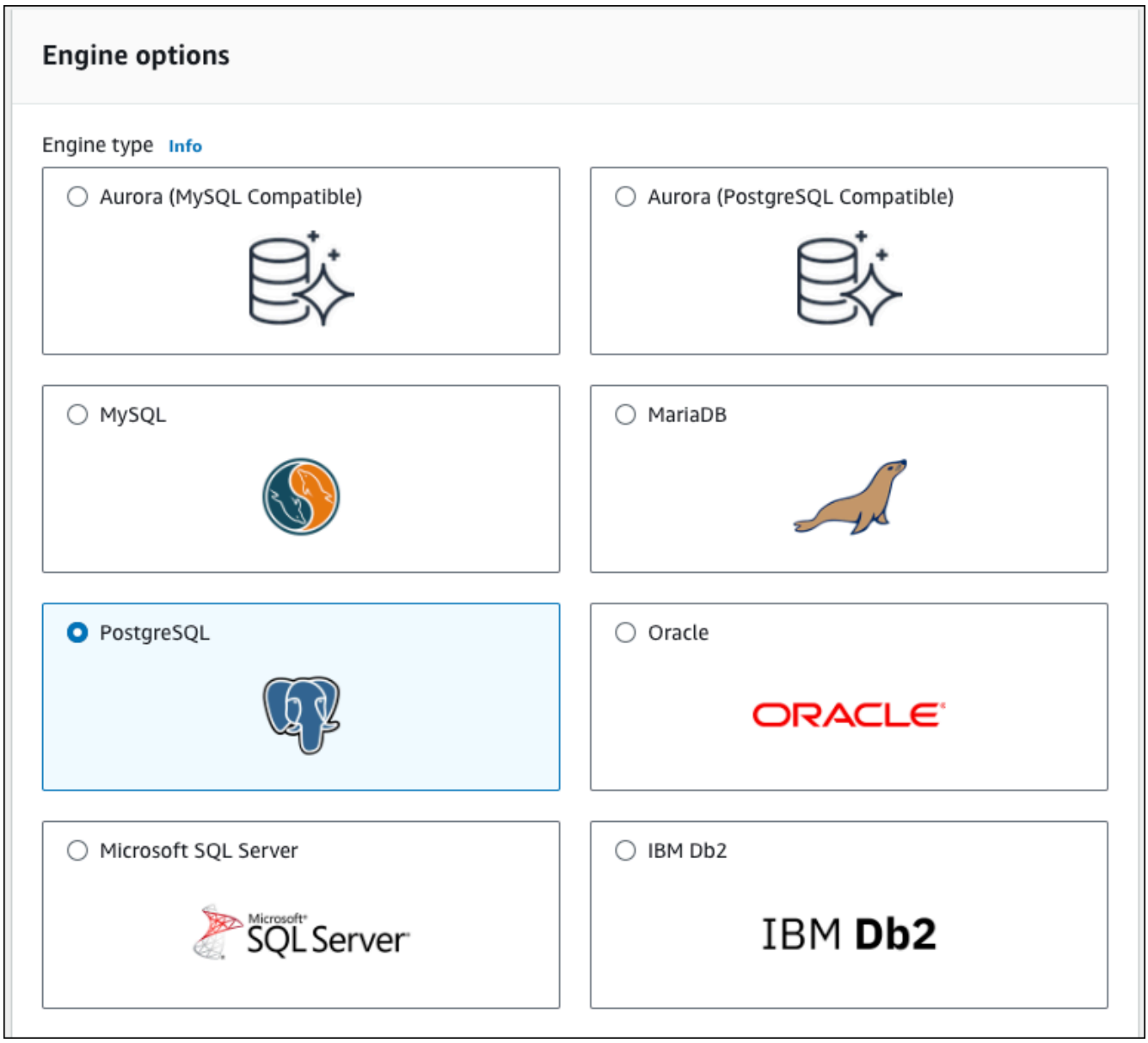
Notez le point de terminaison et le port de votre instance de base de données. Vous utilisez ces informations pour connecter votre serveur web à votre instance de base de données.

18. Termin Installer un serveur web sur votre instance EC2.

RDS for PostgreSQL

Pour créer une instance de base de données PostgreSQL

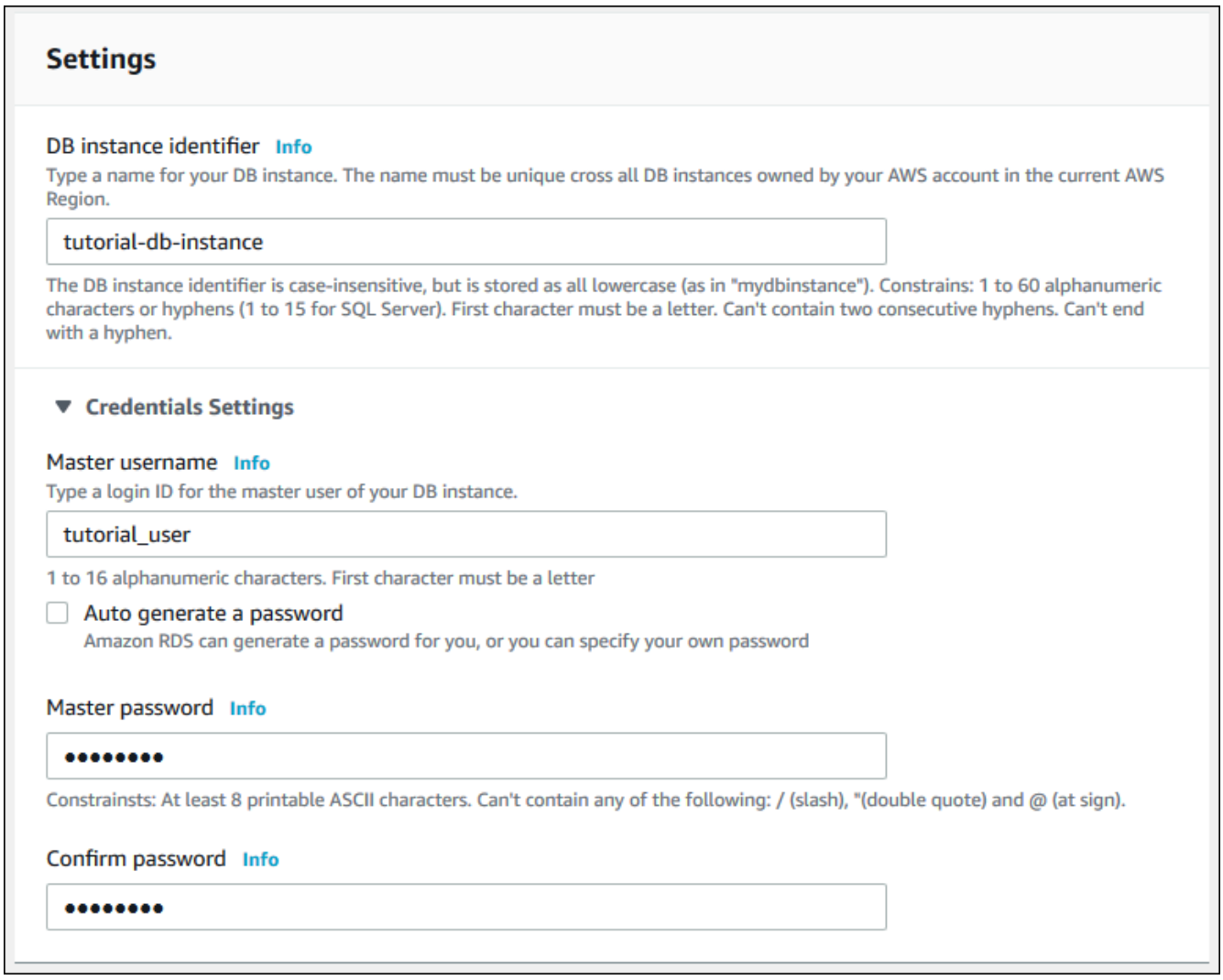
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit du AWS Management Console, cochez la case. Région AWS Il doit être le même que celui dans lequel vous avez créé votre EC2 instance.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).
5. Sur la page Créer une base de données, choisissez Création standard.
6. Pour Options de moteur, choisissez PostgreSQL.



7. Pour les modèles, choisissez le niveau gratuit ou Sandbox. Le niveau gratuit apparaît pour les comptes de plan gratuit. Sandbox apparaît pour les comptes de forfaits payants.

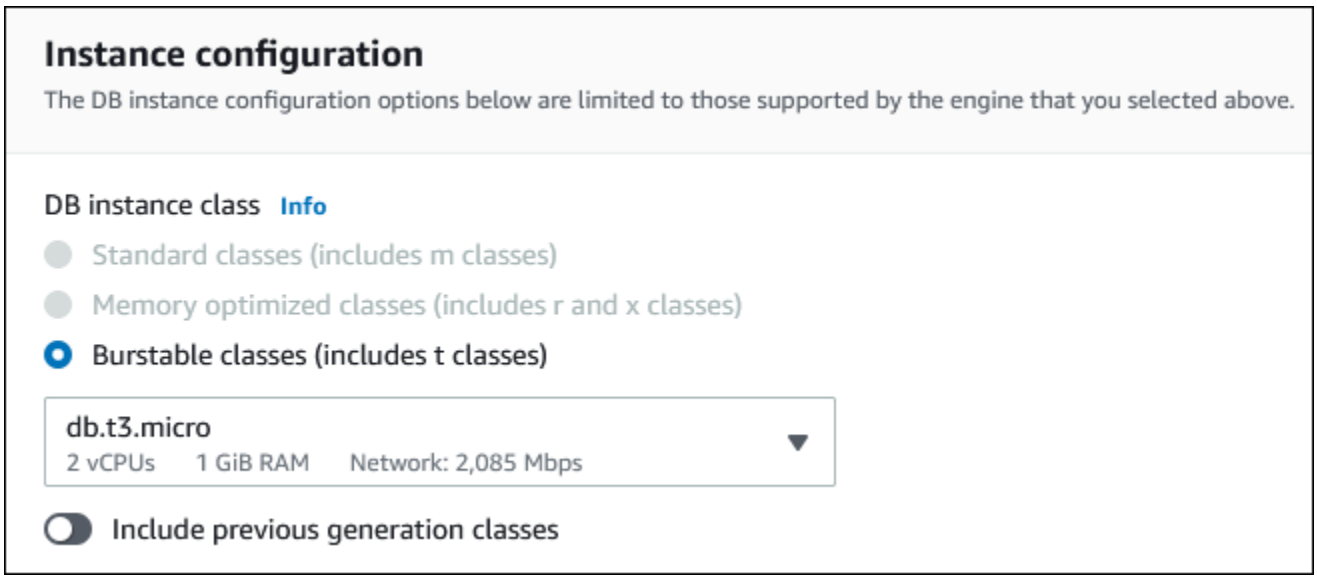


8. Dans la section Availability and durability (Disponibilité et durabilité), conservez les valeurs par défaut.
9. Dans la section Settings (Paramètres), définissez les valeurs suivantes :
   - DB Instance Identifier (Identifiant d'instance de base de données) : saisissez **`tutorial-db-instance`**.
   - Master username (Identifiant principal) : saisissez **`tutorial_user`**.
   - Auto generate a password (Génération automatique d'un mot de passe) : laissez cette option désactivée.
   - Master password (Mot de passe principal) : saisissez un mot de passe.
   - Confirm password (Confirmer le mot de passe) – Saisissez à nouveau le mot de passe.



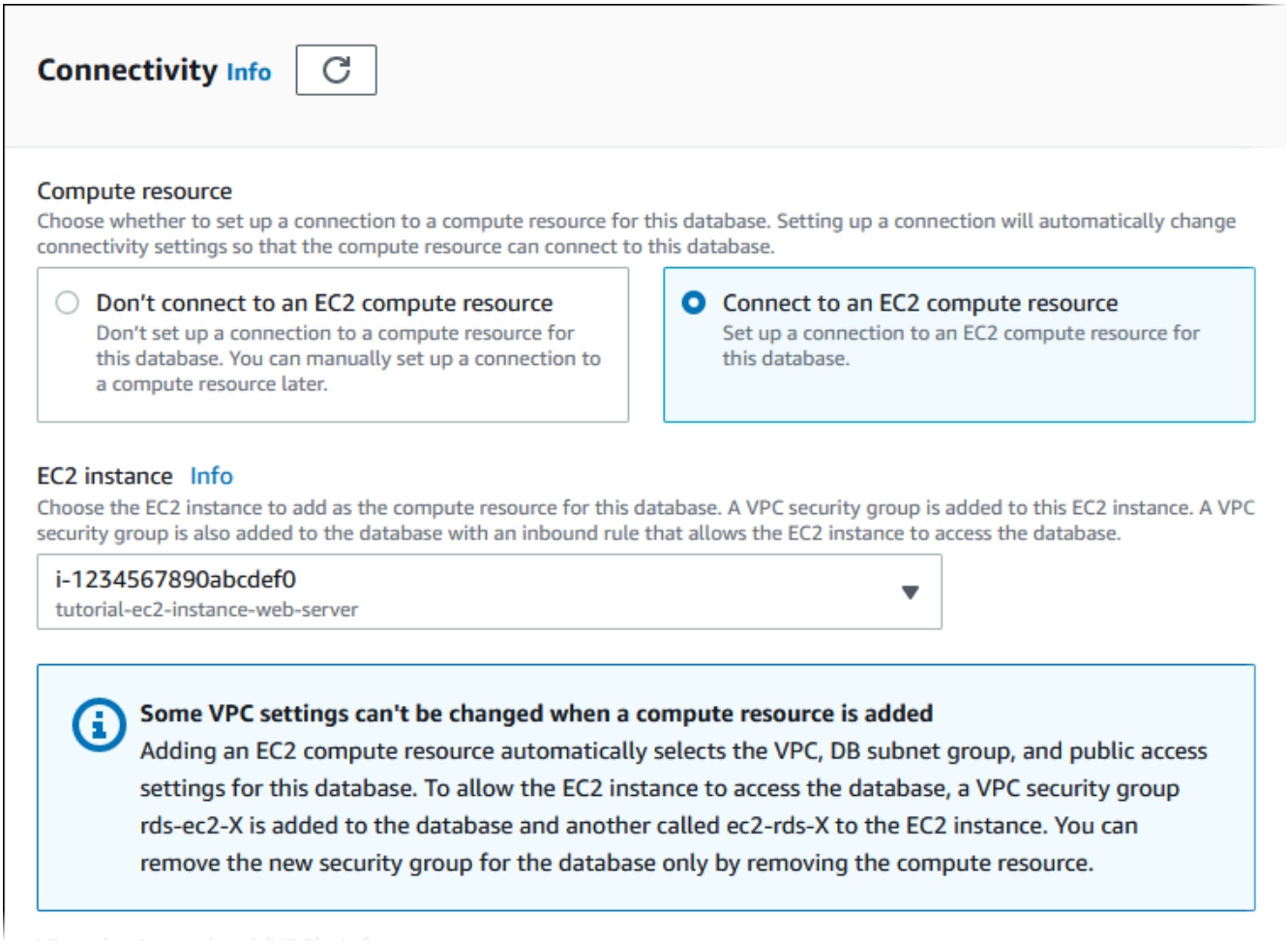
10. Dans la section Instance configuration (Configuration de l'instance), définissez les valeurs suivantes :

    - Classe à capacité extensible (inclut les classes t)
    - db.t3.micro



11. Dans la section Storage (Stockage), conservez les valeurs par défaut.
12. Dans la section Connectivity (Connectivité), définissez ces valeurs et conservez les autres valeurs par défaut :

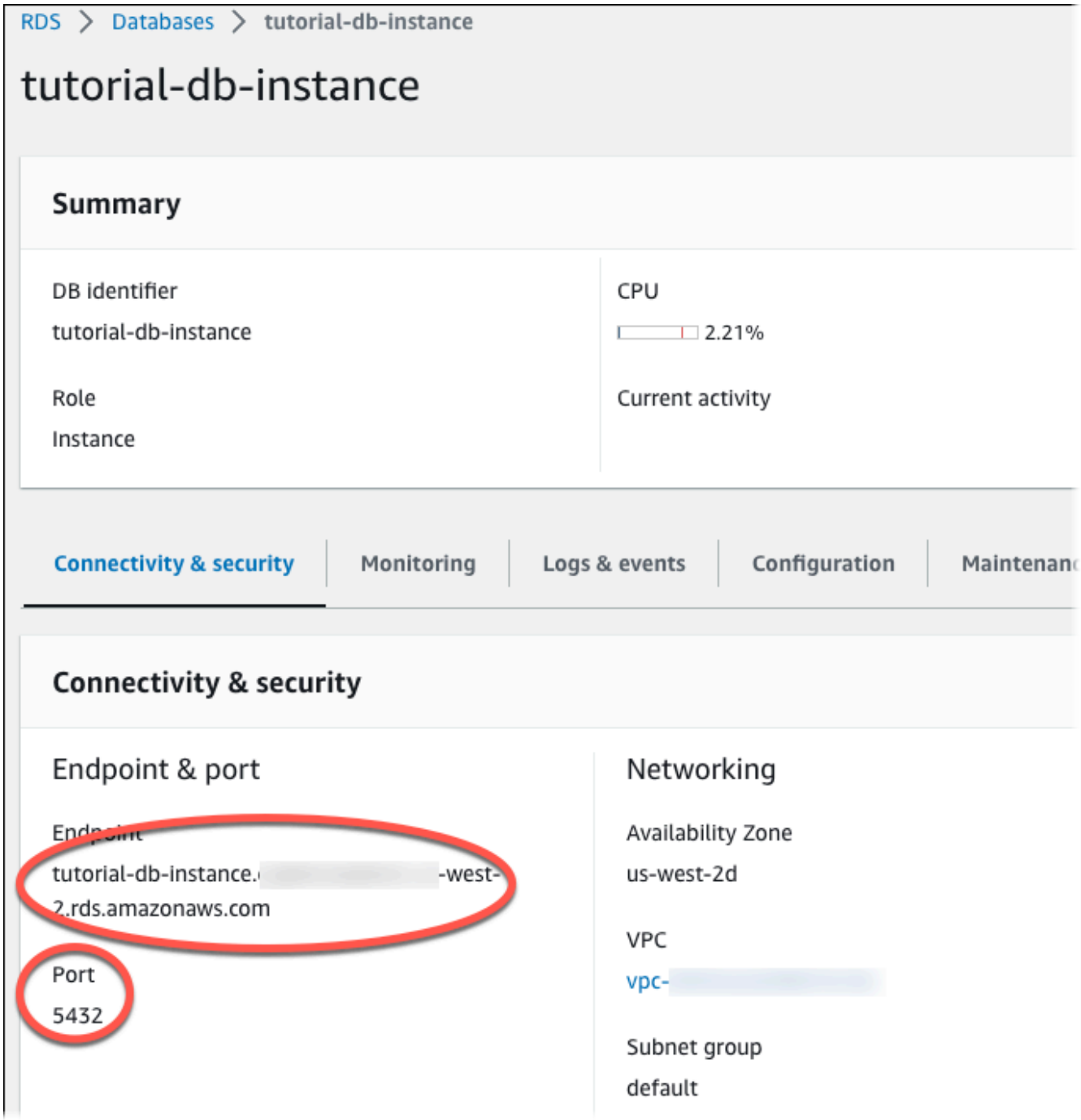
    - Dans le champ Ressource de calcul, choisissez Connect to an EC2 Compute Resource.
    - Par EC2 exemple, choisissez l' EC2 instance que vous avez créée précédemment, telle que tutorial-ec2 -. instance-web-server



13. Dans la section Database authentication (Authentification de base de données, assurez-vous que l'option Password authentication(Authentification par mot de passe) est sélectionnée.
14. Ouvrez la section Additional configuration (Configuration supplémentaire), puis entrez **`sample`** pour Initial database name (Nom de la base de données initiale). Conservez les paramètres par défaut pour les autres options.
15. Pour créer votre instance de base de données PostgreSQL, choisissez Créer une base de données.

    Votre nouvelle instance de base de données apparaît dans la liste Databases (Bases de données) avec l'état Creating (Création en cours).
16. Attendez que le Status (État) de votre instance de base de données affiche Available (Disponible). Sélectionnez ensuite le nom de l'instance de base de données pour afficher les détails.

17. Dans la section Connectivity & security (Connectivité et sécurité), affichez le Endpoint (Point de terminaison) et le Port de l'instance de base de données.



Notez le point de terminaison et le port de votre instance de base de données. Vous utilisez ces informations pour connecter votre serveur web à votre instance de base de données.

18. Termin [Installer un serveur web sur votre instance EC2](#).

Installer un serveur web sur votre instance EC2

Installez un serveur Web sur l'instance EC2 que vous avez créée dans Lancez une EC2 instance pour vous connecter à votre instance de base de données de de bases de données. Le serveur Web se connecte à l'instance de base de données Amazon RDS que vous avez créée dans Créer une instance de base de données Amazon RDS.

Installer un serveur Web Apache avec PHP et MariaDB

Connectez-vous à votre instance EC2 et installez le serveur web.

Pour vous connecter à votre instance EC2 et installer le serveur Web Apache avec PHP

1. Connectez-vous à l'instance EC2 que vous avez créée précédemment en suivant les étapes décrites dans la section Connexion à votre instance Linux dans le guide de l'utilisateur Amazon EC2.

   Nous vous recommandons de vous connecter à votre instance EC2 en utilisant SSH. Si l'utilitaire client SSH est installé sur Windows, Linux ou Mac, vous pouvez vous connecter à l'instance à l'aide du format de commande suivant :

   ```
   ssh -i location_of_pem_file ec2-user@ec2-instance-public-dns-name
   ```

   Supposons, par exemple, que `ec2-database-connect-key-pair.pem` soit stocké dans `/dir1` sur Linux et que le DNS IPv4 public de votre instance EC2 soit `ec2-12-345-678-90.compute-1.amazonaws.com`. Votre commande SSH se présenterait comme suit :

   ```
   ssh -i /dir1/ec2-database-connect-key-pair.pem ec2-
   user@ec2-12-345-678-90.compute-1.amazonaws.com
   ```

2. Obtenez les dernières corrections de bogues et mises à jour de sécurité en mettant à jour le logiciel sur votre instance EC2. Pour ce faire, exécutez la commande suivante.

   > **Note**
   >
   > L'option `-y` installe les mises à jour sans demander de confirmation. Pour examiner les mises à jour avant de les installer, omettez cette option.

```
sudo dnf update -y
```

3. Une fois les mises à jour terminées, installez le serveur web Apache, PHP et le logiciel MariaDB ou PostgreSQL à l'aide des commandes suivantes. Cette commande installe plusieurs packages logiciels et les dépendances connexes au même moment.

   MariaDB & MySQL

   ```
   sudo dnf install -y httpd php php-mysqli mariadb105
   ```

   PostgreSQL

   ```
   sudo dnf install -y httpd php php-pgsql postgresql15
   ```

   Si vous recevez une erreur, votre instance n'a probablement pas été lancée avec une AMI Amazon Linux 2023. Vous utilisez peut-être une AMI Amazon Linux 2 à la place. Vous pouvez afficher votre version d'Amazon Linux avec la commande suivante

   ```
   cat /etc/system-release
   ```

   Pour plus d'informations, consultez Mise à jour du logiciel de l'instance.

4. Démarrez le serveur web avec la commande illustrée ci-dessous.

   ```
   sudo systemctl start httpd
   ```

   Vous pouvez vérifier que votre serveur web est correctement installé et démarré. Pour ce faire, saisissez le nom DNS (Domain Name System) public de votre instance EC2 dans la barre d'adresse d'un navigateur web, par exemple : `http://ec2-42-8-168-21.us-west-1.compute.amazonaws.com`. Si votre serveur Web est en cours d'exécution, vous voyez la page de test Apache.

   Si la page de test Apache ne s'affiche pas, vérifiez vos règles entrantes pour le groupe de sécurité du VPC que vous avez créé dans Tutoriel : Création d'un VPC à utiliser avec un) IPv4 . Assurez-vous que vos règles entrantes incluent un accès HTTP (port 80) à l'adresse IP pour vous connecter au serveur Web.

> **Note**
>
> La page de test Apache apparaît uniquement en l'absence de contenu dans le répertoire racine des documents, `/var/www/html`. Après l'ajout de contenu dans le répertoire racine des documents, votre contenu apparaît à l'adresse DNS publique de votre instance EC2. Avant cela, il apparaît sur la page de test d'Apache.

5. Configurez le serveur web pour qu'il démarre à chaque redémarrage du système à l'aide de la commande `systemctl`.

```
sudo systemctl enable httpd
```

Pour autoriser `ec2-user` à gérer les fichiers dans le répertoire racine par défaut pour votre serveur Web Apache, modifiez l'appartenance et les autorisations du répertoire `/var/www`. Il existe plusieurs façons d'accomplir cette tâche. Dans ce didacticiel, vous ajoutez l'utilisateur `ec2-user` au groupe `apache` pour donner au groupe `apache` la propriété du répertoire `/var/www` et attribuer les autorisations d'écriture au groupe.

Pour définir les autorisations sur les fichiers pour le serveur Web Apache

1. Ajoutez l'utilisateur `ec2-user` au groupe `apache`.

```
sudo usermod -a -G apache ec2-user
```

2. Pour actualiser vos autorisations et inclure le nouveau groupe `apache`, déconnectez-vous.

```
exit
```

3. Reconnectez-vous et vérifiez que le groupe `apache` existe à l'aide de la commande `groups`.

```
groups
```

   Votre sortie se présente comme suit :

```
ec2-user adm wheel apache systemd-journal
```

4. Remplacez le groupe propriétaire du répertoire `/var/www` et de son contenu par le groupe `apache`.

```
sudo chown -R ec2-user:apache /var/www
```

5. Modifiez les autorisations des répertoires `/var/www` et de ses sous-répertoires pour ajouter des autorisations d'écriture de groupe et définir l'ID de groupe sur les sous-répertoires créés à l'avenir.

```
sudo chmod 2775 /var/www
find /var/www -type d -exec sudo chmod 2775 {} \;
```

6. Modifiez de façon récursive les autorisations pour les fichiers figurant dans le répertoire `/var/www` et ses sous-répertoires pour ajouter des autorisations d'écriture de groupe.

```
find /var/www -type f -exec sudo chmod 0664 {} \;
```

Maintenant, `ec2-user` (et tous les futurs membres du groupe `apache`) peut ajouter, supprimer et modifier les fichiers à la racine du document Apache. Cela vous permet d'ajouter du contenu, tel qu'un site Web statique ou une application PHP.

> **Note**
>
> Un serveur web exécutant le protocole HTTP ne fournit aucune sécurité de transport pour les données qu'il envoie ou reçoit. Lorsque vous vous connectez à un serveur HTTP via un navigateur Web, de nombreuses informations peuvent être vues par des personnes malveillantes sur le chemin d'accès réseau. Ces informations incluent les URL que vous visitez, le contenu des pages Web que vous recevez et le contenu (y compris les mots de passe) de tous les formulaires HTML.
> Les bonnes pratiques en matière de sécurisation de votre serveur Web consistent à installer la prise en charge HTTPS (HTTP Secure). Ce protocole protège vos données avec le chiffrement SSL/TLS. Pour plus d'informations, consultez Didacticiel : Configurer SSL/TLS avec l'AMI Amazon Linux dans le Guide de l'utilisateur Amazon EC2.

Connecter le serveur web Apache à l'instance de base de données

Ensuite, vous allez ajouter du contenu à votre serveur web Apache qui se connecte à votre instance de base de données Amazon RDS.

Pour ajouter du contenu au serveur web Apache qui se connecte à votre instance de base de données

1. Alors que vous êtes encore connecté à votre instance EC2, remplacez le répertoire par `/var/www` et créez un sous-répertoire nommé `inc`.

```
cd /var/www
mkdir inc
cd inc
```

2. Créez un fichier dans le répertoire `inc` nommé `dbinfo.inc`, puis modifiez le fichier en appelant nano (ou l'éditeur de votre choix).

```
>dbinfo.inc
nano dbinfo.inc
```

3. Ajoutez le contenu suivant au fichier `dbinfo.inc`. Ici, *`db_instance_endpoint`* est le point de terminaison de votre instance de base de données, sans le port, pour votre instance de base de données.

> **Note**
>
> Nous vous recommandons de placer les informations de nom d'utilisateur et de mot de passe dans un dossier ne faisant pas partie de la racine du document de votre serveur web. Vous réduisez ainsi la possibilité d'exposer vos informations de sécurité. Veillez à remplacer `master password` par un mot de passe approprié dans votre application.

```
<?php

define('DB_SERVER', 'db_instance_endpoint');
define('DB_USERNAME', 'tutorial_user');
define('DB_PASSWORD', 'master password');
define('DB_DATABASE', 'sample');
```

```
?>
```

4. Enregistrez et fermez le fichier `dbinfo.inc`. Si vous utilisez nano, enregistrez et fermez le fichier à l'aide des touches Ctrl+S et Ctrl+X.
5. Remplacez le répertoire par `/var/www/html`.

```
cd /var/www/html
```

6. Créez un fichier dans le répertoire `html` nommé `SamplePage.php`, puis modifiez le fichier en appelant nano (ou l'éditeur de votre choix).

```
>SamplePage.php
nano SamplePage.php
```

7. Ajoutez le contenu suivant au fichier `SamplePage.php` :

   MariaDB & MySQL

```
<?php include "../inc/dbinfo.inc"; ?>
<html>
<body>
<h1>Sample page</h1>
<?php

  /* Connect to MySQL and select the database. */
  $connection = mysqli_connect(DB_SERVER, DB_USERNAME, DB_PASSWORD);

  if (mysqli_connect_errno()) echo "Failed to connect to MySQL: " .
 mysqli_connect_error();

  $database = mysqli_select_db($connection, DB_DATABASE);

  /* Ensure that the EMPLOYEES table exists. */
  VerifyEmployeesTable($connection, DB_DATABASE);

  /* If input fields are populated, add a row to the EMPLOYEES table. */
  $employee_name = htmlentities($_POST['NAME']);
  $employee_address = htmlentities($_POST['ADDRESS']);

  if (strlen($employee_name) || strlen($employee_address)) {
    AddEmployee($connection, $employee_name, $employee_address);
  }
```

```
?>

<!-- Input form -->
<form action="<?PHP echo $_SERVER['SCRIPT_NAME'] ?>" method="POST">
  <table border="0">
    <tr>
      <td>NAME</td>
      <td>ADDRESS</td>
    </tr>
    <tr>
      <td>
        <input type="text" name="NAME" maxlength="45" size="30" />
      </td>
      <td>
        <input type="text" name="ADDRESS" maxlength="90" size="60" />
      </td>
      <td>
        <input type="submit" value="Add Data" />
      </td>
    </tr>
  </table>
</form>

<!-- Display table data. -->
<table border="1" cellpadding="2" cellspacing="2">
  <tr>
    <td>ID</td>
    <td>NAME</td>
    <td>ADDRESS</td>
  </tr>

<?php

$result = mysqli_query($connection, "SELECT * FROM EMPLOYEES");

while($query_data = mysqli_fetch_row($result)) {
  echo "<tr>";
  echo "<td>",$query_data[0], "</td>",
       "<td>",$query_data[1], "</td>",
       "<td>",$query_data[2], "</td>";
  echo "</tr>";
}
?>
```

```
</table>

<!-- Clean up. -->
<?php

  mysqli_free_result($result);
  mysqli_close($connection);

?>

</body>
</html>


<?php

/* Add an employee to the table. */
function AddEmployee($connection, $name, $address) {
   $n = mysqli_real_escape_string($connection, $name);
   $a = mysqli_real_escape_string($connection, $address);

   $query = "INSERT INTO EMPLOYEES (NAME, ADDRESS) VALUES ('$n', '$a');";

   if(!mysqli_query($connection, $query)) echo("<p>Error adding employee data.</
p>");
}

/* Check whether the table exists and, if not, create it. */
function VerifyEmployeesTable($connection, $dbName) {
  if(!TableExists("EMPLOYEES", $connection, $dbName))
  {
     $query = "CREATE TABLE EMPLOYEES (
         ID int(11) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
         NAME VARCHAR(45),
         ADDRESS VARCHAR(90)
       )";

     if(!mysqli_query($connection, $query)) echo("<p>Error creating table.</
p>");
  }
}

/* Check for the existence of a table. */
function TableExists($tableName, $connection, $dbName) {
```

```
  $t = mysqli_real_escape_string($connection, $tableName);
  $d = mysqli_real_escape_string($connection, $dbName);

  $checktable = mysqli_query($connection,
      "SELECT TABLE_NAME FROM information_schema.TABLES WHERE TABLE_NAME = '$t'
 AND TABLE_SCHEMA = '$d'");

  if(mysqli_num_rows($checktable) > 0) return true;

  return false;
}
?>
```

PostgreSQL

```
<?php include "../inc/dbinfo.inc"; ?>

<html>
<body>
<h1>Sample page</h1>
<?php

/* Connect to PostgreSQL and select the database. */
$constring = "host=" . DB_SERVER . " dbname=" . DB_DATABASE . " user=" .
 DB_USERNAME . " password=" . DB_PASSWORD ;
$connection = pg_connect($constring);

if (!$connection){
 echo "Failed to connect to PostgreSQL";
 exit;
}

/* Ensure that the EMPLOYEES table exists. */
VerifyEmployeesTable($connection, DB_DATABASE);

/* If input fields are populated, add a row to the EMPLOYEES table. */
$employee_name = htmlentities($_POST['NAME']);
$employee_address = htmlentities($_POST['ADDRESS']);

if (strlen($employee_name) || strlen($employee_address)) {
  AddEmployee($connection, $employee_name, $employee_address);
}
```

```
?>

<!-- Input form -->
<form action="<?PHP echo $_SERVER['SCRIPT_NAME'] ?>" method="POST">
  <table border="0">
    <tr>
      <td>NAME</td>
      <td>ADDRESS</td>
    </tr>
    <tr>
      <td>
    <input type="text" name="NAME" maxlength="45" size="30" />
      </td>
      <td>
    <input type="text" name="ADDRESS" maxlength="90" size="60" />
      </td>
      <td>
    <input type="submit" value="Add Data" />
      </td>
    </tr>
  </table>
</form>
<!-- Display table data. -->
<table border="1" cellpadding="2" cellspacing="2">
  <tr>
    <td>ID</td>
    <td>NAME</td>
    <td>ADDRESS</td>
  </tr>

<?php

$result = pg_query($connection, "SELECT * FROM EMPLOYEES");

while($query_data = pg_fetch_row($result)) {
  echo "<tr>";
  echo "<td>",$query_data[0], "</td>",
       "<td>",$query_data[1], "</td>",
       "<td>",$query_data[2], "</td>";
  echo "</tr>";
}
?>
</table>
```

```
<!-- Clean up. -->
<?php

  pg_free_result($result);
  pg_close($connection);
?>
</body>
</html>


<?php

/* Add an employee to the table. */
function AddEmployee($connection, $name, $address) {
   $n = pg_escape_string($name);
   $a = pg_escape_string($address);
   echo "Forming Query";
   $query = "INSERT INTO EMPLOYEES (NAME, ADDRESS) VALUES ('$n', '$a');";

   if(!pg_query($connection, $query)) echo("<p>Error adding employee data.</
p>");
}

/* Check whether the table exists and, if not, create it. */
function VerifyEmployeesTable($connection, $dbName) {
  if(!TableExists("EMPLOYEES", $connection, $dbName))
  {
     $query = "CREATE TABLE EMPLOYEES (
         ID serial PRIMARY KEY,
         NAME VARCHAR(45),
         ADDRESS VARCHAR(90)
       )";

     if(!pg_query($connection, $query)) echo("<p>Error creating table.</p>");
  }
}
/* Check for the existence of a table. */
function TableExists($tableName, $connection, $dbName) {
  $t = strtolower(pg_escape_string($tableName)); //table name is case sensitive
  $d = pg_escape_string($dbName); //schema is 'public' instead of 'sample' db
 name so not using that
```

```
  $query = "SELECT TABLE_NAME FROM information_schema.TABLES WHERE TABLE_NAME =
 '$t';";
  $checktable = pg_query($connection, $query);

  if (pg_num_rows($checktable) >0) return true;
  return false;

}
?>
```

8. Enregistrez et fermez le fichier `SamplePage.php`.
9. Vérifiez que votre serveur web se connecte avec succès à votre instance de base de données en ouvrant un navigateur web et en accédant à une page `http://`*`EC2 instance endpoint`*`/SamplePage.php`, par exemple : `http://ec2-12-345-67-890.us-west-2.compute.amazonaws.com/SamplePage.php`.

Vous pouvez utiliser `SamplePage.php` pour ajouter des données à votreinstance de base de données. Les données que vous ajoutez sont ensuite affichées sur la page. Pour vérifier que les données ont été insérées dans la table, installez le client MySQL sur l'instance Amazon EC2. Connectez-vous ensuite à l'instance de bases de données et interrogez la table.

Pour plus d'informations sur l'installation du client MySQL et la connexion à une instance de base de données, consultez Connexion à votre instance de base de données MySQL.

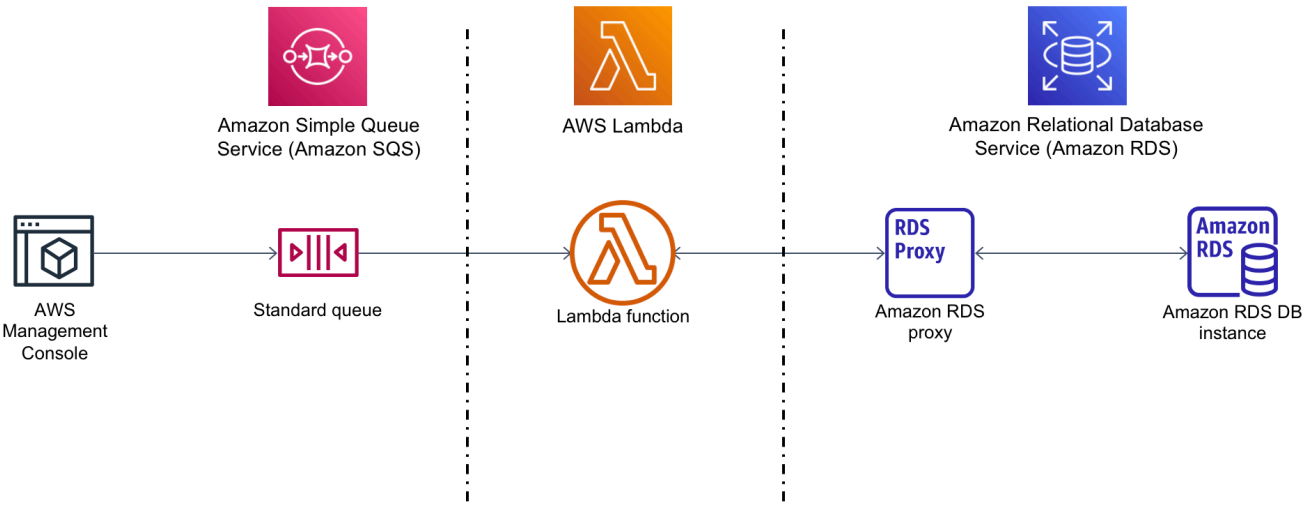
Pour vérifier que la sécurité de votre instance de base de données est assurée, contrôlez que les sources extérieures du VPC ne peuvent pas se connecter à votre instance de base de données.

Après avoir testé votre serveur Web et votre base de données, vous devez supprimer votre instance de base de données et votre instance Amazon EC2.

- Pour supprimer une instance de base de données, suivez les instructions de la section Suppression d'une instance DB. Vous n'avez pas besoin de créer un instantané final.
- Pour résilier une instance Amazon EC2, suivez les instructions de la page Résilier votre instance dans le Guide de l'utilisateur Amazon EC2.

Tutoriel : Utilisation d'une fonction Lambda pour accéder à une base de données Amazon RDS

Dans ce tutoriel, vous allez utiliser une fonction Lambda pour écrire des données dans une base de données Amazon Relational Database Service (Amazon RDS) via un proxy RDS. Votre fonction Lambda lit les enregistrements d'une file d'attente Amazon Simple Queue Service (Amazon SQS) et écrit un nouvel élément dans une table de votre base de données chaque fois qu'un message est ajouté. Dans cet exemple, vous utilisez la AWS Management Console pour ajouter manuellement des messages à votre file d'attente. Le schéma suivant montre les AWS ressources que vous utilisez pour suivre le didacticiel.



Grâce à Amazon RDS, vous pouvez exécuter une base de données relationnelle gérée dans le cloud à l'aide de produits de base de données courants tels que Microsoft SQL Server, MariaDB, MySQL, Oracle Database et PostgreSQL. En utilisant Lambda pour accéder à votre base de données, vous pouvez lire et écrire des données en réponse aux événements, tels que l'enregistrement d'un nouveau client sur votre site web. Votre fonction, votre instance de base de données et votre proxy sont automatiquement mis à l'échelle pour répondre aux périodes de demandes élevées.

Pour compléter ce tutoriel, effectuez les tâches suivantes :

1. Lancez une instance de base de données RDS for MySQL et un proxy dans votre Compte AWS VPC par défaut.

2. Créez et testez une fonction Lambda qui crée une nouvelle table dans votre base de données et y écrit des données.
3. Créez une file d'attente Amazon SQS et configurez-la pour invoquer votre fonction Lambda chaque fois qu'un nouveau message est ajouté.
4. Testez l'ensemble de la configuration en ajoutant des messages à votre file d'attente à l'aide des journaux AWS Management Console et en surveillant les résultats à l'aide CloudWatch des journaux.

En suivant ces étapes, vous apprendrez à :

- Utiliser Amazon RDS pour créer une instance de base de données et un proxy, et connecter une fonction Lambda au proxy.
- Utiliser Lambda pour effectuer des opérations de création et de lecture sur une base de données Amazon RDS.
- Utiliser Amazon SQS pour invoquer une fonction Lambda.

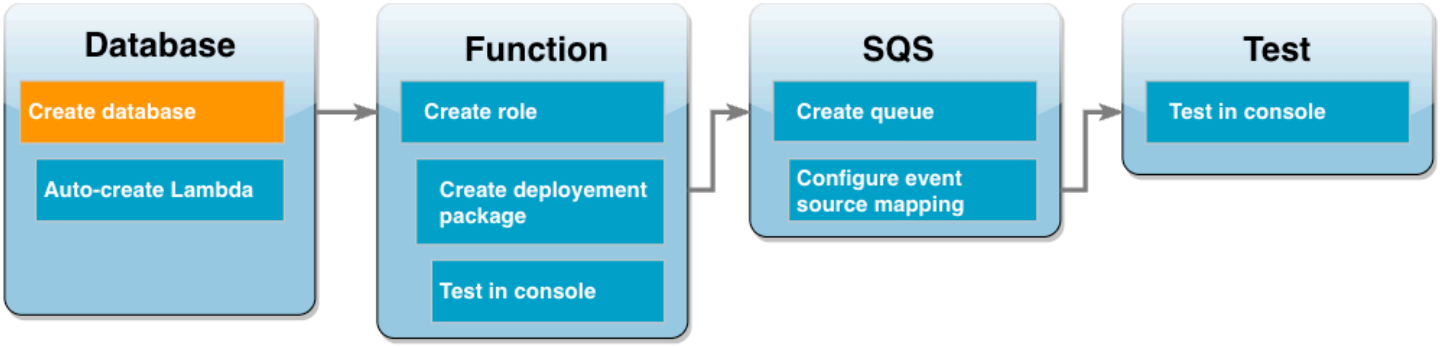
Vous pouvez terminer ce didacticiel en utilisant le AWS Management Console ou le AWS Command Line Interface (AWS CLI).

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

- Inscrivez-vous pour un Compte AWS
- Création d'un utilisateur doté d'un accès administratif

Créer une instance de base de données Amazon RDS



Une instance de base de données Amazon RDS est un environnement de base de données isolé qui s'exécute dans le AWS Cloud. Une instance peut comporter une ou plusieurs bases de données créées par l'utilisateur. Sauf indication contraire de votre part, Amazon RDS crée de nouvelles instances de base de données dans le VPC par défaut inclus dans votre. Compte AWS Pour plus d'informations sur le VPC Amazon, consultez le Guide de l'utilisateur Amazon Virtual Private Cloud.

Dans ce didacticiel, vous allez créer une nouvelle instance dans votre Compte AWS VPC par défaut et créer une base de données nommée `ExampleDB` dans cette instance. Vous pouvez créer votre instance de base de données et votre base de données à l'aide du AWS Management Console ou du AWS CLI.

Pour créer une instance de base de données

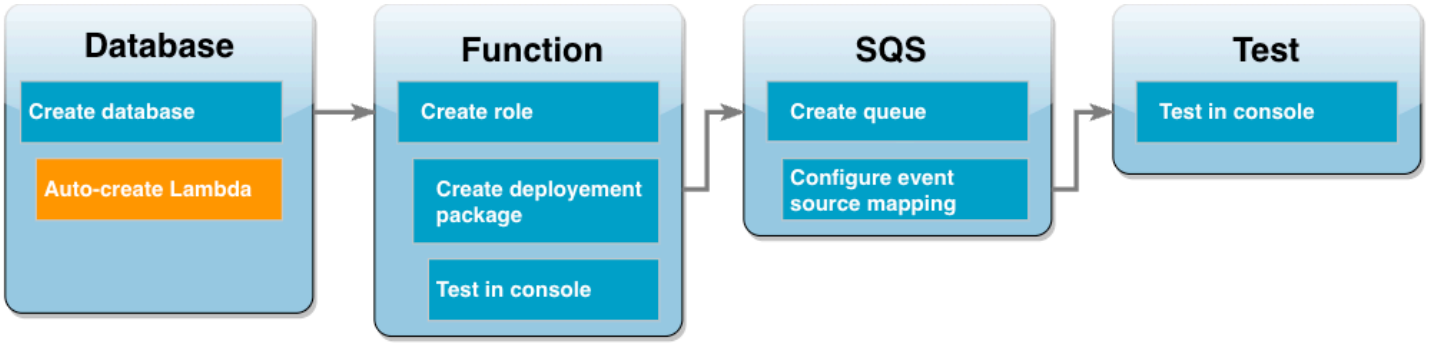
1. Ouvrez la console Amazon RDS et choisissez Créer une base de données.
2. Ne désélectionnez pas l'option Création standard, puis dans Options de moteur, choisissez MySQL.
3. Dans Modèles, sélectionnez Free tier ou Sandbox. Le niveau gratuit apparaît pour les comptes du niveau gratuit. Sandbox apparaît pour les comptes de forfaits payants.
4. Dans Paramètres, pour Identifiant de l'instance de base de données, saisissez **`MySQLForLambda`**.
5. Définissez votre nom d'utilisateur et votre mot de passe en procédant comme suit :

   a. Dans Configuration des informations d'identification, conservez le paramètre Identifiant principal défini sur `admin`.

   b. Pour Mot de passe principal, saisissez et confirmez un mot de passe pour accéder à votre base de données.
6. Spécifiez le nom de la base de données en procédant comme suit :

   - Laissez toutes les autres options par défaut sélectionnées et faites défiler l'affichage vers le bas jusqu'à la section Configuration supplémentaire.
   - Développez cette section et saisissez **`ExampleDB`** comme Nom de la base de données initiale.
7. Ne désélectionnez pas les options par défaut restantes et choisissez Créer une base de données.

Création d'une fonction Lambda et d'un proxy



Vous pouvez utiliser la console RDS pour créer une fonction Lambda et un proxy dans le même VPC que la base de données.

> **Note**
>
> Vous ne pouvez créer ces ressources associées que lorsque la création de votre base de données est terminée et qu'elle a le statut Disponible.

Pour créer une fonction et un proxy associés

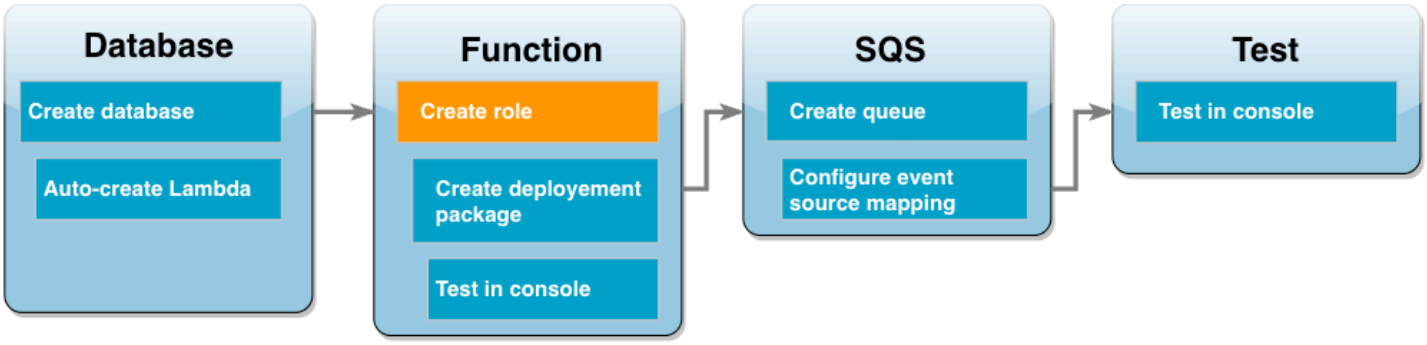
1. Dans la page Bases de données, vérifiez si votre base de données a le statut Disponible. Si tel est le cas, passez à l'étape suivante. Sinon, attendez que votre base de données soit disponible.
2. Sélectionnez votre base de données et choisissez Configurer une connexion Lambda dans Actions.
3. Dans la page Configurer une connexion Lambda, choisissez Créer une nouvelle fonction.

   Définissez Nouveau nom de fonction Lambda sur **`LambdaFunctionWithRDS`**.
4. Dans la section Proxy RDS, sélectionnez l'option Se connecter via un proxy RDS. Choisissez encore Créer un nouveau proxy.

   - Pour Informations d'identification de la base de données, choisissez Nom d'utilisateur et mot de passe de base de données.
   - PourNom d'utilisateur, spécifiez `admin`.
   - Pour Mot de passe, saisissez le mot de passe que vous avez créé pour votre instance de base de données.
5. Sélectionnez Configurer pour terminer la création du proxy et de la fonction Lambda.

L'assistant termine la configuration et fournit un lien vers la console Lambda pour que vous passiez en revue votre nouvelle fonction. Prenez note du point de terminaison du proxy avant de passer à la console Lambda.

Pour créer un rôle d'exécution de fonction



Avant de créer votre fonction Lambda, vous devez créer un rôle d'exécution pour donner à votre fonction les autorisations nécessaires. Pour ce tutoriel, Lambda a besoin d'une autorisation pour gérer la connexion réseau au VPC contenant votre instance de base de données et pour interroger les messages d'une file d'attente Amazon SQS.

Pour donner à votre fonction Lambda les autorisations dont elle a besoin, ce tutoriel utilise des politiques gérées par IAM. Il s'agit de politiques accordant des autorisations pour de nombreux cas d'utilisation courants et disponibles dans votre Compte AWS. Pour en savoir plus sur l'utilisation des politiques gérées, consultez Bonnes pratiques en matière de politiques.

Pour créer le rôle d'exécution Lambda

1. Ouvrez la page Rôles de la console IAM et choisissez Créer un rôle.
2. Pour Type d'entité approuvée, choisissez Service AWS et pour Cas d'utilisation, choisissez Lambda.
3. Choisissez Suivant.
4. Ajoutez les politiques gérées par IAM en procédant comme suit :

    a. À l'aide du champ de recherche de la politique, recherchez **`AWSLambdaSQSQueueExecutionRole`**.
    b. Dans la liste des résultats, cochez la case à côté du rôle, puis choisissez Effacer les filtres.
    c. À l'aide du champ de recherche de la politique, recherchez **`AWSLambdaVPCAccessExecutionRole`**.
    d. Dans la liste des résultats, cochez la case à côté du rôle, puis choisissez Suivant.

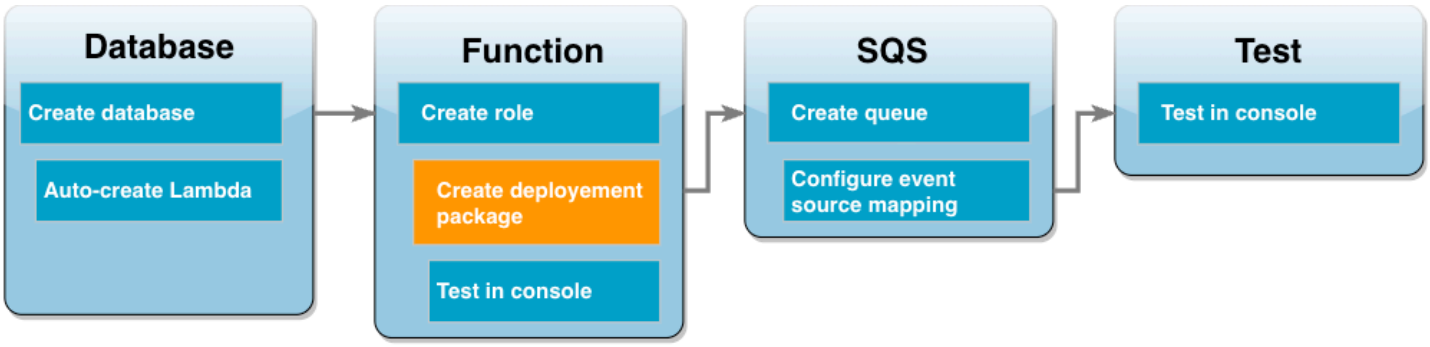
5. Pour Nom du rôle, saisissez **`lambda-vpc-sqs-role`**, puis choisissez Créer un rôle.

Plus loin dans le tutoriel, vous aurez besoin de l'Amazon Resource Name (ARN) du rôle d'exécution que vous venez de créer.

Pour trouver l'ARN du rôle d'exécution

1. Ouvrez la page Rôles de la console IAM et choisissez votre rôle (`lambda-vpc-sqs-role`).
2. Copiez l'ARN affiché dans la section Récapitulatif.

Création d'un package de déploiement Lambda



L'exemple de code Python suivant utilise le package PyMySQL pour ouvrir une connexion à votre base de données. La première fois que vous invoquez votre fonction, elle crée également une nouvelle table invoquée `Customer`. Le tableau utilise le schéma suivant, où `CustID` se trouve la clé primaire :

```
Customer(CustID, Name)
```

La fonction utilise également le PyMy langage SQL pour ajouter des enregistrements à cette table. La fonction ajoute des enregistrements à l'aide des clients IDs et des noms spécifiés dans les messages que vous ajouterez à votre file d'attente Amazon SQS.

Le code crée la connexion à votre base de données en dehors de la fonction de gestionnaire. La création de la connexion dans le code d'initialisation permet de réutiliser la connexion lors des invocations ultérieures de votre fonction et améliore les performances. Dans une application de production, vous pouvez également utiliser la simultanéité provisionnée pour initialiser le nombre requis de connexions à la base de données. Ces connexions sont disponibles dès que votre fonction est invoquée.

```
import sys
import logging
import pymysql
import json
import os

# rds settings
user_name = os.environ['USER_NAME']
password = os.environ['PASSWORD']
rds_proxy_host = os.environ['RDS_PROXY_HOST']
db_name = os.environ['DB_NAME']

logger = logging.getLogger()
logger.setLevel(logging.INFO)

# create the database connection outside of the handler to allow connections to be
# re-used by subsequent function invocations.
try:
        conn = pymysql.connect(host=rds_proxy_host, user=user_name, passwd=password,
 db=db_name, connect_timeout=5)
except pymysql.MySQLError as e:
    logger.error("ERROR: Unexpected error: Could not connect to MySQL instance.")
    logger.error(e)
    sys.exit(1)

logger.info("SUCCESS: Connection to RDS for MySQL instance succeeded")

def lambda_handler(event, context):
    """
    This function creates a new RDS database table and writes records to it
    """
    message = event['Records'][0]['body']
    data = json.loads(message)
    CustID = data['CustID']
    Name = data['Name']

    item_count = 0
    sql_string = f"insert into Customer (CustID, Name) values(%s, %s)"

    with conn.cursor() as cur:
        cur.execute("create table if not exists Customer ( CustID  int NOT NULL, Name
 varchar(255) NOT NULL, PRIMARY KEY (CustID))")
        cur.execute(sql_string, (CustID, Name))
```

```
        conn.commit()
        cur.execute("select * from Customer")
        logger.info("The following items have been added to the database:")
        for row in cur:
            item_count += 1
            logger.info(row)
    conn.commit()

    return "Added %d items to RDS for MySQL table" %(item_count)
```

> **Note**
>
> Dans cet exemple, vos informations d'identification d'accès à la base de données sont stockées sous forme de variables d'environnement. Dans les applications de production, nous vous recommandons d'utiliser AWS Secrets Manager comme option plus sécurisée. Notez que si votre fonction Lambda se trouve dans un VPC, pour vous connecter à Secrets Manager, vous devez créer un point de terminaison de VPC. Consultez Comment se connecter au service Secrets Manager dans un cloud privé virtuel (langue française non garantie) pour en savoir plus.

Pour inclure la dépendance PyMy SQL dans votre code de fonction, créez un package de déploiement .zip. Les commandes suivantes fonctionnent sous Linux, macOS et Unix :

Pour créer un package de déploiement .zip

1. Enregistrez l'exemple de code en tant que fichier nommé `lambda_function.py`.
2. Dans le répertoire dans lequel vous avez créé votre `lambda_function.py` fichier, créez un nouveau répertoire nommé `package` et installez la bibliothèque PyMy SQL.

   ```
   mkdir package
   pip install --target package pymysql
   ```

3. Créez un fichier zip contenant le code de votre application et la bibliothèque PyMy SQL. Sous Linux ou macOS, exécutez les commandes CLI suivantes. Sous Windows, utilisez l'outil zip de votre choix pour créer le fichier `lambda_function.zip`. Votre fichier de code source `lambda_function.py` et les dossiers contenant vos dépendances doivent être installés à la racine du fichier .zip.

```
cd package
zip -r ../lambda_function.zip .
cd ..
zip lambda_function.zip lambda_function.py
```

Vous pouvez également créer votre package de déploiement à l'aide d'un environnement virtuel Python. Consultez Déployer des fonctions Lambda en Python avec des archives de fichiers .zip.

Mise à jour de la fonction Lambda

À l'aide du package .zip que vous venez de créer, vous mettez désormais à jour votre fonction Lambda en utilisant la console Lambda. Pour permettre à votre fonction d'accéder à votre base de données, vous devez également configurer des variables d'environnement avec vos informations d'identification d'accès.

Pour mettre à jour la fonction Lambda

1. Ouvrez la page Fonctions de la console Lambda et choisissez votre fonction `LambdaFunctionWithRDS`.
2. Dans l'onglet Paramètres d'exécution, sélectionnez Modifier pour remplacer le Runtime de la fonction par Python 3.10.
3. Modifiez le paramètre Gestionnaire en spécifiant `lambda_function.lambda_handler`.
4. Dans l'onglet Code, choisissez Charger depuis, puis Fichier .zip.
5. Sélectionnez le fichier `lambda_function.zip` que vous avez créé à l'étape précédente et choisissez Enregistrer.

Configurez maintenant la fonction avec le rôle d'exécution que vous avez créé précédemment. Cela octroie à la fonction les autorisations dont elle a besoin pour accéder à votre instance de base de données et interroger une file d'attente Amazon SQS.

Pour configurer le rôle d'exécution de la fonction

1. Sur la page Fonctions de la console Lambda, sélectionnez l'onglet Configuration, puis choisissez Autorisations.
2. Dans Rôle d'exécution, choisissez Modifier.
3. Dans Rôle existant, choisissez votre rôle d'exécution (`lambda-vpc-sqs-role`).

4. Choisissez Enregistrer.

Pour configurer les variables d'environnement de votre fonction

1. Sur la page Fonctions de la console Lambda, sélectionnez l'onglet Configuration, puis choisissez Variables d'environnement.
2. Choisissez Modifier.
3. Pour ajouter vos informations d'identification d'accès à la base de données, procédez comme suit :

   a. Choisissez Ajouter une variable d'environnement, puis pour Clé saisissez **`USER_NAME`** et pour Valeur saisissez **`admin`**.

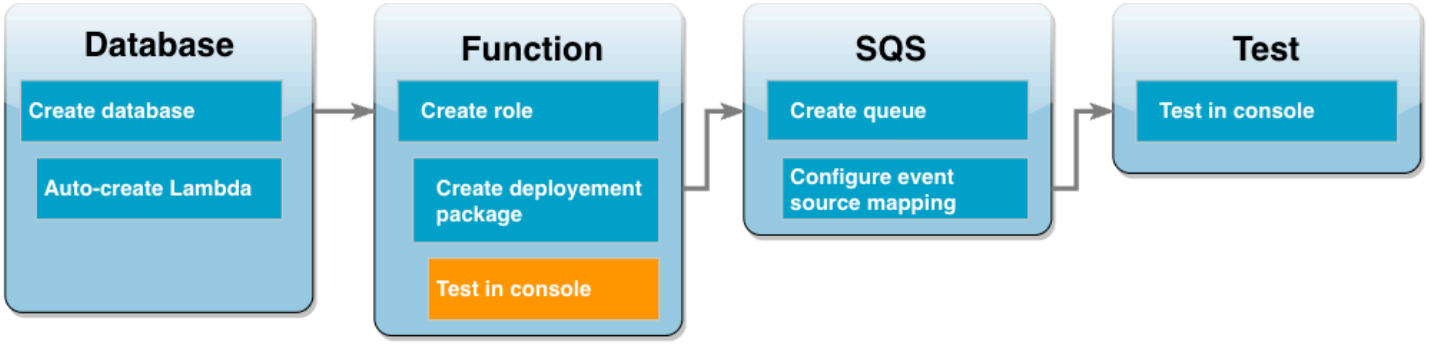
   b. Choisissez Ajouter une variable d'environnement, puis pour Clé saisissez **`DB_NAME`** et pour Valeur saisissez **`ExampleDB`**.

   c. Choisissez Ajouter une variable d'environnement, puis pour Clé saisissez **`PASSWORD`** et pour Valeur saisissez le mot de passe que vous avez choisi lors de la création de votre base de données.

   d. Choisissez Ajouter une variable d'environnement, puis, pour Clé, saisissez **`RDS_PROXY_HOST`**, et pour Valeur, saisissez le point de terminaison du proxy RDS que vous avez noté plus tôt.

   e. Choisissez Enregistrer.

Test de votre fonction Lambda dans la console



Vous pouvez désormais utiliser la console Lambda pour tester votre fonction. Vous créez un événement de test qui imite les données que votre fonction recevra lorsque vous l'invoquerez à l'aide d'Amazon SQS lors de la dernière étape du tutoriel. Votre événement de test contient un objet JSON spécifiant un ID de client et un nom de client à ajouter à la table `Customer` créée par votre fonction.

Pour tester la fonction Lambda

1. Ouvrez la page Fonctions de la console Lambda et choisissez votre fonction.
2. Choisissez la section Tester.
3. Choisissez Créer un événement et entrez **`myTestEvent`** pour le nom de l'événement.
4. Copiez le code suivant dans Event JSON et choisissez Enregistrer.

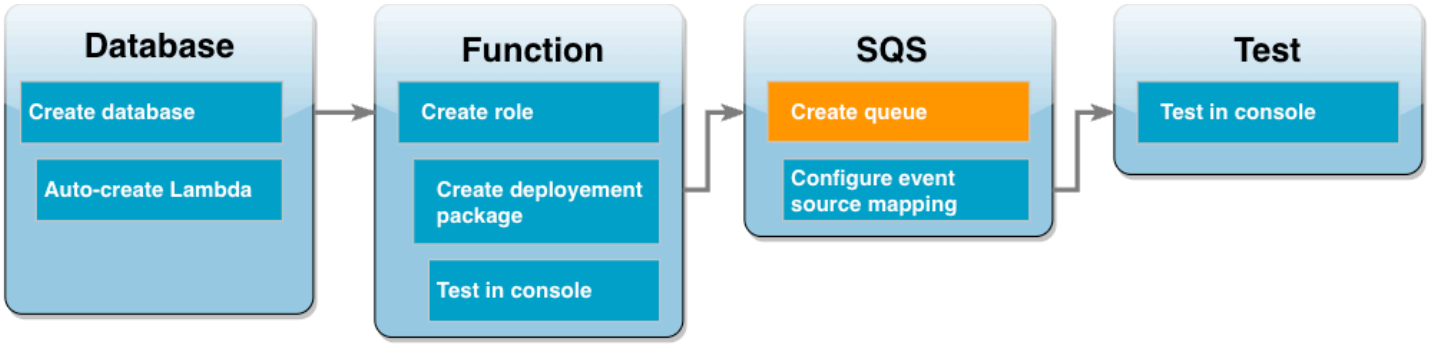
```
{
  "Records": [
    {
      "messageId": "059f36b4-87a3-44ab-83d2-661975830a7d",
      "receiptHandle": "AQEBwJnKyrHigUMZj6rYigCgxlaS3SLy0a...",
      "body": "{\n     \"CustID\": 1021,\n     \"Name\": \"Martha Rivera\"\n}",
      "attributes": {
        "ApproximateReceiveCount": "1",
        "SentTimestamp": "1545082649183",
        "SenderId": "AIDAIENQZJOLO23YVJ4VO",
        "ApproximateFirstReceiveTimestamp": "1545082649185"
      },
      "messageAttributes": {},
      "md5OfBody": "e4e68fb7bd0e697a0ae8f1bb342846b3",
      "eventSource": "aws:sqs",
      "eventSourceARN": "arn:aws:sqs:us-west-2:123456789012:my-queue",
      "awsRegion": "us-west-2"
    }
  ]
}
```

5. Sélectionnez Tester).

Dans l'onglet Résultats de l'exécution, vous devriez voir des résultats similaires aux suivants, affichés dans les journaux de fonctions :

```
[INFO] 2023-02-14T19:31:35.149Z bdd06682-00c7-4d6f-9abb-89f4bbb4a27f The following
 items have been added to the database:
[INFO] 2023-02-14T19:31:35.149Z bdd06682-00c7-4d6f-9abb-89f4bbb4a27f (1021, 'Martha
 Rivera')
```

Créez une file d'attente Amazon SQS.

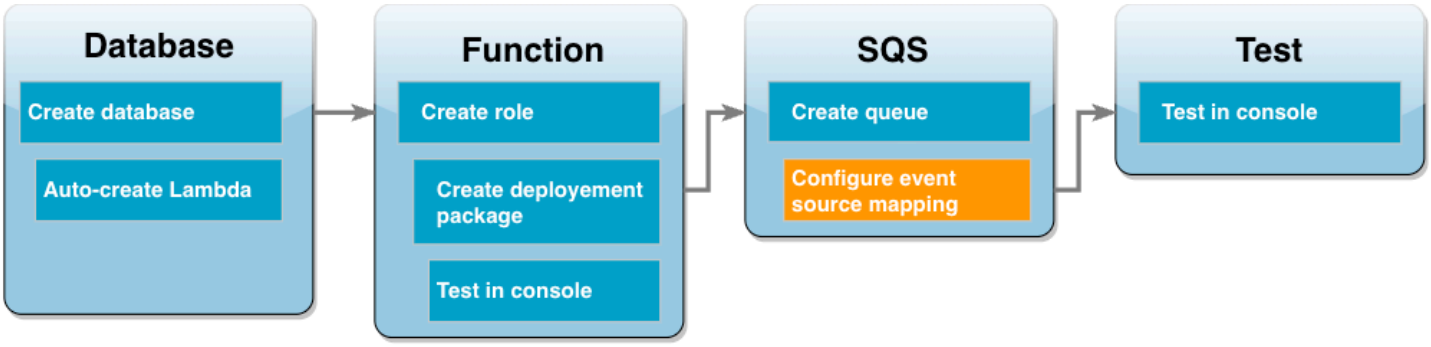


Vous avez réussi à tester l'intégration de votre fonction Lambda et de votre instance de base de données Amazon RDS. Créez maintenant la file d'attente Amazon SQS que vous utiliserez pour invoquer votre fonction Lambda lors de la dernière étape du tutoriel.

Pour créer une file d'attente Amazon SQS (console)

1. Ouvrez la page Files d'attente de la console Amazon SQS et sélectionnez Créer une file d'attente.
2. Conservez le type comme Standard et saisissez **`LambdaRDSQueue`** pour le nom de votre file d'attente.
3. Ne désélectionnez pas les options par défaut et choisissez Créer une file d'attente.

Création d'un mappage des sources d'événements pour invoquer votre fonction Lambda



Un mappage des sources d'événements est une ressource Lambda qui lit des éléments à partir d'un flux ou d'une file d'attente et invoque une fonction Lambda. Lorsque vous configurez un mappage des sources d'événements, vous pouvez spécifier une taille de lot afin que les enregistrements de votre flux ou de votre file d'attente soient regroupés dans une même charge utile. Dans cet exemple,

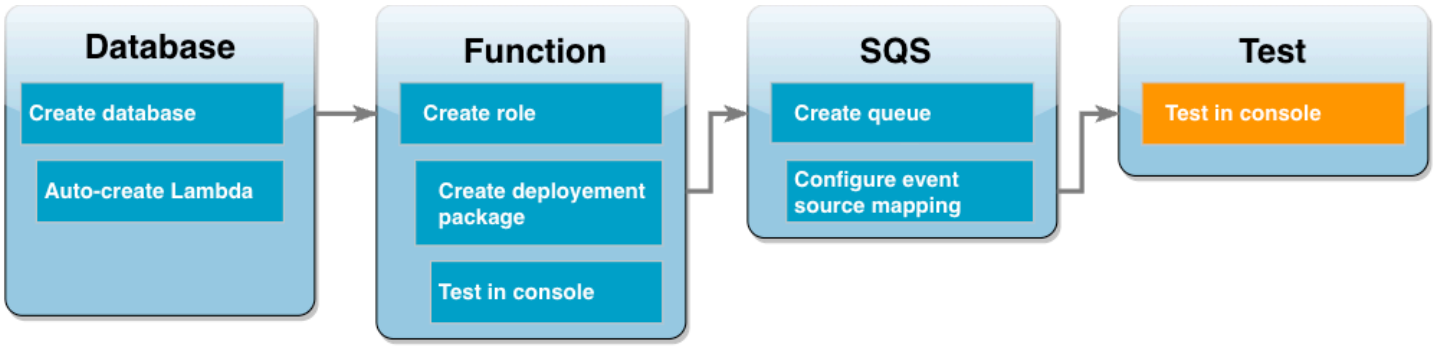
vous définissez une taille de lot sur 1 afin que votre fonction Lambda soit invoquée chaque fois que vous envoyez un message à votre file d'attente. Vous pouvez configurer le mappage des sources d'événements à l'aide de la console Lambda AWS CLI ou de la console Lambda.

Pour créer un mappage des sources d'événements (console)

1. Ouvrez la page Fonctions de la console Lambda et choisissez votre fonction (`LambdaFunctionWithRDS`).
2. Dans la section Présentation de la fonction, choisissez Ajouter un déclencheur.
3. Pour la source, sélectionnez Amazon SQS, puis sélectionnez le nom de votre file d'attente (`LambdaRDSQueue`).
4. Pour la Taille de lot, saisissez **1**.
5. Conservez les valeurs par défaut de toutes les autres options et choisissez Ajouter.

Vous pouvez désormais tester votre configuration complète en ajoutant un message à votre file d'attente Amazon SQS.

Test et surveillance de votre configuration



Pour tester votre configuration complète, ajoutez des messages à votre file d'attente Amazon SQS à l'aide de la console. Vous utilisez ensuite CloudWatch Logs pour confirmer que votre fonction Lambda écrit des enregistrements dans votre base de données comme prévu.

Pour tester et surveiller votre configuration

1. Ouvrez la page Files d'attente de la console Amazon SQS et sélectionnez votre file d'attente (`LambdaRDSQueue`).
2. Choisissez Envoyer et recevoir des messages, puis collez le code JSON suivant dans le champ Corps du message, dans la section Envoyer un message.

```
{
    "CustID": 1054,
    "Name": "Richard Roe"
}
```

3. Choisissez Send Message (Envoyer un message).

   L'envoi de votre message à la file d'attente obligera Lambda à invoquer votre fonction via le mappage des sources d'événements. Pour confirmer que Lambda a invoqué votre fonction comme prévu, utilisez CloudWatch Logs pour vérifier que la fonction a écrit le nom et l'ID du client dans votre table de base de données.
4. Ouvrez la page Groupes de journaux de la CloudWatch console et sélectionnez le groupe de journaux pour votre fonction (`/aws/lambda/LambdaFunctionWithRDS`).
5. Dans la section Flux de journaux, choisissez le flux de journaux le plus récent.

   Votre table doit contenir deux enregistrements clients, un pour chaque invocation de votre fonction. Dans le flux de journaux, vous devriez voir des messages similaires à ce qui suit :

```
[INFO] 2023-02-14T19:06:43.873Z 45368126-3eee-47f7-88ca-3086ae6d3a77 The following
 items have been added to the database:
[INFO] 2023-02-14T19:06:43.873Z 45368126-3eee-47f7-88ca-3086ae6d3a77 (1021, 'Martha
 Rivera')
[INFO] 2023-02-14T19:06:43.873Z 45368126-3eee-47f7-88ca-3086ae6d3a77 (1054,
 'Richard Roe')
```

Nettoyage de vos ressources

Vous pouvez maintenant supprimer les ressources que vous avez créées pour ce didacticiel, sauf si vous souhaitez les conserver. En supprimant AWS les ressources que vous n'utilisez plus, vous évitez des frais inutiles sur votre AWS compte.

Pour supprimer la fonction Lambda

1. Ouvrez la page Functions (Fonctions) de la console Lambda.
2. Sélectionnez la fonction que vous avez créée.
3. Sélectionnez Actions, Supprimer.
4. Sélectionnez Supprimer.

Pour supprimer le rôle d’exécution

1. Ouvrez la page Roles (Rôles) de la console IAM.
2. Sélectionnez le rôle d'exécution que vous avez créé.
3. Choisissez Supprimer le rôle.
4. Choisissez Oui, supprimer.

Pour supprimer l'instance de base de données MySQL

1. Ouvrez la page Bases de données de la console Amazon RDS.
2. Sélectionnez la base de données que vous avez créée.
3. Sélectionnez Actions, Supprimer.
4. Désactivez la case à cocher Create final snapshot (Créer un instantané final).
5. Saisissez **`delete me`** dans la zone de texte.
6. Choisissez Supprimer.

Pour supprimer la file d'attente Amazon SQS

1. Connectez-vous à la console Amazon SQS AWS Management Console et ouvrez-la à l'adresse. https://console.aws.amazon.com/sqs/
2. Sélectionnez la file d’attente que vous avez créée.
3. Choisissez Supprimer.
4. Saisissez **`delete`** dans la zone de texte.
5. Sélectionnez Supprimer.

Tutoriels Amazon RDS et exemple de code

La AWS documentation inclut plusieurs didacticiels qui vous guident à travers les cas d'utilisation courants d'Amazon RDS . La plupart de ces didacticiels vous montrent comment utiliser Amazon RDS () avec d'autres AWS services. En outre, vous pouvez accéder à un exemple de code dans GitHub.

> **Note**
>
> Vous pouvez trouver d'autres tutoriels sur le Blog AWS de base de données. Pour plus d'informations sur la formation, consultez AWS Training and Certification.

Rubriques

- Tutoriels dans ce guide
- Tutoriels dans d'autres AWS guides
- AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS
- AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS
- Tutoriels et exemples de code dans GitHub
- Utilisation de ce service avec un AWS SDK

Tutoriels dans ce guide

Les tutoriels suivants dans ce guide montrent comment exécuter les tâches courantes à l'aide d'Amazon RDS :

- Tutoriel : Création d'un VPC à utiliser avec un) IPv4

  Découvrez comment inclure une instance de bases de données dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC. Dans ce cas, le VPC partage des données avec un serveur Web qui s'exécute sur une EC2 instance Amazon du même VPC.
- Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile)

  Découvrez comment inclure une instance de bases de données dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC. Dans ce cas, le VPC partage des données avec une

EC2 instance Amazon du même VPC. Dans ce tutoriel, vous créez le VPC pour ce scénario qui fonctionne avec une base de données en mode double pile.

- Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS

  Apprenez à installer un serveur web Apache avec PHP et à créer une base de données MySQL. Le serveur Web s'exécute sur une EC2 instance Amazon utilisant Amazon Linux, et la base de données MySQL est une instance de base de données MySQL ou . L' EC2 instance Amazon et le d'instances de base de données s'exécutent dans un Amazon VPC.

- Tutoriel : Restaurer une RDS instance de base de données Amazon à partir d'un instantané de base de données

  Apprenez à restaurer une instance de bases de données à partir d'un instantané de bases de données.

- Tutoriel : Utilisation d'une fonction Lambda pour accéder à une base de données Amazon RDS

  Apprenez à créer une fonction Lambda à partir de la console RDS pour accéder à une base de données, créer une table, ajouter quelques enregistrements et extraire des enregistrements de la table. Vous apprenez également à appeler la fonction Lambda et à vérifier les résultats de la requête.

- Tutoriel : Spécifiez les instances de base de données à arrêter à l'aide de balises

  Apprenez à utiliser des balises pour préciser les instances de bases de données à arrêter.

- Tutoriel : Consigner les modifications de l'état d'une instance de base de données à l'aide EventBridge

  Découvrez comment enregistrer un changement d'état d'une instance de base de données à l'aide d'Amazon EventBridge et AWS Lambda.

- Tutoriel : Création d'une CloudWatch alarme Amazon pour le retard de réplication d'un cluster de bases de données multi-AZ pour Amazon RDS

  Découvrez comment créer une CloudWatch alarme qui envoie un message Amazon SNS lorsque le délai de réplication d'un cluster de bases de données multi-AZ dépasse un seuil. Une alarme surveille la métrique `ReplicaLag` sur la période de temps que vous spécifiez. L'action est une notification envoyée à une rubrique Amazon SNS ou à une politique Amazon EC2 Auto Scaling.

Tutoriels dans d'autres AWS guides

Les didacticiels suivants, présentés dans d'autres AWS guides, vous montrent comment effectuer des tâches courantes avec Amazon RDS () :

- Tutoriel : Rotation d'un secret pour une AWS base de données dans le guide de AWS Secrets Manager l'utilisateur

  Apprenez à créer un secret pour une AWS base de données et à configurer le secret pour qu'il alterne selon un calendrier. Vous déclenchez une rotation manuellement, puis vous vérifiez que la nouvelle version du secret continue de fournir l'accès.

- Tutoriels et exemples dans le Manuel du développeur AWS Elastic Beanstalk

  Découvrez comment déployer des applications qui utilisent des bases de données Amazon RDS avec AWS Elastic Beanstalk.

- Utilisation des données d'une base de données Amazon RDS pour créer une source de données Amazon ML dans le Amazon Machine Learning Developer Guide

  Apprenez à créer un objet de source de données Amazon Machine Learning (Amazon ML) à partir de données stockées dans une instance de bases de données MySQL.

- Activation manuelle de l'accès à une instance Amazon RDS dans un VPC dans le guide de l'utilisateur Amazon QuickSight

  Découvrez comment activer l' QuickSight accès à une instance de base de données Amazon RDS dans un VPC.

AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS

La collection suivante d'ateliers et d'autres contenus pratiques vous permet de mieux comprendre les fonctionnalités et capacités d'Amazon RDS PostgreSQL :

- Création d'une instance de base de données

  Découvrez comment créer une instance de base de données.

- Surveillance des performances avec les outils RDS

Découvrez comment utiliser AWS les outils SQL (Cloudwatch, Enhanced Monitoring, Slow Query Logs, Performance Insights, PostgreSQL Catalog Views) pour comprendre les problèmes de performances et identifier les moyens d'améliorer les performances de votre base de données.

AWS portail de contenu d'atelier et de laboratoire pour Amazon RDS

La collection suivante d'ateliers et d'autres contenus pratiques vous permet de mieux comprendre les fonctionnalités et capacités d'Amazon RDS MySQL :

- Création d'une instance de base de données

  Découvrez comment créer une instance de base de données.

- Utilisation de Performance Insights

  Découvrez comment surveiller et régler votre instance de base de données à l'aide de Performance Insights.

Tutoriels et exemples de code dans GitHub

Les didacticiels et les exemples de code suivants vous GitHub montrent comment effectuer des tâches courantes avec Amazon RDS :

- Création du dispositif de suivi d'élément Amazon Relational Database Service

  Découvrez comment créer une application qui suit et génère des rapports sur les éléments de travail. Cette application utilise Amazon RDS, Amazon Simple Email Service, Elastic Beanstalk et le kit SDK pour Java 2.x.

Utilisation de ce service avec un AWS SDK

AWS des kits de développement logiciel (SDKs) sont disponibles pour de nombreux langages de programmation courants. Chaque SDK fournit une API, des exemples de code et de la documentation qui facilitent la création d'applications par les développeurs dans leur langage préféré.

Documentation SDK	Exemples de code
AWS SDK pour C++	AWS SDK pour C++ exemples de code
AWS CLI	AWS CLI exemples de code
AWS SDK pour Go	AWS SDK pour Go exemples de code
AWS SDK pour Java	AWS SDK pour Java exemples de code
AWS SDK pour JavaScript	AWS SDK pour JavaScript exemples de code
AWS SDK pour Kotlin	AWS SDK pour Kotlin exemples de code
AWS SDK pour .NET	AWS SDK pour .NET exemples de code
AWS SDK pour PHP	AWS SDK pour PHP exemples de code
Outils AWS pour PowerShell	Outils AWS pour PowerShell exemples de code
AWS SDK pour Python (Boto3)	AWS SDK pour Python (Boto3) exemples de code
AWS SDK pour Ruby	AWS SDK pour Ruby exemples de code
Kit AWS SDK pour Rust	Kit AWS SDK pour Rust exemples de code
AWS SDK pour SAP ABAP	AWS SDK pour SAP ABAP exemples de code
Kit AWS SDK pour Swift	Kit AWS SDK pour Swift exemples de code

Pour voir des exemples spécifiques à ce service, consultez Exemples de code pour Amazon RDS utilisant AWS SDKs.

Exemple de disponibilité

Vous n'avez pas trouvé ce dont vous avez besoin ? Demandez un exemple de code en utilisant le lien Provide feedback (Fournir un commentaire) en bas de cette page.

Bonnes pratiques relatives à Amazon RDS.

Découvrez les bonnes pratiques d'utilisation de Amazon RDS. Nous mettrons à jour cette section à mesure que de nouvelles bonnes pratiques seront identifiées.

Rubriques

- Directives opérationnelles de base Amazon RDS
- Recommandations RAM d'une instance de base de données
- Maintien à jour des versions du moteur de base de données
- AWS pilotes de base de données
- Utilisation de la surveillance améliorée pour identifier les problèmes de système d'exploitation
- Utilisation des métriques pour identifier les problèmes de performances
- Réglage des requêtes
- Bonnes pratiques d'utilisation de MySQL
- Bonnes pratiques d'utilisation de MariaDB
- Bonnes pratiques d'utilisation d'Oracle
- Bonnes pratiques pour utiliser les moteurs de stockage PostgreSQL
- Bonnes pratiques pour l'utilisation de SQL Server
- Utilisation des groupes de paramètres DB
- Bonnes pratiques pour automatiser la création d'instances de base de données
- Vidéo sur les nouvelles fonctionnalités d'Amazon RDS

> **Note**
>
> Pour obtenir des recommandations communes pour Amazon RDS, consultez Recommandations d'Amazon RDS.

Directives opérationnelles de base Amazon RDS

Voici les directives opérationnelles de base que toute personne doit suivre lorsqu'elle utilise Amazon RDS. Notez que le contrat de niveau de service (SLA) Amazon RDS exige que vous suiviez ces directives :

- Utilisez des métriques pour surveiller votre mémoire, votre processeur, votre retard de réplica et votre utilisation du stockage. Vous pouvez configurer Amazon CloudWatch pour qu'il vous avertisse lorsque les habitudes d'utilisation changent ou lorsque votre déploiement approche des limites de capacité. Cela vous permet de maintenir les performances et la disponibilité du système.
- Augmentez la capacité de votre instance de base de données lorsque vous atteignez la limite de stockage. Vous devrez disposer de capacités de mémoire et de stockage supplémentaires pour vous adapter aux hausses imprévues des besoins de vos applications.
- Activez les sauvegardes automatiques et configurez-les pour qu'elles s'exécutent pendant la plus faible I/O par seconde en écriture de la journée. C'est à ce moment qu'une sauvegarde perturbe le moins l'utilisation de votre base de données.
- Si la charge de travail de votre base de données est I/O supérieure à ce que vous avez provisionné, la restauration après un basculement ou une défaillance de la base de données sera lente. Pour augmenter la I/O capacité d'une instance de base de données, effectuez l'une ou l'ensemble des opérations suivantes :
  - Migrez vers une autre classe d'instance de base de données à haute I/O capacité.
  - Passez du stockage magnétique au stockage à usage général ou au stockage à IOPS provisionnées, suivant l'augmentation dont vous avez besoin. Pour plus d'informations sur les types de stockage disponibles, consultez Types de stockage Amazon RDS.

    Si vous passez au stockage à IOPS provisionnées, veillez que vous utilisez également une classe d'instance de base de données optimisée pour les IOPS provisionnées. Pour plus d'informations sur les IOPS provisionnées, consultez Stockage SSD d'IOPS par seconde provisionnées.
  - Si vous utilisez déjà un stockage d'IOPS provisionnées, allouez un débit supplémentaire.
- Si votre application cliente met en cache les données DNS (Domain Name Service) de vos instances de base de données, définissez une valeur time-to-live (TTL) inférieure à 30 secondes. L'adresse IP sous-jacente d'une instance de base de données peut changer après un basculement. La mise en cache des données DNS pendant une période prolongée peut donc entraîner des échecs de connexion. Il se peut que votre application essaie de se connecter à une adresse IP qui n'est plus en service.
- Testez le basculement pour votre instance de base de données afin de connaître la durée du processus pour votre cas d'utilisation particulier. Le basculement permet également de veiller à ce que l'application qui accède à votre instance de base de données puisse automatiquement se connecter à la nouvelle instance de base de données suite au basculement.

Recommandations RAM d'une instance de base de données

Une bonne pratique Amazon RDS en matière de performances consiste à attribuer suffisamment de RAM pour que votre ensemble de travail réside presque totalement en mémoire. L'ensemble de travail est les données et les index fréquemment utilisés sur votre instance. Plus vous utilisez l'instance de base de données, plus l'ensemble de travail se développera.

Pour savoir si votre ensemble de travail est presque entièrement en mémoire, vérifiez la métrique ReadiOps (à l'aide d' CloudWatchAmazon) lorsque l'instance de base de données est en charge. La valeur d'I/O par seconde en lecture doit être faible et stable. Dans certains cas, l'augmentation de la classe d'instance de base de données vers une classe disposant de davantage de mémoire RAM entraîne une chute spectaculaire de ReadIOPS. Dans ces cas, votre ensemble de travail n'était pas presque entièrement en mémoire. Continuez à augmenter jusqu'à ce que la valeur d'I/O par seconde en lecture ne chute plus de façon spectaculaire suite à une opération de dimensionnement, ou qu'elle soit réduite au minimum. Pour plus d'informations sur la supervision des métriques d'une instance de base de données, consultez Afficher les métriques dans la RDS console Amazon.

Maintien à jour des versions du moteur de base de données

Mettez régulièrement à niveau la version de votre moteur de base de données pour garantir la sécurité, les performances et la conformité. Amazon RDS publie de nouvelles versions mineures et majeures qui incluent des correctifs de sécurité, des améliorations de performances et de nouvelles fonctionnalités. L'utilisation d'un moteur de base de données obsolète peut exposer vos charges de travail à des vulnérabilités connues, à des problèmes de compatibilité et à une assistance réduite de la part des fournisseurs de bases de données et de la part des AWS fournisseurs de bases de données.

Pour minimiser les perturbations, tenez compte des points suivants lorsque vous planifiez des mises à niveau :

- Test dans un environnement intermédiaire : validez la nouvelle version par rapport à votre charge de travail avant de mettre à niveau les bases de données de production.
- Utilisez les mises à niveau gérées par Amazon RDS : activez les mises à niveau automatiques des versions mineures pour faciliter l'application des correctifs.
- Planifiez les mises à niveau des versions majeures : consultez les notes de mise à jour, testez la compatibilité des applications et planifiez une fenêtre de mise à niveau contrôlée.

Des mises à niveau régulières permettent de garantir que votre base de données reste sécurisée, optimisée et conforme aux AWS meilleures pratiques.

AWS pilotes de base de données

Nous recommandons la AWS suite de pilotes pour la connectivité des applications. Les pilotes ont été conçus pour accélérer les temps de basculement et de basculement, ainsi que pour l'authentification avec AWS Secrets Manager, AWS Identity and Access Management (IAM) et l'identité fédérée. Les AWS pilotes s'appuient sur la surveillance de l'état de l'instance de base de données et sur la connaissance de la topologie de l'instance pour déterminer le nouveau rédacteur. Cette approche réduit les temps de basculement et de basculement à un chiffre, contre des dizaines de secondes pour les pilotes open source.

À mesure que de nouvelles fonctionnalités de service sont introduites, l'objectif de la AWS suite de pilotes est de fournir un support intégré pour ces fonctionnalités de service.

Pour de plus amples informations, veuillez consulter Connexion aux instances de base de données avec les AWS pilotes.

Utilisation de la surveillance améliorée pour identifier les problèmes de système d'exploitation

Lorsque la surveillance améliorée est activée, Amazon RDS fournit des métriques en temps réel pour le système d'exploitation sur lequel votre instance de base de données s'exécute. Vous pouvez afficher les métriques de votre instance de base de données à l'aide de la console. Vous pouvez également utiliser la sortie JSON Enhanced Monitoring d'Amazon CloudWatch Logs dans le système de surveillance de votre choix. Pour plus d'informations sur la surveillance améliorée, consultez la section Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Utilisation des métriques pour identifier les problèmes de performances

Pour identifier les problèmes de performances causés par des ressources insuffisantes et d'autres goulots d'étranglement courants, vous pouvez surveiller les métriques disponibles pour votre instance de base de données Amazon RDS.

Consultation des métriques de performances

Vous devez régulièrement surveiller les métriques de performances pour observer les valeurs moyennes, maximum et minimum pour différents intervalles de temps. De cette façon, vous pouvez identifier quand les performances se dégradent. Vous pouvez également définir des CloudWatch alarmes Amazon pour des seuils métriques spécifiques afin d'être alerté s'ils sont atteints.

Pour résoudre les problèmes de performances, il est important de comprendre les performances de base du système. Lorsque vous configurez une instance de base de données et que vous l'exécutez avec une charge de travail classique, capturez les valeurs moyenne, maximale et minimale de toutes les mesures de performance. Faites-le à différents intervalles (par exemple, une heure, 24 heures, une semaine, deux semaines). Cela vous permet de vous faire une idée de ce qui est normal. Cela permet de comparer l'activité pendant les heures pleines et les heures creuses. Vous pouvez ensuite utiliser ces informations pour identifier quand les performances chutent sous les niveaux standard.

Pour les clusters de bases de données multi-AZ, surveillez la différence de temps entre la dernière transaction sur l'instance de base de données de rédacteur et la dernière transaction appliquée sur une instance de base de données de lecteur. Cette différence s'appelle décalage (retard) de réplica. Pour de plus amples informations, veuillez consulter Retard de réplica et clusters de base de données multi-AZ.

Vous pouvez consulter les CloudWatch statistiques et les statistiques combinées dans le tableau de bord Performance Insights et surveiller votre instance de base de données. Si vous souhaitez utiliser cette vue de surveillance, Performance Insights doit être activé pour votre instance de base de données. Pour obtenir des informations sur cette vue de surveillance, consultez Afficher les indicateurs combinés avec le tableau de bord Performance Insights.

Vous pouvez créer un rapport d'analyse des performances pour une période spécifique et consulter les informations identifiées et les recommandations pour résoudre les problèmes. Pour plus d'informations, voir,Création d'un rapport d'analyse des performances dans Performance Insights.

Pour consulter les métriques de performances

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, sélectionnez Bases de données, choisissez une instance de base de données.
3. Choisissez Surveillance.

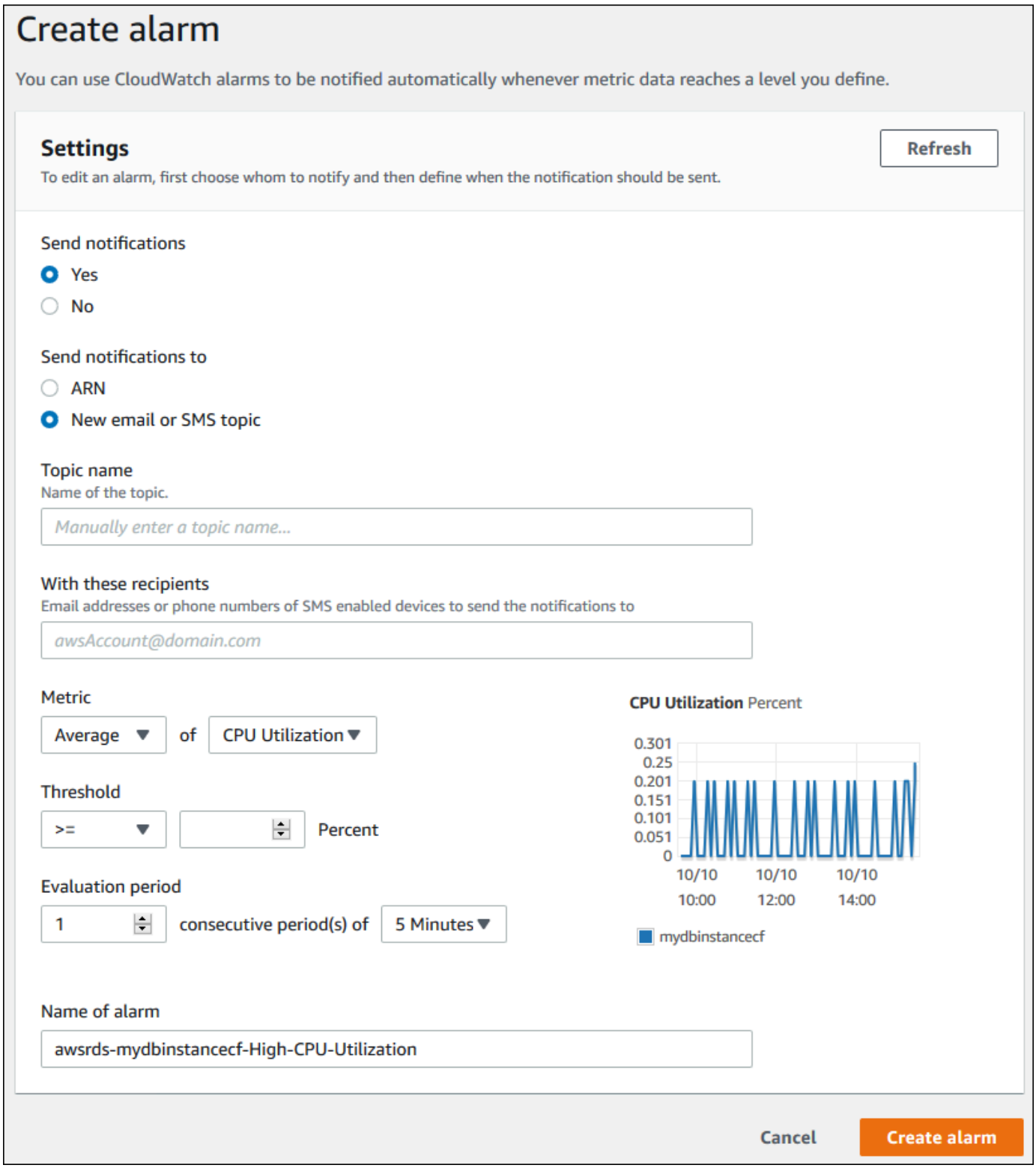
Le tableau de bord fournit les métriques de performances. Les métriques affichent par défaut les informations des trois dernières heures.

4. Utilisez les boutons numérotés dans l'angle supérieur droit de la page pour parcourir les autres métriques ou ajustez les paramètres pour consulter d'autres métriques.
5. Choisissez une métrique de performances pour régler l'intervalle de temps afin de consulter d'autres données que celles du jour même. Vous pouvez changer les valeurs Statistic (Statistique), Time Range (Plage de temps) et Period (Période) pour régler les informations affichées. Par exemple, vous pouvez consulter les valeurs maximales d'une métrique pour chaque jour des deux dernières semaines. Si tel est le cas, définissez Statistic (Statistiques) sur Maximum, Time Range (Plage de temps) sur Last 2 Weeks (Deux dernières semaines) et Period (Période) sur Day (Jour).

Vous pouvez également consulter les métriques de performances à l'aide de la CLI ou de l'API. Pour de plus amples informations, veuillez consulter Afficher les métriques dans la RDS console Amazon.

Pour régler une CloudWatch alarme

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, sélectionnez Bases de données, choisissez une instance de base de données.
3. Choisissez Logs & events (Journaux et événements).
4. Dans la section CloudWatch des alarmes, choisissez Créer une alarme.



5. Pour Envoyer des notifications, choisissez Oui et pour Envoyer des notifications à, choisissez New email or SMS topic (Nouvel e-mail ou nouvelle rubrique SMS).

6. Dans Nom de la rubrique, entrez un nom pour la notification, et pour Avec ces destinataires, entrez une liste séparée par des virgules d'adresses e-mail et de numéros de téléphone.
7. Pour Métrique, choisissez la statistique et la métrique d'alarme à définir.
8. Pour Seuil, indiquez si la métrique doit être supérieure, inférieure ou égale au seuil, et définissez la valeur du seuil.
9. Sous Evaluation period (Période d'évaluation), choisissez la période d'évaluation de l'alarme. Pour consecutive period(s) of (Période(s) consécutive(s) de), choisissez la durée pendant laquelle le seuil doit avoir été atteint pour déclencher l'alarme.
10. Pour Name of alarm (Nom de l'alarme), entrez un nom pour l'alarme.
11. Sélectionnez Create Alarm (Créer une alerte).

L'alarme apparaît dans la section des CloudWatch alarmes.

Évaluation des métriques de performances

Une instance de base de données possède un nombre de catégories différentes de métriques, et la manière de déterminer les valeurs acceptables dépend de la métrique.

CPU

- Utilisation du processeur : pourcentage de capacité de traitement informatique utilisée.

Mémoire

- Mémoire disponible : quantité de RAM disponible sur l'instance de base de données, en octets. La ligne rouge des métriques de l'onglet Monitoring est marqué à 75 % pour les métriques d'UC, de mémoire et de stockage. Si la consommation de la mémoire de l'instance franchit régulièrement cette ligne, cela indique que vous devez vérifier votre charge de travail ou mettre à niveau votre instance.
- Utilisation du swap : quantité d'espace de swap utilisée par l'instance de base de données, en octets.

Espace disque

- Espace de stockage libre : combien d'espace disque n'est pas utilisé actuellement par l'instance de base de données, en octets.

Opérations d'entrée/sortie

- E/S par seconde en lecture, E/S par seconde en écriture : le nombre moyen d'opérations de lecture ou d'écriture sur disque par seconde.
- Latence de lecture, latence d'écriture : la durée moyenne d'une opération de lecture ou d'écriture en millisecondes.
- Débit de lecture, débit d'écriture : le nombre moyen d'octets lus depuis un disque ou écrits sur un disque par seconde.
- Profondeur de la file d'attente : nombre d' I/O opérations en attente d'écriture ou de lecture sur le disque.

Trafic réseau

- Débit réseau reçu, débit réseau transmis : – la vitesse du trafic réseau vers et depuis l'instance de base de données en octets par seconde.

Connexions de la base de données

- Connexions DB : le nombre de sessions client qui sont connectées à l'instance de base de données.

Pour des descriptions individuelles plus détaillées des métriques de performances disponibles, veuillez consulter Surveillance des métriques Amazon RDS () avec Amazon CloudWatch.

En général, les valeurs acceptables pour les métriques de performances dépendent de vos données de référence et de l'activité de votre application. Enquêtez sur les écarts cohérents ou tendanciels de vos données de référence. Voici quelques conseils sur les types spécifiques de métriques :

- Forte utilisation de l'UC et de la RAM – Des valeurs importantes de l'utilisation de l'UC ou de la RAM peuvent être appropriées. Par exemple, elles peuvent être élevées si elles sont conformes aux objectifs pour votre application (comme le débit ou la simultanéité) et sont attendues.
- Utilisation de l'espace disque – Enquêtez sur l'utilisation de l'espace disque si l'espace utilisé est constamment égal ou supérieur à 85 pour cent de l'espace disque total. Voyez s'il est possible de supprimer des données de l'instance ou d'archiver des données sur un système différent pour libérer de l'espace.

- Trafic réseau – Pour le trafic réseau, discutez avec votre administrateur pour connaître le débit attendu pour votre domaine réseau et votre connexion Internet. Enquêtez sur le trafic réseau si le débit est constamment inférieur à vos attentes.
- Connexions de la base de données – Envisagez de limiter les connexions de la base de données si vous constatez un nombre important de connexions utilisateur conjointement avec une baisse des performances de l'instance et des temps de réponse. Le bon nombre de connexions utilisateur pour votre instance de base de données dépendra de votre classe d'instance et de la complexité des opérations exécutées. Pour déterminer le nombre de connexions de la base de données, associez votre instance de base de données à un groupe de paramètres. Dans ce groupe, définissez le paramètre User Connections (Connexions utilisateurs) sur une valeur autre que 0 (illimitée). Vous pouvez utiliser un groupe de paramètres existant ou en créer un nouveau. Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).
- Métriques IOPS – Les valeurs attendues pour les métriques d'IOPS dépendent de la spécification du disque et de la configuration du serveur, donc utilisez vos données de référence pour connaître les caractéristiques typiques. Enquêtez si les valeurs sont constamment différentes de vos données de référence. Pour de meilleures performances d'I/O par seconde, veillez à ce que votre ensemble de travail typique puisse être chargé en mémoire pour minimiser les opérations de lecture et écriture.

Pour les problèmes liés aux indicateurs de performance, la première étape pour améliorer les performances consiste à ajuster les requêtes les plus utilisées et les plus coûteuses. Ajustez-les pour voir si cela réduit la pression sur les ressources du système. Pour de plus amples informations, veuillez consulter Réglage des requêtes.

Si vos requêtes sont ajustées et qu'un problème persiste, pensez à mettre à niveau votre Amazon RDS Classes d'instances de base de données . Vous pouvez le mettre à niveau vers un autre avec une plus grande partie des ressources (processeur, RAM, espace disque, bande passante réseau, I/O capacité) associées au problème.

Réglage des requêtes

L'un des meilleurs moyens d'améliorer les performances d'une instance de base de données est d'ajuster les requêtes les plus communément utilisées et exigeantes en ressources. Ici, vous les ajustez pour les rendre moins onéreuses à utiliser. Pour plus d'informations sur l'amélioration des requêtes, utilisez les ressources suivantes :

- MySQL – ConsultezOptimisation des instructions SELECT dans la documentation MySQL. Pour des ressources supplémentaires de réglage des requêtes, vous pouvez également consulter MySQL Performance Tuning and Optimization Resources (Ressources d'optimisation et de réglage de performance MySQL).
- Oracle : consultez Database SQL Tuning Guide (Guide de paramétrage SQL de base de données) dans la documentation Oracle Database.
- SQL Server – Consultez Analyse d'une requête dans la documentation Microsoft. Vous pouvez également utiliser les vues de gestion des données relatives à l'exécution, à l'index et aux E/S (DMVs) décrites dans la section Vues de gestion dynamique du système de la documentation Microsoft pour résoudre les problèmes liés aux requêtes SQL Server.

  Un aspect courant du réglage de requête est la création d'index efficaces. Pour des améliorations potentielles de l'index de votre instance de base de données, consultez Assistant Paramétrage du moteur de base de données dans la documentation Microsoft. Pour plus d'informations sur l'utilisation de l'Assistant Paramétrage sur RDS for SQL Server, consultez Analyse de la charge de travail de votre base de données sur une instance de base de données Amazon RDS for SQL Server avec Database Engine Tuning Advisor.
- PostgreSQL – Consultez Using EXPLAIN (Utilisation de EXPLAIN) dans la documentation PostgreSQL pour savoir comment analyser un plan de requête. Vous pouvez utiliser ces informations pour modifier une requête ou des tables sous-jacentes afin d'améliorer les performances des requêtes.

  Pour des informations sur la façon dont vous pouvez spécifier des jointures dans votre requête afin d'améliorer les performances, consultez Controlling the planner with explicit JOIN clauses (Contrôler le planificateur avec des clauses JOIN explicites).
- MariaDB – Consultez Optimisations de requête dans la documentation MariaDB.

Bonnes pratiques d'utilisation de MySQL

La taille et le nombre des tables contenues dans une base de données MySQL peuvent tous deux nuire aux performances.

Taille des tables

En règle générale, les contraintes imposées par le système d'exploitation sur la taille des fichiers déterminent la taille maximale effective des tables des bases de données MySQL. Par conséquent, les limites ne dépendent généralement pas de contraintes internes de MySQL.

Sur une instance de base de données MySQL, évitez que les tables de votre base de données deviennent trop volumineuses. Bien que la limite du stockage général soit de 64 Tio,les limites de stockage alloué réduisent la taille maximale d'un fichier de table MySQL à 64 To. Partitionnez vos tables volumineuses pour que la taille des fichiers soit inférieure à la limite de 16 To. Cette approche peut également améliorer les performances et le temps de récupération. Pour plus d'informations, consultez Limites de taille des fichiers MySQL dans Amazon RDS.

Les tables très volumineuses (de plus de 100 Go) peuvent nuire aux performances des lectures et écritures (y compris pour les instructions DML, et en particulier pour les instructions DDL). Sur les tables volumineuses, les index peuvent considérablement améliorer les performances de sélection, mais ils peuvent également dégrader les performances des instructions DML. Les instructions DDL, telles que `ALTER TABLE`, peuvent être significativement plus lentes pour les tables volumineuses car, dans certains cas, ces opérations peuvent entraîner une reconstitution totale des tables. Ces instructions DDL peuvent verrouiller les tables pendant toute la durée de l'opération.

La quantité de mémoire requise par MySQL pour les lectures et les écritures dépend des tables impliquées dans les opérations. Il est recommandé de disposer de suffisamment de RAM pour les index des tables activement utilisées. Pour rechercher les dix tables et index les plus volumineux d'une base de données, utilisez la requête suivante :

```
select table_schema, TABLE_NAME, dat, idx from
(SELECT table_schema, TABLE_NAME,
       ( data_length ) / 1024 / 1024 as dat,
       ( index_length ) / 1024 / 1024 as idx
FROM information_schema.TABLES
order by 3 desc ) a
order by 3 desc
limit 10;
```

Nombre de tables

Votre système de fichiers sous-jacent peut imposer des limites en termes de nombre de fichiers représentant les tables. Cependant, MySQL n'a aucune limite quant au nombre de tables. Cela dit, le nombre total de tables contenues dans le moteur de stockage MySQL InnoDB peut contribuer à la dégradation des performances, quelle que soit la taille de ces tables. Pour limiter l'impact sur le système d'exploitation, vous pouvez répartir les tables entre plusieurs bases de données de la même instance de base de données MySQL. Cela limitera éventuellement le nombre de fichiers contenus dans un répertoire mais ne résoudra pas le problème global.

Toute dégradation des performances liée à la présence d'un grand nombre de tables (plus de 10 000) est due au fait que MySQL intervient sur les fichiers de stockage, notamment sur leur ouverture et leur fermeture. Pour résoudre ce problème, vous pouvez augmenter la taille des paramètres `table_open_cache` et `table_definition_cache`. L'augmentation des valeurs de ces paramètres peut toutefois augmenter la quantité de mémoire utilisée par MySQL, voire utiliser toute la mémoire disponible. Pour plus d'informations, consultez Comment MySQL ouvre et ferme les tables dans la documentation MySQL.

En outre, la présence d'un trop grand nombre de tables peut avoir un impact significatif sur le temps de démarrage de MySQL. L'arrêt et le redémarrage propres ainsi que la reprise sur incident peuvent être affectés, en particulier dans les versions antérieures à MySQL 8.0.

Nous recommandons de maintenir le nombre total de tables en dessous de 10 000 dans toutes les bases de données d'une instance de base de données. Un cas d'utilisation d'un grand nombre de tables dans une base de données MySQL est disponible dans la section Un million de tables dans MySQL 8.0.

Moteur de stockage

Les fonctionnalités de point-in-time restauration et de restauration instantanée d'Amazon RDS for MySQL nécessitent un moteur de stockage récupérable en cas de panne. Ces fonctions sont uniquement prises en charge pour le moteur de stockage InnoDB. Bien que MySQL prenne en charge plusieurs moteurs de stockage avec diverses capacités, toutes ne sont pas optimisées pour la récupération sur incident et la durabilité des données. Par exemple, le moteur de stockage MyISAM ne prend pas en charge une restauration fiable en cas de panne et peut empêcher point-in-time une restauration ou une restauration instantanée de fonctionner comme prévu. Cela peut entraîner la perte ou la corruption de données lors du redémarrage de MySQL après un incident.

InnoDB est le moteur de stockage recommandé et pris en charge pour les instances de base de données MySQL sur Amazon RDS. Les instances InnoDB peuvent également être migrées vers Aurora, au contraire des instances MySQL. Toutefois, les performances de MyISAM sont meilleures qu'InnoDB si vous avez besoin de capacités intenses de recherche en texte intégral. Si vous choisissez d'utiliser MyISAM avec Amazon RDS, suivre les étapes décrites dans Sauvegardes automatisées avec des moteurs My SQL Storage non pris en charge peut être utile dans certaines situations pour la fonctionnalité de restauration d'instantané.

Si vous souhaitez convertir des tables MyISAM existantes en tables InnoDB, vous pouvez utiliser le processus décrit dans Conversion de tables de MyISAM en InnoDB dans la documentation MySQL.

MyISAM et InnoDB ont des forces et des faiblesses différentes, vous devriez donc commencer par évaluer de façon exhaustive l'impact de ce basculement sur vos applications.

De plus, Federated Storage Engine n'est pour l'instant pas pris en charge par Amazon RDS for MySQL.

Bonnes pratiques d'utilisation de MariaDB

La taille et le nombre des tables contenues dans une base de données MariaDB peuvent tous deux nuire aux performances.

Taille des tables

En règle générale, les contraintes imposées par le système d'exploitation sur la taille des fichiers déterminent la taille maximale effective des tables des bases de données MariaDB. Par conséquent, les limites ne dépendent généralement pas de contraintes internes de MariaDB.

Sur une instance de base de données MariaDB, veillez à ce que les tables de votre base de données ne deviennent pas trop volumineuses. Bien que la limite du stockage général soit de 64 Tio, les limites de stockage alloué réduisent la taille maximale d'un fichier de table MariaDB à 16 Tio. Partitionnez vos tables volumineuses pour que la taille des fichiers soit inférieure à la limite de 16 To. Cette approche peut également améliorer les performances et le temps de récupération.

Les tables très volumineuses (de plus de 100 Go) peuvent nuire aux performances des lectures et écritures (y compris pour les instructions DML, et en particulier pour les instructions DDL). Sur les tables volumineuses, les index peuvent considérablement améliorer les performances de sélection, mais ils peuvent également dégrader les performances des instructions DML. Les instructions DDL, telles que `ALTER TABLE`, peuvent être significativement plus lentes pour les tables volumineuses car, dans certains cas, ces opérations peuvent entraîner une reconstitution totale des tables. Ces instructions DDL peuvent verrouiller les tables pendant toute la durée de l'opération.

La quantité de mémoire requise par MariaDB pour les lectures et les écritures dépend des tables impliquées dans les opérations. Il est recommandé de disposer de suffisamment de RAM pour les index des tables activement utilisées. Pour rechercher les dix tables et index les plus volumineux d'une base de données, utilisez la requête suivante :

```
select table_schema, TABLE_NAME, dat, idx from
(SELECT table_schema, TABLE_NAME,
       ( data_length ) / 1024 / 1024 as dat,
```

```
       ( index_length ) / 1024 / 1024 as idx
FROM information_schema.TABLES
order by 3 desc ) a
order by 3 desc
limit 10;
```

Nombre de tables

Votre système de fichiers sous-jacent peut imposer des limites en termes de nombre de fichiers représentant les tables. Cependant, MariaDB n'a aucune limite quant au nombre de tables. Cela dit, le nombre total de tables contenues dans le moteur de stockage MariaDB InnoDB peut contribuer à la dégradation des performances, quelle que soit la taille de ces tables. Pour limiter l'impact sur le système d'exploitation, vous pouvez répartir les tables entre plusieurs bases de données de la même instance de base de données MariaDB. Cela limitera éventuellement le nombre de fichiers contenus dans un répertoire mais ne résoudra pas le problème global.

Toute dégradation des performances liée à la présence d'un grand nombre de tables (plus de 10 000) est due au fait que MariaDB intervient sur les fichiers de stockage. Ce travail inclut l'ouverture et la fermeture de fichiers de stockage par MariaDB. Pour résoudre ce problème, vous pouvez augmenter la taille des paramètres `table_open_cache` et `table_definition_cache`. L'augmentation des valeurs de ces paramètres peut toutefois augmenter la quantité de mémoire utilisée par MariaDB. Elle peut même utiliser toute la mémoire disponible. Pour plus d'informations, consultez Optimisation de table_open_cache dans la documentation MariaDB.

En outre, la présence d'un trop grand nombre de tables peut avoir un impact significatif sur le temps de démarrage de MariaDB. L'arrêt et le redémarrage propres ainsi que la reprise sur incident peuvent être affectés. Nous recommandons de maintenir le nombre total de tables en dessous de dix mille dans toutes les bases de données d'une instance de base de données.

Moteur de stockage

Les fonctionnalités de point-in-time restauration et de restauration de snapshots d'Amazon RDS pour MariaDB nécessitent un moteur de stockage récupérable en cas de panne. Bien que MariaDB prenne en charge plusieurs moteurs de stockage avec diverses capacités, toutes ne sont pas optimisées pour la récupération sur incident et la durabilité des données. Par exemple, bien qu'Aria remplace MyISAM en toute sécurité, il peut tout de même empêcher une point-in-time restauration ou une restauration instantanée de fonctionner comme prévu. Cela peut entraîner la perte ou la corruption de données lors du redémarrage de MariaDB après un incident. InnoDB est le moteur de stockage recommandé et pris en charge pour les instances de base de données MariaDB sur Amazon RDS.

Si vous choisissez d'utiliser Aria avec Amazon RDS, suivre les étapes décrites dans Sauvegardes automatiques avec moteurs de stockage MariaDB non pris en charge peut être utile dans certaines situations pour la fonctionnalité de restauration d'instantané.

Si vous souhaitez convertir des tables MyISAM existantes en tables InnoDB, vous pouvez utiliser le processus décrit dans Conversion de tables de MyISAM en InnoDB dans la documentation MariaDB. MyISAM et InnoDB ont des forces et des faiblesses différentes, vous devriez donc commencer par évaluer de façon exhaustive l'impact de ce basculement sur vos applications.

Bonnes pratiques d'utilisation d'Oracle

Pour de plus amples informations sur les bonnes pratiques d'utilisation de Amazon RDS for Oracle, veuillez consulter les Bonnes pratiques pour l'exécution d'une base de données Oracle sur Amazon Web Services.

Un atelier AWS virtuel organisé en 2020 comprenait une présentation sur l'exécution de bases de données Oracle de production sur Amazon RDS. Une vidéo de la présentation est disponible ici.

Bonnes pratiques pour utiliser les moteurs de stockage PostgreSQL

Parmi les deux principaux domaines dans lesquels vous pouvez améliorer les performances avec RDS for PostgreSQL, l'un est lorsque vous chargez des données dans une instance de base de données. Une autre utilisation est lorsque vous utilisez la fonction autovacuum de PostgreSQL. Les sections suivantes couvrent certaines pratiques que nous recommandons pour ces domaines en particulier.

Pour plus d’informations sur la façon dont Amazon RDS met en œuvre d’autres tâches DBA courantes pour PostgreSQL, consultez Tâches courantes d'administration de bases de données pour Amazon RDS for PostgreSQL.

Chargement des données dans une instance de base de données PostgreSQL

Lors du chargement des données dans une instance de base de données Amazon RDS for PostgreSQL, modifiez vos paramètres d'instance de base de données et vos valeurs de groupe de paramètres de base de données. Définissez-les pour permettre l'importation de données la plus efficace possible dans votre instance de base de données.

Modifiez les paramètres de l'instance de base de données comme suit :

- Désactivez les sauvegardes de l'instance de base de données (affectez la valeur 0 à backup_retention)
- Désactivez le mode multi-AZ

Modifiez votre groupe de paramètres DB pour inclure les paramètres suivants. Testez également les réglages des paramètres pour déterminer les réglages les plus efficaces pour votre instance de base de données.

- Augmentez la valeur du paramètre `maintenance_work_mem`. Pour plus d'informations sur les paramètres de consommation de ressources PostgreSQL, consultez la documentation PostgreSQL.
- Augmentez la valeur des paramètres `max_wal_size` et `checkpoint_timeout` pour réduire le nombre d'écritures dans le journal d'écriture anticipée (WAL).
- Désactivez le paramètre `synchronous_commit`.
- Désactivez le paramètre autovacuum de PostgreSQL.
- Assurez-vous qu'aucune des tables que vous importez n'est pas journalisée. Les données stockées dans les tables non journalisées peuvent être perdues lors d'un basculement. Pour de plus amples informations, consultez CREATE TABLE UNLOGGED (CRÉER UNE TABLE NON JOURNALISÉE).

Utilisez les commandes `pg_dump -Fc` (compressé) ou `pg_restore -j` (parallèle) avec ces paramètres.

Une fois l'opération de chargement terminée, réinitialisez votre instance de base de données et les paramètres de base de données à leurs paramètres normaux.

Utilisation de la fonction autovacuum de PostgreSQL

La fonction autovacuum pour les bases de données PostgreSQL est une fonction que nous vous recommandons vivement d'utiliser pour maintenir l'état de votre instance de bases de données PostgreSQL. Autovacuum automatise l'exécution des commandes VACUUM et ANALYZE ; son utilisation est exigée par PostgreSQL, non imposée par Amazon RDS et essentielle pour garantir de bonnes performances. La fonction est activée par défaut pour toutes les nouvelles instances de bases de données Amazon RDS for PostgreSQL, et les paramètres de configuration associés sont configurés par défaut de manière appropriée.

Votre administrateur de base de données doit connaître et comprendre cette opération de maintenance. Pour accéder à la documentation PostgreSQL sur la fonction autovacuum, consultez The Autovacuum Daemon (Le démon d'autovacuum).

Autovacuum n'est pas une opération « sans utilisation de ressources », mais elle fonctionne en arrière-plan et profite autant que possible aux opérations utilisateur. Lorsqu'elle est activée, la fonction autovacuum vérifie les tables ayant eu un grand nombre de tuples mis à jour ou supprimés. Elle protège également contre la perte de données très anciennes due au renvoi à la ligne de l'ID de transaction. Pour de plus amples informations, veuillez consulter Preventing Transaction ID Wraparound Failures.

Autovacuum ne doit pas être considérée comme une opération aux frais généraux élevés qui peut être réduite pour améliorer les performances. Au contraire, les tables qui mettent à jour et suppriment très rapidement se détériorent vite avec le temps si la fonction autovacuum n'est pas exécutée.

> **Important**
>
> La non-exécution de la fonction autovacuum peut entraîner un arrêt obligatoire ultérieur pour effectuer une opération de nettoyage bien plus intrusive. Dans certains cas, une instance de base de données RDS for PostgreSQL peut devenir indisponible en raison d'une utilisation trop prudente de l'autovacuum. Dans ces cas, la base de données PostgreSQL s'arrête pour se protéger. À ce stade, Amazon RDS doit effectuer un nettoyage single-user-mode complet directement sur l'instance de base de données. Ce vacuum complet peut entraîner une panne de plusieurs heures. Nous vous recommandons donc vivement de ne pas désactiver la fonction autovacuum, qui est activée par défaut.

Les paramètres d'autovacuum déterminent sa fréquence et son intensité de travail. Les paramètres`autovacuum_vacuum_threshold` et `autovacuum_vacuum_scale_factor` déterminent le moment où la fonction autovacuum est exécutée. Les paramètres `autovacuum_max_workers`, `autovacuum_nap_time`, `autovacuum_cost_limit` et `autovacuum_cost_delay` déterminent l'intensité de travail d'autovacuum. Pour de plus amples informations sur la fonction autovacuum, son exécution et les paramètres obligatoires, consultez la Routine Vacuuming dans la documentation PostgreSQL.

La requête suivante indique le nombre de tuples « inactifs » dans une table appelée table1 :

```
SELECT relname, n_dead_tup, last_vacuum, last_autovacuum FROM
```

```
pg_catalog.pg_stat_all_tables
WHERE n_dead_tup > 0 and relname = 'table1';
```

Les résultats de la requête ressembleront à l'exemple ci-dessous :

```
relname | n_dead_tup | last_vacuum | last_autovacuum
---------+------------+-------------+-----------------
 tasks   |   81430522 |             |
(1 row)
```

Vidéo des bonnes pratiques Amazon RDS for PostgreSQL

La conférence AWS re:Invent 2020 comprenait une présentation sur les nouvelles fonctionnalités et les meilleures pratiques d'utilisation de PostgreSQL sur Amazon RDS. Une vidéo de la présentation est disponible ici.

Bonnes pratiques pour l'utilisation de SQL Server

Les bonnes pratiques pour un déploiement multi-AZ avec une instance de base de données SQL Server sont les suivantes :

- Utilisez les événements de base de données Amazon RDS pour surveiller les basculements. Par exemple, vous pouvez être notifié par sms ou e-mail en cas de basculement d'une instance de base de données. Pour de plus amples informations sur les événements Amazon RDS, veuillez consulte Utilisation des notifications d'RDSévénements Amazon.
- Si votre application met en cache des valeurs DNS, configurez leatime-to-live (TTL) à moins de 30 secondes. La configuration de la durée de vie est une bonne pratique en tant que tel en cas de basculement. Lors d'un basculement, l'adresse IP peut changer et la valeur mise en cache peut ne plus être en service.
- Nous vous recommandons de ne pas activer les modes suivants car ils désactivent la journalisation des transactions, qui est obligatoire pour le déploiement multi-AZ :
  - Mode de récupération simple
  - Mode hors ligne
  - Mode lecture seule
- Testez pour déterminer combien de temps votre instance de base de données met-elle pour basculer. Les délais de basculement peuvent varier en raison du type de base de données, la

classe d'instance et le type de stockage utilisé. Vous devez également tester la capacité de votre application à continuer de fonctionner en cas de basculement.

- Pour raccourcir les délais de basculement, procédez comme suit :
  - Veillez à avoir suffisamment d'IOPS provisionnées allouées pour votre charge de travail. InadéquatI/O can lengthen failover times. Database recovery requires I/O.
  - Utilisez de plus petites transactions. La récupération de la base de données repose sur les transactions, donc si vous pouvez séparer d'importantes transactions en plusieurs transactions plus petites, vos délais de basculement devraient diminuer.
- Pensez que lors d'un basculement, les temps de latence sont élevés. Dans le cadre d'un processus de basculement, Amazon RDS réplique automatiquement vos données vers une nouvelle instance de secours. Cette réplication signifie que de nouvelles données sont transférées vers deux instances de base de données différentes. Il peut donc y avoir une certaine latence jusqu'à ce que l'instance de base de données de secours rattrape la nouvelle instance de base de données principale.
- Déployez vos applications dans toutes les zones de disponibilité. Si une zone de disponibilité tombe en panne, vos applications qui se trouvent dans les autres zones de disponibilité seront toujours disponibles.

Lorsque vous travaillez avec un déploiement multi-AZ de SQL Server, rappelez-vous qu'Amazon RDS crée des réplicas pour toutes les bases de données SQL Server sur votre instance. Si vous ne souhaitez pas que des bases de données spécifiques aient des réplicas secondaires, configurez une instance de base de données séparée qui n'utilise pas de déploiement multi-AZ pour ces bases de données.

Vidéo des bonnes pratiques Amazon RDS pour SQL Server

La conférence AWS re:Invent 2019 comprenait une présentation sur les nouvelles fonctionnalités et les meilleures pratiques relatives à l'utilisation de SQL Server sur Amazon RDS. Une vidéo de la présentation est disponible ici.

Utilisation des groupes de paramètres DB

Nous vous recommandons de tester les modifications apportées aux groupes de paramètres de base de données sur une instance de base de données test avant d'appliquer ces modifications à vos instances de base de données de production. La configuration incorrecte de paramètres de moteur DB dans un groupe de paramètres de base de données peut avoir des effets contraires involontaires,

notamment une dégradation de la performance et une instabilité du système. Montrez-vous toujours prudent lorsque vous modifiez des paramètres de moteur DB et sauvegardez votre instance de base de données avant de modifier un groupe de paramètres de base de données.

Pour plus d'informations sur la sauvegarde de votre instance de base de données, consultez Sauvegarde, restauration et exportation de données.

Bonnes pratiques pour automatiser la création d'instances de base de données

Une des bonnes pratiques de Amazon RDS consiste à créer une instance de base de données avec la version mineure préférée du moteur de base de données. Vous pouvez utiliser l' AWS CLI API Amazon RDS ou automatiser la création AWS CloudFormation d'instances de base de données. Lorsque vous utilisez ces méthodes, vous pouvez spécifier uniquement la version principale et Amazon RDS crée automatiquement l'instance avec la version mineure préférée. Par exemple, si PostgreSQL 12.5 est la version mineure préférée, et que vous spécifiez la version 12 avec `create-db-instance`, l'instance de base de données sera à la version 12.5.

Pour déterminer la version mineure préférée, vous pouvez exécuter la commande `describe-db-engine-versions` avec l'option `--default-only`, comme illustré dans l'exemple suivant.

```
aws rds describe-db-engine-versions --default-only --engine postgres

{
    "DBEngineVersions": [
        {
            "Engine": "postgres",
            "EngineVersion": "12.5",
            "DBParameterGroupFamily": "postgres12",
            "DBEngineDescription": "PostgreSQL",
            "DBEngineVersionDescription": "PostgreSQL 12.5-R1",
            ...some output truncated...
        }
    ]
}
```

Pour plus d'informations sur la création d'instances de base de données par programmation, consultez les ressources suivantes :

- En utilisant le AWS CLI — create-db-instance

- Utilisation de l'API Amazon RDS — Create DBInstance
- Utilisation de AWS CloudFormation — AWS : :RDS : : DBInstance

Vidéo sur les nouvelles fonctionnalités d'Amazon RDS

La conférence AWS re:Invent 2023 comprenait une présentation sur les nouvelles fonctionnalités d'Amazon RDS. Une vidéo de la présentation est disponible ici.

Accès programmatique à Amazon RDS Amazon

Amazon RDS met à votre disposition les outils suivants pour gérer vos ressources RDSAmazon de manière programmatique.

AWS Command Line Interface (AWS CLI)

Vous pouvez créer et gérer vos ressources RDS à l'aide de l' AWS CLI interface de ligne de commande. AWS CLI Fournit un accès direct APIs aux quatre AWS services, tels qu'AmazonRDS. Pour obtenir la syntaxe et des exemples de commandes pour AmazonRDS, consultez rdsdans la référence de AWS CLI commande.

AWS CloudFormation

Avec cet outil AWS d'infrastructure en tant que code (IaC), vous pouvez créer des modèles décrivant toutes les ressources RDSAmazon que vous souhaitez, puis AWS CloudFormation provisionner et configurer ces ressources pour vous. Pour de plus amples informations, veuillez consulter the section called “Création de ressources avec AWS CloudFormation”.

AWS kits de développement logiciel (SDKs)

AWS fournit SDKs de nombreuses technologies et langages de programmation populaires. Ils vous permettent d'appeler plus facilement AWS des services depuis vos applications dans ce langage ou cette technologie. Pour plus d'informations à ce sujetSDKs, consultez la section Outils de développement et de gestion d'applications sur AWS.

Amazon RDS API

Il s'APIagit de l'interface au niveau du protocole pour Amazon. RDS Lorsque vous l'utilisezAPI, vous devez formater correctement chaque HTTPS demande et ajouter une signature numérique valide à chaque demande. Pour de plus amples informations, veuillez consulter Référence d'API Amazon RDS.

Console-to-Code

Avec cet outil, vous pouvez générer du code pour les actions que vous effectuez dans la RDS console Amazon et utiliser ce code dans d'autres outils tels que AWS CloudFormation. Pour de plus amples informations, veuillez consulter the section called “Console-to-Code”.

Console-to-CodeÀ utiliser pour générer du code pour les actions de votre console RDSAmazon

La console fournit un parcours guidé pour créer des ressources et tester des prototypes. Si vous souhaitez créer les mêmes ressources à grande échelle, vous aurez besoin d'un code d'automatisation. Console-to-Codeest une fonctionnalité d'Amazon Q Developer qui peut vous aider à démarrer avec votre code d'automatisation. Console-to-Codeenregistre les actions de votre console, y compris les valeurs par défaut et les valeurs de paramètres que vous fournissez. Il utilise ensuite l'IA générative pour suggérer du code dans la langue et le format de votre choix pour les actions que vous choisissez. Étant donné que le flux de travail de la console garantit que les valeurs de paramètres que vous spécifiez sont valides ensemble, le code que vous générez à l'aide de ces valeurs Console-to-Code possède des valeurs de paramètres compatibles. Vous pouvez utiliser le code comme point de départ, puis le personnaliser pour qu'il soit prêt pour la production en fonction de votre cas d'utilisation spécifique.

Par exemple, avec Console-to-Code, vous pouvez enregistrer la création d'un d'instance de base de données et choisir de générer du code au AWS CloudFormation JSON format. Vous pouvez ensuite copier ce code et le personnaliser pour l'utiliser dans votre AWS CloudFormation modèle.

Console-to-Code peut actuellement générer infrastructure-as-code (iAc) dans les langues et formats suivants :

- CDKJava
- CDKPython
- CDK TypeScript
- CloudFormation JSON
- CloudFormation YAML

Pour plus d'informations et des instructions d'utilisation Console-to-Code, consultez Automatiser les AWS services avec Amazon Q Developer Console-to-Code dans le guide de l'utilisateur Amazon Q Developer.

Configuration d'une RDS instance de base de données Amazon

Cette section explique comment configurer votre RDS instance de base de données Amazon. Avant de créer une instance de base de données, choisissez la classe d'instance de base de données qui exécutera l'instance de base de données. Décidez également où l'instance de base de données sera exécutée en choisissant une AWS région. Créez ensuite l'instance de base de données.

Vous pouvez configurer une instance de base de données avec un groupe d'options et un groupe de paramètres de base de données.

- Un groupe d'options spécifie les fonctionnalités, appelées options, disponibles pour une RDS instance de base de données Amazon particulière.
- Un groupe de paramètres de base de données sert de conteneur pour les valeurs de configuration du moteur qui sont appliquées à une ou plusieurs instances de base de données.

Les options et paramètres disponibles dépendent du moteur de base de données et de la version du moteur de base de données. Vous pouvez spécifier un groupe d'options et un groupe de paramètres de base de données lorsque vous créez une instance de base de données. Vous pouvez également modifier une instance de base de données pour les spécifier.

Rubriques

- Création d'une instance de base de données Amazon RDS
- Création de ressources RDSAmazon avec AWS CloudFormation
- Connexion à une RDS instance de base de données Amazon
- Utilisation de groupes d'options
- Groupes de paramètres pour (Amazon RDS)
- Création d'un ElastiCache cache Amazon à l'aide des paramètres de l'instance Amazon RDS DB
- Migration automatique des EC2 bases de données vers) à l'aide RDS de AWS Database Migration Service
- Tutoriel : Création d'un de base de données MySQL avec un paramètre personnalisé et un nouveau groupe d'options

Création d'une instance de base de données Amazon RDS

La composante de base de Amazon RDS est l'instance de base de données dans laquelle vous créez vos bases de données. Vous choisissez les caractéristiques propres au moteur de l'instance de base de données lorsque vous la créez. Vous choisissez également la capacité de stockage, le processeur, la mémoire, etc. de l' AWS instance sur laquelle le serveur de base de données s'exécute.

Rubriques

- Prérequis pour l'instance de base de données
- Création d'une instance de base de données
- Paramètres des instances de base de données

Prérequis pour l'instance de base de données

> **Important**
>
> Avant de pouvoir créer une instance de base de données Amazon RDS, effectuez les tâches indiquées dans Configuration de votre environnement Amazon RDS.

Les conditions préalables suivantes sont requises pour créer une instance de base de données.

Rubriques

- Configurer le réseau pour l'instance de la base de données
- Prérequis supplémentaires

Configurer le réseau pour l'instance de la base de données

Vous ne pouvez créer une instance de base de données Amazon RDS que dans un cloud privé virtuel (VPC) basé sur un service Amazon VPC. En outre, il doit se trouver dans une zone Région AWS comportant au moins deux zones de disponibilité. Le groupe de sous-réseaux de base de données que vous choisissez pour l'instance de bases de données doit couvrir au moins deux zones de disponibilité. Cette configuration vous permet de configurer un déploiement Multi-AZ lorsque vous créez l'instance de base de données ou de passer facilement à un déploiement ultérieur.

Pour configurer la connectivité entre votre nouvelle instance de base de données et une EC2 instance Amazon dans le même VPC, faites-le lors de la création de l'instance de base de données. Pour vous connecter à votre instance de base de données à partir de ressources autres que les EC2 instances du même VPC, configurez les connexions réseau manuellement.

Rubriques

- Configuration de la connectivité réseau automatique avec une EC2 instance
- Configuration manuelle du réseau

Configuration de la connectivité réseau automatique avec une EC2 instance

Lorsque vous créez une instance de base de données RDS, vous pouvez utiliser le AWS Management Console pour configurer la connectivité entre une EC2 instance et la nouvelle instance de base de données. Dans ce cas, RDS configure automatiquement votre VPC et vos paramètres réseau. L'instance de base de données est créée dans le même VPC que l' EC2 instance afin que l'instance puisse accéder à l' EC2 instance de base de données.

Les conditions requises pour connecter une EC2 instance à l'instance de base de données sont les suivantes :

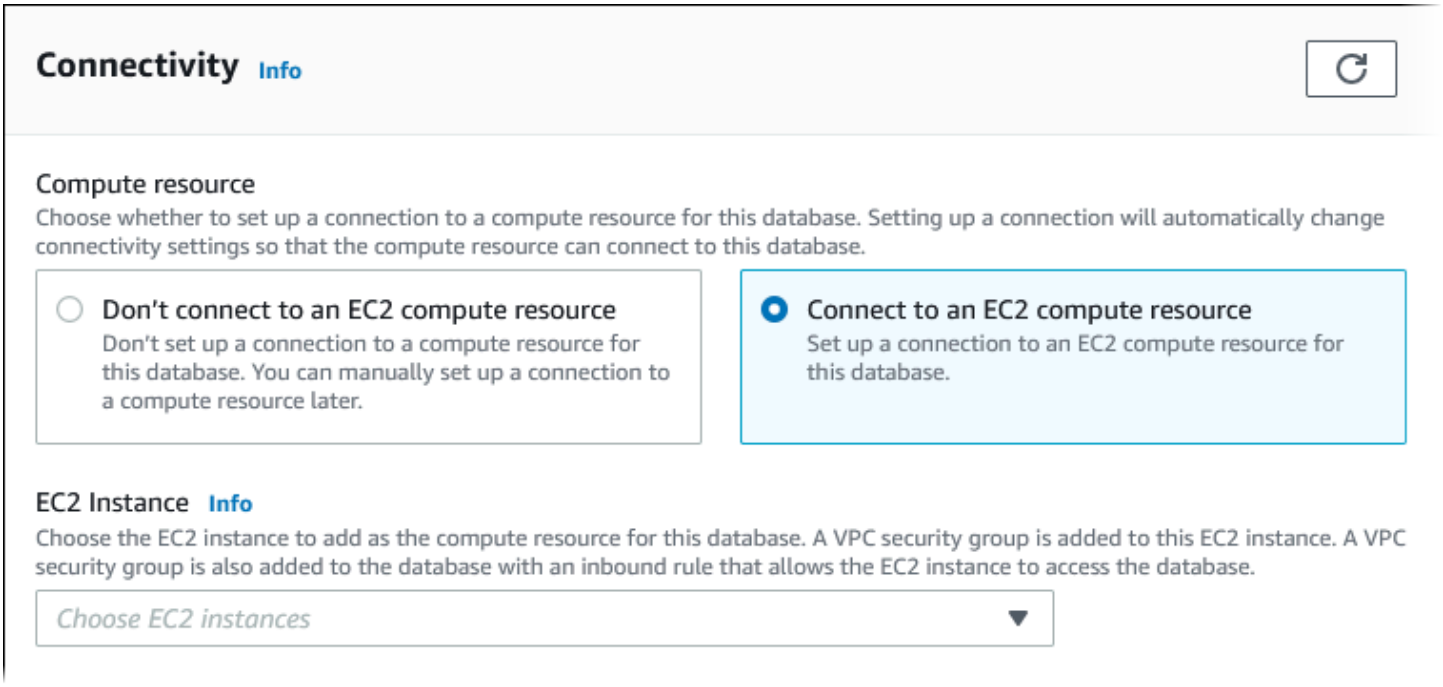
- L' EC2 instance doit exister dans le Région AWS avant de créer l'instance de base de données.

  Si aucune EC2 instance n'existe dans le Région AWS, la console fournit un lien pour en créer une.
- L'utilisateur qui crée l'instance de base de données doit avoir les autorisations nécessaires pour effectuer les opérations suivantes :
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:AuthorizeSecurityGroupIngress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSubnet`
  - `ec2:CreateSecurityGroup`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`
  - `ec2:DescribeSecurityGroups`

- `ec2:DescribeSubnets`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:RevokeSecurityGroupEgress`

Cette option permet de créer une instance de base de données privée. L'instance de base de données utilise un groupe de sous-réseaux de base de données avec uniquement des sous-réseaux privés pour restreindre l'accès aux ressources au sein du VPC.

Pour connecter une EC2 instance à l'instance de base de données, choisissez Se connecter à une ressource de EC2 calcul dans la section Connectivité de la page Créer une base de données.



Lorsque vous choisissez Se connecter à une ressource de EC2 calcul, RDS définit automatiquement les options suivantes. Vous ne pouvez pas modifier ces paramètres sauf si vous choisissez de ne pas configurer la connectivité avec une EC2 instance en choisissant Ne pas vous connecter à une ressource de EC2 calcul.

Option console	Réglage automatique
Network type (Type de réseau)	RDS définit le type de réseau sur IPv4. Actuellement, le mode double pile n'est pas pris en charge lorsque vous configurez

Option console	Réglage automatique
	une connexion entre une EC2 instance et l'instance de base de données.
Virtual Private Cloud (VPC)	RDS définit le VPC sur celui associé EC2 à l'instance.
Groupe de sous-réseaux de base de données	RDS nécessite un groupe de sous-réseaux de base de données avec un sous-réseau privé dans la même zone de disponibilité que l'instance. EC2 Si un groupe de sous-réseau de base de données répondant à cette exigence existe, RDS utilise alors le groupe de sous-réseau de base de données existant. Par défaut, cette option est définie sur Automatic setup (Configur ation automatique). Lorsque vous choisissez Automatic setup (Configuration automatique) et qu'aucun groupe de sous-réseaux de base de données ne répond à cette exigence, l'action suivante se produit. RDS utilise trois sous-réseaux privés disponibles dans trois zones de disponibilité où l'une des zones de disponibilité est identique à l'instance. EC2 Si un sous-réseau privé n'est pas disponible dans une zone de disponibilité, RDS crée un sous-réseau privé dans la zone de disponibilité. RDS crée ensuite le groupe de sous-réseau de base de données. Lorsqu'un sous-réseau privé est disponible, RDS utilise la table de routage qui lui est associée avec le sous-réseau et ajoute les sous-réseaux qu'il crée à cette table de routage. Lorsqu'au cun sous-réseau privé n'est disponible, RDS crée une table de routage sans accès à la passerelle Internet et ajoute les sous-réseaux qu'il crée à la table de routage. RDS vous permet également d'utiliser des groupes de sous-réseaux de base de données existants. Sélectionnez Choose existing (Choisir existants) si vous souhaitez utiliser un groupe de sous-réseaux de base de données existant de votre choix.

Option console	Réglage automatique
Accès public	RDS choisit No (Non) pour que l'instance de base de données ne soit pas publiquement accessible. Pour des raisons de sécurité, il est préférable de garder la base de données privée et de s'assurer qu'elle n'est pas accessible depuis Internet.
VPC security group (firewall) [Groupe de sécurité VPC (pare-feu)]	RDS crée un nouveau groupe de sécurité qui est associé à l'instance de base de données. Le groupe de sécurité est nommé `rds-ec2-`*n*, où *n* est un nombre. Ce groupe de sécurité inclut une règle entrante dont le groupe de sécurité EC2 VPC (pare-feu) est la source. Ce groupe de sécurité associé à l'instance de base de données permet à l' EC2instan ce d'accéder à l'instance de base de données. RDS crée également un nouveau groupe de sécurité associé à l' EC2 instance. Le groupe de sécurité est nommé `ec2-rds-`*n*, où *n* est un nombre. Ce groupe de sécurité comprend une règle de sortie avec le groupe de sécurité VPC de l'instance de base de données comme source. Ce groupe de sécurité permet à l' EC2 instance d'envoyer du trafic vers l'instance de base de données. Vous pouvez ajouter un autre groupe de sécurité en sélection nant Create new (Créer nouveau) et en saisissant le nom du nouveau groupe de sécurité. Vous pouvez ajouter des groupes de sécurité existants en choisissant Choose existing (Choisir existant) et en sélection nant les groupes de sécurité à ajouter.

Option console	Réglage automatique
Zone de disponibilité	Lorsque vous choisissez une instance de base de données unique dans Disponibilité et durabilité (déploiement mono-AZ), RDS choisit la zone de disponibilité de l' EC2 instance. Lorsque vous choisissez une instance de base de données multi-AZ dans Disponibilité et durabilité (déploiement d'instance de base de données multi-AZ), RDS choisit la zone de disponibilité de l' EC2 instance pour une instance de base de données dans le déploiement. RDS choisit de manière aléatoire une zone de disponibilité différente pour l'autre instance de base de données. L'instance de base de données principale ou la réplique de secours est créée dans la même zone de disponibilité que l' EC2instance. Lorsque vous choisissez une instance de base de données multi-AZ, il est possible que des coûts entre zones de disponibilité soient facturés si l'instance de base de données et l' EC2 instance se trouvent dans des zones de disponibilité différentes.

Pour plus d’informations sur ces paramètres, consultez la page Paramètres des instances de base de données.

Si vous modifiez ces paramètres après la création de l'instance de base de données, les modifications peuvent affecter la connexion entre l' EC2 instance et l'instance de base de données.

Configuration manuelle du réseau

Pour vous connecter à votre instance de base de données à partir de ressources autres que les EC2 instances du même VPC, configurez les connexions réseau manuellement. Si vous utilisez le AWS Management Console pour créer votre instance de base de données, Amazon RDS peut créer automatiquement un VPC pour vous. Une autre solution consiste à utiliser un VPC existant ou à créer un VPC pour votre instance de base de données. Quelle que soit l'approche adoptée, votre VPC doit comporter au moins un sous-réseau dans chacune d'au moins deux zones de disponibilité pour que vous puissiez l'utiliser avec une instance de base de données RDS.

Par défaut, Amazon RDS crée l'instance de base de données et la zone de disponibilité automatiquement pour vous. Pour choisir une zone de disponibilité spécifique, vous devez changer le paramètre Availability & durability (Disponibilité et durabilité) en Single DB instance (Instance de base de données unique). Ce faisant, vous exposez un paramètre de zone de disponibilité qui vous permet de choisir parmi les zones de disponibilité de votre VPC. Cependant, si vous choisissez un déploiement Multi-AZ, RDS choisit automatiquement la zone de disponibilité de l'instance de base de données principale ou de l'instance de base de données en écriture, et le paramètre Availability Zone (Zone de disponibilité) n'apparaît pas.

Dans certains cas, vous pouvez ne pas avoir de VPC par défaut ou vous n'avez créé aucun VPC. Dans ces cas, vous pouvez demander à Amazon RDS de créer automatiquement un VPC à votre place lorsque vous créez une instance de base de données à partir de la console. Sinon, procédez comme suit :

- Créez un VPC avec au moins un sous-réseau dans chacune des deux zones de disponibilité dans Région AWS lesquelles vous souhaitez déployer votre instance de base de données. Pour plus d'informations, consultez Utilisation d'un(e) instance de base de données dans un VPC et Tutoriel : Création d'un VPC à utiliser avec un) IPv4 .
- Spécifiez un groupe de sécurité VPC qui autorise les connexions à votre instance de bases de données. Pour plus d'informations, consultez Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC et Contrôle d'accès par groupe de sécurité.
- Spécifiez un groupe de sous-réseaux de base de données RDS définissant au moins deux sous-réseaux du VPC pouvant être utilisés par l'instance de bases de données. Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.

Si vous souhaitez vous connecter à une ressource qui ne se trouve pas dans le même VPC que l'instance de base de données, consultez les scénarios appropriés dans Scénarios d'accès à un d'instances de base de données dans un VPC.

Prérequis supplémentaires

Avant de créer votre instance de base de données, tenez compte des conditions préalables supplémentaires suivantes :

- Si vous vous connectez à AWS l'aide d'informations d'identification AWS Identity and Access Management (IAM), votre AWS compte doit disposer de certaines politiques IAM. Elles accordent les autorisations requises pour effectuer des opérations Amazon RDS. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.

Pour utiliser IAM pour accéder à la console RDS, connectez-vous à l'aide de vos informations d'identification AWS Management Console d'utilisateur IAM. Accédez ensuite à la console Amazon RDS à https://console.aws.amazon.com/rds/l'adresse.

- Pour personnaliser les paramètres de configuration à votre instance de base de données, spécifiez un groupe de paramètres de base de données avec les paramètres requis. Pour plus d'informations sur la création ou la modification d'un groupe de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).

> **Important**
>
> Si vous utilisez le modèle BYOL pour Amazon RDS pour DB2, avant de créer une instance de base de données, vous devez d'abord créer un groupe de paramètres personnalisé contenant votre IBM Site ID and IBM Customer ID. Pour plus d'informations, consultezApportez votre propre licence (BYOL) pour Db2.

- Déterminez le numéro de port TCP/IP à spécifier pour votre instance de base de données. Dans certaines entreprises, les pare-feu bloquent les connexions à ces ports par défaut des instances de base de données RDS. Si le pare-feu de votre entreprise bloque le port par défaut, choisissez un autre port pour votre instance de base de données. Les ports par défaut pour les moteurs de base de données Amazon RDS sont les suivants :

RDS pour Db2	RDS for MariaDB	RDS for MySQL	RDS for Oracle	RDS for PostgreSQL	RDS for SQL Server
50000	3306	3306	1521	5432	1433

Pour RDS for SQL Server, les ports suivants sont réservés et vous ne pouvez pas les utiliser lorsque vous créez une instance de base de données : `1234, 1434, 3260, 3343, 3389, 47001,` et`49152-49156`.

Création d'une instance de base de données

Vous pouvez créer une instance de base de données Amazon RDS à l'aide de l' AWS Management Console API AWS CLI, de ou de l'API RDS.

> **Note**
>
> Pour RDS pour DB2, nous vous recommandons de configurer les éléments nécessaires à votre modèle de licence avant de créer une instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Options de licence Amazon RDS pour DB2.

console

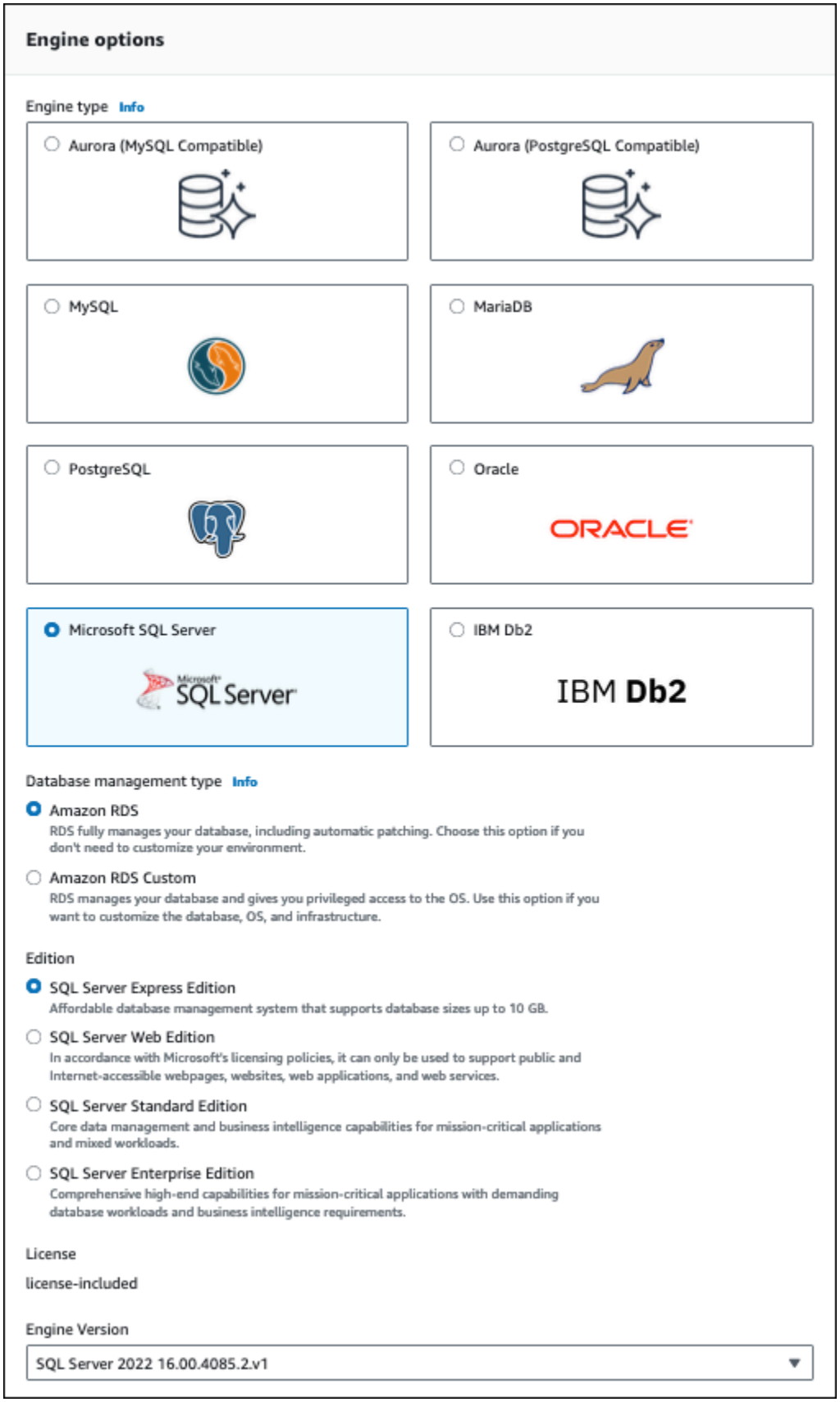
Vous pouvez créer une instance de base de AWS Management Console données en utilisant Easy create activé ou non activé. Lorsque l'option Easy create (Création facile) est activée, vous spécifiez uniquement le type de moteur, la taille de l'instance, ainsi que l'identifiant d'instance de base de données. Easy create (Création facile) utilise les paramètres par défaut pour les autres options de configuration. Lorsque Easy create (Création facile) est désactivé, vous spécifiez davantage d'options de configuration lors de la création d'une base de données, notamment en matière de disponibilité, de sécurité, de sauvegardes et de maintenance.

> **Note**
>
> Dans la procédure suivante, Standard create (Création standard) est activé et Easy create (Création facile) est désactivé. Cette procédure utilise Microsoft SQL Server à titre d'exemple. Pour obtenir des exemples qui utilisent l'option Easy create (Création facile) pour vous aider à créer et à vous connecter à des exemples d'instances de base de données pour chaque moteur, consultez Mise en route avec Amazon RDS.

Pour créer une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la région AWS dans laquelle vous voulez créer l'instance de base de données.
3. Dans la panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Créer une base de données, puis Création standard.
5. Pour le type de moteur, choisissez IBM Db2, MariaDB, Microsoft SQL Server, MySQL, Oracle ou PostgreSQL.

Microsoft SQL Server est illustré ici.


Engine options
Engine type Info
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
MariaDB
PostgreSQL
Oracle
ORACLE
Microsoft SQL Server
SQL Server
IBM Db2
IBM Db2
Database management type Info
Amazon RDS
RDS fully manages your database, including automatic patching. Choose this option if you don't need to customize your environment.
Amazon RDS Custom
RDS manages your database and gives you privileged access to the OS. Use this option if you want to customize the database, OS, and infrastructure.
Edition
SQL Server Express Edition
Affordable database management system that supports database sizes up to 10 GB.
SQL Server Web Edition
In accordance with Microsoft's licensing policies, it can only be used to support public and Internet-accessible webpages, websites, web applications, and web services.
SQL Server Standard Edition
Core data management and business intelligence capabilities for mission-critical applications and mixed workloads.
SQL Server Enterprise Edition
Comprehensive high-end capabilities for mission-critical applications with demanding database workloads and business intelligence requirements.
License
license-included
Engine Version
SQL Server 2022 16.00.4085.2.v1

6. Pour Type de gestion de base de données, si vous utilisez Oracle ou SQL Server, choisissez Amazon RDS ou Amazon RDS Custom.

   Amazon RDS est illustré ici. Pour plus d'informations sur RDS Custom, consultez Amazon RDS Custom.

7. Pour Edition, si vous utilisez Db2, Oracle ou SQL Server, choisissez l'édition du moteur de base de données que vous souhaitez utiliser.

   MySQL n'a qu'une seule option pour l'édition, tandis que MariaDB et PostgreSQL n'en ont aucune.

8. Dans Version, choisissez la version du moteur.

9. Dans Templates (Modèles), sélectionnez le modèle qui correspond à votre cas d'utilisation. Si vous choisissez Production, les éléments suivants sont présélectionnés lors d'une étape ultérieure :

   - Option de basculement Multi-AZ
   - Option de stockage Provisioned IOPS SSD (io1) (SSD à IOPS provisionnés (io1))
   - Option Enable deletion protection (Activer la protection contre la suppression)

   Ces fonctions sont recommandées pour tous les environnements de production.

   > **Note**
   >
   > Les choix de modèles varient selon l'édition.

10. Dans la section Settings (Paramètres), ouvrez Credential Settings (Paramètres des informations d'identification). Ensuite, procédez comme suit :

    a. (Facultatif) Changez la valeur de Master username (Nom d'utilisateur principal).

    b. Choisissez l'une des options de gestion des informations d'identification suivantes :

       - Géré dans AWS Secrets Manager

         Dans Sélectionnez la clé de chiffrement, choisissez soit une clé KMS créée par Secrets Manager, soit une clé que vous avez créée.

> **Note**
>
> C'est la technique la plus sûre AWS Secrets Manager que nous recommandons pour gérer les informations d'identification. Des frais supplémentaires seront facturés. Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.

- Autogéré

  Pour spécifier un mot de passe, désactivez la case à cocher Générer automatiquement un mot de passe si elle est sélectionnée. Entrez le même mot de passe dans Mot de passe principal et Confirmez le mot de passe principal.

11. (Facultatif) Configurez une connexion à une ressource de calcul pour cette instance de base de données.

    Vous pouvez configurer la connectivité entre une EC2 instance Amazon et la nouvelle instance de base de données lors de la création de l'instance de base de données. Pour de plus amples informations, veuillez consulter Configuration de la connectivité réseau automatique avec une EC2 instance.

12. Dans la section Connectivité sous Groupe de sécurité VPC (pare-feu), si vous sélectionnez Créer, un groupe de sécurité VPC est créé avec une règle entrante qui autorise l'adresse IP de votre ordinateur local à accéder à la base de données.

13. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour plus d'informations sur chaque paramètre, consultez Paramètres des instances de base de données.

14. Choisissez Create database (Créer une base de données).

    Si vous choisissez de générer un mot de passe automatiquement, le bouton View credential details (Afficher les informations d'identification) apparaît sur la page Databases (Bases de données).

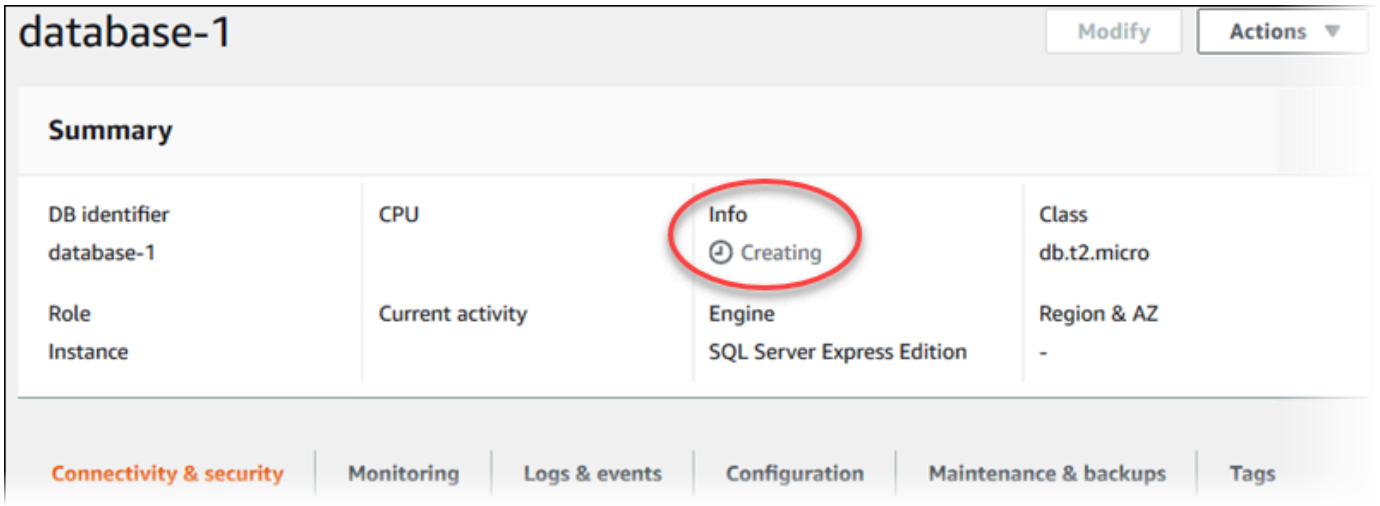
    Pour afficher l'identifiant principal et le mot de passe pour l'instance de base de données, choisissez View credential details (Afficher les informations d'identification).

    Pour vous connecter à l'instance de base de données en tant qu'utilisateur principal, utilisez le nom d'utilisateur et le mot de passe affichés.

> **Important**
>
> Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier. Si vous devez changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, faites-le en modifiant l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

15. Pour Databases (Bases de données), choisissez le nom de la nouvelle instance de base de données.

    Sur la console RDS, les détails de la nouvelle instance de base de données s'affichent. L'instance de base de données a le statut Creating (Création en cours) jusqu'à ce qu'elle soit créée et prête à l'emploi. Lorsque le statut devient Available (Disponible), vous pouvez vous connecter à l'instance de base de données. En fonction du stockage et de la classe d'instance de base de données alloués, la mise à disposition de la nouvelle instance de base de données peut nécessiter plusieurs minutes.



AWS CLI

> **Note**
>
> Si vous souhaitez utiliser la licence DB2 via AWS Marketplace, vous devez d'abord vous abonner AWS Marketplace et vous enregistrer auprès d'IBM en utilisant le AWS Management

Console. Pour de plus amples informations, veuillez consulter Abonnement aux listes de DB2 Marketplace et inscription auprès de IBM.

Pour créer une instance de base de données à l'aide de AWS CLI, appelez la create-db-instancecommande avec les paramètres suivants :

- `--db-instance-identifier`
- `--db-instance-class`
- `--vpc-security-group-ids`
- `--db-subnet-group`
- `--engine`
- `--master-username`
- `--master-user-password` ou `--manage-master-user-password`
- `--allocated-storage`
- `--backup-retention-period`

Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Cet exemple utilise Microsoft SQL Server.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --engine sqlserver-se \
    --db-instance-identifier mymsftsqlserver \
    --allocated-storage 250 \
    --db-instance-class db.t3.large \
    --vpc-security-group-ids mysecuritygroup \
    --db-subnet-group mydbsubnetgroup \
    --master-username masterawsuser \
    --manage-master-user-password \
    --backup-retention-period 3
```

Dans Windows:

```
aws rds create-db-instance ^
    --engine sqlserver-se ^
    --db-instance-identifier mydbinstance ^
    --allocated-storage 250 ^
    --db-instance-class db.t3.large ^
    --vpc-security-group-ids mysecuritygroup ^
    --db-subnet-group mydbsubnetgroup ^
    --master-username masterawsuser ^
    --manage-master-user-password ^
    --backup-retention-period 3
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
DBINSTANCE  mydbinstance  db.t3.large  sqlserver-se  250  sa  creating  3  ****  n
 10.50.2789
SECGROUP  default  active
PARAMGRP  default.sqlserver-se-14  in-sync
```

API RDS

> **Note**
>
> Si vous souhaitez utiliser la licence DB2 via AWS Marketplace, vous devez d'abord vous abonner AWS Marketplace et vous enregistrer auprès d'IBM en utilisant le AWS Management Console. Pour de plus amples informations, veuillez consulter Abonnement aux listes de DB2 Marketplace et inscription auprès de IBM.

Pour créer une instance de base de données à l'aide de l'API Amazon RDS, appelez l'DBInstanceopération Create.

Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Paramètres des instances de base de données

Le tableau suivant contient des détails sur les paramètres que vous choisissez lors de la création d'une instance de base de données. Le tableau répertorie également les moteurs de base de données pour lesquels chaque paramètre est pris en charge.

Vous pouvez créer une instance de base de données à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Stockage alloué	Quantité de stockage à allouer pour votre instance de base de données (en gibioctets). Dans certains cas, l'allocation d'une quantité de stockage supérieure à la taille de votre base de données pour votre instance de base de données peut améliorer les I/O performances.
	
	Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.
		Option de l'interface CLI :
		
		`--allocated-storage`
		
		Paramètre de l'API :
		
		`AllocatedStorage`
			Tous

Paramètres de l'architecture	Si vous choisissez Architecture multilocataire Oracle, RDS for Oracle crée une base de données de conteneur (CDB). Si vous ne choisissez pas cette option, RDS for Oracle crée une base de données non-CDB. Une base de données non CDB utilise l'architecture de base de données Oracle traditionnelle. Un CDB peut contenir des bases de données enfichables (PDBs) alors qu'un non-CDB ne le peut pas. Oracle Database 21c utilise uniquement l'architecture CDB. Oracle Database 19c peut utiliser une architecture CDB ou non CDB. Les versions inférieures à	Option de l'interface CLI : `--engine oracle-ee-cdb` (architecture multilocataire Oracle) `--engine oracle-se2-cdb` (architecture multilocataire Oracle) `--engine oracle-ee` (traditionnel) `--engine oracle-se2` (traditionnel) Paramètre de l'API :	Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	Oracle Database 19c utilisent uniquement l'architecture non CDB. Pour de plus amples informations, veuillez consulter Présentation de RDS pour Oracle CDBs.	`Engine`	

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
Configuration de l'architecture	Ces paramètres ne sont valides que lorsque vous choisissez Architecture multilocataire Oracle pour Paramètres d'architecture. Choisissez l'un des paramètres supplémentaires suivants : • Avec la configuration multi-locataires, votre instance RDS pour Oracle CDB peut contenir de 1 à 30 bases de données mutualisées, selon l'édition de la base de données et les licences d'option requises. Dans le contexte d'une base de données Oracle, une base de données locataire est une PDB. L'application PDBs et le proxy PDBs ne sont pas pris en charge. Votre instance de base de données est créée avec une base de données locataire initiale. Choisissez des valeurs pour les champs Nom de la base de données locataire, Nom d'utilisateur principal de la base de données locataire, Mot de passe principal de base de données locataire et Jeu de caractères de base de données locataire. La configuration à locataires multiples est définitive. Par conséquent, vous	Option de l'interface CLI : `--multi-tenant` (configuration à locataires multiples) `--no-multi-tenant` (configuration à locataire unique) Paramètre de l'API : `MultiTenant`	Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	ne pouvez pas reconvertir la configuration à locataires multiples en configuration à locataire unique. La mise à jour de version (RU) minimale prise en charge pour la configuration à locataires multiples est 19.0.0.0.ru-2022-01.rur-2022.r1. **Note** La configuration Amazon RDS est appelée « multi-tenant » plutôt que « multitenant » car il s'agit d'une fonctionnalité d'Amazon RDS, et pas seulement du moteur de base de données Oracle. De même, le terme « locataire » du RDS fait référence à tout locataire d'une configuration RDS, et pas seulement à Oracle. PDBs Dans la documentation RDS, le terme sans trait d'union « Oracle multitenant » fait exclusivement référence à l'architecture CDB de la base de données Oracle, qui est compatible avec les déploiements sur site et RDS.		

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	• Avec la configuration à locataire unique, votre CDB RDS for Oracle contient une seule PDB. Il s'agit de la configuration par défaut lorsque vous créez une CDB. Vous ne pouvez pas supprimer le PDB initial ni en ajouter d'autres PDBs. Vous pouvez convertir ultérieurement la configuration à locataire unique de votre CDB en configuration à locataires multiples, mais vous ne pouvez pas ensuite la reconvertir en configuration à locataire unique. Quelle que soit la configuration que vous choisissez, votre CDB contient une seule PDB initiale. Dans la configuration multi-locataires, vous pouvez en créer d'autres PDBs ultérieurement à l'aide de APIs RDS. Pour de plus amples informations, veuillez consulter Présentation de RDS pour Oracle CDBs.		

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Mise à niveau automatique de versions mineures	Choisissez Activer la mise à niveau automatique des versions mineures pour permettre à votre instance de base de données de recevoir automatiquement les mises à niveau des versions mineures préférées du moteur de base de données lorsqu'elles sont disponibles. Il s'agit du comportement de par défaut. Amazon RDS effectue les mises à niveau automatiques des versions mineures dans la fenêtre de maintenance. Si vous ne sélectionnez pas Activer la mise à niveau automatique des versions mineures, votre instance de base de données n'est pas mise à niveau automatiquement lorsque de nouvelles versions mineures sont disponibles.
	
	Pour de plus amples informations, veuillez consulter Mise à niveau automatique de la version mineure du moteur.
		Option de l'interface CLI :
		
		`--auto-minor-version-upgrade`
		
		`--no-auto-minor-version-upgrade`
		
		Paramètre de l'API :
		
		`AutoMinorVersionUpgrade`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Zone de disponibilité	Zone de disponibilité de votre instance de base de données. Utilisez la valeur par défaut No Preference (Aucune préférence), sauf si vous souhaitez spécifier une zone de disponibilité.
	
	Pour plus d'informations, consultez Régions, zones de disponibilité et zones locales.
		Option de l'interface CLI :
		
		`--availability-zone`
		
		Paramètre de l'API :
		
		`AvailabilityZone`
			Tous

AWS KMS key	Disponible uniquement si l'option Chiffrement est définie sur Activer le chiffrement. Choisissez la AWS KMS key à utiliser pour le chiffrement de cette instance de bases de données. Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.	Option de l'interface CLI :
		
		`--kms-key-id`
		
		Paramètre de l'API :
		
		`KmsKeyId`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
AWS License Manager configuration	Entrez le nom d'une configuration AWS License Manager de licence. Le nom doit comporter 100 caractères ou moins et inclure uniquement les caractères a-z, A-Z et 0-9. Pour de plus amples informations, veuillez consulter the section called “Intégration avec AWS License Manager”.	Option de l'interface CLI : Pour de plus amples informations, veuillez consulter AWS License Manager CLI. Paramètre de l'API : Pour de plus amples informations, veuillez consulter AWS License Manager API.	Db2
Réplication des sauvegardes	Choisissez Activer la réplication dans une autre région AWS pour créer des sauvegardes dans une autre région à des fins de reprise après sinistre. Sélectionnez ensuite la Région de destination des sauvegardes supplémentaires.	Non disponible lors de la création d'une instance de base de données. Pour plus d'informations sur l'activation des sauvegardes entre régions à l'aide de l'API AWS CLI ou RDS, consultez. Activation des sauvegardes automatisées entre régions pour Amazon RDS	Db2 Oracle PostgreSQL SQL Server

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Période de rétention des sauvegardes	Nombre de jours durant lesquels les sauvegardes automatiques de votre instance de base de données doivent être conservées. Pour une instance de base de données importante, définissez cette valeur sur **1** ou une valeur supérieure.
	
	Pour plus d'informations, consultez Présentation des sauvegardes.
		Option de l'interface CLI :
		
		`--backup-retention-period`
		
		Paramètre de l'API :
		
		`BackupRetentionPeriod`
			Tous

Cible de sauvegarde	Choisissez AWS Cloudde stocker les sauvegardes automatisées et les instantanés manuels dans la AWS région parent. Choisissez Outposts (sur site) pour les stocker en local sur votre Outpost.
	
	Ce paramètre s'applique uniquement à RDS sur Outposts. Pour de plus amples informations, veuillez consulter Création d'instances de base de données pour Amazon RDS sur AWS Outposts.
		Option de l'interface CLI :
		
		`--backup-target`
		
		Paramètre de l'API :
		
		`BackupTarget`
			MySQL, PostgreSQL, SQL Server

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Fenêtre de sauvegarde	Période de temps durant laquelle Amazon RDS effectue automatiquement une sauvegarde de votre instance de base de données. Si vous n'avez pas besoin que votre base de données soit sauvegardée à un moment précis, utilisez la valeur par défaut No Preference (Aucune préférence).
	
	Pour plus d'informations, consultez Présentation des sauvegardes.
		Option de l'interface CLI :
		
		`--preferred-backup-window`
		
		Paramètre de l'API :
		
		`PreferredBackupWindow`
			Tous

Autorité de certification	L'autorité de certification (CA) pour le certificat de serveur utilisé par l'instance de base de données.
	
	Pour de plus amples informations, veuillez consulter .
		Option de l'interface CLI :
		
		`--ca-certificate-identifier`
		
		Paramètre de l'API RDS :
		
		`CACertificateIdentifier`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Jeu de caractères	Jeu de caractères pour l'instance de base de données. La valeur par défaut du jeu AL32UTF8de caractères de base de données correspond au jeu de caractères universel Unicode 5.0 UTF-8. Vous ne pouvez pas modifier le jeu de caractères de base de données après avoir créé l'instance de base de données.
	
	Dans une configuration à locataire unique, un jeu de caractères de base de données autre que par défaut n'affecte que la base de données enfichable (PDB) et pas la base de données de conteneur (CDB). Pour plus d'informations, consultez Configuration à locataire unique de l'architecture CDB.
	
	Le jeu de caractères de base de données est différent du jeu de caractères national, appelé jeu de caractères NCHAR. Contrairement au jeu de caractères de base de données, le jeu de caractères NCHAR spécifie le codage des colonnes de types de données NCHAR (NCHAR, NVARCHAR2, et NCLOB) sans affecter les métadonnées de la base de données.
		Option de l'interface CLI :
		
		`--character-set-name`
		
		Paramètre de l'API :
		
		`CharacterSetName`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	Pour de plus amples informations, veuillez consulter Jeux de caractères RDS for Oracle.		

Classement (Collation)	Classement de niveau serveur pour votre instance de base de données.
	
	Pour plus d'informations, consultez Classement de niveau serveur pour Microsoft SQL Server.
		Option de l'interface CLI :
		
		`--character-set-name`
		
		Paramètre de l'API :
		
		`CharacterSetName`
			SQL Server

Copier les balises aux instantanés	Cette option copie les balises de l'instance de base de données dans un instantané de base de données au moment où vous créez un instantané.
	
	Pour plus d'informations, consultez Marquage d' Amazon RDS.
		Option de l'interface CLI :
		
		`--copy-tags-to-snapshot`
		
		`--no-copy-tags-to-snapshot`
		
		Paramètre de l'API RDS :
		
		`CopyTagsToSnapshot`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Authentification de base de données	L'option d'authentification de base de données que vous souhaitez utiliser.
	
	Choisissez Authentification par mot de passe pour authentifier les utilisateurs de base de données avec des mots de passe de base de données uniquement.
	
	Choisissez Password and IAM DB authentication (Mot de passe et authentification de base de données IAM) pour authentifier les utilisateurs de base de données avec des mots de passe de base de données et des informations d'identification utilisateur via des utilisateurs et rôles. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL. Cette option n'est prise en charge que pour MySQL et PostgreSQL.
	
	Choisissez l'authentification par mot de passe et Kerberos pour authentifier les utilisateurs de base de données à l'aide de mots de passe de base de données et l'authentification Kerberos via un outil créé avec. AWS Managed Microsoft AD AWS Directory Service Ensuitez,
		IAM :
		
		Option de l'interface CLI :
		
		`--enable-iam-database-authentication`
		
		`--no-enable-iam-database-authentication`
		
		Paramètre de l'API RDS :
		
		`EnableIAMDatabaseAuthentication`
		
		Kerberos :
		
		Option de l'interface CLI :
		
		`--domain`
		
		`--domain-iam-role-name`
		
		Paramètre de l'API RDS :
		
		`Domain`
		
		`DomainIAMRoleName`
			Varie selon le type d'authentification

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	choisissez le répertoire ou Créer un nouveau répertoire. Pour plus d'informations, consultez les étapes suivantes : • Utilisation de Kerberos l'authentification pour Amazon RDS pour Db2 • Utilisation de Kerberos l'authentification pour Amazon RDS for MySQL • Configuration de l'authentification Kerberos pour Amazon RDS for Oracle • Utilisation de l'authentification Kerberos avec Amazon RDS for PostgreSQL		
Type de gestion de base de données	Choisissez Amazon RDS si vous n'avez pas besoin de personnaliser votre environnement. Choisissez Amazon RDS Custom si vous souhaitez personnaliser la base de données, le système d'exploitation et l'infrastructure. Pour de plus amples informations, veuillez consulter Amazon RDS Custom.	Pour l'interface de ligne de commande (CLI) et l'API, vous devez spécifier le type de moteur de base de données.	Oracle SQL Server

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Port de la base de données	Port par lequel vous souhaitez accéder à l'instance de base de données. La valeur par défaut du port est indiquée.
	
	Note
	Dans certaines entreprises, les pare-feux bloquent les connexions aux ports MariaDB, MySQL et PostgreSQL par défaut. Si le pare-feu de votre entreprise bloque le port par défaut, entrez un autre port pour votre instance de base de données.
		Option de l'interface CLI :
		
		`--port`
		
		Paramètre de l'API RDS :
		
		`Port`
			Tous

Version du moteur de base de données	Version du moteur de base de données que vous souhaitez utiliser.	Option de l'interface CLI :
		
		`--engine-version`
		
		Paramètre de l'API RDS :
		
		`EngineVersion`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Classe d'instances de base de données	Configuration pour votre instance de base de données. Par exemple, une classe d'instance de base de données db.t3.small possède 2 GiB de mémoire, 2 V, 1 cœur virtuelCPUs, un ECU variable et une capacité d'E/S modérée.
	
	Dans la mesure du possible, choisisse z une classe d'instance de base de données suffisamment grande pour qu'un ensemble de travail de requête classique puisse tenir dans la mémoire. Lorsque les ensembles de travail sont en mémoire, le système peut éviter d'écrire sur le disque, ce qui améliore les performances. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .
	
	Dans RDS for Oracle, vous pouvez sélectionner Include additional memory configurations (Inclure des configura tions de mémoire supplémentaire). Ces configurations sont optimisées pour obtenir un ration mémoire-vCPU élevé. Par exemple, db.r5.6xlarge.tpc2.mem4x est une instance de base de données db.r5.8x dotée de 2 threads par cœur (tpc2) et de 4 fois plus de mémoire qu'une instance de base de données
		Option de l'interface CLI :
		
		`--db-instance-clas s`
		
		Paramètre de l'API RDS :
		
		`DBInstanceClass`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	db.r5.6xlarge standard. Pour de plus amples informations, veuillez consulter RDS pour les classes d'instance de base de données Oracle.		

Identifiant d'instance de base de données	Nom de votre instance de base de données. Nommez vos instances de base de données de la même façon que vos serveurs sur site. L'identifiant de votre instance de base de données peut contenir jusqu'à 63 caractères alphanumériques et doit être unique pour votre compte dans la AWS région que vous avez choisie.	Option de l'interface CLI :
		`--db-instance-identifier`
		Paramètre de l'API RDS :
		`DBInstanceIdentifier`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Groupe de paramètres de base de données	Groupe de paramètres pour l'instanc e de base de données. Vous pouvez soit choisir le groupe de paramètre s par défaut, soit créer un groupe de paramètres personnalisé.
	
	Si vous utilisez le modèle BYOL pour RDS pour DB2, avant de créer une instance de base de données, vous devez d'abord créer un groupe de paramètres personnalisé contenant votre et. IBM Site ID IBM Customer ID Pour de plus amples informations, veuillez consulter Apportez votre propre licence (BYOL) pour Db2.
	
	Pour de plus amples informations, veuillez consulter Groupes de paramètre s pour (Amazon RDS).
		Option de l'interface CLI :
		
		`--db-parameter-gro up-name`
		
		Paramètre de l'API RDS :
		
		`DBParameterGroupNa me`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Groupe de sous-réseaux de base de données	Le groupe de sous-réseaux de base de données à utiliser pour le cluster de bases de données.
	Sélectionnez Choose existing (Choisir existants) pour utiliser un groupe de sous-réseaux de base de données. Choisissez ensuite le groupe de sous-réseaux requis dans la liste déroulante Existing DB subnet groups (Groupes de sous-réseaux de base de données existants).
	
	Choisissez Automatic setup (Configuration automatique) pour permettre à RDS de sélectionner un groupe de sous-réseaux de base de données compatible. S'il n'en existe aucun, RDS crée un nouveau groupe de sous-réseaux pour votre cluster.
	
	Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.
		Option de l'interface CLI :
		
		`--db-subnet-group-name`
		
		Paramètre de l'API RDS :
		
		`DBSubnetGroupName`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Volumes dédiés aux journaux	Utilisez un volume dédié aux journaux (DLV) pour stocker les journaux de transactions de base de données sur un volume de stockage distinct du volume contenant les tables de base de données.
	
	Pour de plus amples informations, veuillez consulter Utilisation d'un volume dédié aux journaux (DLV).
		Option de l'interface CLI :
		
		`--dedicated-log-volume`
		
		Paramètre de l'API RDS :
		
		`DedicatedLogVolume`
			Tous

Deletion protection (Protection contre la suppression)	Enable deletion protection (Activer la protection contre la suppression) vise à empêcher la suppression de votre instance de base de données. Si vous créez une instance de base de données de production avec le AWS Management Console, la protection contre la suppression est activée par défaut.
	
	Pour de plus amples informations, veuillez consulter Suppression d'une instance DB.
		Option de l'interface CLI :
		
		`--deletion-protection`
		
		`--no-deletion-protection`
		
		Paramètre de l'API RDS :
		
		`DeletionProtection`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Chiffrement	Activer le chiffrement si vous souhaitez activer le chiffrement au repos pour cette instance de base de données.
	
	Pour plus d'informations, consultez Chiffrement des ressources Amazon RDS.
		Option de l'interface CLI :
		
		`--storage-encrypted`
		
		`--no-storage-encrypted`
		
		Paramètre de l'API RDS :
		
		`StorageEncrypted`
			Tous

Surveillance améliorée	Activer la surveillance améliorée permet d'activer la collecte des métriques en temps réel pour le système d'exploitation sur lequel votre instance de base de données s'exécute.
	
	Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.
		Options d'interface de ligne de commande :
		
		`--monitoring-interval`
		
		`--monitoring-role-arn`
		
		Paramètres de l'API RDS :
		
		`MonitoringInterval`
		
		`MonitoringRoleArn`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Engine type (Type de moteur)	Choisissez le nom du moteur de base de données à utiliser pour cette instance de base de données de base de données.	Option de l'interface CLI :
		`--engine`
		Paramètre de l'API RDS :
		`Engine`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Nom de la base de données initiale	Nom de la base de données dans votre instance de base de données. Si vous ne fournissez pas de nom, Amazon RDS ne crée pas de base de données sur l'instance de base de données (sauf pour Oracle et PostgreSQL). Le nom ne peut pas être un mot réservé par le moteur de base de données et possède d'autres contraintes selon le moteur de base de données.
	
	DB2 :
	
	• Il doit contenir entre 1 et 8 caractères alphanumériques.
	• Il doit commencer par a-z, A-Z, @, $ ou #, et être suivi de a-z, A-Z, 0-9, _, @, # ou $.
	• Il ne doit pas contenir d'espace.
	• Pour de plus amples informations, veuillez consulter Considérations supplémentaires.
	
	MariaDB et MySQL :
	
	•
		Option de l'interface CLI :
		
		`--db-name`
		
		Paramètre de l'API RDS :
		
		`DBName`
			Tous sauf SQL Server

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	Il doit contenir entre 1 et 64 caractères alphanumériques. Oracle : • Il doit contenir entre 1 et 8 caractères alphanumériques. • Il ne peut pas être `NULL`. La valeur par défaut est `ORCL`. • Il doit commencer par une lettre. PostgreSQL : • Il doit contenir entre 1 et 63 caractères alphanumériques. • Il doit commencer par une lettre ou un trait de soulignement. Les caractère s suivants peuvent être des lettres, des traits de soulignement ou des chiffres (0-9). • Le nom initial de la base de données est `postgres`.		

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Licence	Valeurs valides pour le modèle de licence :
	• bring-your-own-licenseou licence Marketplace pour Db2.
	• general-public-licensepour MariaDB.
	• license-included pour Microsoft SQL Server.
	• general-public-licensepour MySQL.
	• licence incluse ou bring-your-own-lic ensepour Oracle.
	• postgresql-license pour PostgreSQL.
		Option de l'interface CLI :
		`--license-model`
		Paramètre de l'API RDS :
		`LicenseModel`
			Tous

Exportati ons des journaux	Les types de fichiers journaux de base de données à publier sur Amazon CloudWatch Logs.
	Pour de plus amples informations, veuillez consulter Publication des journaux de base de données sur Amazon CloudWatch Logs.
		Option de l'interface CLI :
		`--enable-cloudwatc h-logs-exports`
		Paramètre de l'API RDS :
		`EnableCloudwatchLo gsExports`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Fenêtre de maintenance	Créneau de 30 minutes pendant lequel les modifications en attente pour votre instance de base de données sont appliquées. Si la période n'a pas d'importance, choisissez No Preference (Aucune préférence).
	
	Pour plus d'informations, consultez Fenêtre de maintenance Amazon RDS.
		Option de l'interface CLI :
		
		`--preferred-maintenance-window`
		
		Paramètre de l'API RDS :
		
		`PreferredMaintenanceWindow`
			Tous

Gérez les informations d'identification principales dans AWS Secrets Manager	Sélectionnez Gérer les informations d'identification principales dans AWS Secrets Manager pour gérer le mot de passe d'utilisateur principal dans un secret, dans Secrets Manager.
	
	Vous pouvez éventuellement choisir une clé KMS à utiliser pour protéger le secret. Choisissez l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
	
	Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.
		Option de l'interface CLI :
		
		`--manage-master-user-password | --no-manage-master-user-password`
		
		`--master-user-secret-kms-key-id`
		
		Paramètre de l'API RDS :
		
		`ManageMasterUserPassword`
		
		`MasterUserSecretKmsKeyId`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Mot de passe principal	Mot de passe de votre compte utilisateur principal. Le mot de passe comporte le nombre suivant de caractères ASCII imprimables (à l'exclusion des caractères /, ", espace et @) selon le moteur de base de données :
	• DB2 : 8—255
	• Oracle : entre 8 et 30
	• MariaDB et MySQL : entre 8 et 41
	• SQL Server et PostgreSQL : entre 8 et 128
		Option de l'interface CLI :
		`--master-user-password`
		Paramètre de l'API RDS :
		`MasterUserPassword`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Identifiant principal	Nom que vous utilisez comme nom d'utilisateur principal pour vous connecter à votre instance de base de données avec tous les privilèges de base de données. Notez les restrictions d'attribution de noms suivantes :
	• Le nom peut contenir entre 1 et 16 caractères alphanumériques et des traits de soulignement.
	• Le premier caractère doit être une lettre.
	• Le nom ne peut pas être un mot réservé par le moteur de base de données.
	
	Vous ne pouvez pas changer le nom d'utilisateur principal après la création de l'instance de base de données.
	
	Pour DB2, nous vous recommandons d'utiliser le même nom d'utilisateur principal que le nom de votre instance Db2 autogérée.
	
	Pour en savoir plus sur les privilèges accordés à l'utilisateur principal, consultez Privilèges du compte utilisateur principal.
		Option de l'interface CLI :
		
		`--master-username`
		
		Paramètre de l'API RDS :
		
		`MasterUsername`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Authentification Windows Microsoft SQL Server	Activez l'authentification Windows pour Microsoft SQL Server, puis parcourez le répertoire pour choisir le répertoire dans lequel vous souhaitez permettre aux utilisateurs de domaine autorisés de s'authentifier auprès de cette instance SQL Server à l'aide de l'authentification Windows.	Options d'interface de ligne de commande :
		`--domain`
		`--domain-iam-role-name`
		Paramètres de l'API RDS :
		`Domain`
		`DomainIAMRoleName`
			SQL Server

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

déploiement multi-AZ	Create a standby instance (Créer une instance de secours) permet de créer un réplica secondaire passif de votre instance de base de données dans une autre zone de disponibilité pour la prise en charge du basculement. Nous recommandons Multi-AZ pour les charges de travail de production afin de maintenir une haute disponibilité.
	
	Pour le développement et les tests, vous pouvez choisir Do not create a standby instance (Ne pas créer d'instance de secours).
	
	Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
		Option de l'interface CLI :
		
		`--multi-az`
		
		`--no-multi-az`
		
		Paramètre de l'API RDS :
		
		`MultiAZ`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Jeu de caractères national (NCHAR)	Jeu de caractères national pour votre instance de base de données, communément appelé jeu de caractères NCHAR. Vous pouvez définir le jeu de caractères national sur AL16 UTF16 (par défaut) ou sur UTF-8. Vous ne pouvez pas modifier le jeu de caractères national après avoir créé l'instance de base de données.
	
	Le jeu de caractères national est différent du jeu de caractères de base de données. Contrairement au jeu de caractères de base de données, le jeu de caractères national spécifie le codage uniquement pour les colonnes de types de données NCHAR (NCHAR, NVARCHAR2, et NCLOB) sans affecter les métadonnées de la base de données.
	
	Pour de plus amples informations, veuillez consulter Jeux de caractères RDS for Oracle.
		Option de l'interface CLI :
		
		`--nchar-character-set-name`
		
		Paramètre de l'API :
		
		`NcharCharacterSetName`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Network type (Type de réseau)	Les protocoles d'adressage IP pris en charge par l'instance de la base de données.
	
	IPv4(valeur par défaut) pour spécifier que les ressources peuvent communiquer avec l'instance de base de données uniquement via le protocole d'adressage Internet Protocol version 4 (IPv4).
	
	Mode double pile pour spécifier que les ressources peuvent communiquer avec l'instance de base de données via IPv4 le protocole Internet version 6 (IPv6) ou les deux. Utilisez le mode double pile si vous avez des ressources qui doivent communiquer avec votre instance de base de données via le protocole d' IPv6 adressage. Assurez-vous également d'associer un bloc IPv6 CIDR à tous les sous-réseaux du groupe de sous-réseaux de base de données que vous spécifiez.
	
	Pour de plus amples informations, veuillez consulter Adressage IP Amazon RDS.
		Option de l'interface CLI :
		
		`--network-type`
		
		Paramètre de l'API RDS :
		
		`NetworkType`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Groupe d'options	Groupe d'options pour l'instance de base de données. Vous pouvez choisir le groupe d'options par défaut, ou vous pouvez créer un groupe d'options personnalisé.
	
	Pour plus d'informations, consultez Utilisation de groupes d'options.
		Option de l'interface CLI :
		
		`--option-group-name`
		
		Paramètre de l'API RDS :
		
		`OptionGroupName`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Performance Insights	Activer Performance Insights permet de surveiller la charge de votre instance de base de données et ainsi d'analyser les performances de votre base de données et de résoudre les problèmes associés.
	
	Choisissez une période de conservation pour déterminer l'historique des données de Performance Insights à conserver. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.
	
	Choisissez la clé KMS à utiliser pour protéger la clé servant à chiffrer ce volume de base de données. Choisissez l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
	
	Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.
		Options d'interface de ligne de commande :
		
		`--enable-performance-insights`
		
		`--no-enable-performance-insights`
		
		`--performance-insights-retention-period`
		
		`--performance-insights-kms-key-id`
		
		Paramètres de l'API RDS :
		
		`EnablePerformanceInsights`
		
		`PerformanceInsightsRetentionPeriod`
		
		`PerformanceInsightsKMSKeyId`
			Tous sauf Db2

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

IOPS provisionnés	Valeur d'IOPS provisionnés (opérations d'I/O par seconde) pour l'instance de base de données. Ce paramètre n'est disponible que si vous choisissez l'une des options suivantes pour Storage type (Type de stockage) :
	• General purpose SSD (gp3) (SSD à usage général (gp3))
	• Provisioned IOPS SSD (io1) (SSD à IOPS provisionnés (io1))
	• SSD IOPS provisionné (io2)
	
	Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.
		Option de l'interface CLI :
		`--iops`
		Paramètre de l'API RDS :
		`Iops`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Accès public	La valeur Oui dote l'instance de base de données d'une adresse IP publique, ce qui signifie qu'elle est accessible en dehors du VPC. Pour être accessible au public, l'instance de base de données doit aussi se trouver dans un sous-rése au public du VPC.
	
	La valeur Non rend l'instance de base de données accessible uniquement au sein du VPC.
	
	Pour de plus amples informations, veuillez consulter Masquer un(e) instance de base de données dans un VPC depuis Internet.
	
	Pour se connecter à une instance de base de données depuis l'extérieur de son VPC, l'instance de base de données doit être accessible publiquement. En outre, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité de l'instance de base de données. En outre, d'autres exigences doivent être respectées. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.
		Option de l'interface CLI :
		
		`--publicly-accessible`
		
		`--no-publicly-accessible`
		
		Paramètre de l'API RDS :
		
		`PubliclyAccessible`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	Si votre instance de base de données n'est pas accessible au public, utilisez une connexion AWS Site-to-Site VPN ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour de plus amples informations, veuillez consulter Confidentialité du trafic inter-réseau.		

Support étendu RDS	Sélectionnez Activer le support étendu RDS pour permettre aux versions principales du moteur prises en charge de continuer à fonctionner après la date de fin du support standard RDS.
	
	Lorsque vous créez une instance de base de données, Amazon RDS utilise par défaut RDS Extended Support. Pour empêcher la création d'une nouvelle instance de base de données après la date de fin du support standard RDS et pour éviter les frais liés au support étendu RDS, désactivez ce paramètre . Vos instances de base de données existantes ne seront pas facturées avant la date de début des tarifs du Support étendu RDS.
	
	Pour de plus amples informations, veuillez consulter .
		Option de l'interface CLI :
		
		`--engine-lifecycle-support`
		
		Paramètre de l'API RDS :
		
		`EngineLifecycleSupport`
			MySQL
			
			PostgreSQL

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
RDS Proxy (Proxy RDS)	Choisissez Create an RDS Proxy (Créer un proxy RDS) pour créer un proxy pour votre instance de base de données. Amazon RDS crée automatiquement un rôle IAM et un secret Secrets Manager pour le proxy. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .	Non disponible lors de la création d'une instance de base de données.	MariaDB MySQL PostgreSQL

Dimensionnement automatique du stockage	Enable storage autoscaling (Activer le dimensionnement automatique du stockage) permet à Amazon RDS d'augmenter automatiquement l'espace de stockage quand cela est nécessaire pour éviter que votre instance de base de données en manque.
	
	Utilisez Maximum storage threshold (Seuil de stockage maximum) pour définir la limite supérieure au-delà de laquelle Amazon RDS augmente automatiquement l'espace de stockage pour votre instance de base de données. La valeur par défaut est 1 000 GiO.
	
	Pour plus d'informations, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.
		Option de l'interface CLI :
		
		`--max-allocated-storage`
		
		Paramètre de l'API RDS :
		
		`MaxAllocatedStorage`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Storage throughput (Débit de stockage)	La valeur de débit de stockage de l'instance de base de données. Ce paramètre n'est disponible que si vous choisissez General purpose SSD (gp3) (SSD à usage général (gp3)) pour Storage type (Type de stockage).
	
	Pour de plus amples informations, veuillez consulter Stockage GP3 (recommandé).
		Option de l'interface CLI :
		
		`--storage-throughput`
		
		Paramètre de l'API RDS :
		
		`StorageThroughput`
			Tous

Type de stockage	Type de stockage pour votre instance de base de données.
	
	Si vous choisissez General Purpose SSD (gp3) (SSD à usage général (gp3)), vous pouvez allouer des IOPS provisionnés et un débit de stockage supplémentaires sous Advanced settings (Paramètres avancés).
	
	Si vous choisissez Provisioned IOPS SSD (io1) ou Provisioned IOPS SSD (io2), entrez la valeur Provisioned IOPS.
	
	Pour de plus amples informations, veuillez consulter Types de stockage Amazon RDS.
		Option de l'interface CLI :
		
		`--storage-type`
		
		Paramètre de l'API RDS :
		
		`StorageType`
			Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Groupe de sous-réseaux	Groupe de sous-réseaux de base de données à associer à cette instance de base de données.
	
	Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.
		Option de l'interface CLI :
		
		`--db-subnet-group-name`
		
		Paramètre de l'API RDS :
		
		`DBSubnetGroupName`
			Tous

Nom de la base de données locataire	Le nom de votre PDB initiale dans la configuration à locataires multiples de l'architecture Oracle. Ce paramètre n'est disponible que si vous choisissez Configuration à locataires multiples pour Configuration de l'architecture.
	
	Le nom de la base de données locataire doit être différent de celui de votre CDB, à savoir `RDSCDB`. Vous ne pouvez pas changer le nom de la CDB.
		Option de l'interface CLI :
		
		`--db-name`
		
		Paramètre de l'API RDS :
		
		`DBName`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Nom d'utilisateur principal de la base de données locataire	Le nom que vous utilisez comme nom d'utilisateur principal pour vous connecter à votre base de données locataire (PDB) avec tous les privilèges de base de données. Ce paramètre n'est disponibl e que si vous choisissez Configuration à locataires multiples pour Configuration de l'architecture.
	
	Notez les restrictions d'attribution de noms suivantes :
	
	• Le nom peut contenir entre 1 et 16 caractères alphanumériques et des traits de soulignement.
	• Le premier caractère doit être une lettre.
	• Le nom ne peut pas être un mot réservé par le moteur de base de données.
	
	Vous ne pouvez pas exécuter les actions suivantes :
	
	• Modifiez le nom d'utilisateur principal du locataire après avoir créé la base de données locataire.
	•
		Option de l'interface CLI :
		
		`--master-username`
		
		Paramètre de l'API RDS :
		
		`MasterUsername`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge
	Connectez-vous à la CDB avec le nom d'utilisateur principal du locataire.		

Mot de passe principal de base de données locataire	Le mot de passe du compte d'utilisateur principal de votre base de données locataire (PDB). Ce paramètre n'est disponible que si vous choisissez Configuration à locataires multiples pour Configuration de l'architecture.
	
	Le mot de passe comporte de 8 à 30 caractères ASCII imprimables, à l'exception de `/`, de `"`, d'un espace et de `@`.
		Option de l'interface CLI :
		
		`--master-password`
		
		Paramètre de l'API RDS :
		
		`MasterPassword`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Jeu de caractères de base de données locataire	Le jeu de caractères de la base de données locataire initiale. Ce paramètre n'est disponible que si vous choisissez Configuration à locataires multiples pour Configuration de l'architecture. Seules les instances de CDB RDS for Oracle sont prises en charge.
	
	La valeur par défaut du jeu de caractères de la base de données mutualisée correspond au jeu de caractères universel Unicode 5.0 UTF-8 AL32UTF8. Vous pouvez choisir un jeu de caractères de base de données locataire différent du jeu de caractères de la CDB.
	
	Pour de plus amples informations, veuillez consulter Jeux de caractères RDS for Oracle.
		Option de l'interface CLI :
		
		`--character-set-name`
		
		Paramètre de l'API RDS :
		
		`CharacterSetName`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Jeu de caractères national de base de données locataire	Jeu de caractères national pour votre base de données locataire, communément appelé jeu de caractères NCHAR. Ce paramètre n'est disponible que si vous choisissez Configuration à locataires multiples pour Configuration de l'architecture. Seules les instances de CDB RDS for Oracle sont prises en charge.
	
	Vous pouvez définir le jeu de caractères national sur AL16UTF16(par défaut) ou sur UTF-8. Vous ne pouvez pas modifier le jeu de caractères national après la création de la base de données locataire.
	
	Le jeu de caractères national de base de données locataire est différent du jeu de caractères de la base de données locataire. Le jeu de caractères national spécifie l'encodage uniquement pour les colonnes qui utilisent le type de données NCHAR (NCHAR, NVARCHAR2 et NCLOB) et n'affecte pas les métadonnées de la base de données.
	
	Pour de plus amples informations, veuillez consulter Jeux de caractères RDS for Oracle.
		Option de l'interface CLI :
		
		`--nchar-character-set-name`
		
		Paramètre de l'API :
		
		`NcharCharacterSetName`
			Oracle

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

Fuseau horaire	Fuseau horaire de votre instance de base de données. Si vous ne choisissez pas de fuseau horaire, votre instance de base de données utilise le fuseau horaire par défaut. Vous ne pouvez pas modifier le fuseau horaire après la création de l'instance de base de données.
	
	Pour plus d'informations, consultez Fuseau horaire local pour Amazon RDS pour les instances de base de données DB2 et Fuseau horaire local pour les instances de base de données Microsoft SQL Server.
		Option de l'interface CLI :
		
		`--timezone`
		
		Paramètre de l'API RDS :
		
		`Timezone`
			Db2
			
			SQL Server
			
			RDS Custom for SQL Server

Virtual Private Cloud (VPC)	Un VPC basé sur le service Amazon VPC à associer à cette instance de base de données. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.	Pour la CLI et l'API, vous spécifiez le groupe de sécurité VPC. IDs	Tous

Paramètre de la console	Description du paramètre	Option de CLI et paramètre de l'API RDS	Moteurs de base de données pris en charge

VPC security group (firewall) [Groupe de sécurité VPC (pare-feu)]	Groupe de sécurité à associer à l'instance de base de données.
	
	Pour de plus amples informations, veuillez consulter Présentation des groupes de sécurité VPC.
		Option de l'interface CLI :
		
		`--vpc-security-group-ids`
		
		Paramètre de l'API RDS :
		
		`VpcSecurityGroupIds`
			Tous

Création de ressources RDSAmazon avec AWS CloudFormation

Amazon RDS est intégré à AWS CloudFormation un service qui vous aide à modéliser et à configurer vos AWS ressources afin que vous puissiez passer moins de temps à créer et à gérer vos ressources et votre infrastructure. Vous créez un modèle qui décrit toutes les AWS ressources souhaitées (telles que les instances de base de données et les groupes de paramètres de base de données) puis vous AWS CloudFormation provisionnez et configurez ces ressources pour vous.

Lorsque vous l'utilisez AWS CloudFormation, vous pouvez réutiliser votre modèle pour configurer vos ressources RDS de manière cohérente et répétée. Décrivez vos ressources une seule fois, puis fournissez les mêmes ressources à plusieurs reprises dans plusieurs AWS comptes et régions.

RDS et AWS CloudFormation modèles

AWS CloudFormation les modèles sont des fichiers texte formatés dans JSON ouYAML. Ces modèles décrivent les ressources que vous souhaitez mettre à disposition dans vos AWS CloudFormation piles. Si vous ne connaissez pas JSON ouYAML, vous pouvez utiliser AWS CloudFormation Designer pour vous aider à démarrer avec les AWS CloudFormation modèles. Pour plus d'informations, consultez Qu'est-ce que AWS CloudFormation Designer ? dans le AWS CloudFormation Guide de l'utilisateur.

RDS prend en charge la création de ressources dans AWS CloudFormation. Pour plus d'informations, y compris des exemples JSON et des YAML modèles pour ces ressources, consultez la référence au type de RDS ressource dans le guide de AWS CloudFormation l'utilisateur.

En savoir plus sur AWS CloudFormation

Pour en savoir plus AWS CloudFormation, consultez les ressources suivantes :

- AWS CloudFormation
- AWS CloudFormation Guide de l'utilisateur
- AWS CloudFormation APIRéférence
- AWS CloudFormation Guide de l'utilisateur de l'interface de ligne de commande

Connexion à une RDS instance de base de données Amazon

Avant de pouvoir vous connecter à une instance de base de données, vous devez créer l'instance de base de données. Pour plus d'informations, veuillez consulter Création d'une instance de base de données Amazon RDS. Une fois qu'Amazon a RDS approvisionné votre instance de base de données, utilisez une application ou un utilitaire client standard pour que votre moteur de base de données se connecte à l'instance de base de données. Dans la chaîne de connexion, spécifiez l'DNSadresse du point de terminaison de l'instance de base de données en tant que paramètre d'hôte. Vous spécifiez également le numéro de port du point de terminaison de l'instance de base de données en tant que paramètre de port.

Pour plus d'informations sur la recherche d'informations de connexion pour une RDS instance de base de données Amazon ou sur les scénarios d'accès à une instance de base de données dans une instance de base de donnéesVPC, consultez les rubriques suivantes.

- Recherche des informations de connexion pour une RDS instance de base de données Amazon
- Scénarios d'accès à une instance de base de données dans un VPC

Recherche des informations de connexion pour une RDS instance de base de données Amazon

Les informations de connexion d'une instance de base de données incluent son point de terminaison, son port et un utilisateur de base de données valide, tel que l'utilisateur principal. Par exemple, pour une instance My SQL DB, supposons que la valeur du point de terminaison soit`mydb.123456789012.us-east-1.rds.amazonaws.com`. Dans ce cas, la valeur du port est `3306`, et l'utilisateur de base de données est `admin`. Compte tenu de ces informations, vous spécifiez les valeurs suivantes dans une chaîne de connexion :

- Pour l'hôte ou le nom ou le DNS nom d'hôte, spécifiez`mydb.123456789012.us-east-1.rds.amazonaws.com`.
- Pour un port, spécifiez `3306`.
- Pour l'utilisateur, spécifiez `admin`.

Le point de terminaison est unique pour chaque instance de base de données, et les valeurs du port et de l'utilisateur peuvent varier. La liste suivante présente le port le plus courant pour chaque moteur de base de données :

- DB2 — 50 000
- MariaDB – 3306
- Microsoft SQL Server — 1433
- Mai SQL — 3306
- Oracle – 1521
- Poster SQL — 5432

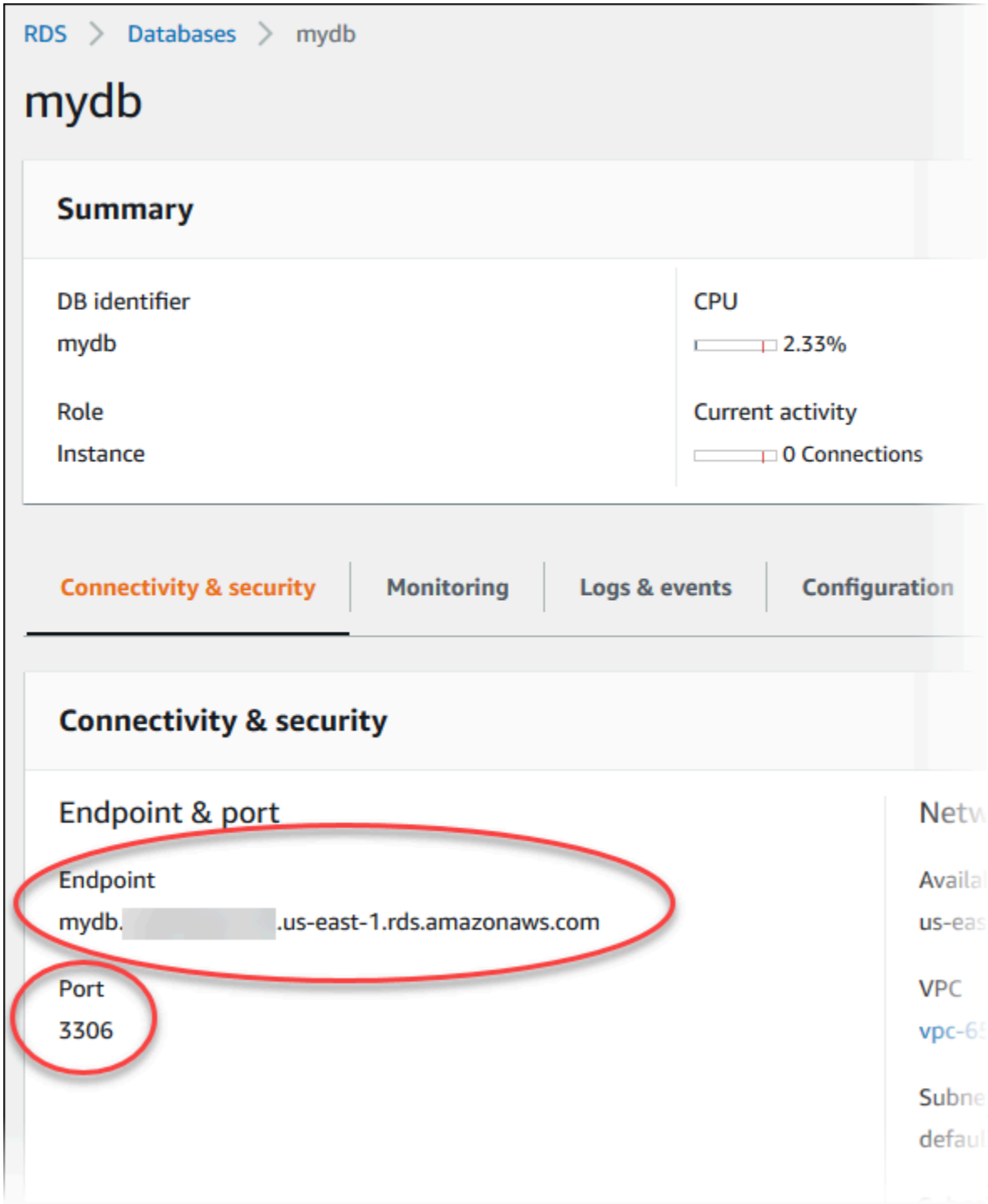
Pour vous connecter à une instance de base de données, utilisez n'importe quel client pour un moteur de base de données. Par exemple, vous pouvez utiliser l'utilitaire mysql pour vous connecter à une instance MariaDB ou SQL My DB. Vous pouvez utiliser Microsoft SQL Server Management Studio pour vous connecter à une instance de base de données de SQL serveur. Vous pouvez utiliser Oracle SQL Developer pour vous connecter à une instance de base de données Oracle. De même, vous pouvez utiliser l'utilitaire de ligne de commande psql pour vous connecter à une instance de SQL base de données Postgre.

Pour rechercher les informations de connexion d'une instance de base de données, utilisez la AWS Management Console. Vous pouvez également utiliser la describe-db-instancescommande AWS Command Line Interface (AWS CLI) ou l'escribeDBInstancesopération RDS API D.

Console

Pour trouver les informations de connexion d'une instance de base de données dans AWS Management Console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases (Bases de données) pour afficher la liste de vos instances de base de données.
3. Choisissez le nom de l'instance de base de données pour afficher ses détails.
4. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



5. Si vous devez rechercher le nom d'utilisateur principal, choisissez l'onglet Configuration et affichez la valeur Master username (Identifiant principal.

AWS CLI

Pour rechercher les informations de connexion d'une instance de base de données à l'aide de AWS CLI, appelez la describe-db-instancescommande. Dans l'appel, recherchez l'ID d'instance de base de données, le point de terminaison, le port et l'identifiant principal.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-instances \
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Dans Windows:

```
aws rds describe-db-instances ^
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Votre sortie doit ressembler à ce qui suit.

```
[
    [
        "mydb",
        "mydb.123456789012.us-east-1.rds.amazonaws.com",
        3306,
        "admin"
    ],
    [
        "myoracledb",
        "myoracledb.123456789012.us-east-1.rds.amazonaws.com",
        1521,
        "dbadmin"
    ],
    [
        "mypostgresqldb",
        "mypostgresqldb.123456789012.us-east-1.rds.amazonaws.com",
        5432,
        "postgresadmin"
    ]
]
```

RDS API

Pour trouver les informations de connexion d'une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'escribeDBInstancesopération D. Dans la sortie, recherchez les valeurs de l'adresse du point de terminaison, du port du point de terminaison et du nom d'utilisateur principal.

Scénarios d'accès à une instance de base de données dans un VPC

À l'aide d'Amazon Virtual Private Cloud (AmazonVPC), vous pouvez lancer AWS des ressources, telles que des RDS instances de base de données Amazon, dans un cloud privé virtuel (VPC). Lorsque vous utilisez AmazonVPC, vous contrôlez votre environnement réseau virtuel. Vous pouvez choisir votre propre plage d'adresses IP, créer des sous-réseaux et configurer le routage et les listes de contrôle d'accès.

Un groupe VPC de sécurité contrôle l'accès aux instances de base de données au sein d'unVPC. Chaque règle VPC de groupe de sécurité permet à une source spécifique d'accéder à une instance de base de données associée à ce groupe VPC de sécurité. VPC La source peut être une plage d'adresses (par exemple, 203.0.113.0/24) ou un autre groupe de sécurité. VPC En spécifiant un groupe VPC de sécurité comme source, vous autorisez le trafic entrant en provenance de toutes les instances (généralement des serveurs d'applications) qui utilisent le groupe VPC de sécurité source.

Avant de tenter de vous connecter à votre instance de base de données, configurez-la en VPC fonction de votre cas d'utilisation. Les scénarios suivants sont courants pour accéder à une instance de base de données dans un VPC :

- Une instance de base de données dans une EC2 instance VPC accessible par Amazon dans la même instance VPC — Une utilisation courante d'une instance de base de données dans un VPC est de partager des données avec un serveur d'applications qui s'exécute dans une EC2 instance de la même instanceVPC. L'EC2instance peut exécuter un serveur Web avec une application qui interagit avec l'instance de base de données.
- Une instance de base de données dans une EC2 instance VPC accessible par une instance différente VPC — Dans certains cas, votre instance de base de données se trouve dans une EC2 instance VPC différente de celle que vous utilisez pour y accéder. Si tel est le cas, vous pouvez utiliser le VPC peering pour accéder à l'instance de base de données.
- Une instance de base de données dans une application cliente VPC accessible via Internet — Pour accéder à une instance de base de données VPC depuis une application client via Internet, vous configurez une instance VPC avec un seul sous-réseau public. Vous configurez également une passerelle Internet pour permettre la communication via Internet.

Pour se connecter à une instance de base de données depuis l'extérieur de celle-ciVPC, l'instance de base de données doit être accessible au public. En outre, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité de l'instance de base de données, et d'autres exigences doivent être satisfaites. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

- Une instance de base de données dans un réseau privé VPC accessible — Si votre instance de base de données n'est pas accessible au public, vous pouvez utiliser l'une des options suivantes pour y accéder depuis un réseau privé :
  - Une AWS Site-to-Site VPN connexion
  - Une AWS Direct Connect connexion
  - Une AWS Client VPN connexion

Pour de plus amples informations, veuillez consulter Scénarios d'accès à un d'instances de base de données dans un VPC.

Connexion aux instances de base de données avec les AWS pilotes

La AWS suite de pilotes a été conçue pour accélérer les temps de basculement et de basculement, ainsi que pour l'authentification avec AWS Secrets Manager, AWS Identity and Access Management (IAM) et Federated Identity. Les AWS pilotes s'appuient sur la surveillance de l'état de l'instance de base de données et sur la connaissance de la topologie de l'instance pour déterminer la nouvelle instance principale. Cette approche réduit les temps de basculement et de basculement à un chiffre, contre des dizaines de secondes pour les pilotes open source.

Le tableau suivant répertorie les fonctionnalités prises en charge pour chacun des pilotes. À mesure que de nouvelles fonctionnalités de service sont introduites, l'objectif de la AWS suite de pilotes est de fournir un support intégré pour ces fonctionnalités de service.

Fonctionnalité	AWS JDBCChauffeur	AWS Pilote Python	AWS ODBCDriver for My SQL
Assistance en cas de basculement	Oui	Oui	Oui

Fonctionnalité	AWS JDBCChauffeur	AWS Pilote Python	AWS ODBCDriver for My SQL
Surveillance améliorée du basculement	Oui	Oui	Oui
Séparation en lecture/écriture	Oui	Oui	Non
Connexion aux métadonnées du pilote	Oui	N/A	N/A
Télémétrie	Oui	Oui	Non
Secrets Manager	Oui	Oui	Oui
IAMauthentification	Oui	Oui	Oui
Identité fédérée (AD FS)	Oui	Oui	Non
Identité fédérée (Okta)	Oui	Non	Non
Clusters de base de données multi-AZ	Oui	Oui	Non

Pour plus d'informations sur les AWS pilotes, consultez le pilote de langue correspondant à votre instance de base de données RDSpour MariaDB RDS, pour SQL My RDSou pour SQL Postgre.

> **Note**
>
> Les seules fonctionnalités prises en charge RDS pour MariaDB sont l'authentification AWS Secrets Manager avec AWS Identity and Access Management , IAM () et l'identité fédérée.

Connexion à une instance de base de données qui exécute un moteur de base de données spécifique

Pour savoir comment vous connecter à une instance de base de données qui exécute un moteur de base de données spécifique, suivez les instructions relatives à votre moteur de base de données :

- RDSpour Db2
- RDSpour MariaDB
- RDSpour SQL serveur
- RDSpour My SQL
- RDSpour Oracle
- RDSpour Postgre SQL

Gestion des connexions avec RDS Proxy

Vous pouvez également utiliser Amazon RDS Proxy RDS pour gérer les connexions aux instances de base de données MariaDB, RDS SQL Microsoft ServerRDS, SQL My RDS et Postgre. SQL RDSLe proxy permet aux applications de regrouper et de partager des connexions à des bases de données afin d'améliorer l'évolutivité. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .

Options d'authentification de base de données

Amazon RDS prend en charge les méthodes suivantes pour authentifier les utilisateurs de la base de données :

- Authentification par mot de passe, – Votre instance de base de données effectue toute l'administration des comptes d'utilisateurs. Vous créez des utilisateurs et spécifiez des mots de passe à l'aide d'SQLinstructions. Les SQL instructions que vous pouvez utiliser dépendent de votre moteur de base de données.
- AWS Identity and Access Management (IAM) authentification de base de données : vous n'avez pas besoin d'utiliser de mot de passe lorsque vous vous connectez à une instance de base de données. En revanche, un jeton d'authentification est nécessaire.
- Authentification Kerberos – Vous utilisez l'authentification externe des utilisateurs de base de données avec Kerberos et Microsoft Active Directory. Kerberos est un protocole d'authentification réseau qui utilise les tickets et la cryptographie de clé symétrique pour vous éviter d'acheminer vos

mots de passe via le réseau. Intégré dans Active Directory, Kerberos est conçu pour authentifier les utilisateurs sur les ressources réseau, par exemple les bases de données.

IAMl'authentification de base de données et l'authentification Kerberos ne sont disponibles que pour des moteurs et des versions de base de données spécifiques.

Pour de plus amples informations, veuillez consulter Authentification de base de données avec Amazon RDS.

Connexions chiffrées

Vous pouvez utiliser Secure Socket Layer (SSL) ou Transport Layer Security (TLS) depuis votre application pour chiffrer une connexion à une instance de base de données. Chaque moteur de base de données possède son propre processus pour implémenterSSL/TLS. Pour de plus amples informations, veuillez consulter .

Utilisation de groupes d'options

Certains moteurs de base de données offrent des fonctions supplémentaires qui facilitent la gestion des données et des bases de données ainsi que la fourniture d'une sécurité supplémentaire pour votre base de données. Amazon RDS utilise des groupes d'options pour activer et configurer ces fonctionnalités. Un groupe d'options peut spécifier des fonctionnalités, appelées options, disponibles pour une RDS instance de base de données Amazon particulière. Les options peuvent avoir des paramètres spécifiant le mode de fonctionnement de l'option. Lorsque vous associez une instance de base de données a un groupe d'options, les paramètres d'options et les options spécifiés sont activés pour cette instance de base de données.

Amazon RDS prend en charge les options pour les moteurs de base de données suivants :

Moteur de base de données	Documentation utile
`Db2`	Options RDS pour les instances de base de données DB2
`MariaDB`	Options pour le moteur de base de données MariaDB
`Microsoft SQL Server`	Options pour le moteur de base de données Microsoft SQL Server
`MySQL`	Options pour les instances My SQL DB
`Oracle`	Ajout d'options aux instances de base de données Oracle
`PostgreSQL`	Postgre n'utilise SQL pas d'options ni de groupes d'options. Postgre SQL utilise des extensions et des modules pour fournir des fonctionnalités supplémentaires. Pour de plus amples informations, veuillez consulter Versions d'SQLextension Postgre prises en charge.

Présentation des groupes d'options

Amazon RDS fournit un groupe d'options par défaut vide pour chaque nouvelle instance de base de données. Vous ne pouvez ni modifier ni supprimer ce groupe d'options par défaut, mais tout nouveau groupe d'options que vous créez tire ses paramètres du groupe d'options par défaut. Pour appliquer une option à une instance de base de données, vous devez effectuer les opérations suivantes :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajouter une ou plusieurs options au groupe.
3. Associez le groupe d'options à l'instance de base de données.

   Pour associer un groupe d'options à une instance de base de données, modifiez l'instance de base de données. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Les instances de base de données et les snapshots DB peuvent tous deux être associés à un groupe d'options. Dans certains cas, vous pouvez effectuer une restauration à partir d'un instantané de base de données ou effectuer une point-in-time restauration pour une instance de base de données. Dans ces cas, le groupe d'options associé à l'instantané de base de données ou à l'instance de base de données est associé, par défaut, à l'instance de base de données restaurée. Vous pouvez associer un groupe d'options différent à une instance de base de données restaurée. Toutefois, le nouveau groupe d'options doit contenir toutes les options persistantes ou permanentes qui étaient incluses dans le groupe d'origine. Les options persistantes et permanentes sont décrites ci-dessous.

Les options nécessitent d'avantage de mémoire pour s'exécuter sur une instance de base de données. Ainsi, selon votre utilisation actuelle de l'instance de base de données, il vous faudra peut-être lancer une instance de plus grande taille pour les utiliser. Par exemple, Oracle Enterprise Manager Database Control utilise environ 300 Mo deRAM. Si vous activez cette option pour une petite instance de base de données, vous risquez de rencontrer des problèmes de performances ou out-of-memory des erreurs.

Options persistantes et permanentes

Deux types d'options, persistantes et permanentes, nécessitent une considération spéciale lorsque vous les ajoutez à un groupe d'options.

Les options persistantes ne peuvent pas être supprimées d'un groupe d'options tant que des instances de base de données sont associées au groupe d'options. L'option de chiffrement transparent des données de Microsoft SQL Server (TDE) est un exemple d'TDEoption persistante. Pour qu'une option persistante puisse être supprimée, vous devez dissocier toutes les instances de base de données du groupe d'options. Dans certains cas, vous pouvez restaurer ou effectuer une point-in-time restauration à partir d'un instantané de base de données. Dans ces cas, si le groupe d'options associé à cet instantané de base de données contient une option persistante, vous ne pouvez associer l'instance de base de données restaurée qu'à ce groupe d'options.

Les options permanentes, telles que l'TDEoption pour Oracle Advanced SecurityTDE, ne peuvent jamais être supprimées d'un groupe d'options. Vous pouvez cependant modifier le groupe d'options d'une instance de base de données qui utilise l'option permanente. Notez cependant que le groupe d'options associé à l'instance de base de données doit inclure la même option permanente. Dans certains cas, vous pouvez restaurer ou effectuer une point-in-time restauration à partir d'un instantané de base de données. Dans ces cas, si le groupe d'options associé à cet instantané de base de données contient une option permanente, vous ne pouvez associer l'instance de base de données restaurée qu'à un groupe d'options avec cette option permanente.

Pour les instances de base de données Oracle, vous pouvez copier les instantanés de bases de données partagés ayant les options `Timezone` ou `OLS` (ou les deux). Pour ce faire, spécifiez un groupe d'options cibles qui inclut ces options lorsque vous copiez l'instantané de bases de données. L'OLSoption est permanente et persistante uniquement pour les instances de base de données Oracle exécutant Oracle version 12.2 ou supérieure. Pour de plus amples informations sur ces options, veuillez consulter Fuseau horaire Oracle et Oracle Label Security.

VPCconsidérations

Le groupe d'options associé à l'instance de base de données est lié à celui de l'instance de base de donnéesVPC. Cela signifie que vous ne pouvez pas utiliser le groupe d'options attribué à une instance de base de données si vous essayez de restaurer l'instance sur une autre instanceVPC. Si vous restaurez une instance de base de données sur une autre instanceVPC, vous pouvez effectuer l'une des opérations suivantes :

- Affectez le groupe d'options par défaut à l'instance de base de données.
- Attribuez un groupe d'options qui y est liéVPC.
- Créez un nouveau groupe d'options et affectez-le à l'instance de base de données.

Avec les options persistantes ou permanentes, telles qu'OracleTDE, vous devez créer un nouveau groupe d'options. Ce groupe d'options doit inclure l'option persistante ou permanente lors de la restauration d'une instance de base de données dans une autre instanceVPC.

Les paramètres d'options contrôlent le comportement d'une option. Par exemple, l'option Oracle Advanced Security `NATIVE_NETWORK_ENCRYPTION` a un paramètre que vous pouvez utiliser pour spécifier l'algorithme de chiffrement pour le trafic réseau vers et depuis l'instance de base de données. Certains paramètres d'options sont optimisés pour une utilisation avec Amazon RDS et ne peuvent pas être modifiés.

Options mutuellement exclusives

Certaines options sont mutuellement exclusives. Vous pouvez utiliser l'une ou l'autre, mais pas les deux en même temps. Les options suivantes sont mutuellement exclusives :

- Oracle Enterprise Manager Database Express et Oracle Management Agent pour Enterprise Manager Cloud Control.
- Oracle NNE (Native Network Encryption) et Oracle Secure Sockets Layer (SSL).

Création d'un groupe d'options

Vous pouvez créer un nouveau groupe d'options qui tire ses paramètres du groupe d'options par défaut. Vous ajoutez ensuite une ou plusieurs options au nouveau groupe d'options. Ou si vous avez déjà un groupe d'options existant, vous pouvez également copier ce groupe avec toutes ses options dans un nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Copie d'un groupe d'options.

Une fois que vous avez créé un groupe d'options, il ne contient pas d'options. Pour savoir comment ajouter des options au groupe d'options, consultez la page Ajout d'une option à un groupe d'options. Une fois que vous avez ajouté les options que vous voulez, vous pouvez associer une instance de base de données au groupe d'options. De cette façon, les options deviennent disponibles sur l'instance de base de données. Pour plus d'informations sur l'association d'un groupe d'options à une instance de base de données, consultez la documentation pour votre moteur dans Utilisation de groupes d'options.

Console

Une manière de créer un groupe d'options consiste à utiliser la AWS Management Console.

Pour créer un nouveau groupe d'options à l'aide de la console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :

    a. Dans Nom, saisissez un nom unique au sein de votre AWS compte pour le groupe d'options. Le nom ne peut contenir que des lettres, des chiffres et des tirets.

b. Pour Description, saisissez une brève description du groupe d'options. La description est utilisée à des fins d'affichage.

c. Dans la zone Moteur, choisissez le moteur de base de données que vous souhaitez.

d. Dans la zone Version majeure du moteur, choisissez la version majeure du moteur de base de données que vous souhaitez.

5. Pour continuer, choisissez Créer. Pour annuler l'opération à la place, choisissez Cancel (Annuler).

AWS CLI

Pour créer un groupe d'options, utilisez la AWS CLI `create-option-group`commande avec les paramètres obligatoires suivants.

- `--option-group-name`
- `--engine-name`
- `--major-engine-version`
- `--option-group-description`

Example

L'exemple suivant crée un groupe d'options nommé `testoptiongroup` qui est associé au moteur de base de données Oracle Enterprise Edition. La description est entre guillemets.

Pour LinuxmacOS, ou Unix :

```
aws rds create-option-group \
    --option-group-name testoptiongroup \
    --engine-name oracle-ee \
    --major-engine-version 19 \
    --option-group-description "Test option group for Oracle Database 19c EE"
```

Dans Windows :

```
aws rds create-option-group ^
    --option-group-name testoptiongroup ^
    --engine-name oracle-ee ^-
    --major-engine-version 19 ^
    --option-group-description "Test option group for Oracle Database 19c EE"
```

RDS API

Pour créer un groupe d'options, appelez l'RDSAPI`CreateOptionGroup`opération Amazon. Incluez les paramètres suivants :

- `OptionGroupName`
- `EngineName`
- `MajorEngineVersion`
- `OptionGroupDescription`

Copie d'un groupe d'options

Vous pouvez utiliser le groupe d'options AWS CLI ou Amazon RDS API copier un groupe d'options. La copie d'un groupe d'options peut s'avérer pratique. Par exemple, lorsque vous avez un groupe d'options existant et que vous souhaitez inclure la plupart de ses valeurs et paramètres personnalisés dans un nouveau groupe d'options. Vous pouvez aussi faire une copie d'un groupe d'options que vous utilisez en production, puis modifier la copie pour tester d'autres paramètres d'options.

> **Note**
>
> Actuellement, vous ne pouvez pas copier un groupe d'options dans une autre AWS région.

AWS CLI

Pour copier un groupe d'options, utilisez la AWS CLI copy-option-groupcommande. Inclure les options requises suivantes :

- `--source-option-group-identifier`
- `--target-option-group-identifier`

- `--target-option-group-description`

Example

L'exemple suivant crée un groupe d'options nommé `new-option-group` qui est une copie locale du groupe d'options `my-option-group`.

Pour LinuxmacOS, ou Unix :

```
aws rds copy-option-group \
    --source-option-group-identifier my-option-group \
    --target-option-group-identifier new-option-group \
    --target-option-group-description "My new option group"
```

Dans Windows :

```
aws rds copy-option-group ^
    --source-option-group-identifier my-option-group ^
    --target-option-group-identifier new-option-group ^
    --target-option-group-description "My new option group"
```

RDS API

Pour copier un groupe d'options, appelez l'RDSAPICopyOptionGroupopération Amazon. Incluez les paramètres requis suivants.

- `SourceOptionGroupIdentifier`
- `TargetOptionGroupIdentifier`
- `TargetOptionGroupDescription`

Ajout d'une option à un groupe d'options

Vous pouvez ajouter une option à un groupe d'options existant. Une fois que vous avez ajouté les options que vous voulez, vous pouvez associer une instance de base de données au groupe d'options afin que ces dernières deviennent disponibles sur l'instance de base de données. Pour plus d'informations sur l'association d'un groupe d'options à une instance de base de données, consultez la documentation pour votre moteur de base de données spécifique répertorié à l'adresse Utilisation de groupes d'options.

Les modifications apportées au groupe d'options doivent être appliquées immédiatement dans deux cas :

- Lorsque vous ajoutez une option qui ajoute ou met à jour une valeur de port, telle que l'option `OEM`.
- Lorsque vous ajoutez ou supprimez un groupe d'options avec une option qui inclut une valeur de port.

Dans ces cas, choisissez l'option Appliquer immédiatement dans la console. Vous pouvez également inclure l'`--apply-immediately`option lorsque vous utilisez le AWS CLI ou définir le `ApplyImmediately` paramètre sur `true` lorsque vous utilisez Amazon RDSAPI. Les options qui n'incluent pas de valeur de port peuvent être appliquées immédiatement ou pendant la fenêtre de maintenance suivant pour l'instance de base de données.

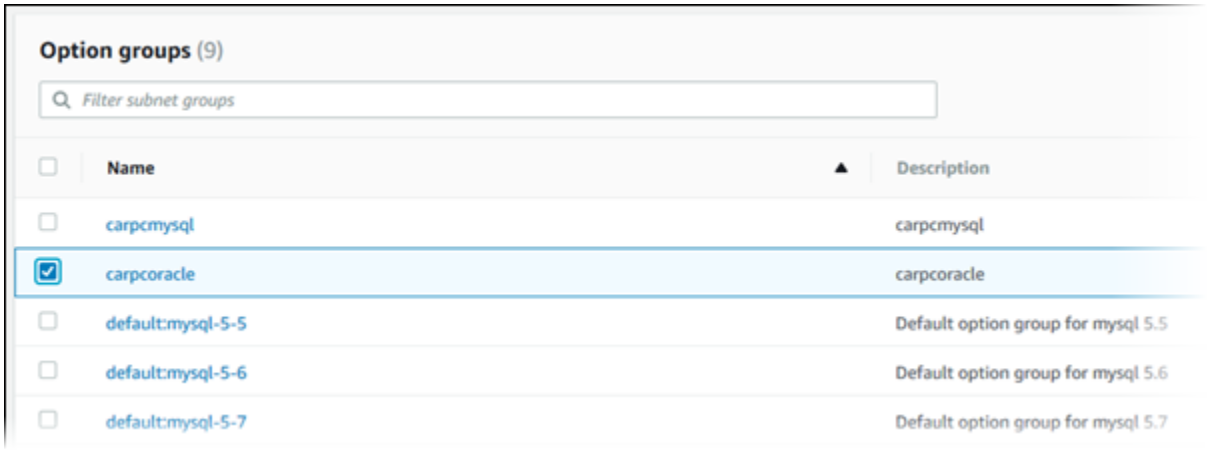
> **Note**
>
> Si vous spécifiez un groupe de sécurité comme valeur d'une option dans un groupe d'options, gérez le groupe de sécurité en modifiant le groupe d'options. Vous ne pouvez pas modifier ou supprimer ce groupe de sécurité en modifiant une instance de base de données. De plus, le groupe de sécurité n'apparaît pas dans les détails de l'instance de base de données dans AWS Management Console ou dans la sortie de la AWS CLI commande`describe-db-instances`.

Console

Vous pouvez utiliser le AWS Management Console pour ajouter une option à un groupe d'options.

Pour ajouter une option à un groupe d'options à l'aide de la console

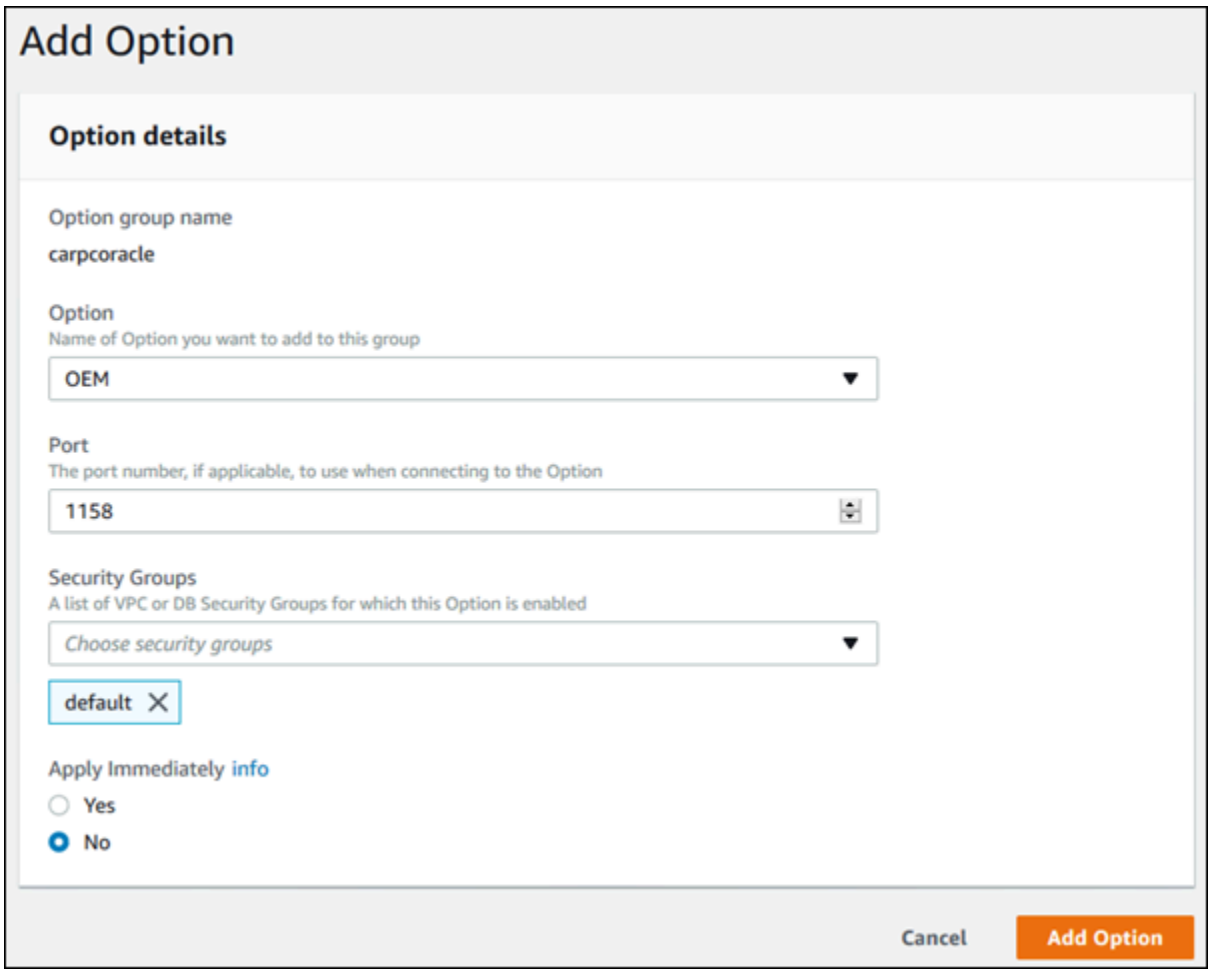
1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Cochez la case pour le groupe d'options que vous souhaitez modifier, puis choisissez Ajouter une option.



4. Dans la fenêtre Ajouter une option, procédez comme suit :

   a. Choisissez l'option que vous souhaitez ajouter. Selon l'option que vous sélectionnez, vous devrez peut-être fournir des valeurs supplémentaires. Par exemple, lorsque vous sélectionnez l'option `OEM`, vous devez également saisir une valeur de port et spécifier un groupe de sécurité.

   b. Pour activer l'option sur toutes les instances de base de données associées dès que vous l'ajoutez, pour Apply Immediately (Appliquer immédiatement), choisissez Oui. Si vous choisissez Non (valeur par défaut), l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante.



5. Lorsque les paramètres vous conviennent, choisissez Ajouter une option.

AWS CLI

Pour ajouter une option à un groupe d'options, exécutez la commande AWS CLI add-option-to-option-group avec l'option que vous souhaitez ajouter. Pour activer immédiatement la nouvelle option sur toutes les instances de base de données associées, incluez le paramètre `--apply-immediately`. Par défaut, l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Incluez le paramètre requis suivant :

- `--option-group-name`

Example

L'exemple suivant ajoute l'`Timezone`option, avec le `America/Los_Angeles` paramètre, à un groupe d'options nommé `testoptiongroup` et l'active immédiatement.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name testoptiongroup \
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=America/
Los_Angeles}]" \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name testoptiongroup ^
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=America/
Los_Angeles}]" ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
...{
       "OptionName": "Timezone",
       "OptionDescription": "Change time zone",
       "Persistent": true,
       "Permanent": false,
       "OptionSettings": [
           {
               "Name": "TIME_ZONE",
               "Value": "America/Los_Angeles",
               "DefaultValue": "UTC",
               "Description": "Specifies the timezone the user wants to change the
 system time to",
               "ApplyType": "DYNAMIC",
               "DataType": "STRING",
               "AllowedValues": "Africa/Cairo,...",
               "IsModifiable": true,
               "IsCollection": false
           }
       ],
```

```
        "DBSecurityGroupMemberships": [],
        "VpcSecurityGroupMemberships": []
    }...
```

Example

L'exemple suivant ajoute l'OEMoption Oracle à un groupe d'options. Il spécifie également un port personnalisé et une paire de groupes de EC2 VPC sécurité Amazon à utiliser pour ce port.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name testoptiongroup \
    --options OptionName=OEM,Port=5500,VpcSecurityGroupMemberships="sg-test1,sg-test2"
 \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name testoptiongroup ^
    --options OptionName=OEM,Port=5500,VpcSecurityGroupMemberships="sg-test1,sg-test2"
 ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
OPTIONGROUP  False  oracle-ee  19 arn:aws:rds:us-east-1:1234567890:og:testoptiongroup
 Test Option Group  testoptiongroup vpc-test
OPTIONS Oracle 12c EM Express   OEM     False   False   5500
VPCSECURITYGROUPMEMBERSHIPS     active  sg-test1
VPCSECURITYGROUPMEMBERSHIPS     active  sg-test2
```

Example

L'exemple suivant ajoute l'option Oracle `NATIVE_NETWORK_ENCRYPTION` à un groupe d'options et spécifie les paramètres de l'option. Si aucun paramètre d'options n'est spécifié, les valeurs par défaut sont utilisées.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
```

```
    --option-group-name testoptiongroup \
    --options '[{"OptionSettings":
[{"Name":"SQLNET.ENCRYPTION_SERVER","Value":"REQUIRED"},
{"Name":"SQLNET.ENCRYPTION_TYPES_SERVER","Value":"AES256,AES192,DES"}],"OptionName":"NATIVE_NET
 \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name testoptiongroup ^
    --options "OptionSettings"=[{"Name"="SQLNET.ENCRYPTION_SERVER","Value"="REQUIRED"},
{"Name"="SQLNET.ENCRYPTION_TYPES_SERVER","Value"="AES256\,AES192\,DES"}],"OptionName"="NATIVE_N
 ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
...{
    "OptionName": "NATIVE_NETWORK_ENCRYPTION",
    "OptionDescription": "Native Network Encryption",
    "Persistent": false,
    "Permanent": false,
    "OptionSettings": [
        {
            "Name": "SQLNET.ENCRYPTION_TYPES_SERVER",
             "Value": "AES256,AES192,DES",
            "DefaultValue":
 "RC4_256,AES256,AES192,3DES168,RC4_128,AES128,3DES112,RC4_56,DES,RC4_40,DES40",
            "Description": "Specifies list of encryption algorithms in order of
 intended use",
            "ApplyType": "STATIC",
            "DataType": "STRING",
            "AllowedValues":
 "RC4_256,AES256,AES192,3DES168,RC4_128,AES128,3DES112,RC4_56,DES,RC4_40,DES40",
            "IsModifiable": true,
            "IsCollection": true
        },
        {
            "Name": "SQLNET.ENCRYPTION_SERVER",
            "Value": "REQUIRED",
            "DefaultValue": "REQUESTED",
            "Description": "Specifies the desired encryption behavior",
```

```
            "ApplyType": "STATIC",
            "DataType": "STRING",
            "AllowedValues": "ACCEPTED,REJECTED,REQUESTED,REQUIRED",
            "IsModifiable": true,
            "IsCollection": false
        },...
```

RDS API

Pour ajouter une option à un groupe d'options à l'aide d'Amazon RDSAPI, appelez l'ModifyOptionGroupopération avec l'option que vous souhaitez ajouter. Pour activer immédiatement la nouvelle option sur toutes les instances de base de données associées, incluez le paramètre `ApplyImmediately` et affectez-lui la valeur `true`. Par défaut, l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Incluez le paramètre requis suivant :

- `OptionGroupName`

Liste des options et des paramètres d'options pour un groupe d'options

Vous pouvez répertorier toutes les options et tous les paramètres d'options pour un groupe d'options.

Console

Vous pouvez utiliser le AWS Management Console pour répertorier toutes les options et tous les paramètres d'options d'un groupe d'options.

Pour répertorier les options et les paramètres d'options pour un groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Sélectionnez le nom du groupe d'options pour afficher ses informations détaillées. Les options et les paramètres d'options du groupe d'options sont affichés.

AWS CLI

Pour répertorier les options et les paramètres d'options d'un groupe d'options, utilisez la AWS CLI `describe-option-groups`commande. Spécifiez le nom du groupe d'options dont vous souhaitez

afficher les options et les paramètres. Si vous ne spécifiez pas un nom de groupe d'options, tous les groupes d'options sont décrits.

Example

L'exemple suivant répertorie les options et les paramètres d'options pour tous les groupes d'options.

```
aws rds describe-option-groups
```

Example

L'exemple suivant répertorie les options et les paramètres d'options pour un groupe d'options nommé `testoptiongroup`.

```
aws rds describe-option-groups --option-group-name testoptiongroup
```

RDS API

Pour répertorier les options et les paramètres d'options d'un groupe d'options, utilisez l'RDSAPI`DescribeOptionGroups`opération Amazon. Spécifiez le nom du groupe d'options dont vous souhaitez afficher les options et les paramètres. Si vous ne spécifiez pas un nom de groupe d'options, tous les groupes d'options sont décrits.

Modification d'un paramètre d'option

Après que vous avez ajouté une option dont les paramètres d'options sont modifiables, vous pouvez modifier les paramètres à tout moment. Si vous modifiez des options ou des paramètres d'options dans un groupe d'options, ces modifications sont appliquées à toutes les instances DB qui sont associées à ce groupe d'options. Pour plus d'informations sur les paramètres disponibles pour les différentes options, consultez la documentation pour votre moteur dans Utilisation de groupes d'options.

Les modifications apportées au groupe d'options doivent être appliquées immédiatement dans deux cas :

- Lorsque vous ajoutez une option qui ajoute ou met à jour une valeur de port, telle que l'option `OEM`.
- Lorsque vous ajoutez ou supprimez un groupe d'options avec une option qui inclut une valeur de port.

Dans ces cas, choisissez l'option Appliquer immédiatement dans la console. Vous pouvez également inclure l'`--apply-immediately`option lorsque vous utilisez le AWS CLI ou définir le `ApplyImmediately` paramètre sur `true` lors de l'utilisation du RDSAPI. Les options qui n'incluent pas de valeur de port peuvent être appliquées immédiatement ou pendant la fenêtre de maintenance suivant pour l'instance de base de données.

> **Note**
>
> Si vous spécifiez un groupe de sécurité comme valeur d'une option dans un groupe d'options, vous gérez le groupe de sécurité en modifiant le groupe d'options. Vous ne pouvez pas modifier ou supprimer ce groupe de sécurité en modifiant une instance de base de données. De plus, le groupe de sécurité n'apparaît pas dans les détails de l'instance de base de données dans AWS Management Console ou dans la sortie de la AWS CLI commande`describe-db-instances`.

Console

Vous pouvez utiliser le AWS Management Console pour modifier un paramètre d'option.

Pour modifier un paramètre d'option à l'aide de la console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Sélectionnez le groupe d'options que vous souhaitez modifier, et choisissez Modify option (Modifier une option).
4. Dans la fenêtre Modify option (Modifier une option), dans la zone Options installées, choisissez l'option dont vous souhaitez modifier le paramètre. Apportez les modifications que vous souhaitez.
5. Pour activer l'option dès que vous l'ajoutez, pour Apply Immediately (Appliquer immédiatement), choisissez Oui. Si vous choisissez Non (valeur par défaut), l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante.
6. Lorsque les paramètres vous conviennent, choisissez Modify Option (Modifier l'option).

AWS CLI

Pour modifier un paramètre d'option, utilisez la AWS CLI [add-option-to-option-group](#)commande avec le groupe d'options et l'option que vous souhaitez modifier. Par défaut, l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Pour appliquer la modification immédiatement à toutes les instances de base de données associées, incluez le paramètre `--apply-immediately`. Pour modifier un paramètre d'option, utilisez l'argument `--settings`.

Example

L'exemple suivant modifie le port utilisé par Oracle Enterprise Manager Database Control (OEM) dans un groupe d'options nommé `testoptiongroup` et applique immédiatement la modification.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name testoptiongroup \
    --options OptionName=OEM,Port=5432,DBSecurityGroupMemberships=default \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name testoptiongroup ^
    --options OptionName=OEM,Port=5432,DBSecurityGroupMemberships=default ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
OPTIONGROUP   False  oracle-ee  19  arn:aws:rds:us-east-1:1234567890:og:testoptiongroup
   Test Option Group    testoptiongroup
OPTIONS Oracle 12c EM Express   OEM     False   False   5432
DBSECURITYGROUPMEMBERSHIPS   default  authorized
```

Example

L'exemple suivant modifie l'option Oracle `NATIVE_NETWORK_ENCRYPTION` et ses paramètres.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
```

```
    --option-group-name testoptiongroup \
    --options '[{"OptionSettings":
[{"Name":"SQLNET.ENCRYPTION_SERVER","Value":"REQUIRED"},
{"Name":"SQLNET.ENCRYPTION_TYPES_SERVER","Value":"AES256,AES192,DES,RC4_256"}],"OptionName":"NA
 \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name testoptiongroup ^
    --options "OptionSettings"=[{"Name"="SQLNET.ENCRYPTION_SERVER","Value"="REQUIRED"},
{"Name"="SQLNET.ENCRYPTION_TYPES_SERVER","Value"="AES256\,AES192\,DES
\,RC4_256"}],"OptionName"="NATIVE_NETWORK_ENCRYPTION" ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
OPTIONGROUP   False  oracle-ee  19  arn:aws:rds:us-east-1:1234567890:og:testoptiongroup
   Test Option Group    testoptiongroup
OPTIONS Oracle Advanced Security - Native Network Encryption
 NATIVE_NETWORK_ENCRYPTION       False   False
OPTIONSETTINGS
 RC4_256,AES256,AES192,3DES168,RC4_128,AES128,3DES112,RC4_56,DES,RC4_40,DES40 STATIC
 STRING
   RC4_256,AES256,AES192,3DES168,RC4_128,AES128,3DES112,RC4_56,DES,RC4_40,DES40
 Specifies list of encryption algorithms in order of intended use
   True     True    SQLNET.ENCRYPTION_TYPES_SERVER  AES256,AES192,DES,RC4_256
OPTIONSETTINGS  ACCEPTED,REJECTED,REQUESTED,REQUIRED    STATIC  STRING  REQUESTED
 Specifies the desired encryption behavior  False   True  SQLNET.ENCRYPTION_SERVER
 REQUIRED
OPTIONSETTINGS  SHA1,MD5   STATIC  STRING  SHA1,MD5    Specifies list of
 checksumming algorithms in order of intended use      True    True
 SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER  SHA1,MD5
OPTIONSETTINGS  ACCEPTED,REJECTED,REQUESTED,REQUIRED  STATIC  STRING
 REQUESTED     Specifies the desired data integrity behavior   False   True
 SQLNET.CRYPTO_CHECKSUM_SERVER  REQUESTED
```

RDS API

Pour modifier un paramètre d'option, utilisez la RDS API [ModifyOptionGroup](#)commande Amazon avec le groupe d'options et l'option que vous souhaitez modifier. Par défaut, l'option est activée pour

chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Pour appliquer la modification immédiatement à toutes les instances de base de données associées, incluez le paramètre `ApplyImmediately` et affectez-lui la valeur `true`.

Suppression d'une option d'un groupe d'options

Certaines options peuvent être supprimées d'un groupe d'options, et certaines ne le peuvent pas. Il n'est pas possible de supprimer une option persistante d'un groupe d'options tant que l'ensemble des instances de base de données associées à ce groupe d'options est dissocié. Une option permanente ne peut jamais être supprimée d'un groupe d'options. Pour plus d'informations sur les options pouvant être supprimées, consultez la documentation pour votre moteur spécifique répertoriée à l'adresse Utilisation de groupes d'options.

Si vous supprimez toutes les options d'un groupe d'options, Amazon RDS ne supprime pas le groupe d'options. Les instances de base de données qui sont associées au groupe d'options vide continuent de l'être. Elles ne comportent simplement plus aucune option active. Pour supprimer toutes les options d'une instance de base de données simultanément, vous pouvez aussi associer l'instance de base de données au groupe d'options par défaut (vide).

Console

Vous pouvez utiliser le AWS Management Console pour supprimer une option d'un groupe d'options.

Pour supprimer une option à partir d'un groupe d'options à l'aide de la console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Sélectionnez le groupe d'options que vous souhaitez supprimer, et choisissez Delete option (Supprimer l'option).
4. Dans la fenêtre Delete option (Supprimer l'option), procédez comme suit :
   - Cochez la case correspondant à l'option à supprimer.
   - Pour que la suppression entre en vigueur dès que vous l'effectuez, pour Apply immediately (Appliquer immédiatement), choisissez Oui. Si vous choisissez Non (valeur par défaut), l'option est supprimée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante.



5. Lorsque les paramètres vous conviennent, choisissez Yes, Delete (Oui, supprimer).

AWS CLI

Pour supprimer une option d'un groupe d'options, utilisez la AWS CLI `remove-option-from-option-group`commande associée à l'option que vous souhaitez supprimer. Par défaut, l'option est supprimée de chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Pour appliquer immédiatement la modification, incluez le paramètre `--apply-immediately`.

Example

L'exemple suivant supprime l'option Oracle Enterprise Manager Database Control (OEM) d'un groupe d'options nommé `testoptiongroup` et applique immédiatement la modification.

Pour LinuxmacOS, ou Unix :

```
aws rds remove-option-from-option-group \
    --option-group-name testoptiongroup \
    --options OEM \
    --apply-immediately
```

Dans Windows :

```
aws rds remove-option-from-option-group ^
    --option-group-name testoptiongroup ^
    --options OEM ^
    --apply-immediately
```

Le résultat de la commande est similaire à ce qui suit :

```
OPTIONGROUP    testoptiongroup oracle-ee   19    Test option group
```

RDS API

Pour supprimer une option d'un groupe d'options, utilisez l'RDSAPI`ModifyOptionGroup`action Amazon. Par défaut, l'option est supprimée de chaque instance de base de données associée pendant sa fenêtre de maintenance suivante. Pour appliquer la modification immédiatement, incluez le paramètre `ApplyImmediately` et affectez-lui la valeur `true`.

Incluez les paramètres suivants :

- `OptionGroupName`
- `OptionsToRemove.OptionName`

Suppression d'un groupe d'options

Vous ne pouvez supprimer un groupe d'options que s'il répond aux critères suivants :

- Il n'est associé à aucune RDS ressource Amazon. Un groupe d'options peut être associé à une instance de base de données, un instantané de base de données manuel ou un instantané de base de données automatique.
- Il ne s'agit pas d'un groupe d'options par défaut.

Pour identifier les groupes d'options utilisés par vos instances de base de données et vos instantanés de base de données, vous pouvez utiliser les CLI commandes suivantes :

```
aws rds describe-db-instances \
    --query 'DBInstances[*].
[DBInstanceIdentifier,OptionGroupMemberships[].OptionGroupName]'
```

```
aws rds describe-db-snapshots | jq -r '.DBSnapshots[] | "\(.DBInstanceIdentifier),
\(.OptionGroupName)"' | sort | uniq
```

Si vous essayez de supprimer un groupe d'options associé à une RDS ressource, une erreur comme celle-ci est renvoyée.

```
An error occurred (InvalidOptionGroupStateFault) when calling the DeleteOptionGroup
 operation: The option group 'optionGroupName' cannot be deleted because it is in use.
```

Pour rechercher les RDS ressources Amazon associées à un groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Sélectionnez le nom du groupe d'options pour afficher ses informations détaillées.
4. Consultez la section Instances associées et instantanés pour connaître les RDS ressources Amazon associées.

Si une instance de base de données est associée au groupe d'options, modifiez-la afin qu'elle utilise un autre groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Si un instantané de base de données manuel est associé au groupe d'options, modifiez-le afin qu'il utilise un autre groupe d'options. Vous pouvez le faire à l'aide de la AWS CLI `modify-db-snapshot`commande.

> **Note**
>
> Vous ne pouvez pas modifier le groupe d'options d'un instantané de base de données automatique.

Console

Une manière de créer un groupe d'options consiste à utiliser l AWS Management Console.

Pour supprimer un groupe d'options à l'aide de la console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options.
4. Choisissez Supprimer un groupe.
5. Dans la page de confirmation, choisissez Supprimer pour terminer la suppression du groupe d'options ou Annuler pour annuler la suppression.

AWS CLI

Pour supprimer un groupe d'options, utilisez la AWS CLI `delete-option-group`commande avec le paramètre obligatoire suivant.

- `--option-group-name`

Example

L'exemple suivant supprime un groupe d'options nommé `testoptiongroup`.

Pour LinuxmacOS, ou Unix :

```
aws rds delete-option-group \
    --option-group-name testoptiongroup
```

Dans Windows :

```
aws rds delete-option-group ^
    --option-group-name testoptiongroup
```

RDS API

Pour supprimer un groupe d'options, appelez l'RDSAPI`DeleteOptionGroup`opération Amazon. Incluez le paramètre suivant :

- `OptionGroupName`

Groupes de paramètres pour (Amazon RDS)

Les paramètres de base de données spécifient comment la base de données est configurée. Par exemple, les paramètres de base de données peuvent spécifier la quantité de ressources, telles que la mémoire, à allouer à une base de données.

Vous gérez la configuration de votre base de données en associant vos instances de base de données et clusters de base de données Multi-AZ à des groupes de paramètres. Amazon RDS définit des groupes de paramètres avec des paramètres par défaut. Vous pouvez également définir vos propres groupes de paramètres à l'aide de paramètres personnalisés.

> **Note**
>
> Certains moteurs de base de données offrent des fonctions supplémentaires que vous pouvez ajouter à votre base de données en tant qu'options dans un groupe d'options. Pour plus d'informations sur les groupes d'options, veuillez consulter Utilisation de groupes d'options.

Rubriques

- Présentation des groupes de paramètres
- Groupes de paramètres de base de données pour les instances de base de données RDSAmazon
- Utilisation des groupes de paramètres de clusters de base de données pour les clusters de base de données Multi-AZ
- Comparaison des groupes de paramètres de bases de données
- Spécification des paramètres de base de données

Présentation des groupes de paramètres

Un groupe de paramètres de base de données sert de conteneur pour les valeurs de configuration du moteur qui sont appliquées à une ou plusieurs instances de base de données.

Les groupes de paramètres du cluster de bases de données ne s'appliquent qu'aux clusters de bases de données multi-AZ. Dans un cluster de bases de données multi-AZ, le groupe de paramètres du cluster de bases de données s'applique à toutes les instances de base de données du cluster. Le groupe de paramètres de base de données par défaut pour le moteur et la version du moteur de base de données est utilisé pour chaque instance de base de données du cluster de bases de données.

Rubriques

- Groupes de paramètres par défaut et personnalisés
- Paramètres d'instance de bases de données statiques et dynamiques
- Paramètres de cluster de bases de données statiques et dynamiques
- Paramètres de jeu de caractères
- Paramètres et valeurs de paramètres pris en charge

Groupes de paramètres par défaut et personnalisés

Si vous créez une instance de base de données sans spécifier de groupe de paramètres de base de données, l'instance de base de données utilise un groupe de paramètres de base de données par défaut. De même, si vous créez un cluster de base de données Multi-AZ sans spécifier de groupe de paramètres de cluster de base de données, le cluster de base de données utilise un groupe de paramètres de cluster de base de données par défaut. Chaque groupe de paramètres par défaut contient les valeurs par défaut du moteur de base de données, ainsi que celles du système Amazon RDS en fonction du moteur, de la classe de calcul et de l'espace de stockage alloué de l'instance.

Vous ne pouvez pas modifier les valeurs de paramètre d'un groupe de paramètres de base de données par défaut. Au lieu de cela, vous pouvez effectuer les actions suivantes :

1. Créez un groupe de paramètres.
2. Modifiez les paramètres souhaités. Il n'est pas possible de modifier tous les paramètres du moteur de base de données dans un groupe de paramètres.
3. Modifiez votre instance de base de données ou votre cluster de base de données pour associer le nouveau groupe de paramètres.

   Lorsque vous associez un nouveau groupe de paramètres de base de données à une instance de base de données, l'association se produit immédiatement. Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon. Pour plus d'informations sur la modification d'un cluster de bases de données multi-AZ, consultez Modification d'un cluster de base de données multi-AZ pour Amazon RDS.

   > **Note**
   >
   > Si vous avez modifié votre instance de base de données pour utiliser un groupe de paramètres personnalisés et que vous démarrez l'instance de base de données, RDS redémarre automatiquement l'instance de base de données dans le cadre du processus

de démarrage. Pour les instances RDS for SQL Server Multi-AZ dont l'option de mise en miroir est activée AlwaysOn ou dont l'option de mise en miroir est activée, un basculement est attendu lorsque l'instance est redémarrée après le processus de démarrage.

RDS applique les paramètres statiques et dynamiques modifiés dans un groupe de paramètres nouvellement associé uniquement après le redémarrage de l'instance de base de données. Toutefois, si vous modifiez des paramètres dynamiques dans le groupe de paramètres de base de données après l'avoir associé à l'instance de base de données, ces modifications sont appliquées immédiatement sans redémarrage. Pour de plus amples informations sur la modification du groupe de paramètres de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Si vous mettez à jour les paramètres d'un groupe de paramètres de base de données, les modifications effectuées s'appliquent à toutes les instances de base de données qui sont associées à ce groupe de paramètres. De même, si vous mettez à jour les paramètres d'un groupe de paramètres de cluster de bases de données multi-AZ, les modifications effectuées s'appliquent à tous les clusters de bases de données Aurora qui sont associés à ce groupe de paramètres du cluster de bases de données.

Si vous ne souhaitez pas créer un groupe de paramètres à partir de zéro, vous pouvez copier un groupe de paramètres existant à l'aide de la AWS CLI copy-db-parameter-groupcommande command ou de la commande copy-db-cluster-parameter-group. Vous trouverez peut-être utile de copier un groupe de paramètres dans certains cas. Par exemple, vous pouvez vouloir inclure la plupart des valeurs et paramètres personnalisés d'un groupe de paramètres de bases de données dans un nouveau groupe de paramètres de base de données.

Paramètres d'instance de bases de données statiques et dynamiques

Les paramètres d'instance de base de données sont statiques ou dynamiques. Ils diffèrent comme suit :

- Lorsque vous modifiez un paramètre statique et que vous enregistrez le groupe de paramètres de base de données, la modification du paramètre est appliquée après le redémarrage manuel des instances de base de données associées. Pour les paramètres statiques, la console utilise toujours `pending-reboot` pour `ApplyMethod`.
- Lorsque vous modifiez un paramètre dynamique, par défaut, la modification du paramètre s'applique immédiatement, sans nécessiter de redémarrage. Lorsque vous utilisez le AWS

Management Console pour modifier les valeurs des paramètres d'une instance de base de données, il l'utilise `immediate` toujours `ApplyMethod` pour les paramètres dynamiques. Pour différer la modification des paramètres jusqu'au redémarrage d'une instance de base de données associée, utilisez l'API AWS CLI ou RDS. Définissez `ApplyMethod` sur `pending-reboot` pour le changement de paramètre.

> **Note**
>
> L'utilisation `pending-reboot` de paramètres dynamiques dans l'API AWS CLI ou RDS sur les instances de base de données RDS pour SQL Server génère une erreur. Utilisez `apply-immediately` sur RDS for SQL Server.

Pour plus d'informations sur l'utilisation du AWS CLI pour modifier la valeur d'un paramètre, consultez modify-db-parameter-group. Pour plus d'informations sur l'utilisation de l'API RDS pour modifier la valeur d'un paramètre, consultez Modifier le DBParameter groupe.

Si une instance de base de données n'utilise pas les dernières modifications apportées à son groupe de paramètres de base de données associé, la console affiche le statut pending-reboot pour le groupe de paramètres de base de données. Le statut n'entraîne pas de redémarrage automatique lors de la fenêtre de maintenance suivante. Pour appliquer les modifications de paramètre les plus récentes apportées à cette instance de base de données, vous devez la redémarrer manuellement.

Paramètres de cluster de bases de données statiques et dynamiques

Les paramètres de cluster de base de données sont statiques ou dynamiques. Ils diffèrent comme suit :

- Lorsque vous modifiez un paramètre statique et que vous enregistrez le groupe de paramètres de base de données d'un cluster, la modification du paramètre prend effet après le redémarrage manuel des clusters de base de données associés. Pour les paramètres statiques, la console utilise toujours `pending-reboot` pour `ApplyMethod`.
- Lorsque vous modifiez un paramètre dynamique, par défaut, la modification du paramètre s'applique immédiatement, sans nécessiter de redémarrage. Lorsque vous utilisez le AWS Management Console pour modifier les valeurs des paramètres du cluster de bases de données, il l'utilise `immediate` toujours `ApplyMethod` pour les paramètres dynamiques. Pour différer la modification des paramètres jusqu'au redémarrage d'un cluster de base de données associé,

utilisez l'API AWS CLI ou RDS. Définissez `ApplyMethod` sur `pending-reboot` pour le changement de paramètre.

Pour plus d'informations sur l'utilisation de AWS CLI pour modifier la valeur d'un paramètre, consultez modify-db-cluster-parameter-group. Pour plus d'informations sur l'utilisation de l'API RDS pour modifier la valeur d'un paramètre, consultez Modifier DBCluster ParameterGroup.

Paramètres de jeu de caractères

Avant de créer l'instance ou le cluster de bases de données multi-AZ, définissez tous les paramètres relatifs au jeu de caractères ou au classement de votre base de données dans votre groupe de paramètres. Faites-le également avant d'y créer une base de données. Cela garantit que la base de données par défaut et les nouvelles bases de données utilisent les valeurs de jeu de caractères et de classement que vous spécifiez. Si vous modifiez les paramètres de jeu de caractères ou de classement, les modifications de paramètre ne sont pas appliquées aux bases de données existantes.

Pour certains moteurs de base de données, vous pouvez modifier les valeurs de jeu de caractères ou de classement pour une base de données existante à l'aide de la commande `ALTER DATABASE`, par exemple :

```
ALTER DATABASE database_name CHARACTER SET character_set_name COLLATE collation;
```

Pour plus d'informations sur le changement de jeu de caractères ou de valeurs de classement d'une base de données, consultez la documentation de votre moteur de base de données.

Paramètres et valeurs de paramètres pris en charge

Pour déterminer les paramètres pris en charge pour votre moteur de base de données, affichez les paramètres du groupe de paramètres de base de données et du groupe de paramètres de cluster de bases de données utilisés par l'instance de base de données ou le cluster de bases de données. Pour plus d'informations, consultez Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon et Affichage des valeurs de paramètres pour un groupe de paramètres de cluster de base de données .

Dans la plupart des cas, vous pouvez spécifier des valeurs de paramètres entiers et booléens au moyen d'expressions, de formules et de fonctions. Les fonctions peuvent inclure une expression de journal mathématique. Cependant, tous les paramètres ne prennent pas en charge les expressions,

les formules et les fonctions des valeurs de paramètres. Pour de plus amples informations, veuillez consulter Spécification des paramètres de base de données.

La configuration incorrecte de paramètres dans un groupe de paramètres peut avoir des effets contraires involontaires, dont une dégradation de la performance et une instabilité du système. Montrez-vous toujours prudent lorsque vous modifiez des paramètres de base de données et sauvegardez vos données avant de modifier un groupe de paramètres. Essayez de modifier les paramètres des groupes de paramètres sur une instance de base de données ou un cluster de bases de données de test avant d'appliquer ces modifications à une instance de base de données ou un cluster de bases de données de production.

Groupes de paramètres de base de données pour les instances de base de données RDSAmazon

Les instances de base de données utilisent des groupes de paramètres de base de données. Les sections suivantes décrivent la configuration et la gestion des groupes de paramètres d'une instance de base de données.

Rubriques

- Création d'un groupe de paramètres de base de données dans RDSAmazon
- Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon
- Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora)
- Réinitialisation des paramètres d'un groupe de paramètres de base de données à leurs valeurs par défaut dans Amazon RDS Amazon
- Copier un groupe de paramètres de base de données dans RDSAmazon
- Répertorier les groupes de paramètres de base de données dans RDSAmazon
- Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon
- Suppression d'un groupe de paramètres de base de données dans RDSAmazon

Création d'un groupe de paramètres de base de données dans RDSAmazon

Vous pouvez créer un nouveau groupe de paramètres de base de données en utilisant le AWS Management Console, le AWS CLI, ou le RDSAPI.

Les limites suivantes s'appliquent aux noms de groupes de paramètres de base de données :

- Ces noms doivent comporter entre 1 et 255 lettres, chiffres ou traits d'union.

  Les noms des groupes de paramètres par défaut peuvent inclure un point, par exemple `default.mysql8.0`. Toutefois, les noms de groupes de paramètres personnalisés ne peuvent pas inclure de point.
- Le premier caractère doit être une lettre.
- Les noms ne peuvent pas se terminer par un trait d'union ni contenir deux traits d'union consécutifs.

Console

Pour créer un groupe de paramètres de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans Nom du groupe de paramètres, entrez le nom de votre nouveau groupe de paramètres de base de données.
5. Dans Description, entrez une description pour votre nouveau groupe de paramètres de base de données.
6. Pour Type de moteur, choisissez votre moteur de base de données.
7. Pour Famille de groupes de paramètres, choisissez une famille de groupes de paramètres de base de données.
8. Pour Type, le cas échéant, choisissez DB Parameter Group.
9. Sélectionnez Create (Créer).

AWS CLI

Pour créer un groupe de paramètres de base de données, utilisez la AWS CLI `create-db-parameter-group`commande. L'exemple suivant crée un groupe de paramètres de base de données nommé mydbparametergroup pour My SQL version 8.0 avec une description de « Mon nouveau groupe de paramètres ».

Incluez les paramètres requis suivants :

- `--db-parameter-group-name`
- `--db-parameter-group-family`
- `--description`

Pour répertorier toutes les familles de groupes de paramètres, utilisez la commande suivante :

```
aws rds describe-db-engine-versions --query "DBEngineVersions[].DBParameterGroupFamily"
```

> **Note**
>
> La sortie contient des doublons.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --db-parameter-group-family MySQL8.0 \
    --description "My new parameter group"
```

Dans Windows :

```
aws rds create-db-parameter-group ^
    --db-parameter-group-name mydbparametergroup ^
    --db-parameter-group-family MySQL8.0 ^
    --description "My new parameter group"
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
DBPARAMETERGROUP  mydbparametergroup  mysql8.0  My new parameter group
```

RDS API

Pour créer un groupe de paramètres de base de données, utilisez l'RDSAPI`CreateDBParameterGroup`opération.

Incluez les paramètres requis suivants :

- `DBParameterGroupName`
- `DBParameterGroupFamily`
- `Description`

Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon

Vous pouvez créer vos propres groupes de paramètres de base de données avec des paramètres personnalisés. Vous pouvez associer un groupe de paramètres de base de données à une instance de base de données en utilisant le AWS Management Console AWS CLI, le, ou le RDSAPI. Vous pouvez le faire lorsque vous créez ou modifiez une instance de base de données.

Pour plus d'informations sur la création d'un groupe de paramètres de base de données, consultez Création d'un groupe de paramètres de base de données dans RDSAmazon. Pour plus d'informations sur la création d'une instance de base de données, consultez Création d'une instance de base de données Amazon RDS. Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

> **Note**
>
> Lorsque vous associez un nouveau groupe de paramètres de base de données à une instance de base de données, les paramètres statiques et dynamiques modifiés sont appliqués uniquement après que l'instance de base de données est redémarrée. Toutefois, si vous modifiez des paramètres dynamiques dans le groupe de paramètres de base de données après l'avoir associé à l'instance de base de données, ces modifications sont appliquées immédiatement sans redémarrage.

Console

Associer un groupe de paramètres de base de données à une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.

3. Sélectionnez Modify (Modifier). La page Modifier l'instance de base de données s'affiche.
4. Modifiez le paramètre DB parameter group (groupe de paramètres de base de données).
5. Choisissez Continuer et vérifiez le récapitulatif des modifications.
6. (Facultatif) Choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. La sélection de cette option peut entraîner une interruption de service dans certains cas. Pour plus d'informations, consultez Utilisation du paramètre de modification du calendrier.
7. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modify DB instance (Modifier l'instance de base de données) pour enregistrer vos modifications.

   Ou choisissez Retour pour revoir vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour associer un groupe de paramètres de base de données à une instance de base de données, utilisez la AWS CLI `modify-db-instance`commande avec les options suivantes :

- `--db-instance-identifier`
- `--db-parameter-group-name`

L'exemple suivant associe le groupe de paramètres de base de données `mydbpg` à l'instance de base de données `database-1`. Les modifications sont appliquées immédiatement en utilisant `--apply-immediately`. Utilisez `--no-apply-immediately` pour appliquer les modifications pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier database-1 \
    --db-parameter-group-name mydbpg \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
```

```
    --db-instance-identifier database-1 ^
    --db-parameter-group-name mydbpg ^
    --apply-immediately
```

RDS API

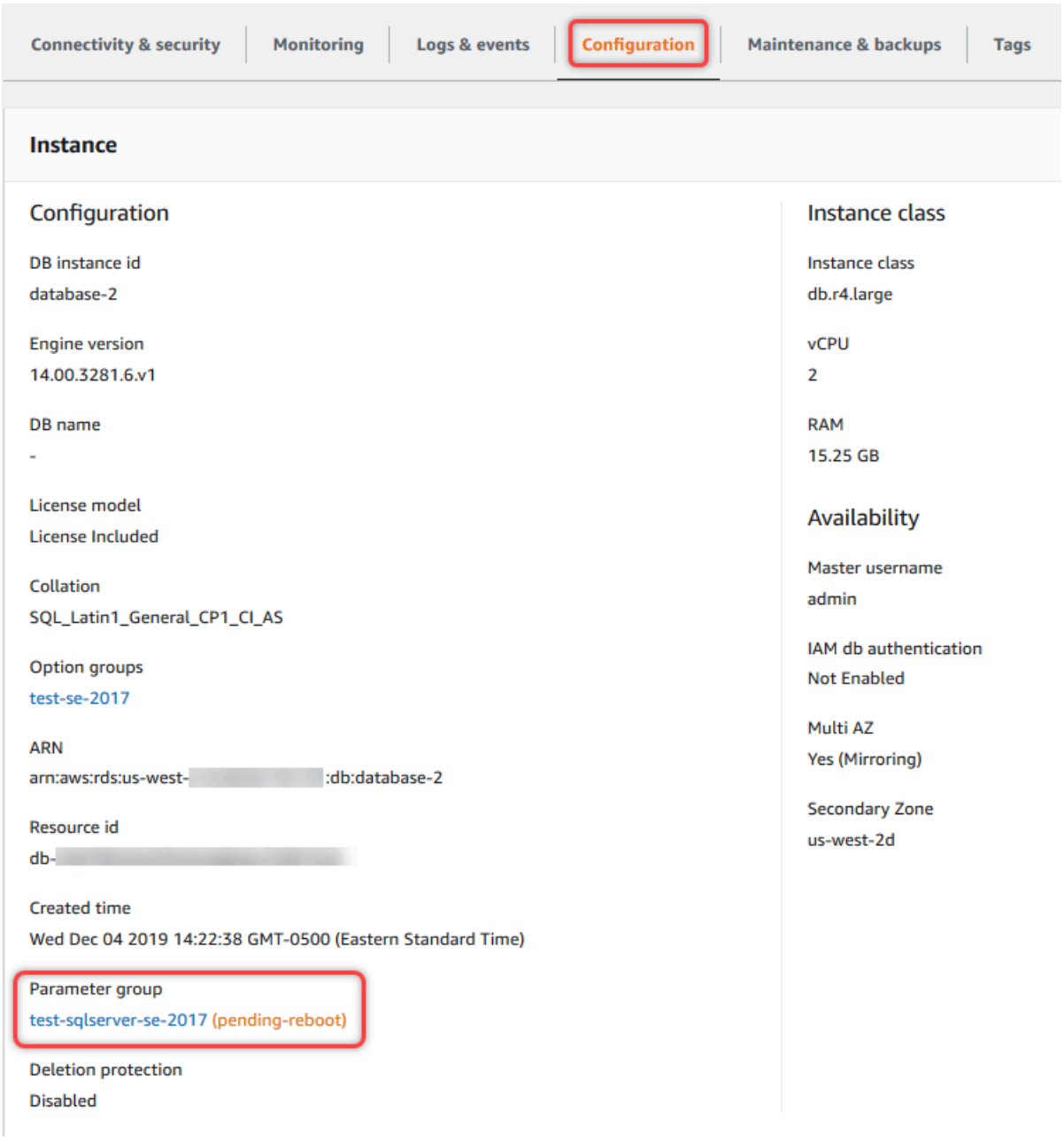
Pour associer un groupe de paramètres de base de données à une instance de base de données, utilisez l'RDSAPIModifyDBInstanceopération avec les paramètres suivants :

- `DBInstanceName`
- `DBParameterGroupName`

Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora)

Vous pouvez modifier des valeurs de paramètres dans un groupe de paramètres de base de données créé par le client. Par contre, vous ne pouvez pas modifier les valeurs de paramètres dans un groupe de paramètres de base de données par défaut. Les modifications apportées à des paramètres dans un groupe de paramètres DB créé par le client sont appliquées à toutes les instances de base de données qui sont associées au groupe de paramètres DB.

Les modifications apportées à certains paramètres sont appliquées immédiatement à l'instance de base de données sans redémarrage. Les modifications apportées à d'autres paramètres s'appliquent uniquement après le redémarrage de l'instance de base de données. La console RDS affiche le statut du groupe de paramètres de base de données associé à une instance de base de données dans l'onglet Configuration. Par exemple, supposons que l'instance de base de données n'utilise pas les dernières modifications apportées à son groupe de paramètres de base de données associé. Si tel est le cas, la console RDS affiche le groupe de paramètres de base de données avec le statut suivant : pending-reboot. Pour appliquer les modifications de paramètre les plus récentes apportées à cette instance de base de données, vous devez la redémarrer manuellement. Nous vous recommandons de toujours redémarrer l'instance de base de données après avoir modifié les paramètres. Si vous ne redémarrez pas l'instance de base de données, une opération de basculement peut prendre plus de temps que d'habitude.



console

Pour modifier les paramètres d'un groupe de paramètres de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Dans la liste, choisissez le nom du groupe de paramètres que vous souhaitez modifier.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Edit (Modifier).

5. Modifiez les valeurs des paramètres que vous souhaitez remplacer. Vous pouvez parcourir les paramètres en utilisant les touches fléchées en haut à droite de la boîte de dialogue.

   Vous ne pouvez pas modifier les valeurs dans un groupe de paramètres par défaut.

6. Sélectionnez Save Changes.

AWS CLI

Pour modifier un groupe de paramètres de base de données, utilisez la AWS CLI `modify-db-parameter-group`commande avec les options requises suivantes :

- `--db-parameter-group-name`
- `--parameters`

L'exemple suivant modifie les valeurs `max_connections` et `max_allowed_packet` dans le groupe de paramètres de base de données nommé mydbparametergroup.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --parameters
 "ParameterName=max_connections,ParameterValue=250,ApplyMethod=immediate" \

 "ParameterName=max_allowed_packet,ParameterValue=1024,ApplyMethod=immediate"
```

Dans Windows :

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name mydbparametergroup ^
    --parameters
 "ParameterName=max_connections,ParameterValue=250,ApplyMethod=immediate" ^

 "ParameterName=max_allowed_packet,ParameterValue=1024,ApplyMethod=immediate"
```

La commande produit un résultat similaire à ce qui suit :

```
DBPARAMETERGROUP  mydbparametergroup
```

API RDS

Pour modifier un groupe de paramètres de base de données, utilisez l'opération d'API RDS `ModifyDBParameterGroup` avec les paramètres requis suivants :

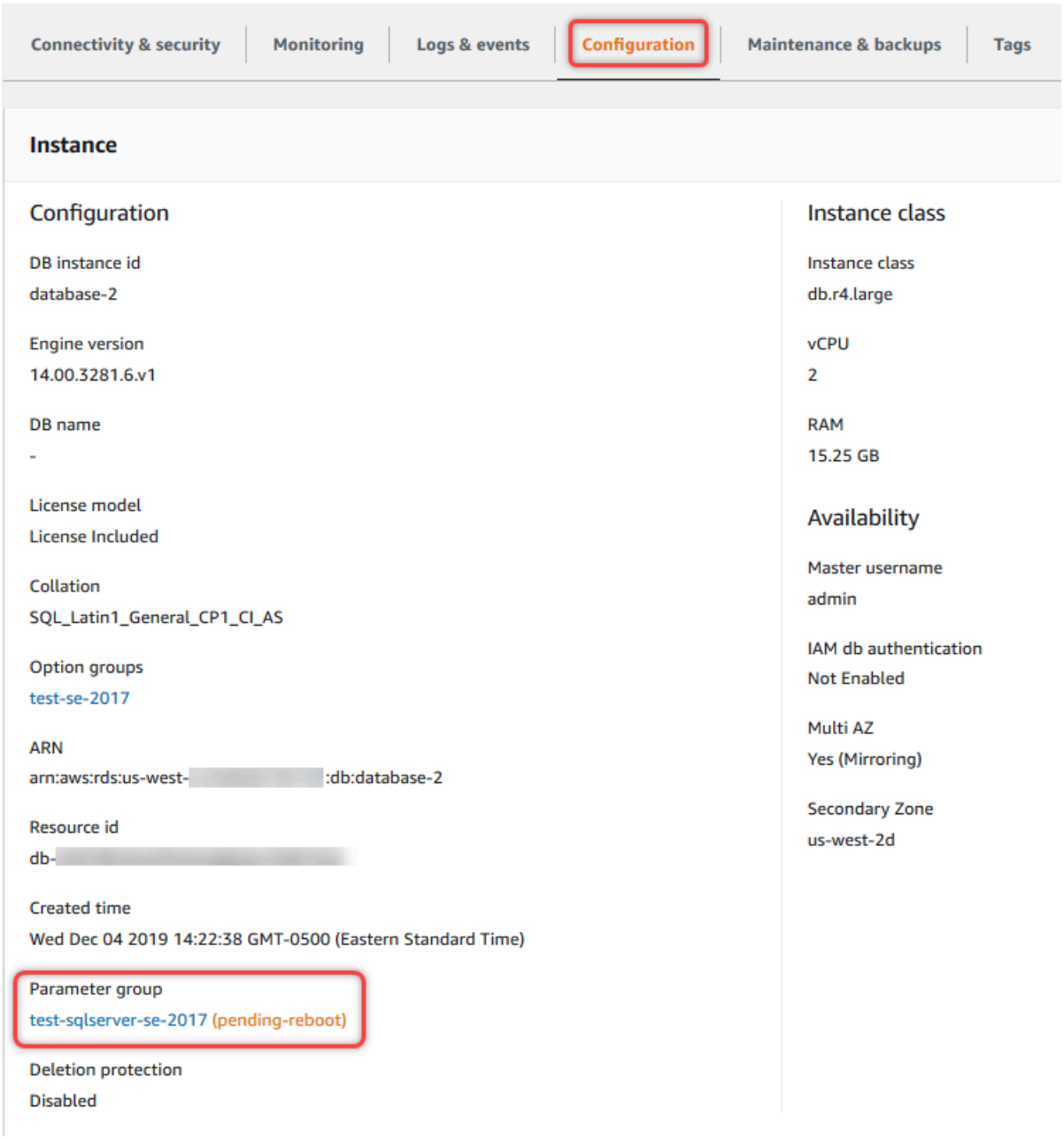
- `DBParameterGroupName`
- `Parameters`

Réinitialisation des paramètres d'un groupe de paramètres de base de données à leurs valeurs par défaut dans Amazon RDS Amazon

Vous pouvez rétablir les valeurs par défaut des paramètres d'un groupe de paramètres de base de données créé par le client. Les modifications apportées à des paramètres dans un groupe de paramètres DB créé par le client sont appliquées à toutes les instances de base de données qui sont associées au groupe de paramètres DB.

Lorsque vous utilisez la console, vous pouvez rétablir les valeurs par défaut de paramètres spécifiques. Cependant, vous ne pouvez pas facilement réinitialiser tous les paramètres du groupe de paramètres de base de données simultanément. Lorsque vous utilisez le AWS CLI ou RDSAPI, vous pouvez rétablir les valeurs par défaut de certains paramètres. Vous pouvez également réinitialiser tous les paramètres du groupe de paramètres de base de données simultanément.

Les modifications apportées à certains paramètres sont appliquées immédiatement à l'instance de base de données sans redémarrage. Les modifications apportées à d'autres paramètres s'appliquent uniquement après le redémarrage de l'instance de base de données. La RDS console affiche l'état du groupe de paramètres de base de données associé à une instance de base de données dans l'onglet Configuration. Par exemple, supposons que l'instance de base de données n'utilise pas les dernières modifications apportées à son groupe de paramètres de base de données associé. Dans ce cas, la RDS console affiche le groupe de paramètres de base de données avec un statut en attente de redémarrage. Pour appliquer les modifications de paramètre les plus récentes apportées à cette instance de base de données, vous devez la redémarrer manuellement.



> **Note**
>
> Dans un groupe de paramètres de base de données par défaut, les paramètres sont toujours définis sur leurs valeurs par défaut.

Console

Pour réinitialiser les valeurs par défaut des paramètres d'un groupe de paramètres de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, sélectionnez le groupe de paramètres.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Edit (Modifier).
5. Sélectionnez les paramètres que vous souhaitez réinitialiser à leurs valeurs par défaut. Vous pouvez parcourir les paramètres en utilisant les touches fléchées en haut à droite de la boîte de dialogue.

   Vous ne pouvez pas réinitialiser les valeurs dans un groupe de paramètres par défaut.
6. Choisissez Réinitialiser, puis confirmez en sélectionnant Réinitialiser les paramètres.

AWS CLI

Pour réinitialiser certains ou tous les paramètres d'un groupe de paramètres de base de données, utilisez la AWS CLI `reset-db-parameter-group`commande avec l'option requise suivante :`--db-parameter-group-name`.

Pour réinitialiser tous les paramètres du groupe de paramètres de base de données, spécifiez l'option `--reset-all-parameters`. Pour réinitialiser des paramètres spécifiques, spécifiez l'option `--parameters`.

L'exemple suivant réinitialise tous les paramètres du groupe de paramètres de base de données nommé mydbparametergroup à leurs valeurs par défaut.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds reset-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --reset-all-parameters
```

Dans Windows :

```
aws rds reset-db-parameter-group ^
    --db-parameter-group-name mydbparametergroup ^
    --reset-all-parameters
```

L'exemple suivant réinitialise les valeurs par défaut des options `max_connections` et `max_allowed_packet` du groupe de paramètres de base de données mydbparametergroup.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds reset-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --parameters "ParameterName=max_connections,ApplyMethod=immediate" \
                 "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

Dans Windows :

```
aws rds reset-db-parameter-group ^
    --db-parameter-group-name mydbparametergroup ^
    --parameters "ParameterName=max_connections,ApplyMethod=immediate" ^
                 "ParameterName=max_allowed_packet,ApplyMethod=immediate"
```

La commande produit un résultat similaire à ce qui suit :

```
DBParameterGroupName  mydbparametergroup
```

RDS API

Pour rétablir les valeurs par défaut des paramètres d'un groupe de paramètres de base de données, utilisez la RDS API `ResetDBParameterGroup`commande avec le paramètre obligatoire suivant :`DBParameterGroupName`.

Pour réinitialiser tous les paramètres du groupe de paramètres de base de données, définissez le `ResetAllParameters` paramètre sur `true`. Pour réinitialiser des paramètres spécifiques, spécifiez le paramètre `Parameters`.

Copier un groupe de paramètres de base de données dans RDSAmazon

Vous pouvez copier des groupes de paramètres DB personnalisés que vous créez. La copie d'un groupe de paramètres peut s'avérer une solution pratique. Par exemple, lorsque vous avez créé un

groupe de paramètres de base de données et que vous souhaitez inclure la plupart de ses valeurs et paramètres personnalisés dans un nouveau groupe de paramètres de base de données. Vous pouvez copier un groupe de paramètres de base de données à l'aide du AWS Management Console. Vous pouvez également utiliser la AWS CLI copy-db-parameter-groupcommande ou l'opération RDS API C opyDBParameter Group.

Après avoir copié un groupe de paramètres de base de données, patientez au moins 5 minutes avant de créer votre première instance de base de données utilisant ce groupe comme groupe de paramètres par défaut. Cela permet RDS à Amazon de terminer complètement l'action de copie avant que le groupe de paramètres ne soit utilisé. Cela est particulièrement important pour les paramètres qui sont essentiels lors de la création de la base de données par défaut d'une instance de base de données. Parmi ces paramètres, citons par exemple le jeu de caractères de la base de données par défaut défini par le paramètre `character_set_database`. Utilisez l'option Groupes de paramètres de la RDSconsole Amazon ou la describe-db-parameterscommande pour vérifier que votre groupe de paramètres de base de données est créé.

> **Note**
>
> Vous ne pouvez pas copier un groupe de paramètres par défaut. Toutefois, vous pouvez créer un nouveau groupe de paramètres basé sur un groupe de paramètres par défaut. Vous ne pouvez pas copier un groupe de paramètres de base de données vers un autre Compte AWS ou Région AWS.

Console

Pour copier un groupe de paramètres DB

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, choisissez le groupe de paramètres personnalisé que vous souhaitez copier.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Copy (Copier).
5. Dans New DB parameter group identifier (Nouvel identifiant de groupe de paramètres de base de données), saisissez un nom pour le nouveau groupe de paramètres.
6. Dans Description, saisissez une description pour le nouveau groupe de paramètres.
7. Choisissez Copy.

AWS CLI

Pour copier un groupe de paramètres de base de données, utilisez la AWS CLI `copy-db-parameter-group`commande avec les options requises suivantes :

- `--source-db-parameter-group-identifier`
- `--target-db-parameter-group-identifier`
- `--target-db-parameter-group-description`

L'exemple suivant crée un groupe de paramètres DB nommé `mygroup2` qui est une copie du groupe de paramètres DB `mygroup1`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds copy-db-parameter-group \
    --source-db-parameter-group-identifier mygroup1 \
    --target-db-parameter-group-identifier mygroup2 \
    --target-db-parameter-group-description "DB parameter group 2"
```

Dans Windows :

```
aws rds copy-db-parameter-group ^
    --source-db-parameter-group-identifier mygroup1 ^
    --target-db-parameter-group-identifier mygroup2 ^
    --target-db-parameter-group-description "DB parameter group 2"
```

RDS API

Pour copier un groupe de paramètres de base de données, utilisez l'RDSAPI`CopyDBParameterGroup`opération avec les paramètres obligatoires suivants :

- `SourceDBParameterGroupIdentifier`
- `TargetDBParameterGroupIdentifier`
- `TargetDBParameterGroupDescription`

Répertorier les groupes de paramètres de base de données dans RDSAmazon

Vous pouvez répertorier les groupes de paramètres de base de données que vous avez créés pour votre AWS compte.

> **Note**
>
> Les groupes de paramètres par défaut sont automatiquement créés à partir d'un modèle de paramètre par défaut lorsque vous créez une instance de base de données pour une version et un moteur de base de données spécifiques. Ces groupes de paramètres par défaut contiennent des valeurs de paramètres préférentielles et ne peuvent pas être modifiés. Lorsque vous créez un groupe de paramètres personnalisé, vous pouvez modifier les réglages des paramètres.

Console

Pour répertorier tous les groupes de paramètres de base de données pour un AWS compte

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres DB s'affichent dans une liste.

AWS CLI

Pour répertorier tous les groupes de paramètres de base de données d'un AWS compte, utilisez la AWS CLI `describe-db-parameter-groups`commande.

Example

L'exemple suivant répertorie tous les groupes de paramètres DB disponibles pour un compte AWS .

```
aws rds describe-db-parameter-groups
```

La commande renvoie une réponse telle que la suivante :

```
DBPARAMETERGROUP  default.mysql8.0     mysql8.0  Default parameter group for MySQL8.0
```

```
DBPARAMETERGROUP  mydbparametergroup   mysql8.0  My new parameter group
```

L'exemple suivant décrit le groupe de paramètres mydbparamgroup1.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-parameter-groups \
    --db-parameter-group-name mydbparamgroup1
```

Dans Windows :

```
aws rds describe-db-parameter-groups ^
    --db-parameter-group-name mydbparamgroup1
```

La commande renvoie une réponse telle que la suivante :

```
DBPARAMETERGROUP  mydbparametergroup1  mysql8.0  My new parameter group
```

RDS API

Pour répertorier tous les groupes de paramètres de base de données d'un AWS compte, utilisez l'RDSAPIDescribeDBParameterGroupsopération.

Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon

Vous pouvez obtenir une liste de tous les paramètres dans un groupe de paramètres DB et de leurs valeurs.

Console

Pour afficher les valeurs de paramètres pour un groupe de paramètres DB

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres DB s'affichent dans une liste.
3. Choisissez le nom du groupe de paramètres pour consulter la liste des paramètres associée.

AWS CLI

Pour afficher les valeurs des paramètres d'un groupe de paramètres de base de données, utilisez la AWS CLI `describe-db-parameters`commande avec le paramètre obligatoire suivant.

- `--db-parameter-group-name`

Example

L'exemple suivant répertorie les paramètres et les valeurs de paramètres pour un groupe de paramètres de base de données nommé mydbparametergroup.

```
aws rds describe-db-parameters --db-parameter-group-name mydbparametergroup
```

La commande renvoie une réponse telle que la suivante :

```
DBPARAMETER  Parameter Name            Parameter Value  Source           Data Type
 Apply Type  Is Modifiable
DBPARAMETER  allow-suspicious-udfs                      engine-default   boolean
 static      false
DBPARAMETER  auto_increment_increment                   engine-default   integer
 dynamic     true
DBPARAMETER  auto_increment_offset                      engine-default   integer
 dynamic     true
DBPARAMETER  binlog_cache_size         32768            system           integer
 dynamic     true
DBPARAMETER  socket                    /tmp/mysql.sock  system           string
 static      false
```

RDS API

Pour afficher les valeurs des paramètres d'un groupe de paramètres de base de données, utilisez la RDS API `DescribeDBParameters`commande avec le paramètre obligatoire suivant.

- `DBParameterGroupName`

Suppression d'un groupe de paramètres de base de données dans RDSAmazon

Vous pouvez supprimer un groupe de paramètres de base de données à l'aide de AWS Management Console AWS CLI, ou RDSAPI. Un groupe de paramètres ne peut être supprimé que s'il n'est pas associé à une instance de base de données.

Console

Pour supprimer un groupe de paramètres de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres DB s'affichent dans une liste.
3. Choisissez le nom des groupes de paramètres à supprimer.
4. Choisissez Actions, puis Supprimer.
5. Vérifiez les noms des groupes de paramètres, puis choisissez Supprimer.

AWS CLI

Pour supprimer un groupe de paramètres de base de données, utilisez la AWS CLI `delete-db-parameter-group`commande avec le paramètre obligatoire suivant.

- `--db-parameter-group-name`

Example

L'exemple suivant supprime un groupe de paramètres de base de données nommé mydbparametergroup.

```
aws rds delete-db-parameter-group --db-parameter-group-name mydbparametergroup
```

RDS API

Pour supprimer un groupe de paramètres de base de données, utilisez la RDS API `DeleteDBParameterGroup`commande avec le paramètre obligatoire suivant.

- `DBParameterGroupName`

Utilisation des groupes de paramètres de clusters de base de données pour les clusters de base de données Multi-AZ

Les clusters de base de données Multi-AZ utilisent des groupes de paramètres de cluster de base de données. Les sections suivantes décrivent la configuration et la gestion des groupes de paramètres de cluster de bases de données.

Rubriques

- Création d'un groupe de paramètres de cluster de base de données
- Modification des paramètres d'un groupe de paramètres de cluster de base de données
- Réinitialisation des paramètres d'un groupe de paramètres de cluster de base de données
- Copie d'un groupe de paramètres de cluster de base de données
- Répertorier les groupes de paramètres de clusters de bases de données
- Affichage des valeurs de paramètres pour un groupe de paramètres de cluster de base de données
- Suppression d'un groupe de paramètres de cluster de base de données

Création d'un groupe de paramètres de cluster de base de données

Vous pouvez créer un nouveau groupe de paramètres de cluster de base de données à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

Après avoir créé un groupe de paramètres de base de données, attendez au moins cinq minutes avant de créer un cluster de base de données qui utilise ce groupe de paramètres de base de données. Cela permet à Amazon RDS de créer entièrement le groupe de paramètres avant qu'il ne soit utilisé par le nouveau cluster de base de données. Vous pouvez utiliser la page Groupes de paramètres de la console Amazon RDS ou la describe-db-cluster-parameterscommande pour vérifier que votre groupe de paramètres de cluster de base de données est créé.

Les limites suivantes s'appliquent aux noms de groupes de paramètres de cluster de bases de données :

- Ces noms doivent comporter entre 1 et 255 lettres, chiffres ou traits d'union.

  Les noms des groupes de paramètres par défaut peuvent inclure un point, par exemple `default.mysql5.7`. Toutefois, les noms de groupes de paramètres personnalisés ne peuvent pas inclure de point.

- Le premier caractère doit être une lettre.
- Les noms ne peuvent pas se terminer par un trait d'union ni contenir deux traits d'union consécutifs.

console

Pour créer un groupe de paramètres de cluster de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans Nom du groupe de paramètres, entrez le nom du nouveau groupe de paramètres du cluster de base de données.
5. Dans Description, entrez une description pour le nouveau groupe de paramètres du cluster de base de données.
6. Pour Type de moteur, choisissez votre moteur de base de données.
7. Pour Famille de groupes de paramètres, choisissez une famille de groupes de paramètres de base de données.
8. Choisissez Créer.

AWS CLI

Pour créer un groupe de paramètres de cluster de base de données, utilisez la AWS CLI `create-db-cluster-parameter-group`commande.

L'exemple suivant crée un groupe de paramètres de cluster de base de données nommé mydbclusterparametergroup pour MySQL version 8.0 avec une description de « My new cluster parameter group » (Mon nouveau groupe de paramètres de cluster).

Incluez les paramètres requis suivants :

- `--db-cluster-parameter-group-name`
- `--db-parameter-group-family`
- `--description`

Pour répertorier toutes les familles de groupes de paramètres, utilisez la commande suivante :

```
aws rds describe-db-engine-versions --query "DBEngineVersions[].DBParameterGroupFamily"
```

> **Note**
>
> La sortie contient des doublons.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterparametergroup \
    --db-parameter-group-family mysql8.0 \
    --description "My new cluster parameter group"
```

Dans Windows :

```
aws rds create-db-cluster-parameter-group ^
    --db-cluster-parameter-group-name mydbclusterparametergroup ^
    --db-parameter-group-family mysql8.0 ^
    --description "My new cluster parameter group"
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
{
    "DBClusterParameterGroup": {
        "DBClusterParameterGroupName": "mydbclusterparametergroup",
        "DBParameterGroupFamily": "mysql8.0",
        "Description": "My new cluster parameter group",
        "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-
pg:mydbclusterparametergroup2"
    }
}
```

API RDS

Pour créer un groupe de paramètres de cluster de base de données, utilisez l'action d'API RDS `CreateDBClusterParameterGroup`.

Incluez les paramètres requis suivants :

- `DBClusterParameterGroupName`
- `DBParameterGroupFamily`
- `Description`

Modification des paramètres d'un groupe de paramètres de cluster de base de données

Vous pouvez modifier les valeurs des paramètres dans un groupe de paramètres de cluster base de données créé par le client. Vous ne pouvez pas modifier les valeurs des paramètres dans un groupe de paramètres de cluster de base de données par défaut. Les modifications apportées à des paramètres dans un groupe de paramètres de cluster de base de données créé par le client sont appliquées à tous les clusters de base de données qui sont associés au groupe de paramètres de cluster de base de données.

console

Pour modifier un groupe de paramètres de cluster de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, sélectionnez le groupe de paramètres que vous souhaitez modifier.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Edit (Modifier).
5. Modifiez les valeurs des paramètres que vous souhaitez remplacer. Vous pouvez parcourir les paramètres en utilisant les touches fléchées en haut à droite de la boîte de dialogue.

   Vous ne pouvez pas modifier les valeurs dans un groupe de paramètres par défaut.
6. Sélectionnez Enregistrer les modifications.
7. Redémarrez l' cluster pour y appliquer les modifications.

   Si vous ne redémarrez pas le cluster, l'opération de basculement peut prendre plus de temps que d'habitude.

AWS CLI

Pour modifier un groupe de paramètres de cluster de base de données, utilisez la AWS CLI [modify-db-cluster-parameter-group](#)commande avec les paramètres obligatoires suivants :

- `--db-cluster-parameter-group-name`
- `--parameters`

L'exemple suivant modifie les valeurs `server_audit_logging` et `server_audit_logs_upload` dans le groupe de paramètres de cluster de base de données nommé mydbclusterparametergroup.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterparametergroup \
    --parameters
 "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate" \

 "ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate"
```

Dans Windows :

```
aws rds modify-db-cluster-parameter-group ^
    --db-cluster-parameter-group-name mydbclusterparametergroup ^
    --parameters
 "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate" ^

 "ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate"
```

La commande produit un résultat similaire à ce qui suit :

```
DBCLUSTERPARAMETERGROUP  mydbclusterparametergroup
```

API RDS

Pour modifier un groupe de paramètres de cluster de base de données, utilisez la commande d'API RDS [ModifyDBClusterParameterGroup](#) avec les paramètres requis suivants :

- `DBClusterParameterGroupName`
- `Parameters`

Réinitialisation des paramètres d'un groupe de paramètres de cluster de base de données

Vous pouvez réinitialiser les paramètres à leurs valeurs par défaut dans un groupe de paramètres de cluster de bases de données créé par le client. Les modifications apportées à des paramètres dans un groupe de paramètres de cluster de base de données créé par le client sont appliquées à tous les clusters de base de données qui sont associés au groupe de paramètres de cluster de base de données.

> **Note**
>
> Dans un groupe de paramètres de cluster de bases de données par défaut, les paramètres sont toujours définis sur leurs valeurs par défaut.

console

Pour réinitialiser les paramètres d'un groupe de paramètres de cluster de bases de données à leurs valeurs par défaut

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, sélectionnez le groupe de paramètres.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Edit (Modifier).
5. Sélectionnez les paramètres que vous souhaitez réinitialiser à leurs valeurs par défaut. Vous pouvez parcourir les paramètres en utilisant les touches fléchées en haut à droite de la boîte de dialogue.

   Vous ne pouvez pas réinitialiser les valeurs dans un groupe de paramètres par défaut.
6. Choisissez Réinitialiser, puis confirmez en sélectionnant Réinitialiser les paramètres.
7. Redémarrez l' base de données.

AWS CLI

Pour rétablir les valeurs par défaut des paramètres d'un groupe de paramètres de cluster de base de données, utilisez la AWS CLI `reset-db-cluster-parameter-group`commande avec l'option requise suivante :`--db-cluster-parameter-group-name`.

Pour réinitialiser tous les paramètres du groupe de paramètres du cluster de bases de données, spécifiez l'option `--reset-all-parameters`. Pour réinitialiser des paramètres spécifiques, spécifiez l'option `--parameters`.

L'exemple suivant réinitialise tous les paramètres du groupe de paramètres de base de données nommé mydbparametergroup à leurs valeurs par défaut.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbparametergroup \
    --reset-all-parameters
```

Dans Windows :

```
aws rds reset-db-cluster-parameter-group ^
    --db-cluster-parameter-group-name mydbparametergroup ^
    --reset-all-parameters
```

L'exemple suivant modifie les valeurs `server_audit_logging` et `server_audit_logs_upload` dans le groupe de paramètres de cluster de bases de données nommé mydbclusterparametergroup.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds reset-db-cluster-parameter-group \
    --db-cluster-parameter-group-name mydbclusterparametergroup \
    --parameters "ParameterName=server_audit_logging,ApplyMethod=immediate" \
                 "ParameterName=server_audit_logs_upload,ApplyMethod=immediate"
```

Dans Windows :

```
aws rds reset-db-cluster-parameter-group ^
    --db-cluster-parameter-group-name mydbclusterparametergroup ^
    --parameters
 "ParameterName=server_audit_logging,ParameterValue=1,ApplyMethod=immediate" ^

 "ParameterName=server_audit_logs_upload,ParameterValue=1,ApplyMethod=immediate"
```

La commande produit un résultat similaire à ce qui suit :

```
DBClusterParameterGroupName  mydbclusterparametergroup
```

API RDS

Pour réinitialiser les paramètres d'un groupe de paramètres de cluster de bases de données à leurs valeurs par défaut, utilisez la commande `ResetDBClusterParameterGroup` de l'API RDS avec le paramètre obligatoire suivant : `DBClusterParameterGroupName`.

Pour réinitialiser tous les paramètres du groupe de paramètres du cluster de bases de données, définissez le paramètre `ResetAllParameters` sur `true`. Pour réinitialiser des paramètres spécifiques, spécifiez le paramètre `Parameters`.

Copie d'un groupe de paramètres de cluster de base de données

Vous pouvez copier des groupes de paramètres de cluster de base de données personnalisés que vous créez. La copie d'un groupe de paramètres est une solution pratique lorsque vous avez déjà créé un groupe de paramètres de cluster de base de données et que vous souhaitez inclure la plupart des valeurs et des paramètres personnalisés de ce groupe dans un nouveau groupe de paramètres de cluster de base de données. Vous pouvez copier un groupe de paramètres de cluster de base de données à l'aide de la commande AWS CLI copy-db-cluster-parameter-group ou de l'opération RDS API Copy DBCluster ParameterGroup.

Après avoir copié un groupe de paramètres de base de données, attendez au moins cinq minutes avant de créer un cluster de base de données qui utilise ce groupe de paramètres de base de données. Cela permet à Amazon RDS de copier entièrement le groupe de paramètres avant qu'il ne soit utilisé par le nouveau cluster de base de données. Vous pouvez utiliser la page Groupes de paramètres de la console Amazon RDS ou la describe-db-cluster-parameterscommande pour vérifier que votre groupe de paramètres de cluster de base de données est créé.

> **Note**
>
> Vous ne pouvez pas copier un groupe de paramètres par défaut. Toutefois, vous pouvez créer un nouveau groupe de paramètres basé sur un groupe de paramètres par défaut.
> Vous ne pouvez pas copier un groupe de paramètres de cluster de base de données vers un autre Compte AWS ou Région AWS.

console

Pour copier un groupe de paramètres de cluster de bases de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, choisissez le groupe de paramètres personnalisé que vous souhaitez copier.
4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Copy (Copier).
5. Dans New DB parameter group identifier (Nouvel identifiant de groupe de paramètres de base de données), saisissez un nom pour le nouveau groupe de paramètres.
6. Dans Description, saisissez une description pour le nouveau groupe de paramètres.
7. Choisissez Copy.

AWS CLI

Pour copier un groupe de paramètres de cluster de base de données, utilisez la AWS CLI `copy-db-cluster-parameter-group`commande avec les paramètres obligatoires suivants :

- `--source-db-cluster-parameter-group-identifier`
- `--target-db-cluster-parameter-group-identifier`
- `--target-db-cluster-parameter-group-description`

L'exemple suivant crée un groupe de paramètres de cluster de bases de données nommé `mygroup2` qui est une copie du groupe de paramètres de cluster de bases de données `mygroup1`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds copy-db-cluster-parameter-group \
    --source-db-cluster-parameter-group-identifier mygroup1 \
    --target-db-cluster-parameter-group-identifier mygroup2 \
    --target-db-cluster-parameter-group-description "DB parameter group 2"
```

Dans Windows :

```
aws rds copy-db-cluster-parameter-group ^
    --source-db-cluster-parameter-group-identifier mygroup1 ^
    --target-db-cluster-parameter-group-identifier mygroup2 ^
    --target-db-cluster-parameter-group-description "DB parameter group 2"
```

API RDS

Pour copier un groupe de paramètres de cluster de base de données, utilisez l'opération d'API RDS `CopyDBClusterParameterGroup` avec les paramètres requis suivants :

- `SourceDBClusterParameterGroupIdentifier`
- `TargetDBClusterParameterGroupIdentifier`
- `TargetDBClusterParameterGroupDescription`

Répertorier les groupes de paramètres de clusters de bases de données

Vous pouvez répertorier les groupes de paramètres de cluster de base de données que vous avez créés pour votre AWS compte.

> **Note**
>
> Les groupes de paramètres par défaut sont automatiquement créés à partir d'un modèle de paramètre par défaut lorsque vous créez un cluster de base de données pour une version et un moteur de base de données spécifiques. Ces groupes de paramètres par défaut contiennent des valeurs de paramètres préférentielles et ne peuvent pas être modifiés. Lorsque vous créez un groupe de paramètres personnalisé, vous pouvez modifier les réglages des paramètres.

console

Pour répertorier tous les groupes de paramètres de cluster de base de données pour un AWS compte

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres de cluster de bases de données apparaissent dans la liste avec Groupe de paramètres de cluster de base de données pour le Type.

AWS CLI

Pour répertorier tous les groupes de paramètres de cluster de base de données pour un AWS compte, utilisez la AWS CLI `describe-db-cluster-parameter-groups`commande.

Example

L'exemple suivant répertorie tous les groupes de paramètres de cluster de bases de données disponibles pour un compte AWS .

```
aws rds describe-db-cluster-parameter-groups
```

L'exemple suivant décrit le groupe de paramètres mydbclusterparametergroup.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-cluster-parameter-groups \
    --db-cluster-parameter-group-name mydbclusterparametergroup
```

Dans Windows :

```
aws rds describe-db-cluster-parameter-groups ^
    --db-cluster-parameter-group-name mydbclusterparametergroup
```

La commande renvoie une réponse telle que la suivante :

```
{
    "DBClusterParameterGroups": [
```

```
        {
            "DBClusterParameterGroupName": "mydbclusterparametergroup2",
            "DBParameterGroupFamily": "mysql8.0",
            "Description": "My new cluster parameter group",
            "DBClusterParameterGroupArn": "arn:aws:rds:us-east-1:123456789012:cluster-
pg:mydbclusterparametergroup"
        }
    ]
}
```

API RDS

Pour répertorier tous les groupes de paramètres de cluster de base de données pour un AWS compte, utilisez l'`DescribeDBClusterParameterGroups`action API RDS.

Affichage des valeurs de paramètres pour un groupe de paramètres de cluster de base de données

Vous pouvez obtenir une liste de tous les paramètres dans un groupe de paramètres de cluster de bases de données et de leurs valeurs.

console

Pour afficher les valeurs de paramètres pour un groupe de paramètres de cluster de bases de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres de cluster de bases de données apparaissent dans la liste avec Groupe de paramètres de cluster de base de données pour le Type.
3. Choisissez le nom du groupe de paramètres du cluster de base de données pour afficher la liste des paramètres associée.

AWS CLI

Pour afficher les valeurs des paramètres d'un groupe de paramètres de cluster de base de données, utilisez la AWS CLI `describe-db-cluster-parameters`commande avec le paramètre obligatoire suivant.

- `--db-cluster-parameter-group-name`

Example

L'exemple suivant répertorie les paramètres et les valeurs de paramètres pour un groupe de paramètres de cluster de bases de données nommé mydbclusterparametergroup au format JSON.

La commande renvoie une réponse telle que la suivante :

```
aws rds describe-db-cluster-parameters --db-cluster-parameter-group-
name mydbclusterparametergroup
```


```
{
    "Parameters": [
        {
            "ParameterName": "activate_all_roles_on_login",
            "ParameterValue": "0",
            "Description": "Automatically set all granted roles as active after the
 user has authenticated successfully.",
            "Source": "engine-default",
            "ApplyType": "dynamic",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": true,
            "ApplyMethod": "pending-reboot",
            "SupportedEngineModes": [
                "provisioned"
            ]
        },
        {
            "ParameterName": "allow-suspicious-udfs",
            "Description": "Controls whether user-defined functions that have only an
 xxx symbol for the main function can be loaded",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot",
            "SupportedEngineModes": [
                "provisioned"
            ]
```

```
        },
...
```

API RDS

Pour afficher les valeurs de paramètre d'un groupe de paramètres de cluster de base de données, utilisez la commande d'API RDS `DescribeDBClusterParameters` avec le paramètre requis suivant.

- `DBClusterParameterGroupName`

Dans certains cas, les valeurs autorisées pour un paramètre ne sont pas affichées. Il s'agit toujours de paramètres dont la source est la valeur par défaut du moteur de base de données.

Pour afficher les valeurs de ces paramètres, vous pouvez exécuter les instructions SQL suivantes :

- MySQL :

```
-- Show the value of a particular parameter
mysql$ SHOW VARIABLES LIKE '%parameter_name%';

-- Show the values of all parameters
mysql$ SHOW VARIABLES;
```

- PostgreSQL :

```
-- Show the value of a particular parameter
postgresql=> SHOW parameter_name;

-- Show the values of all parameters
postgresql=> SHOW ALL;
```

Suppression d'un groupe de paramètres de cluster de base de données

Vous pouvez supprimer un groupe de paramètres de cluster de base de données à l'aide de l'API AWS Management Console AWS CLI, ou RDS. Un groupe de paramètres de groupe de paramètres de cluster de base de données est éligible à la suppression uniquement s'il n'est pas associé à un cluster de base de données.

console

Pour supprimer des groupes de paramètres

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres apparaissent dans une liste.
3. Choisissez le nom des groupes de paramètres du cluster de base de données à supprimer.
4. Choisissez Actions, puis Supprimer.
5. Passez en revue les noms des groupes de paramètres, puis choisissez Supprimer.

AWS CLI

Pour supprimer un groupe de paramètres de cluster de base de données, utilisez la AWS CLI `delete-db-cluster-parameter-group`commande avec le paramètre obligatoire suivant.

- `--db-parameter-group-name`

Example

L'exemple suivant supprime un groupe de paramètres de cluster de base de données nommé mydbparametergroup.

```
aws rds delete-db-cluster-parameter-group --db-parameter-group-name mydbparametergroup
```

API RDS

Pour supprimer un groupe de paramètres de cluster de base de données, utilisez la `DeleteDBClusterParameterGroup`commande API RDS avec le paramètre obligatoire suivant.

- `DBParameterGroupName`

Comparaison des groupes de paramètres de bases de données

Vous pouvez utiliser le AWS Management Console pour visualiser les différences entre deux groupes de paramètres de base de données.

Les groupes de paramètres doivent tous deux être des groupes de paramètres de base de données, ou bien des groupes de paramètres de cluster de bases de données. Cela est vrai même si le moteur de base de données et la version sont identiques. Par exemple, vous ne pouvez pas comparer un groupe de paramètres de base de données `aurora-mysql8.0` (Aurora My SQL version 3) et un groupe de paramètres de `aurora-mysql8.0` cluster de bases de données.

Vous pouvez comparer les groupes de paramètres Aurora My SQL et RDS for My SQL DB, même pour différentes versions, mais vous ne pouvez pas comparer les groupes de paramètres Aurora Postgre SQL et RDS pour Postgre SQL DB.

Pour comparer deux groupes de paramètres de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Dans la liste, choisissez les deux groupes de paramètres que vous souhaitez comparer.

   > **Note**
   >
   > Pour comparer un groupe de paramètres par défaut à un groupe de paramètres personnalisé, choisissez d'abord le groupe de paramètres par défaut dans l'onglet Par défaut, puis choisissez le groupe de paramètres personnalisés dans l'onglet Personnalisé.

4. Dans Actions, sélectionnez Comparer.

Spécification des paramètres de base de données

Les types de paramètres de base de données sont les suivants :

- Entier
- Booléen
- Chaîne
- Long
- Double
- Horodatage

- Objet d'autres types de données définis
- Tableau de valeurs de type entier, booléen, chaîne, long, double, horodatage ou objet

Vous pouvez également spécifier des paramètres entiers et booléens au moyen d'expressions, de formules et de fonctions.

Pour le moteur Oracle, vous pouvez utiliser la variable de formule `DBInstanceClassHugePagesDefault` pour spécifier un paramètre de base de données booléen. Consultez Variables de formule de paramètre de bases de données.

Pour le SQL moteur Postgre, vous pouvez utiliser une expression pour spécifier un paramètre de base de données booléen. Consultez Expressions de paramètre de base de données booléennes.

Table des matières

- Des formules de paramètre de bases de données
  - Variables de formule de paramètre de bases de données
  - Opérateurs de formule de paramètre de bases de données
- Fonctions de paramètre de bases de données
- Expressions de paramètre de base de données booléennes
- Expressions de journal des paramètres de base de données
- Exemples de valeurs de paramètre de bases de données

Des formules de paramètre de bases de données

Une formule de paramètre de base de données est une expression qui se réduit à une valeur entière ou booléenne. Vous insérez l'expression entre des accolades : {}. Vous pouvez utiliser une formule pour une valeur de paramètre de base de données ou en tant qu'argument pour une fonction de paramètre de base de données.

Syntaxe

```
{FormulaVariable}
{FormulaVariable*Integer}
{FormulaVariable*Integer/Integer}
{FormulaVariable/Integer}
```

Variables de formule de paramètre de bases de données

Chaque variable de formule renvoie une valeur entière ou booléenne. Les noms des variables sont sensibles à la casse.

AllocatedStorage

Renvoie un entier qui représente la taille du volume de données en octets.

DBInstanceClassHugePagesDefault

Renvoie une valeur booléenne. Actuellement, la prise en charge ne concerne que les moteurs Oracle.

Pour de plus amples informations, veuillez consulter Activation de HugePages pour une instance RDS for Oracle.

DBInstanceClassMemory

Renvoie un entier correspondant au nombre d'octets de mémoire disponibles pour le processus de base de données. Ce nombre est calculé en interne, en commençant par la quantité totale de mémoire pour la classe d'instance de base de données. Le calcul en soustrait la mémoire réservée au système d'exploitation et aux RDS processus qui gèrent l'instance. Par conséquent, ce nombre est toujours légèrement inférieur aux chiffres de mémoire affichés dans les tables de classes d'instance dans Classes d'instances de base de données . La valeur exacte dépend d'une combinaison de facteurs. Il s'agit notamment de la classe d'instance, du moteur de base de données et du fait que cela s'applique à une RDS instance ou à une instance faisant partie d'un cluster Aurora.

DBInstanceVCPU

Renvoie un entier représentant le nombre d'unités centrales virtuelles (vCPUs) utilisées par Amazon RDS pour gérer l'instance.

EndPointPort

Renvoie un entier qui représente le port utilisé lors de la connexion à l'instance de base de données.

TrueIfReplica

Renvoie `1` si l'instance de base de données est un réplica en lecture et `0` si ce n'est pas le cas. Il s'agit de la valeur par défaut du `read_only` paramètre dans MySQL.

Opérateurs de formule de paramètre de bases de données

Les formules de paramètre DB prennent en charge deux opérateurs : division et multiplication.

Opérateur de division : /

Divise le dividende par le diviseur, en renvoyant un quotient entier. Les décimales dans le quotient sont tronquées, pas arrondies.

Syntaxe

```
dividend / divisor
```

Les arguments de dividende et de diviseur doivent être des expressions entières.

Opérateur de multiplication : *

Multiplie les expressions, affichant ainsi le résultat des expressions. Les décimales dans les expressions sont tronquées, pas arrondies.

Syntaxe

```
expression * expression
```

Les deux expressions doivent être des entiers.

Fonctions de paramètre de bases de données

Vous spécifiez les arguments des fonctions de paramètres de base de données sous la forme d'entiers ou de formules. Chaque fonction doit avoir au moins un argument. Spécifiez plusieurs arguments sous la forme d'une liste séparée par des virgules. Cette liste ne peut pas contenir de membres vides, tels que argument1,,argument3. Les noms de fonctions ne sont pas sensibles à la casse.

IF

Renvoie un argument.

Actuellement, la prise en charge ne concerne que les moteurs Oracle et `{DBInstanceClassHugePagesDefault}` est le seul premier argument pris en charge. Pour plus d'informations, consultez Activation de HugePages pour une instance RDS for Oracle.

Syntaxe

```
IF(argument1, argument2, argument3)
```

Renvoie le deuxième argument si le premier a la valeur true. Sinon, renvoie le troisième argument.

GREATEST

Renvoie la plus grande valeur depuis une liste d'entiers ou de formules de paramètres.

Syntaxe

```
GREATEST(argument1, argument2,...argumentn)
```

Renvoie un entier.

LEAST

Renvoie la plus petite valeur depuis une liste d'entiers ou de formules de paramètres.

Syntaxe

```
LEAST(argument1, argument2,...argumentn)
```

Renvoie un entier.

SUM

Ajoute les valeurs des formules de paramètres ou d'entiers spécifiés.

Syntaxe

```
SUM(argument1, argument2,...argumentn)
```

Renvoie un entier.

Expressions de paramètre de base de données booléennes

Une expression de paramètre de base de données booléenne se réduit à une valeur booléenne de 1 ou 0. L'expression est insérée entre guillemets.

> **Note**
>
> Les expressions de paramètres de base de données booléennes ne sont prises en charge que pour le moteur SQL Postgre.

Syntaxe

```
"expression operator expression"
```

Les deux expressions doivent se réduire à des entiers. Une expression peut être l'un des éléments suivants :

- Une constante entière
- Une formule de paramètre de base de données
- Une fonction de paramètre de base de données
- Une variable de paramètre de base de données

Les expressions de paramètres de base de données booléennes prennent en charge les opérateurs d'inégalité suivants :

L'opérateur supérieur à : **>**

Syntaxe

```
"expression > expression"
```

L'opérateur inférieur à : **<**

Syntaxe

```
"expression < expression"
```

Les opérateurs supérieur ou égal à : **>=**, **=>**

Syntaxe

```
"expression >= expression"
"expression => expression"
```

Les opérateurs inférieur ou égal à : <=, =<

Syntaxe

```
"expression <= expression"
"expression =< expression"
```

Example utilisation d'une expression de paramètre de base de données booléenne

L'exemple d'expression de paramètre de base de données booléenne suivant compare le résultat d'une formule de paramètre à un entier. Il le fait pour modifier le paramètre de base de données booléen `wal_compression` pour une instance de base de données PostgreSQL. L'expression du paramètre compare le nombre de vCPUs à la valeur 2. Si le nombre de vCPUs est supérieur à 2, le paramètre `wal_compression` DB est défini sur true.

```
aws rds modify-db-parameter-group --db-parameter-group-name group-name \
--parameters "ParameterName=wal_compression,ParameterValue=\"{DBInstanceVCPU} > 2\" "
```

Expressions de journal des paramètres de base de données

Vous pouvez définir une valeur de paramètre de base de données entier à une expression de journal. Vous insérez l'expression entre des accolades : {}. Exemples :

```
{log(DBInstanceClassMemory/8187281418)*1000}
```

La fonction `log` représente la base du journal 2. Cet exemple utilise également la variable de formule `DBInstanceClassMemory`. Consultez Variables de formule de paramètre de bases de données.

> **Note**
>
> Actuellement, vous ne pouvez pas spécifier le SQL `innodb_log_file_size` paramètre My avec une valeur autre qu'un entier.

Exemples de valeurs de paramètre de bases de données

Ces exemples montrent l'utilisation de formules, de fonctions et d'expressions pour les valeurs des paramètres de base de données.

> **Warning**
>
> La définition incorrecte des paramètres d'un groupe de paramètres de base de données peut avoir des effets indésirables involontaires. Cela peut se manifester par une dégradation des performances et l'instabilité du système. Agissez avec prudence lorsque vous modifiez des paramètres de base de données et sauvegardez vos données avant de modifier votre groupe de paramètres de base de données. Testez les modifications de groupe de paramètres sur une instance de base de données de test, créée à l'aide de point-in-time-restores, avant d'appliquer ces modifications de groupe de paramètres à vos instances de base de données de production.

Example en utilisant la fonction de paramètre de base de données GREATEST

Vous pouvez spécifier la fonction `GREATEST` dans un paramètre de processus Oracle. Utilisez-la pour définir le nombre de processus utilisateur sur le plus grand des deux nombres : 80 ou `DBInstanceClassMemory` divisé par 9 868 951.

```
GREATEST({DBInstanceClassMemory/9868951},80)
```

Example en utilisant la fonction de paramètre de base de données LEAST

Vous pouvez spécifier la `LEAST` fonction dans une valeur Mon SQL `max_binlog_cache_size` paramètre. Utilisez-le pour définir la taille de cache maximale qu'une transaction peut utiliser dans une SQL instance My à 1 Mo ou `DBInstanceClass` /256.

```
LEAST({DBInstanceClassMemory/256},10485760)
```

Création d'un ElastiCache cache Amazon à l'aide des paramètres de l'instance Amazon RDS DB

ElastiCache est un service de mise en cache en mémoire entièrement géré qui fournit des latences de lecture et d'écriture de l'ordre de l'ordre de l'ordre de l'ordre de la microseconde pour des cas d'utilisation flexibles en temps réel. ElastiCache peut vous aider à accélérer les performances des applications et des bases de données. Vous pouvez l'utiliser ElastiCache comme magasin de données principal pour les cas d'utilisation qui ne nécessitent pas la durabilité des données, tels que les classements de jeu, le streaming et l'analyse des données. ElastiCache contribue à éliminer la complexité associée au déploiement et à la gestion d'un environnement informatique distribué. Pour plus d'informations, consultez les sections Cas ElastiCache d'utilisation courants et How ElastiCache Can Help for Memcached et Cases d' ElastiCache utilisation courants et How ElastiCache Can Help for Redis. OSS Vous pouvez utiliser la RDS console Amazon pour créer ElastiCache du cache.

Vous pouvez utiliser Amazon ElastiCache sous deux formats. Vous pouvez commencer avec un cache sans serveur ou choisir de concevoir votre propre cluster de cache. Si vous choisissez de concevoir votre propre cluster de cache, il ElastiCache fonctionne à la fois avec les moteurs Redis OSS et Memcached. Si vous ne savez pas quel moteur vous souhaitez utiliser, consultez Comparaison de Memcached et RedisOSS. Pour plus d'informations sur Amazon ElastiCache, consultez le guide de ElastiCache l'utilisateur Amazon.

Rubriques

- Vue d'ensemble de la création du ElastiCache cache avec les paramètres de l'instance de base de données du RDS cluster
- Création d'un ElastiCache cache avec les paramètres d'une instance de base de données du cluster de RDS base

Vue d'ensemble de la création du ElastiCache cache avec les paramètres de l'instance de base de données du RDS cluster

Vous pouvez créer un ElastiCache cache depuis Amazon RDS en utilisant les mêmes paramètres de configuration qu'une instance de base de données de nouvellement créée ou existante.

Voici quelques cas d'utilisation pour associer un ElastiCache cache à votre instance de base de données de de base de données :

- Vous pouvez réduire les coûts et améliorer vos performances en utilisant ElastiCache avec RDS plutôt que de fonctionner RDS seul.

  Par exemple, vous pouvez économiser jusqu'à 55 % sur les coûts et obtenir des performances de lecture jusqu'à 80 fois plus rapides en utilisant « ElastiCache avec RDS pour moi » SQL plutôt que « RDS pour moi » SQL uniquement.
- Vous pouvez utiliser le ElastiCache cache comme magasin de données principal pour les applications qui ne nécessitent pas la durabilité des données. Vos applications qui utilisent Redis OSS ou Memcached peuvent l'utiliser pratiquement ElastiCache sans aucune modification.

Lorsque vous créez un ElastiCache cache à partir deRDS, le ElastiCache cache hérite des paramètres suivants de l'instance de base de données du associée RDS :

- ElastiCache paramètres de connectivité
- ElastiCache paramètres de sécurité

Vous pouvez également définir les paramètres de configuration du cache en fonction de vos besoins.

Configuration ElastiCache dans vos applications

Vos applications doivent être configurées pour utiliser ElastiCache le cache. Vous pouvez également optimiser et améliorer les performances du cache en configurant vos applications pour qu'elles utilisent des stratégies de mise en cache en fonction de vos besoins.

- Pour accéder à votre ElastiCache cache et commencer, consultez Getting started with ElastiCache (RedisOSS) et Getting started with ElastiCache (Memcached).
- Pour plus d'informations sur les stratégies de mise en cache, consultez Stratégies de mise en cache et meilleures pratiques pour Memcached et stratégies de mise en cache et meilleures pratiques pour Redis. OSS
- Pour plus d'informations sur la haute disponibilité dans les clusters ElastiCache (RedisOSS), consultez Haute disponibilité à l'aide de groupes de réplication.
- Vous pouvez encourir des coûts liés au stockage des sauvegardes, au transfert de données au sein ou entre les régions, ou à l'utilisation de AWS Outposts. Pour plus de détails sur les prix, consultez ElastiCache les tarifs Amazon.

Création d'un ElastiCache cache avec les paramètres d'une instance de base de données du cluster de RDS base

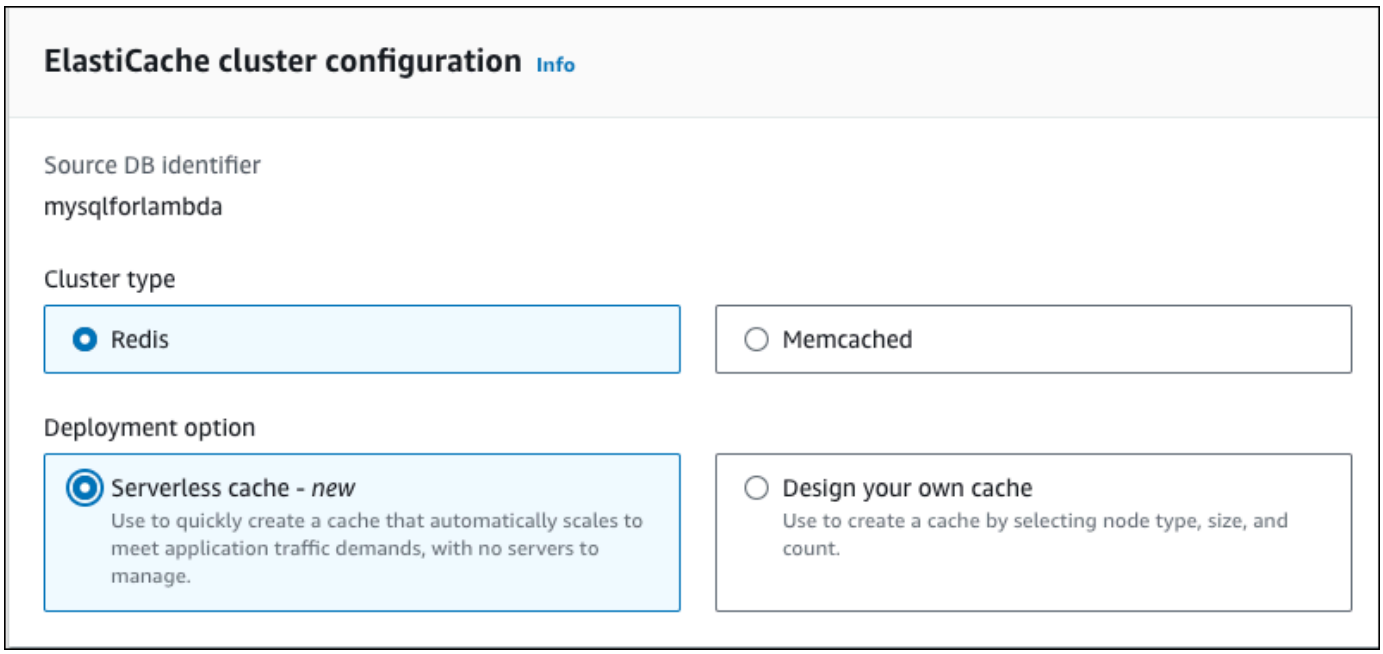
Vous pouvez créer un ElastiCache cache pour les instances de base de données de vos clusters de RDS base avec des paramètres hérités de l'instance de de base de données.

Création d'un ElastiCache cache avec les paramètres d'une instance de de base de données

1. Pour créer une instance de base de données, suivez les instructions de la section Création d'une instance de base de données Amazon RDS.
2. Après avoir créé une instance de base de données du RDS cluster , la console affiche la fenêtre Extensions suggérées. Sélectionnez Créer un ElastiCache cluster à RDS partir de vos paramètres de base de données.

   Pour une base de données existante, dans la page Bases de données, sélectionnez l'instance de de base de données requise. Dans le menu déroulant Actions, choisissez Create ElastiCache cluster pour créer un ElastiCache cache doté des mêmes paramètres RDS que votre instance DB de RDS DB existante.

   Dans la section ElastiCache de configuration, l'identifiant de base de données source indique de quelle instance de de base de données le ElastiCache cache hérite des paramètres.
3. Choisissez si vous souhaitez créer un cluster Redis OSS ou Memcached. Pour plus d'informations, consultez Comparaison de Memcached et RedisOSS.



4. Ensuite, choisissez si vous souhaitez créer un cache sans serveur ou concevoir votre propre cache. Pour plus d'informations, consultez la section Choix entre les options de déploiement.

   Si vous choisissez le cache sans serveur :

   a. Dans les paramètres du cache, entrez les valeurs du nom et de la description.

   b. Sous Afficher les paramètres par défaut, conservez les paramètres par défaut pour établir la connexion entre votre cache et l'instance de de base de données.

   c. Vous pouvez également modifier les paramètres par défaut en choisissant Personnaliser les paramètres par défaut. Sélectionnez les paramètres de ElastiCache connectivité, les paramètres ElastiCache de sécurité et les limites d'utilisation maximales.

5. Si vous choisissez Concevez votre propre cache :

   a. Si vous avez choisi le OSScluster Redis, choisissez si vous souhaitez conserver le mode cluster activé ou désactivé. Pour plus d'informations, consultez Réplication : Redis OSS (mode cluster désactivé) contre Redis OSS (mode cluster activé).

   b. Saisissez des valeurs pour Nom, Description et Version du moteur.

      Pour Version du moteur, la valeur par défaut recommandée est la dernière version du moteur. Vous pouvez également choisir la version du moteur qui répond le mieux à vos besoins pour le ElastiCache cache.

   c. Choisissez le type de nœud dans l'option Type de nœud. Pour plus d'informations, consultez Gestion des nœuds.

      Si vous choisissez de créer un OSS cluster Redis avec le mode cluster défini sur Activé, entrez le nombre de partitions (partitions/groupes de nœuds) dans l'option Nombre de partitions.

      Saisissez le nombre de réplicas de chaque partition dans Nombre de réplicas.

      > **Note**
      >
      > Le type de nœud sélectionné, le nombre de partitions et le nombre de répliques ont tous une incidence sur les performances de votre cache et le coût des ressources. Veillez à ce que ces paramètres correspondent aux besoins de votre base de données. Pour plus d'informations sur les tarifs, consultez ElastiCache les tarifs Amazon.

d. Sélectionnez les paramètres ElastiCache de connectivité et les paramètres ElastiCache de sécurité. Vous pouvez conserver les paramètres par défaut ou les personnaliser selon vos besoins.

6. Vérifiez les paramètres par défaut et hérités de votre ElastiCache cache. Certains paramètres ne peuvent pas être modifiés après la création.

> **Note**
>
> RDSpeut ajuster la fenêtre de sauvegarde de votre ElastiCache cache pour répondre à la durée minimale requise de 60 minutes. La fenêtre de sauvegarde de votre base de données source reste la même.

7. Lorsque vous êtes prêt, choisissez Create ElastiCache cache.

La console affiche une bannière de confirmation pour la création du ElastiCache cache. Suivez le lien figurant dans la bannière menant à la ElastiCache console pour afficher les détails du cache. La ElastiCache console affiche le ElastiCache cache nouvellement créé.

Migration automatique des EC2 bases de données vers) à l'aide RDS de AWS Database Migration Service

Vous pouvez utiliser la RDSconsole pour migrer une EC2 base de données vers . utilise AWS Database Migration Service (AWS DMS) pour migrer votre EC2 base de données source. AWS DMS vous permet de migrer des bases de données relationnelles vers votre AWS Cloud. Pour plus d'informations AWS Database Migration Service, voir Qu'est-ce que c'est AWS Database Migration Service ? dans le guide de AWS Database Migration Service l'utilisateur.

Pour commencer la migration, vous devez créer une instance de base de données du RDS de base de données équivalente vers laquelle migrer les données. Après avoir créé votre base de données cible, vous pouvez y importer votre EC2 base de données.

Présentation

La RDSconsole vous permet de migrer des bases de données vers des EC2 bases de RDSdonnées équivalentes. Vous devez créer une RDSbase de données pour permettre la migration depuis la console.

Vous pouvez migrer des EC2 bases de données pour les moteurs de base de données suivants :

- Mon SQL
- MariaDB
- Poster SQL

Le processus de migration comprend les étapes suivantes :

- Créez une base de données équivalente dans . Pour que les bases de données soient équivalentes, elles doivent avoir le même moteur de base de données et des versions de moteur compatibles. Ils doivent également être dans le même sacVPC. Pour obtenir des instructions sur la création de votre base de données, consultez Création d'une instance de base de données Amazon RDS.
- Choisissez le type de réplication pour votre base de données :
  - Migration à charge complète : copie l'intégralité de la base de données source vers la base de données cible, en créant de nouvelles tables dans la cible si nécessaire.

> **Note**
>
> Cette option entraîne une panne de votre RDSbase de données .

- Migration à chargement complet et capture des données modifiées (CDC) : similaire à la migration à chargement complet, avec cette option, copie l'intégralité de la base de données source vers la base de données cible. Toutefois, après la migration à chargement complet, applique les modifications capturées dans la source à la base de données cible. La capture des données de modification collecte les modifications apportées aux journaux de base de données en utilisant le moteur de base de données natifAPI.

  > **Note**
  >
  > Cette option entraîne une panne de votre RDSbase de données .

- Modifier la capture des données (CDC) : utilisez cette option pour que votre base de données cible reste disponible pendant la migration. migre les modifications en cours dans votre base de données source vers la base de données cible.

- crée les ressources réseau nécessaires pour faciliter la migration. Une fois qu' a RDScréé les ressources requises, elle vous informe des ressources créées et vous permet de lancer le transfert de données.

  Le temps nécessaire pour effectuer la migration dépend du type de réplication et de la taille de la base de données source.

Prérequis

Moi SQL et MariaDB

Avant de commencer à utiliser une base de données My SQL ou MariaDB comme base de données source, assurez-vous que vous remplissez les conditions préalables suivantes. Ces conditions préalables s'appliquent aux sources AWS gérées.

Vous devez disposer d'un compte AWS DMS doté du rôle d'administrateur de réplication. Ce rôle nécessite les privilèges suivants :

- REPLICATIONCLIENT— Ce privilège n'est requis que pour CDC les tâches. En d'autres termes, full-load-only les tâches ne nécessitent pas ce privilège.
- REPLICATIONSLAVE— Ce privilège n'est requis que pour CDC les tâches. En d'autres termes, full-load-only les tâches ne nécessitent pas ce privilège.

L' AWS DMS utilisateur doit également disposer de SELECT privilèges pour les tables sources destinées à la réplication.

Accordez les privilèges suivants si vous utilisez les évaluations de prémigration SQL spécifiques à My.

```
grant select on mysql.user to <dms_user>;
grant select on mysql.db to <dms_user>;
grant select on mysql.tables_priv to <dms_user>;
grant select on mysql.role_edges to <dms_user>  #only for MySQL version 8.0.11 and
 higher
```

Poster SQL

Avant de migrer des données depuis une base de données SQL source Postgre AWS gérée, procédez comme suit :

- Nous vous recommandons d'utiliser un compte AWS utilisateur avec les autorisations minimales requises pour l'SQLinstance de base de données Postgre comme compte utilisateur pour le point de terminaison SQL source Postgre pour. AWS DMS L'utilisation du compte principal n'est pas recommandée. Le compte doit avoir le rôle `rds_superuser` et le rôle `rds_replication`. Le rôle `rds_replication` accorde les autorisations permettant de gérer des emplacements logiques et de diffuser les données à l'aide d'emplacements logiques.

> **Note**
>
> Certaines AWS DMS transactions restent inactives pendant un certain temps avant que le DMS moteur ne les réutilise. En utilisant ce paramètre `idle_in_transaction_session_timeout` dans les SQL versions 9.6 et supérieures de Postgre, vous pouvez provoquer l'expiration et l'échec des transactions inactives.

Limites

Les limites suivantes s'appliquent au processus de migration automatique :

- Le statut de votre base de données cible doit être Disponible pour commencer la migration de la base de données source.
- Lorsque vous migrez depuis une base de données My SQL source, votre RDScompte doit avoir le rôle d'administrateur de réplication. Vous devez également disposer des privilèges appropriés pour ce rôle.
- Votre EC2 instance et votre base de données cible doivent être identiquesVPC.
- Vous ne pouvez pas migrer votre EC2 base de données vers les bases de données cibles suivantes lorsque vous utilisez l'action Migrer les données depuis la EC2 base de données :
  - Base de données membre d'un cluster
  - Bases de données Oracle, SQL Server et DB2
  - Bases de données dont SQL la version My est inférieure à 5.7
  - Bases de données dont la SQL version Postgre est inférieure à 10.4
  - Bases de données avec une version de MariaDB inférieure à 10.2

Création de ressources IAM pour des migrations homogènes

RDS les utilise AWS DMS pour migrer vos données. Pour accéder à vos bases de données et migrer les données, AWS DMS créez un environnement sans serveur pour des migrations de données homogènes. Dans cet environnement, AWS DMS nécessite un accès au peering VPC, aux tables de routage, aux groupes de sécurité et à d'autres ressources. AWS AWS DMS Stocke également les journaux, les statistiques et la progression de chaque migration de données sur Amazon CloudWatch. Pour créer un projet de migration de données, il AWS DMS faut avoir accès à ces services.

AWS DMS Nécessite également l'accès aux secrets qui représentent un ensemble d'informations d'identification utilisateur pour authentifier la connexion à la base de données pour les connexions source et cible.

> **Note**
>
> En utilisant l'action Migrer les données depuis l' EC2 instance, vous pouvez utiliser la console RDS pour générer ces ressources IAM. Ignorez cette étape si vous utilisez les ressources IAM générées par la console.

Vous avez besoin des ressources IAM suivantes pour ce processus :

Rubriques

- Création d'une politique IAM pour des migrations de données homogènes
- Création d'un rôle IAM pour des migrations de données homogènes
- Création d'une politique et d'un rôle d'accès secrets
- Création d'un IAM rôle AWS DMS pour gérer Amazon VPC

Création d'une politique IAM pour des migrations de données homogènes

Au cours de cette étape, vous créez une politique IAM qui donne accès AWS DMS à Amazon EC2 et à ses CloudWatch ressources. Ensuite, créez un rôle IAM et attachez cette politique.

Pour créer une politique IAM pour la migration des données

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le volet de navigation, choisissez Politiques.
3. Choisissez Create Policy (Créer une politique).
4. Sur la page Créer une politique, choisissez l’onglet JSON.
5. Collez le code JSON suivant dans l’éditeur.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
```

```
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeVpcPeeringConnections",
                "ec2:DescribeVpcs",
                "ec2:DescribePrefixLists",
                "logs:DescribeLogGroups"
            ],
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "servicequotas:GetServiceQuota"
            ],
            "Resource": "arn:aws:servicequotas:*:*:vpc/L-0EA8095F"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogGroup",
                "logs:DescribeLogStreams"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:dms-data-migration-*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "logs:CreateLogStream",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:dms-data-migration-*:log-
stream:dms-data-migration-*"
        },
        {
            "Effect": "Allow",
            "Action": "cloudwatch:PutMetricData",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateRoute",
                "ec2:DeleteRoute"
            ],
```

```
            "Resource": "arn:aws:ec2:*:*:route-table/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:security-group/*",
                "arn:aws:ec2:*:*:security-group-rule/*",
                "arn:aws:ec2:*:*:route-table/*",
                "arn:aws:ec2:*:*:vpc-peering-connection/*",
                "arn:aws:ec2:*:*:vpc/*"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:AuthorizeSecurityGroupIngress"
            ],
            "Resource": "arn:aws:ec2:*:*:security-group-rule/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:AuthorizeSecurityGroupIngress",
                "ec2:RevokeSecurityGroupEgress",
                "ec2:RevokeSecurityGroupIngress"
            ],
            "Resource": "arn:aws:ec2:*:*:security-group/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "ec2:AcceptVpcPeeringConnection",
                "ec2:ModifyVpcPeeringConnectionOptions"
            ],
            "Resource": "arn:aws:ec2:*:*:vpc-peering-connection/*"
        },
        {
            "Effect": "Allow",
            "Action": "ec2:AcceptVpcPeeringConnection",
```

```
            "Resource": "arn:aws:ec2:*:*:vpc/*"
        }
    ]
}
```

6. Choisissez Next: Tags (Suivant : Balises), puis Next: Review (Suivant : Vérification).
7. Entrez **`HomogeneousDataMigrationsPolicy`** pour Nom*, puis choisissez Créer une politique.

Création d'un rôle IAM pour des migrations de données homogènes

Au cours de cette étape, vous créez un rôle IAM qui donne accès à AWS Secrets Manager Amazon EC2 et CloudWatch.

Pour créer un rôle IAM pour les migrations de données

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Sélectionnez Create role (Créer un rôle).
4. Sur la page Sélectionner une entité de confiance, pour Type d'entité approuvée, choisissez Service AWS . Pour Cas d'utilisation d'autres services AWS , choisissez DMS.
5. Cochez la case DMS et choisissez Suivant.
6. Sur la page Ajouter des autorisations, choisissez HomogeneousDataMigrationsPolicycelle que vous avez créée auparavant. Choisissez Suivant.
7. Sur la page Nommer, vérifier et créer, entrez **`HomogeneousDataMigrationsRole`** pour Nom du rôle et choisissez Créer un rôle.
8. Sur la page Rôles, entrez **`HomogeneousDataMigrationsRole`** pour Nom du rôle. Sélectionnez HomogeneousDataMigrationsRole.
9. Sur la HomogeneousDataMigrationsRolepage, choisissez l'onglet Relations de confiance. Choisissez Edit trust policy (Modifier la politique d'approbation).
10. Sur la page Modifier la politique d'approbation, collez le code JSON suivant dans l'éditeur en remplaçant le texte existant.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "",
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "dms-data-migrations.amazonaws.com",
                    "dms.your_region.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Dans l'exemple précédent, remplacez *your_region* par le nom de votre Région AWS.

La politique basée sur les ressources précédente donne aux responsables de AWS DMS service les autorisations nécessaires pour effectuer des tâches conformément à la politique gérée par le client. HomogeneousDataMigrationsPolicy

11. Choisissez Mettre à jour une politique.

Création d'une politique et d'un rôle d'accès secrets

Suivez les procédures ci-dessous pour créer votre politique d'accès secret et votre rôle permettant à DMS d'accéder aux informations d'identification utilisateur de vos bases de données source et cible.

Pour créer la politique et le rôle d'accès secret, qui permettent à Amazon RDS d'accéder AWS Secrets Manager à votre secret approprié

1. Connectez-vous à la console AWS Identity and Access Management (IAM) AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Sélectionnez Politiques, puis Créer une politique.

3. Choisissez JSON et entrez la politique suivante pour permettre d'accéder à votre secret et de le déchiffrer.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "arn:aws:secretsmanager:us-
east-1:111122223333:secret:SecretName-ABCDEF"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt",
                "kms:DescribeKey"
            ],
            "Resource": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

   *secret_arn*Voici l'ARN de votre secret, que vous pouvez obtenir de l'un ou l'autre `SecretsManagerSecretId` selon le cas, et *kms_key_arn* l'ARN de la AWS KMS clé que vous utilisez pour chiffrer votre secret, comme dans l'exemple suivant.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "arn:aws:secretsmanager:us-
east-2:123456789012:secret:MySQLTestSecret-qeHamH"
```

```
        },
        {
             "Effect": "Allow",
             "Action": [
                        "kms:Decrypt",
                        "kms:DescribeKey"
                      ],
             "Resource": "arn:aws:kms:us-
east-2:123456789012:key/761138dc-0542-4e58-947f-4a3a8458d0fd"
        }
    ]
}
```

> **Note**
>
> Si vous utilisez la clé de chiffrement par défaut créée par AWS Secrets Manager, il n'est pas nécessaire de spécifier les AWS KMS autorisations pour*kms_key_arn*.
> Si vous souhaitez que votre politique donne accès aux deux secrets, il vous suffit de spécifier un objet de ressource JSON supplémentaire pour l'autre*secret_arn*.

4. Vérifiez et créez la politique avec un nom convivial et une description facultative.
5. Choisissez Rôles, puis Créer un rôle.
6. Choisissez Service AWS comme type d'entité de confiance.
7. Choisissez DMS dans la liste des services comme service de confiance, puis choisissez Suivant : Autorisations.
8. Recherchez et attachez la politique que vous avez créée à l'étape 4, puis ajoutez des balises et passez en revue votre rôle. À ce stade, modifiez les relations de confiance pour le rôle afin d'utiliser votre principal de service régional Amazon RDS comme entité de confiance. Ce principal a le format suivant.

```
dms.region-name.amazonaws.com
```

Ici, *region-name* est le nom de votre région, par exemple `us-east-1`. Voici donc un principe de service régional Amazon RDS pour cette région.

```
dms.us-east-1.amazonaws.com
dms-data-migrations.amazonaws.com
```

Création d'un IAM rôle AWS DMS pour gérer Amazon VPC

Vous devez créer un IAM rôle AWS DMS pour gérer les VPC paramètres de vos ressources. Ce rôle doit être disponible pour que la migration soit réussie.

Création du **`dms-vpc-role`** pour la migration de base de données

<result>

Cela crée le rôle permettant DMS de gérer les VPC paramètres de la migration.
</result>

1. Connectez-vous à la IAM console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation de la console, choisissez Rôles, puis Créer un rôle.
3. Choisissez l'Service AWSoption Sélectionner une entité de confiance.

   Pour Cas d'utilisation, sélectionnez DMS.
4. Pour l'étape Ajouter des autorisations, sélectionnez `AmazonDMSVPCManagementRole` et choisissez Next.
5. Dans la page Nom, révision et création, définissez le nom du rôle sur `dms-vpc-role` et choisissez Créer un rôle.

Configuration de la migration des données pour la EC2 base de données

Pour commencer à migrer des données depuis votre base de données EC2 source, vous devez créer une RDSbase de données équivalente. Pour obtenir des instructions sur la création de votre base de données, consultez Création d'une instance de base de données Amazon RDS.

Après avoir créé votre base de données cible, procédez comme suit pour configurer la migration des données :

Configurer un projet de migration de données

1. Sélectionnez la base de données cible sur la page Bases de données de la RDS console.
2. Choisissez le menu déroulant Actions et sélectionnez l'option Migrer les données depuis la EC2 base de données. Pour connaître les bases de données cibles prises en charge, consultezLimites.

3. Dans la section Sélectionner la EC2 base de données source :

    1. Vérifiez le type de moteur et assurez-vous qu'il est identique à celui de votre base de données source.

       Vérifiez également si les versions du moteur sont compatibles.
    2. Par EC2exemple, choisissez l'EC2instance dans laquelle réside votre base de données source.
    3. Pour Port, entrez le port sur lequel votre base de données source autorise le trafic.
    4. Pour Secret, choisissez Créer et utilisez un nouveau secret si vous n'en avez pas déjà un. Entrez le nom d'utilisateur et le mot de passe de votre base de données source. Choisissez également la KMS clé avec laquelle chiffrer votre secret.

       Si vous avez un secret existant, sélectionnez Utiliser un secret existant, puis choisissez secret dans le menu déroulant.
    5. Pour un IAMrôle par secret, si vous avez un IAM rôle existant, sélectionnez Utiliser un IAM rôle existant et choisissez un IAM rôle dans la liste déroulante qui peut accéder à l'ID secret de l'étape précédente.

       Si vous n'avez pas de IAM rôle existant, choisissez Créer et utiliser un nouveau IAM rôle. Entrez le nouveau nom de votre rôle pour le nom du IAM rôle. Vous pouvez consulter les autorisations associées à ce rôle dans le lien ci-dessous.
4. Dans la section Afficher la RDS base de données cible :

    1. Confirmez les paramètres de votre base de données cible en haut de la section.
    2. Pour Secret, choisissez Créer et utilisez un nouveau secret si aucun secret ne contient les informations d'identification de votre base de données cible.

       Si vous possédez déjà un secret, sélectionnez-le dans le menu déroulant.
    3. Pour un IAMrôle par secret, sélectionnez un IAM rôle qui peut accéder au secret depuis l'étape précédente. Vous pouvez également créer un nouveau IAM rôle si vous n'en avez pas déjà IAM un.

       Si le menu déroulant ne renseigne pas les IAM rôles, spécifiez le IAMrôle ARN dans le format. `arn:aws:iam:`*`account_id`*`:role/`*`roleName`*
5. Dans la section Configurer la migration des données :

1. Sélectionnez le type de migration de données en choisissant entre Chargement complet, Capture de données de chargement complet et de modification (CDC) ou Capture de données de modification (CDC). Pour plus d'informations sur ces options, consultez Présentation.

   Vous ne pouvez pas modifier le type de migration une fois celle-ci lancée.
2. Pour IAMle rôle pour la migration des données, si vous avez un IAM rôle existant, sélectionnez Utiliser un IAM rôle existant et choisissez un IAM rôle dans la liste déroulante qui accorde DMS les autorisations nécessaires pour créer les ressources nécessaires à la migration. Si vous n'avez pas de IAM rôle existant, choisissez Créer et utiliser un nouveau IAM rôle.
6. Vérifiez que l'onglet Afficher les paramètres de migration affiche les paramètres requis pour que votre migration de données soit correctement configurée.
7. Sélectionnez Migrer pour terminer la configuration de la migration.

Une fois ces étapes terminées, vous pouvez voir les ressources en cours de configuration pour la migration des données en choisissant Afficher les détails dans la bannière de progression de la console. Une fois les ressources requises configurées, la migration démarre automatiquement. Si vous créez

Pour migrer plusieurs bases de données vers la base de données cible, recommencez ce processus en fournissant des informations sur la nouvelle EC2 base de données.

Gestion des migrations de données

Après avoir utilisé l'action Migrer les données depuis la EC2 base de données depuis la RDS console, lance automatiquement la migration.

Si vous avez utilisé la AWS DMS console pour créer les ressources de migration, vous pouvez démarrer le processus de migration.

Démarrage de la migration des données

Pour démarrer la migration des données, procédez comme suit :

Démarrage d'une migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.

3. Sous l'onglet Migrations de données, les migrations de données associées répertorie les migrations de données disponibles.

   Les migrations configurées à l'aide de la RDSconsole démarrent automatiquement une fois que les ressources requises sont configurées.

   Les migrations configurées à l'aide de la DMS console sont définies sur Prêt.

   Pour commencer ces migrations, sélectionnez le menu déroulant Actions, puis sélectionnez Démarrer.
4. La migration des données de votre EC2 base de données commence alors.

Arrêt de la migration des données

Pour les migrations de données dont le type de réplication est à pleine charge, l'arrêt de la migration entraîne l'arrêt du processus et ne peut pas être repris. Une fois la migration arrêtée, vous devez redémarrer la migration.

Pour les migrations dont le type de réplication est défini pour modifier la capture des données (CDC) ou le chargement completCDC, vous pouvez arrêter le processus de réplication continue et le reprendre ultérieurement.

Arrêter une migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.
3. Sous l'onglet Migrations de données, les migrations de données associées répertorie les migrations de données en cours.

   Pour arrêter une migration, sélectionnez une migration de données, puis sélectionnez Arrêter dans le menu déroulant Actions.
4. Cela arrête la migration des données pour votre EC2 base de données.

Reprise de la migration des données

Pour les migrations de données dont le type de réplication est le chargement complet et la migration de capture de données de modification (CDC) ou de capture de données de modification (CDC), vous pouvez reprendre le CDC processus à partir du dernier point d'arrêt.

Reprise d'une migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.
3. Sous l'onglet Migrations de données, les migrations de données associées répertorie les migrations de données arrêtées.

   Pour reprendre une migration, sélectionnez une migration de données, puis sélectionnez Reprendre le traitement dans le menu déroulant Actions.
4. Cela reprend la migration des données pour votre EC2 base de données.

Supprimer la migration des données

Pour supprimer une migration de données associée, suivez les instructions suivantes

Supprimer une migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.
3. Pour supprimer une migration, sélectionnez une migration de données, puis sélectionnez Supprimer dans le menu déroulant Actions.
4. Cela supprime la migration des données.

La suppression d'une migration de données en cours n'a aucune incidence sur les données déjà chargées dans la base de données cible.

Redémarrer la migration des données

Pour redémarrer une migration de données associée depuis un point de CDC départ, suivez les instructions suivantes

Redémarrer une migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.
3. Pour redémarrer une migration, sélectionnez une migration de données, puis sélectionnez Redémarrer dans le menu déroulant Actions.

4. Cela redémarre la migration des données à partir d'un point de CDC départ.

Le redémarrage d'une migration de données en cours n'a aucune incidence sur les données déjà chargées dans la base de données cible.

Surveillance de vos migrations de données

Une fois les migrations de données lancées, vous pouvez suivre leur statut et leur progression. Les migrations de données de grands ensembles de données prennent des heures. Pour gérer la fiabilité, la disponibilité et les performances de votre migration de données, surveillez régulièrement sa progression.

Pour vérifier le statut et la progression de votre migration de données

1. Choisissez la base de données cible sur la page Bases de données de la RDS console.
2. Sur la page des détails de la base de données, choisissez l'onglet Migrations de données.
3. La section Migrations de données associées répertorie vos migrations de données. Vérifiez la colonne État.
4. Pour exécuter des migrations de données, la colonne Processus de migration affiche le pourcentage de données migrées.
5. Pour surveiller le processus dans CloudWatch, utilisez le lien dans la CloudWatchcolonne in.

Statuts de migration

Pour chaque migration de données que vous exécutez, la RDSconsole affiche le statut. La liste suivante inclut les statuts :

- `Ready`: La migration des données est prête à démarrer.
- `Starting`: crée RDSl'environnement sans serveur pour la migration de vos données.
- `Load running`: effectue la migration à chargement complet.
- `Load complete, replication ongoing`: a terminé le chargement complet et reproduit désormais les modifications en cours. Ce statut ne s'applique qu'aux migrations à chargement complet et aux migrations CDC de type.
- `Replication ongoing`: reproduit les modifications en cours. Ce statut s'applique uniquement aux migrations CDC de type.

- `Stopping`: arrête les migrations de données. Ce statut s'applique lorsque vous choisissez d'arrêter la migration des données dans le menu Actions.
- `Stopped`: a arrêté la migration des données.
- `Failed`: La migration des données a échoué. Pour plus d’informations, consultez les fichiers journaux.
- `Restarting`: La migration des données a relancé une réplication de données en cours depuis un point de CDC départ.

Tutoriel : Création d'un de base de données MySQL avec un paramètre personnalisé et un nouveau groupe d'options

Dans ce didacticiel, vous allez créer un de base de données MySQL avec un paramètre personnalisé et un nouveau groupe d'options. Pour plus d'informations sur les groupes de paramètres et d'options personnalisés, reportez-vous Groupes de paramètres pour (Amazon RDS)aux sections etUtilisation de groupes d'options.

> **Important**
>
> La création d'un AWS compte est gratuite. Cependant, en suivant ce didacticiel, les AWS ressources que vous utilisez peuvent vous coûter cher. Vous pouvez supprimer ces ressources après avoir terminé le didacticiel si elles ne sont plus nécessaires.

Pour créer un de base de données avec des configurations et des paramètres personnalisés, vous pouvez utiliser des paramètres personnalisés et de nouveaux groupes d'options. Les paramètres personnalisés et les nouveaux groupes d'options sont particulièrement utiles si vous travaillez avec plusieurs bases de données et souhaitez configurer leurs paramètres de manière uniforme.

En suivant ces étapes, vous apprendrez à :

- Comment utiliser Amazon RDS Amazon pour créer un de base de données MySQL avec un paramètre personnalisé et un nouveau groupe d'options.
- Comment utiliser des paramètres et des options spécifiques pour les instances de base de données MySQL et les de données.

Pour suivre ce didacticiel, effectuez les tâches suivantes :

1. Créez un groupe de paramètres de personnalisé avec le de paramètres MySQL `default_password_lifetime` et `disconnect_on_expired_password`.
2. Créez un nouveau groupe d'options avec la fonctionnalité d'options MySQL`MariaDB Audit Plugin`. Pour connaître les étapes de création d'un groupe d'options, voirUtilisation de groupes d'options.
3. Créez un de base de données MySQL avec le groupe de paramètres de personnalisé et le nouveau groupe d'options que vous avez créés.

Rubriques

- Prérequis
- Création d'un groupe de paramètres de
- Modifier la des paramètres dans votre groupe de paramètres personnalisé
- Création d'un nouveau groupe d'options Amazon RDS
- Ajoutez une option à votre nouveau groupe d'options
- Créez un de base de données MySQL avec un paramètre de personnaliséet un nouveau groupe d'options

Prérequis

Ce didacticiel nécessite que vous disposiez d'un Compte AWS et d'un utilisateur disposant d'un accès administratif. Si vous ne les avez pas déjà configurés, suivez les étapes décrites dans les sections suivantes :

- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Création d'un groupe de paramètres de

Le `default_password_lifetime` paramètre contrôle le nombre de jours avant l'expiration automatique du mot de passe du client. Le `disconnect_on_expired_password` paramètre contrôle si l'instance de base de données MySQL déconnecte le client lorsque le mot de passe expire. Pour plus d'informations sur les autres paramètres disponibles pour les de base de données MySQL, consultez Paramètres pour My SQL.

Pour créer un groupe de paramètres

1. Ouvrez la console Amazon RDS et choisissez Groupes de paramètres.
2. Pour les groupes de paramètres personnalisés, choisissez Créer un groupe de paramètres.
3. Définissez les détails du groupe de paramètres.
    1. Entrez un nom pour le groupe de paramètres.
    2. Entrez une description du groupe de paramètres.
    3. Pour le type de moteur, choisissez MySQL Community.

4. Pour la famille de groupes de paramètres, choisissez MySQL 8.0.
4. Sélectionnez Create (Créer).

Le nouveau groupe de paramètres de base de données apparaît sur la page Groupes de paramètres de la console Amazon RDS. Les étapes suivantes montrent comment modifier les valeurs des paramètres pour personnaliser votre groupe de paramètres.

Modifier la des paramètres dans votre groupe de paramètres personnalisé

Procédez comme suit pour modifier les des paramètres dans le groupe de paramètres que vous avez créé dansCréation d'un groupe de paramètres de .

Pour modifier les valeurs des paramètres de votre groupe de paramètres

1. Ouvrez la console Amazon RDS et choisissez Groupes de paramètres.
2. Pour les groupes de paramètres personnalisés, choisissez le nom du groupe de paramètres du de base de données que vous avez créé.
3. Choisissez Modifier.
4. Dans la zone de recherche des paramètres du filtre, recherchez le paramètre personnalisé`default_password_lifetime`.
5. Cochez la case à côté du paramètre et entrez une valeur correspondant au nombre de jours à définir pour ce paramètre de durée de vie du mot de passe.
6. Sélectionnez Save Changes (Enregistrer les modifications).
7. Répétez les mêmes étapes pour le paramètre `disconnect_on_expired_password`. Lorsque vous choisissez ce paramètre, vous êtes invité à sélectionner une valeur de 0 ou 1 dans le menu déroulant. Sélectionnez 1 pour vous déconnecter en cas de mot de passe expiré.

Le groupe de paramètres personnalisé peut désormais être associé à l'instance de base de données Amazon RDS for MySQL 8.0. Créez ensuite un nouveau groupe d'options pour votre instance de base de données.

Création d'un nouveau groupe d'options Amazon RDS

Créez un nouveau groupe d'options avec l'option MariaDB Audit Plugin. Ce plugin enregistre l'activité du serveur pour des raisons de sécurité et de conformité. Pour plus d'informations sur

les autres options personnalisées disponibles pour les instances de base de données MySQL, consultezOptions pour les instances My SQL DB.

Pour créer un groupe d'options

1. Ouvrez la console Amazon RDS et choisissez Groupes d'options.
2. Pour Groupes d'options, choisissez Créer un groupe.
3. Définissez les détails du groupe d'options.
   - Entrez un nom pour le groupe d'options.
   - Entrez une description du groupe d'options.
   - Pour Engine, sélectionnez mysql.
   - Pour la version principale du moteur, sélectionnez 8.0.
4. Sélectionnez Create (Créer).

Le nouveau groupe d'options apparaît sur la page Groupes d'options de la console Amazon RDS. Les étapes suivantes indiquent comment ajouter des options au groupe d'options.

Ajoutez une option à votre nouveau groupe d'options

Suivez les étapes ci-dessous pour ajouter une option au nouveau groupe d'options que vous avez créé dansCréation d'un nouveau groupe d'options Amazon RDS.

Pour ajouter une option à votre groupe d'options

1. Ouvrez la console Amazon RDS et choisissez Groupes d'options.
2. Pour les groupes d'options, sélectionnez le nom du groupe d'options que vous avez créé.
3. Sous Options, choisissez Ajouter une option.
4. Définissez les détails du groupe d'options.
   - Pour le nom de l'option, choisissez l'option Plugin d'audit MariaDB, MARIADB_AUDIT_PLUGIN.
   - Pour les paramètres des options, laissez toutes les options par défaut sélectionnées.
   - Pour Appliquer immédiatement, choisissez Oui.
5. Choisissez l'option Créer.

Le nouveau groupe d'options devrait désormais être disponible pour toutes les instances de base de données associées. Créez ensuite une instance de base de données MySQL avec le paramètre personnalisé et le nouveau groupe d'options.

Créez un de base de données MySQL avec un paramètre de personnaliséet un nouveau groupe d'options

Enfin, créez un de base de données MySQL avec le paramètre personnalisé et le nouveau groupe d'options que vous avez créés lors des étapes précédentes. Les étapes suivantes montrent comment créer le de base de données MySQL avec votre paramètre personnalisé et un nouveau groupe d'options.

Pour créer un de base de données avec un paramètre personnalisé et un nouveau groupe d'options

1. Ouvrez la console Amazon RDS et choisissez Databases.
2. Choisissez Créer une base de données.
3. Pour Choisir une méthode de création de base de données, choisissez Standard Create.
4. Pour les options du moteur, choisissez MySQL .
5. Pour la disponibilité et la durabilité, choisissez une instance de base de données unique. Cette étape est nécessaire pour prendre en charge un paramètre personnalisé ou un nouveau groupe d'options.
6. Sélectionnez Configuration supplémentaire.
   - Pour Nom de base de données initial, choisissez un nom pour le de base de données.
   - Dans le menu déroulant du groupe de paramètres de de base de données, sélectionnez le nom du groupe de paramètres de personnalisé que vous avez créé précédemment.
   - Dans la liste déroulante des groupes d'options, sélectionnez le nom du nouveau groupe d'options que vous avez créé précédemment.
7. Pour ce didacticiel, vous pouvez conserver les paramètres par défaut pour tous les autres paramètres de base de données ou les modifier selon vos besoins.
8. Choisissez Créer une base de données.

RDS crée un nouveau de base de données MySQL avec un groupe de paramètres personnalisé et un nouveau groupe d'options. Pour plus d'informations sur cette base de données, consultez la page Bases de données de la console Amazon RDS.

Dans ce didacticiel, vous avez configuré un de base de données MySQL avec des paramètres personnalisés à l'aide d'un paramètre personnalisé et d'un nouveau groupe d'options. Ce de base de données MySQL nouvellement créé gère la durée de vie du mot de passe utilisateur à l'aide du paramètre`default_password_lifetime`. Ce de base de données déconnecte également les utilisateurs qui se connectent avec un mot de passe expiré à l'aide du paramètre`disconnect_on_expired_password`. Vous pouvez également utiliser cette option `MariaDB Audit Plugin` pour suivre l'activité du serveur. Pour optimiser votre base de données, vous pouvez appliquer des paramètres supplémentaires à votre groupe de paramètres personnalisé et ajouter des options.

Une fois que vous avez fini de créer votre de base de données personnalisé, vous devez supprimer vos ressources pour éviter d'encourir des coûts indésirables. Pour supprimer un de base de données, suivez les instructions figurant dans Suppression d'une instance DB.

Gestion d'une RDS instance de base de données Amazon

Vous trouverez ci-dessous des instructions pour gérer et maintenir votre RDS instance de base de données Amazon.

Rubriques

- Arrêt temporaire d'une instance de bases de données Amazon RDS
- Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée
- Redémarrage d'une instance de base de données
- Connexion automatique d'une EC2 instance et d'une instance de base de données à
- Connexion automatique d'une fonction Lambda et d'une instance de base de données
- Modification d'une RDS instance de base de données Amazon
- Entretien d'une instance de base de données
- Mise à niveau de la version du moteur d'une instance de base de données
- Affectation d'un nouveau nom à une instance DB
- Utilisation des réplicas en lecture d'instance de base de données
- Marquage d' Amazon RDS
- Noms des ressources Amazon (ARNs) dans Amazon RDS
- Utilisation du stockage pour les RDS instances de base de données Amazon
- Suppression d'une instance DB
- Tutoriel : Gestion d'un environnement d'instance My SQL DB, du développement à la production

Arrêt temporaire d'une instance de bases de données Amazon RDS

Vous pouvez arrêter votre instance de base de données par intermittence pour des tests temporaires ou pour une activité de développement quotidienne, pendant un maximum de 7 jours consécutifs. Le cas d'utilisation le plus courant est l'optimisation des coûts.

Le délai d'arrêt de votre instance de base de données varie en fonction de facteurs tels que la classe d'instance, l'état du réseau, le type de moteur de base de données et l'état de la base de données. Le processus peut prendre plusieurs minutes. Le service doit effectuer les actions suivantes :

- Arrêtez les processus du moteur de base de données.
- Arrêtez les processus de la plateforme RDS.
- Détachez les volumes de stockage EBS associés à votre instance de base de données.
- Mettez fin à l' EC2 instance Amazon sous-jacente.

> **Warning**
>
> Le démarrage d'une instance de base de données nécessite la restauration de l'instance et peut prendre de quelques minutes à quelques heures. Par conséquent, si la disponibilité des instances est un problème, soyez prudent avant d'arrêter temporairement une instance de production. Pour de plus amples informations, veuillez consulter Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée.

Pour arrêter et démarrer votre instance de base de données au cours de la même opération, redémarrez-la. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Rubriques

- Cas d'utilisation pour arrêter votre instance de base de données
- Moteurs de base de données, classes d'instances et régions pris en charge
- Arrêt d'une instance de base de données dans un déploiement multi-AZ
- Comment fonctionne l'arrêt d'une instance de base de données
- Limites liées à l'arrêt de votre instance de base de données

- Considérations relatives au groupe d'options et de paramètres
- Considérations relatives aux adresses IP publiques
- Arrêt temporaire d'une instance de base de données : étapes de base

Cas d'utilisation pour arrêter votre instance de base de données

L'arrêt et le démarrage d'une instance de base de données sont plus rapides que la création d'un instantané de base de données, la suppression de votre instance de base de données, puis la restauration du cliché lorsque vous souhaitez accéder à l'instance. Les cas d'utilisation courants pour arrêter une instance sont les suivants :

- Optimisation des coûts : pour les bases de données hors production, vous pouvez arrêter temporairement votre instance de base de données Amazon RDS pour économiser de l'argent. Lorsque l'instance est arrêtée, les heures d'instance de base de données ne vous sont pas facturées.

  > **Important**
  >
  > Pendant que votre instance de base de données est arrêtée, le stockage provisionné vous est facturé (y compris les IOPS provisionnés). Le stockage de sauvegarde vous est également facturé, notamment le stockage des instantanés manuels et des sauvegardes automatisées dans la fenêtre de conservation que vous avez spécifiée. Les heures de l'instance de base de données ne vous sont toutefois pas facturées. Pour plus d'informations, consultez la section Facturation FAQs.

- Développement quotidien — Si vous gérez une instance de base de données à des fins de développement, vous pouvez démarrer l'instance lorsque cela est nécessaire, puis arrêter l'instance lorsqu'elle n'est pas nécessaire.
- Tests : vous pourriez avoir besoin d'une instance de base de données temporaire pour tester les procédures de sauvegarde et de restauration, les migrations, les mises à niveau d'applications ou les activités connexes. Dans ces cas d'utilisation, vous pouvez arrêter l'instance de base de données lorsqu'elle n'est pas nécessaire.
- Formation : si vous suivez une formation sur RDS, vous devrez peut-être démarrer des instances de base de données pendant la session de formation, puis les arrêter par la suite.

Moteurs de base de données, classes d'instances et régions pris en charge

Vous pouvez arrêter et démarrer les instances de base de données Amazon RDS qui exécutent les moteurs de base de données suivants :

- Db2
- MariaDB
- Microsoft SQL Server, y compris RDS Custom pour SQL Server
- MySQL
- Oracle
- PostgreSQL

L'arrêt et le démarrage d'une instance de bases de données sont pris en charge pour tous les classes d'instance de bases de données et dans toutes les régions AWS .

Arrêt d'une instance de base de données dans un déploiement multi-AZ

Vous pouvez arrêter et démarrer une instance de base de données dans un déploiement multi-AZ. Prenez en compte les limitations suivantes :

- Vous ne pouvez créer un déploiement multi-AZ que si votre moteur de base de données le prend en charge. Pour plus d'informations sur le support du moteur, consultezRégions et moteurs de base de données pris en charge pour les clusters de bases de données multi-AZ dans Amazon RDS.
- RDS pour SQL Server ne prend pas en charge l'arrêt d'une instance de base de données dans un déploiement multi-AZ. Pour de plus amples informations, veuillez consulter Notes, limitations et recommandations concernant le déploiement multi-AZ de Microsoft SQL Server.
- L'arrêt d'une instance de base de données peut prendre un certain temps. Si vous disposez d'au moins une sauvegarde après un précédent basculement, vous pouvez accélérer l'opération d'arrêt en effectuant un redémarrage avec opération de basculement. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Comment fonctionne l'arrêt d'une instance de base de données

L'opération d'arrêt se déroule selon les étapes suivantes :

1. L'instance de base de données lance le processus d'arrêt normal.

   Le statut de l'instance de base de données devient `stopping`.
2. L'instance cesse de fonctionner, jusqu'à un maximum de 7 jours consécutifs.

   Le statut de l'instance de base de données devient `stopped`.

Caractéristiques d'une instance de base de données arrêtée

Lorsqu'elle est arrêtée, votre instance de base de données présente les caractéristiques suivantes :

- Votre instance de base de données arrêtée conserve les éléments suivants :
  - ID d’instance
  - Point de terminaison DNS (Domain Name Server)
  - Groupe de paramètres
  - Groupe de sécurité
  - Groupe d'options
  - Journaux de transactions Amazon S3 (nécessaires pour une point-in-time restauration)

  Lorsque vous démarrez une instance de base de données, sa configuration est la même que lorsque vous l'avez arrêtée.
- Tous les volumes de stockage restent attachés à l'instance de bases de données et leurs données sont conservées. RDS supprime toutes les données stockées dans la RAM de l'instance de base de données.

  Pendant que votre instance de base de données est arrêtée, le stockage provisionné vous est facturé (y compris les IOPS provisionnés). Le stockage de sauvegarde vous est également facturé, notamment le stockage des instantanés manuels et des sauvegardes automatisées dans la fenêtre de conservation que vous avez spécifiée.
- RDS supprime les actions en attente, y compris les mises à jour de maintenance planifiées, à l'exception des actions en attente pour le groupe d'options ou le groupe de paramètres de base de données de l'instance de base de données.

> **Note**
>
> Parfois, une instance de base de données RDS for PostgreSQL ne s'arrête pas correctement. Si cela se produit, vous constatez que l'instance passe par un processus de récupération lorsque vous la redémarrez ultérieurement. Il s'agit d'un comportement attendu du moteur de base de données, destiné à protéger l'intégrité de la base de données. Certaines statistiques et compteurs basés sur la mémoire ne conservent pas l'historique et sont réinitialisés après le redémarrage, afin de capturer la charge de travail opérationnelle à l'avenir.

Redémarrage automatique d'une instance de base de données arrêtée

Si vous ne démarrez pas manuellement votre instance de base de données après sept jours d'arrêt, RDS démarre automatiquement votre instance de base de données pour vous. Ainsi, votre instance n'est pas en retard par rapport aux mises à jour de maintenance requises. Pour apprendre à arrêter et démarrer votre instance selon un calendrier, consultez Comment puis-je utiliser Step Functions pour arrêter une instance Amazon RDS pendant plus de 7 jours ?.

Limites liées à l'arrêt de votre instance de base de données

Certaines limites de l'opération d'arrêt sont les suivantes :

- Vous ne pouvez pas arrêter une instance de base de données qui possède un réplica en lecture ou qui est un réplica en lecture.
- Vous ne pouvez pas modifier une instance de bases de données arrêtée.
- Vous ne pouvez pas supprimer un groupe d'options associé à une instance de bases de données arrêtée.
- Vous ne pouvez pas supprimer un groupe de paramètres de base de données associé à une instance de bases de données arrêtée.
- Dans un déploiement multi-AZ, notez les limites suivantes :
  - Vous ne pouvez pas arrêter une instance de base de données RDS pour SQL Server.
  - Les zones de disponibilité principales et secondaires peuvent être commutées après le démarrage de l'instance de base de données.

Des limites supplémentaires s'appliquent à RDS Custom for SQL Server. Pour de plus amples informations, veuillez consulter Démarrage et arrêt d'une instance de base de données RDS personnalisée pour SQL serveur.

Considérations relatives au groupe d'options et de paramètres

Vous ne pouvez pas supprimer les options persistantes (y compris les options permanentes) d'un groupe d'options si des instances de bases de données sont associées à ce groupe d'options. Cette fonctionnalité vaut également pour les instances de bases de données ayant l'état `stopping`, `stopped`, ou `starting`.

Vous ne pouvez pas modifier le groupe d'options ou le groupe de paramètres de base de données associé à une instance de bases de données arrêtée. Toutefois, la modification ne sera appliquée qu'au prochain démarrage de l'instance de bases de données. Si vous avez choisi d'appliquer les modifications immédiatement, elles sont appliquées au démarrage de l'instance de bases de données. Dans le cas contraire, la modification est appliquée au cours de la fenêtre de maintenance suivante, après le démarrage de l'instance de bases de données.

Considérations relatives aux adresses IP publiques

Lorsque vous arrêtez une instance de bases de données, elle conserve son point de terminaison DNS. Si vous arrêtez une instance de base de données qui a une adresse IP publique, Amazon RDS libère son adresse IP publique. Lorsque l'instance de base de données est redémarrée, elle a une adresse IP publique différente.

> **Note**
>
> Vous devez toujours vous connecter à une instance de bases de données à l'aide du point de terminaison DNS, et non pas de l'adresse IP.

Arrêt temporaire d'une instance de base de données : étapes de base

Vous pouvez arrêter une base de données à l'aide de l'API AWS Management Console AWS CLI, de, ou de l'API RDS.

Console

Pour arrêter une instance de bases de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous voulez arrêter.
3. Pour Actions, choisissez Stop temporarily (Arrêter temporairement).
4. Dans la fenêtre Stop DB instance temporarily (Arrêter temporairement l'instance de base de données), sélectionnez l'accusé de réception indiquant que l'instance de base de données redémarrera automatiquement au bout de 7 jours.
5. (Facultatif) Sélectionnez Save the DB instance in a snapshot (Enregistrer l'instance de base de données dans un instantané) et saisissez le nom de l'instantané pour Snapshot name (Nom d'instantané). Choisissez cette option si vous souhaitez créer un instantané de l'instance de base de données avant de l'arrêter.
6. Choisissez Stop temporarily (Arrêter temporairement) pour arrêter l'instance de base de données ou choisissez Cancel (Annuler) pour annuler l'opération.

AWS CLI

Pour arrêter une instance de base de données à l'aide de AWS CLI, appelez la stop-db-instancecommande avec l'option suivante :

- `--db-instance-identifier` – le nom de l'instance de base de données.

Example

```
aws rds stop-db-instance --db-instance-identifier mydbinstance
```

API RDS

Pour arrêter une instance de base de données à l'aide de l'API Amazon RDS, appelez l'DBInstanceopération Stop avec le paramètre suivant :

- `DBInstanceIdentifier` – le nom de l'instance de base de données.

Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée

Cette rubrique explique comment démarrer une instance Amazon RDS DB précédemment arrêtée, en décrivant les étapes nécessaires et les principaux points à prendre en compte pour reprendre l'utilisation de la base de données.

Vous pouvez arrêter temporairement votre instance Amazon RDS DB pour économiser de l'argent. Une fois votre instance de bases de données arrêtée, vous pouvez la redémarrer et recommencer à l'utiliser. Pour plus d'informations sur l'arrêt d'une instance de base de données, consultezArrêt temporaire d'une instance de bases de données Amazon RDS.

Lorsque vous démarrez une instance de base de données que vous avez précédemment arrêtée, l'instance de base de données conserve des informations telles que les suivantes :

- ID d’instance
- Point de terminaison du serveur de noms de domaine (DNS)
- Groupe de paramètres de base de données
- VPCgroupe de sécurité
- Groupe d'options DB

Amazon RDS facture par tranches d'une seconde pour les instances de base de données et le stockage associé. Une charge minimale de 10 minutes est facturée au démarrage d'une instance.

Pour démarrer l'instance, le RDS service Amazon exécute des actions telles que les suivantes :

- Provisionnement de l'instance Amazon EC2 sous-jacente
- Démarrage des RDS processus
- Démarrage des processus du moteur de base de données
- Attacher les volumes EBS de stockage
- Activation de Performance Insights s'il a déjà été activé
- Restauration de l'instance de base de données (la restauration a lieu même après un arrêt normal)

L'heure de démarrage de votre instance de base de données varie en fonction de facteurs tels que la classe d'instance, l'état du réseau, le type de moteur de base de données, la taille de la base

de données et l'état de la base de données au moment de l'arrêt de l'instance. Le processus de démarrage peut prendre de quelques minutes à quelques heures. Nous vous recommandons de tenir compte de la variabilité du temps de démarrage lors de la création de votre plan de disponibilité.

Console

Pour démarrer une instance de bases de données

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous voulez démarrer.
3. Pour Actions, choisissez Start (Démarrer).

AWS CLI

Pour démarrer une instance de base de données à l'aide du AWS CLI, appelez la start-db-instancecommande avec l'option suivante :

- `--db-instance-identifier` – Nom de l'instance de base de données.

Example

```
aws rds start-db-instance --db-instance-identifier mydbinstance
```

RDS API

Pour démarrer une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'tartDBInstanceopération S avec le paramètre suivant :

- `DBInstanceIdentifier` – Nom de l'instance de base de données.

Redémarrage d'une instance de base de données

Vous pouvez arrêter et démarrer le service de base de données sur votre RDS instance de base de données en une seule opération, appelée redémarrage. Le redémarrage peut être nécessaire pour appliquer des modifications de configuration, résoudre des problèmes mineurs ou résoudre des problèmes réseau sans avoir à effectuer un redémarrage complet ou une migration de votre base de données.

> **Note**
>
> Cette rubrique s'applique uniquement au redémarrage d'une instance de base de données mono-AZ ou multi-AZ. Pour obtenir des instructions sur le redémarrage d'un cluster de base de données multi-AZ, consultezthe section called “Redémarrage d'un cluster de base de données multi-AZ”.

Rubriques

- Cas d'utilisation pour le redémarrage d'une instance de base de données cluster de base de données
- Comment fonctionne le redémarrage d'une instance de base de données base de données
- Comment fonctionne le redémarrage d'une instance de base de données dans un déploiement multi-AZ
- Considérations relatives au redémarrage d'une instance de base de données cluster de base de données
- Conditions préalables au redémarrage d'une instance de base de données un cluster de base de données
- Redémarrage d'une instance de base de données de base

Cas d'utilisation pour le redémarrage d'une instance de base de données cluster de base de données

Généralement, vous redémarrez votre instance de base de données pour des raisons de maintenance afin que vos modifications prennent effet. Les cas d'utilisation suivants sont courants :

- Association d'un nouveau groupe de paramètres de base de données : lorsque vous associez un nouveau groupe de paramètres de base de données à une instance de base de données, RDS applique les paramètres statiques et dynamiques modifiés uniquement après le redémarrage de l'instance de base de données. Toutefois, si vous modifiez les paramètres dynamiques du groupe de paramètres de base de données après l'avoir associé à l'instance de base de données, ces modifications sont appliquées immédiatement sans redémarrage. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).
- Appliquer une modification à un paramètre statique dans un groupe de paramètres de base de données existant : lorsque vous modifiez un paramètre statique et que vous enregistrez le groupe de paramètres de base de données, le statut des instances de base de données associées à ce groupe de paramètres dans la console passe à pending-reboot. La modification des paramètres ne prend effet qu'après le redémarrage des instances de base de données associées. Lorsque vous modifiez un paramètre dynamique dans un groupe de paramètres existant, la modification prend effet immédiatement par défaut, sans nécessiter de redémarrage.

  > **Note**
  >
  > L'état en attente de redémarrage n'entraîne pas de redémarrage automatique lors de la fenêtre de maintenance suivante. Pour appliquer les dernières modifications de paramètres à votre instance de base de données, redémarrez-la manuellement. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

- Résolution des problèmes : vous pouvez rencontrer des problèmes de performance ou d'autres problèmes opérationnels nécessitant un redémarrage. Par exemple, il se peut que votre instance de base de données ne réponde pas.

Comment fonctionne le redémarrage d'une instance de base de données base de données

Lorsque Amazon RDS redémarre votre instance de base de données, il exécute les tâches séquentielles suivantes :

1. Arrête le service de base de données sur votre instance de base de données
2. Démarre le service de base de données sur votre instance de base de données

Le processus de redémarrage entraîne une brève interruption de service. Pendant cette panne, l'état de l'instance de base de données est en train de redémarrer. Une panne se produit à la fois pour un déploiement mono-AZ et un déploiement multi-AZ d'instance de base de données, même lorsque vous redémarrez avec un basculement.

Comment fonctionne le redémarrage d'une instance de base de données dans un déploiement multi-AZ

Si l'RDSinstance de base de données Amazon est dans un déploiement multi-AZ, vous pouvez redémarrer avec un basculement. Cette opération est utile pour simuler la défaillance d'une instance de base de données ou pour restaurer les opérations dans la zone de disponibilité d'origine après un basculement.

Lors du redémarrage avec basculement, Amazon RDS effectue les opérations suivantes

- Interrompt brusquement la base de données. L'instance de base de données et ses sessions clientes peuvent ne pas avoir le temps de s'arrêter correctement.

  > ⚠ Warning
  >
  > Pour éviter toute perte de données, nous vous recommandons d'arrêter les transactions sur votre instance de base de données avant de redémarrer avec un basculement.

- Effectue une restauration après incident de la base de données si nécessaire.
- Bascule automatiquement vers une réplique de secours dans une autre zone de zone de disponibilité. La modification de l'AZ peut ne pas être reflétée dans le AWS Management Console et dans les appels vers RDS API le AWS CLI et pendant plusieurs minutes.
- Met à jour l'DNSenregistrement de l'instance de base de données afin qu'il pointe vers l'instance de base de données de secours. Par conséquent, vous devez nettoyer et rétablir toutes les connexions existantes à votre instance de bases de données. Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
- Crée un RDS événement Amazon après le redémarrage.

Activé RDS pour Microsoft SQL Server, le basculement redémarre uniquement l'instance de base de données principale. Après le basculement, l'instance de base de données principale devient la nouvelle instance de base de données secondaire. Les paramètres peuvent ne pas être mis à jour pour les instances multi-AZ. Pour le redémarrage sans basculement, les instances de

base de données principale et secondaire redémarrent, et les paramètres sont mis à jour après le redémarrage. Si l'instance de base de données ne répond pas, nous vous recommandons de procéder à un redémarrage sans basculement.

Considérations relatives au redémarrage d'une instance de base de données cluster de base de données

Avant de redémarrer votre instance, tenez compte des points suivants :

- Pour une instance de base de données avec des réplicas en lecture, vous pouvez redémarrer indépendamment l'instance de base de données source et ses réplicas en lecture. Une fois le redémarrage terminé, la réplication reprend automatiquement.
- Le temps de redémarrage dépend du processus de restauration après incident, de l'activité de la base de données au moment du redémarrage et du comportement de votre moteur de base de données spécifique. Pour améliorer le temps de redémarrage, nous vous recommandons de réduire au maximum l'activité de la base de données pendant le redémarrage. Cette technique réduit l'activité d'annulation pour les transactions en transit.

Conditions préalables au redémarrage d'une instance de base de données un cluster de base de données

Assurez-vous de remplir les conditions préalables suivantes :

- Votre instance de base de données doit être dans l'état `available`. Votre base de données peut être indisponible pour plusieurs raisons, telles qu'une sauvegarde en cours, une modification précédemment demandée ou une opération de fenêtre de maintenance.
- Si vous forcez un basculement vers une autre zone de disponibilité, votre instance de base de données doit être configurée pour le mode multi-AZ.
- Si vous forcez un basculement vers une autre zone de disponibilité, nous vous recommandons d'arrêter d'abord les transactions sur votre instance de base de données afin d'éviter toute perte de données.

Redémarrage d'une instance de base de données de base

Vous pouvez redémarrer votre instance de base de données à l'aide de AWS Management Console AWS CLI, ou RDSAPI.

Console

Pour redémarrer une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données à redémarrer.
3. Pour Actions, choisissez Redémarrer.

   La page Redémarrer l'instance de base de données s'affiche.
4. (Facultatif) Sélectionnez Redémarrer avec basculement ? pour imposer un basculement d'une zone de disponibilité vers une autre.
5. Choisissez Redémarrer pour redémarrer votre instance de bases de données.

   Sinon, choisissez Annuler.

AWS CLI

Pour redémarrer une instance de base de données à l'aide de AWS CLI, appelez la `reboot-db-instance`commande.

Example Redémarrage simple

Dans Linux, macOS, ou Unix:

```
aws rds reboot-db-instance \
    --db-instance-identifier mydbinstance
```

Dans Windows:

```
aws rds reboot-db-instance ^
    --db-instance-identifier mydbinstance
```

Example Redémarrer avec basculement

Pour forcer un basculement d'un AZ à l'autre dans un cluster de base de données multi-AZ, utilisez le `--force-failover` paramètre.

Dans Linux, macOS, ou Unix:

```
aws rds reboot-db-instance \
    --db-instance-identifier mydbinstance \
    --force-failover
```

Dans Windows:

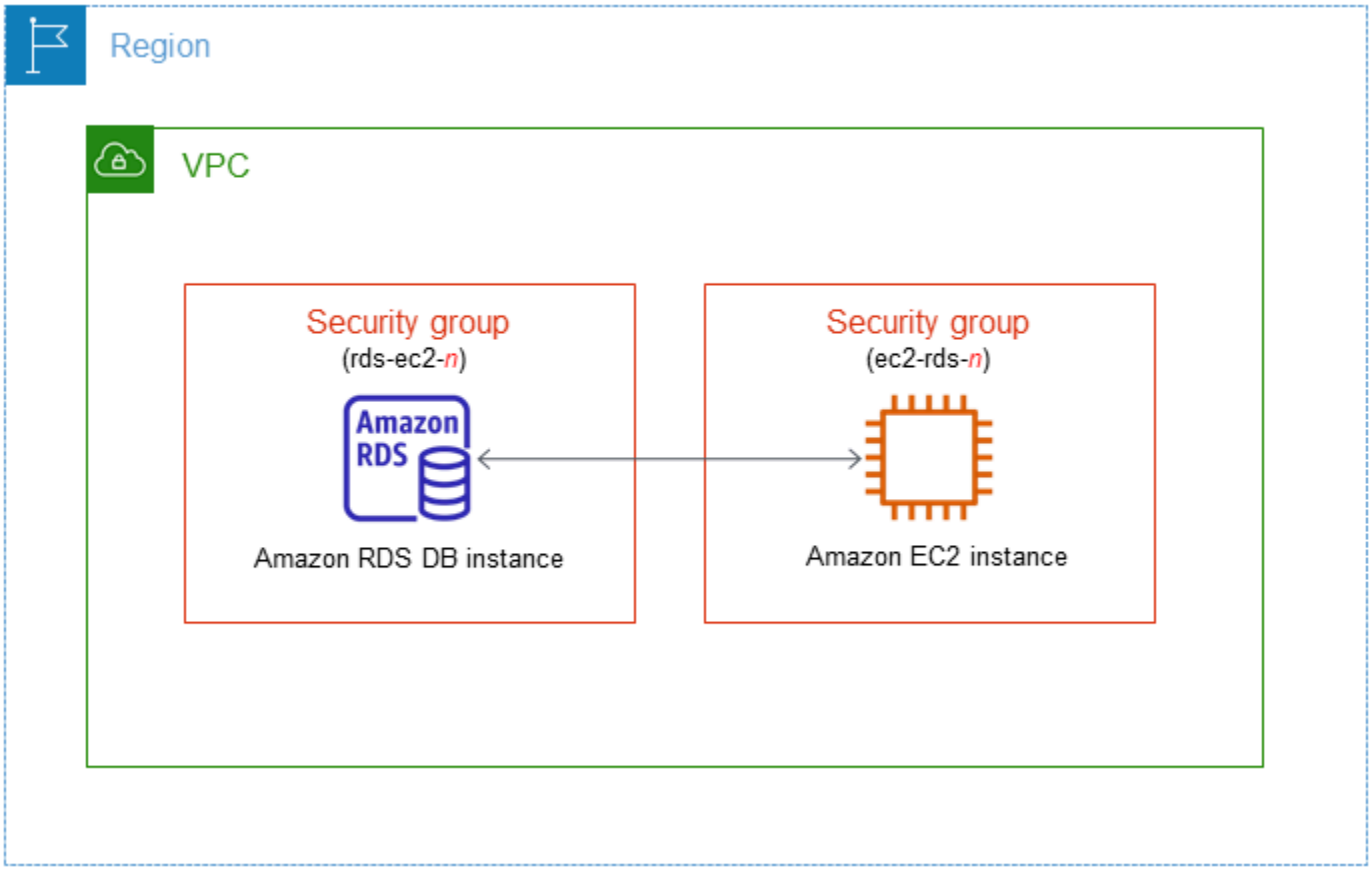
```
aws rds reboot-db-instance ^
    --db-instance-identifier mydbinstance ^
    --force-failover
```

RDS API

Pour redémarrer une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'RebootDBInstanceopération.

Connexion automatique d'une EC2 instance et d'une instance de base de données à

Vous pouvez utiliser la RDS console Amazon pour simplifier la configuration d'une connexion entre une instance Amazon Elastic Compute Cloud (AmazonEC2) et une instance de base de données, . Souvent, le de base de données se trouve dans un sous-réseau privé et votre EC2 instance se trouve dans un sous-réseau public au sein d'un. VPC Vous pouvez utiliser un SQL client sur votre EC2 instance pour vous connecter au de base de données. L'EC2instance peut également exécuter des serveurs Web ou des applications qui accèdent à votre de base de données privée. Pour obtenir des instructions sur la configuration d'une connexion entre une EC2 instance et un cluster de base de données multi-AZ, consultezthe section called “Connexion d'une instance EC2 et d'un cluster de base de données multi-AZ”.



Si vous souhaitez vous connecter à une EC2 instance qui ne se trouve pas dans la même instance VPC que le cluster , consultez les scénarios dansScénarios d'accès à un d'instances de base de données dans un VPC.

Rubriques

- Vue d'ensemble de la connectivité automatique avec une EC2 instance
- Connexion automatique d'une EC2 instance et d'un
- Affichage des ressources de calcul connectées
- Connexion à une instance de base de données qui exécute un moteur de base de données spécifique

Vue d'ensemble de la connectivité automatique avec une EC2 instance

Lorsque vous configurez une connexion entre une EC2 instance et un , Amazon RDSautomatically configure le groupe de VPC sécurité pour votre EC2 instance et pour votre de base de RDSdonnées de base de données.

Les conditions requises pour connecter une EC2 instance à un sont les suivantes :

- L'EC2instance doit exister au même endroit VPC que le de base de données.

  Si aucune EC2 instance n'existe dans la même instanceVPC, la console fournit un lien pour en créer une.
- L'utilisateur qui configure la connectivité doit être autorisé à effectuer les EC2 opérations Amazon suivantes :
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:AuthorizeSecurityGroupIngress`
  - `ec2:CreateSecurityGroup`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeSecurityGroups`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

Si l'instance de base de données et l'EC2instance se trouvent dans des zones de disponibilité différentes, votre compte peut entraîner des coûts liés aux zones de disponibilité croisée.

Lorsque vous configurez une connexion à une EC2 instance, Amazon RDS agit conformément à la configuration actuelle des groupes de sécurité associés au de RDSbase de données et à l'EC2instance, comme décrit dans le tableau suivant.

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle EC2 du groupe de sécurité	RDSaction
Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle `rds-ec2-`*n* (où se *n* trouve un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante dont le groupe de VPC sécurité de l'EC2instance est la source.	Un ou plusieurs groupes de sécurité sont associés à l'EC2instance dont le nom correspond au modèle `ec2-rds-`*n* (où *n* figure un chiffre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité n'a qu'une seule règle sortante avec le groupe de VPC sécurité du de RDSbase de données comme source.	RDSne prend aucune mesure. Une connexion a déjà été configurée automatiquement entre l'EC2instance et le de base de données. Comme une connexion existe déjà entre l'EC2instance et la RDS base de données, les groupes de sécurité ne sont pas modifiés.
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. • Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. Amazon ne RDS peut toutefois utiliser aucun de ces groupes de sécurité pour la connexion avec l'EC2instance. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à l'EC2inst ance dont le nom correspon d au modèle`ec2-rds-`*n*. • Un ou plusieurs groupes de sécurité sont associés à l'EC2instance dont le nom correspond au modèle`ec2-rds-`*n*. Cependant, Amazon ne RDS peut utiliser aucun de ces groupes de sécurité pour la connexion avec le de base de données. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle sortante avec le groupe	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle EC2 du groupe de sécurité	RDSaction
pas de règle entrante avec le groupe de VPC sécurité de l'EC2instance comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié. Des exemples de modifications incluent l'ajout d'une règle ou la modification du port d'une règle existante.	de VPC sécurité du de RDSbase de données comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	
Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante dont le groupe de VPC sécurité de l'EC2instance est la source.	Un ou plusieurs groupes de sécurité sont associés à l'EC2instance dont le nom correspond au modèle`ec2-rds-`*n*. Cependant, Amazon ne RDS peut utiliser aucun de ces groupes de sécurité pour la connexion avec le de base de données. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle sortante avec le groupe de VPC sécurité du de RDSbase de données comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle EC2 du groupe de sécurité	RDSaction
Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante dont le groupe de VPC sécurité de l'EC2instance est la source.	Un groupe EC2 de sécurité valide existe pour la connexion , mais il n'est pas associé à l'EC2instance. Le nom de ce groupe de sécurité correspond au modèle `ec2-rds-`*n*. Il n'a pas été modifié. Il n'a qu'une seule règle sortante avec le groupe de VPC sécurité du de RDSbase de données comme source.	RDS action: associate EC2 security group

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle EC2 du groupe de sécurité	RDSaction
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. • Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle`rds-ec2-`*n*. Amazon ne RDS peut toutefois utiliser aucun de ces groupes de sécurité pour la connexion avec l'EC2instance. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle entrante avec le groupe de VPC sécurité de l'EC2instance comme source. Amazon RDS ne peut pas non plus utiliser le groupe de sécurité qui a été modifié.	Un ou plusieurs groupes de sécurité sont associés à l'EC2instance dont le nom correspond au modèle`ec2-rds-`*n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité n'a qu'une seule règle sortante avec le groupe de VPC sécurité du de RDSbase de données comme source.	RDS action: create new security groups

RDSaction : créer de nouveaux groupes de sécurité

Amazon RDS prend les mesures suivantes :

- Crée un nouveau groupe de sécurité qui correspond au modèle `rds-ec2-`*n*. Ce groupe de sécurité possède une règle entrante dont le groupe de VPC sécurité de l'EC2instance est la

source. Ce groupe de sécurité est associé au de RDSbase de données et permet à l'EC2instance d'accéder au de RDSbases de données de base de données.

- Crée un nouveau groupe de sécurité qui correspond au modèle `ec2-rds-`*n*. Ce groupe de sécurité dispose d'une règle sortante avec le groupe de VPC sécurité du de RDSbase de données comme cible. Ce groupe de sécurité est associé à l'EC2instance et permet à celle-ci d'envoyer du EC2 trafic vers le de base de RDSdonnées de base de données.

RDSaction : associer un groupe EC2 de sécurité

Amazon RDS associe le groupe de EC2 sécurité existant valide à l'EC2instance. Ce groupe de sécurité permet à l'EC2instance d'envoyer du trafic vers le de base de données.

Connexion automatique d'une EC2 instance et d'un

Avant de configurer une connexion entre une EC2 instance et un , assurez-vous de répondre aux exigences décrites dansVue d'ensemble de la connectivité automatique avec une EC2 instance.

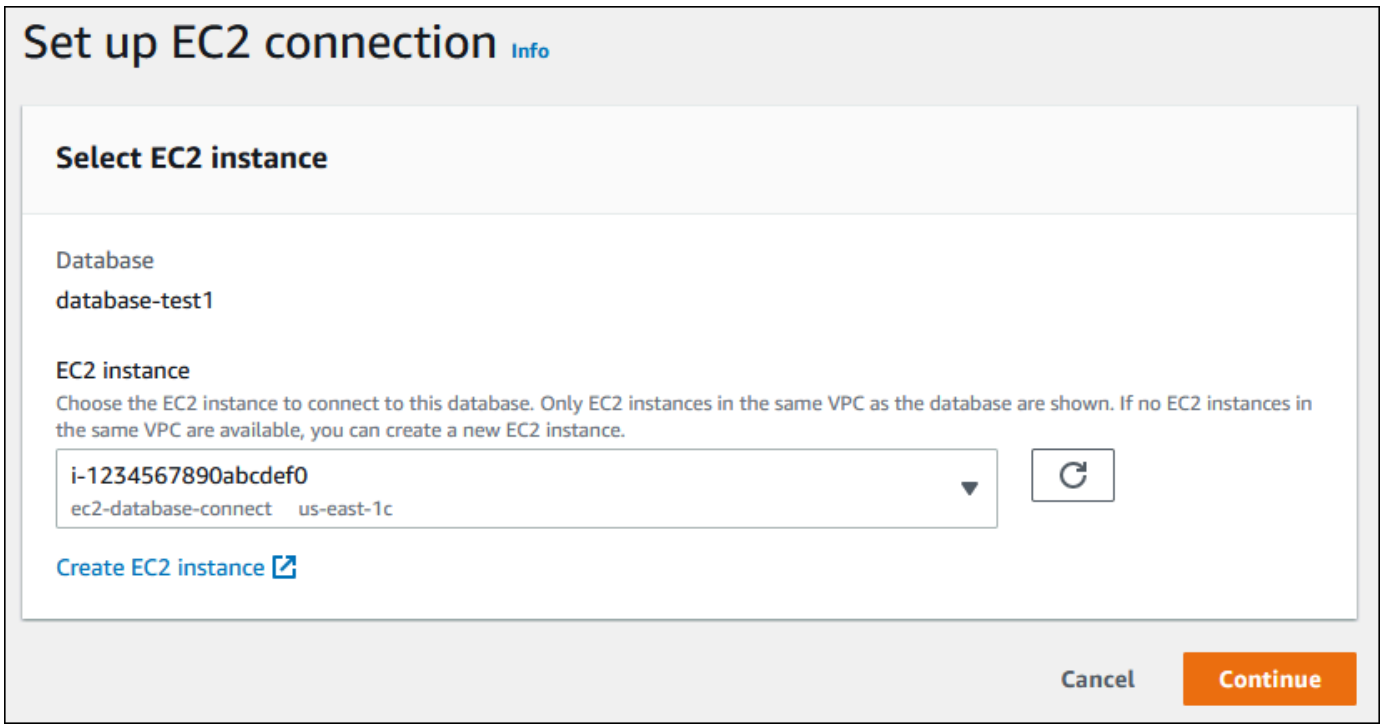
Si vous modifiez les groupes de sécurité après avoir configuré la connectivité, ces modifications peuvent affecter la connexion entre l'EC2instance et le .

> **Note**
>
> Vous ne pouvez configurer automatiquement une connexion entre une EC2 instance et un qu'à l'aide du AWS Management Console. Vous ne pouvez pas configurer une connexion automatiquement avec le AWS CLI ou RDSAPI.

Pour connecter automatiquement une EC2 instance et un

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez le de bases de données.
3. Dans Actions, choisissez Configurer la EC2 connexion.

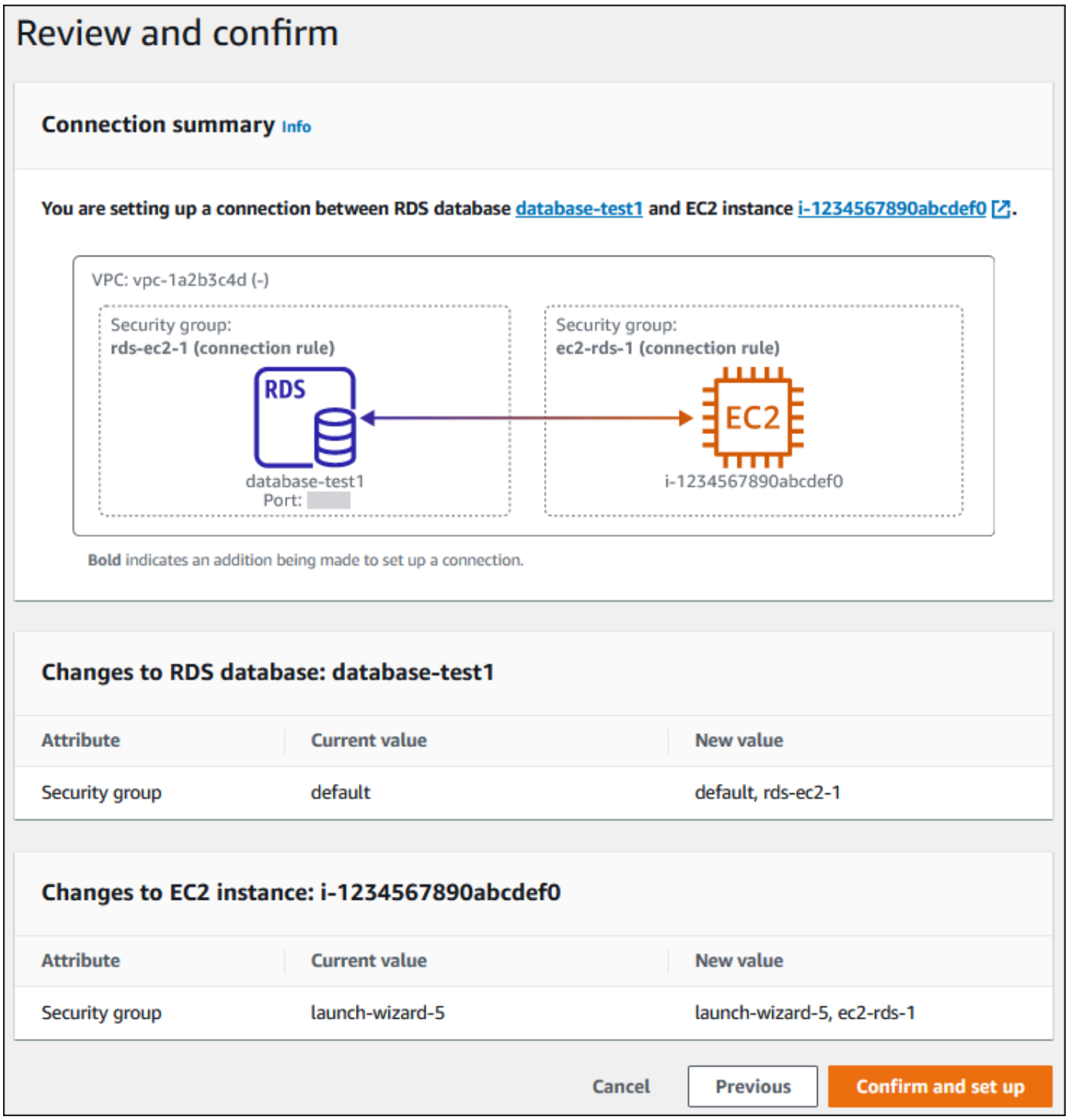
   La page Configurer EC2 la connexion apparaît.
4. Sur la page Configurer la EC2 connexion, choisissez l'EC2instance.



Si aucune EC2 instance n'existe dans la même instanceVPC, choisissez Create EC2 instance pour en créer une. Dans ce cas, assurez-vous que la nouvelle EC2 instance se trouve dans le même de base de données VPC que celui RDSde la base de données.

5. Choisissez Continuer.

   La page Review and confirm (Vérifier et confirmer) s'affiche.



6. Sur la page Vérifier et confirmer, passez en revue les modifications qui RDS seront apportées pour configurer la connectivité avec l'EC2instance.

   Si les modifications sont correctes, choisissez Confirmer et configurer.

   Si les modifications ne sont pas correctes, choisissez Previous (Précédent) ou Cancel (Annuler).

Affichage des ressources de calcul connectées

Vous pouvez utiliser le AWS Management Console pour afficher les ressources de calcul connectées à un . Les ressources affichées comprennent les connexions de ressources de calcul qui ont été configurées automatiquement. Vous pouvez configurer automatiquement la connectivité avec les ressources de calcul de la manière suivante :

- Vous pouvez sélectionner la ressource de calcul lorsque vous créez la base de données.

  Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS et Création d'un cluster de base de données multi-AZ pour Amazon RDS.
- Vous pouvez configurer la connectivité entre une base de données existante et une ressource de calcul.

  Pour de plus amples informations, veuillez consulter Connexion automatique d'une EC2 instance et d'un .

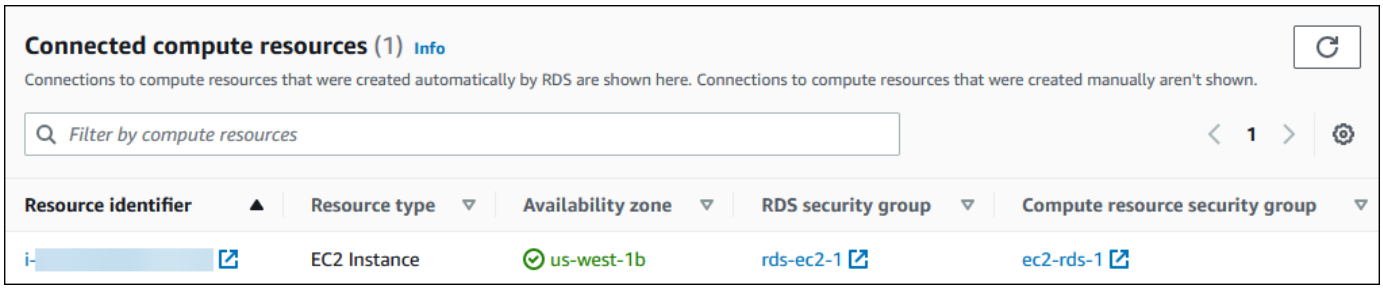
Les ressources de calcul répertoriées n'incluent pas celles qui ont été connectées manuellement à la base de données. Par exemple, vous pouvez autoriser une ressource de calcul à accéder manuellement à une base de données en ajoutant une règle au groupe VPC de sécurité associé à la base de données.

Pour qu'une ressource de calcul soit répertoriée, les conditions suivantes doivent s'appliquer :

- Le nom du groupe de sécurité associé à la ressource de calcul correspond au modèle `ec2-rds-`*n* (où *n* est un nombre).
- Le groupe de sécurité associé à la ressource de calcul possède une règle sortante dont la plage de ports est définie sur le port utilisé par le de RDSbase de données de base de données.
- Le groupe de sécurité associé à la ressource de calcul possède une règle sortante, la source étant définie sur un groupe de sécurité associé au de bases de RDSdonnées de base de données.
- Le nom du groupe de sécurité associé au de base de données correspond au modèle `rds-ec2-`*n* (où *n* figure un nombre).
- Le groupe de sécurité associé au de RDSbase de données possède une règle entrante dont la plage de ports est définie sur le port utilisé par le de base de RDSdonnées de base de données.
- Le groupe de sécurité associé au de base de données possède une règle entrante dont la source est définie sur un groupe de sécurité associé à la ressource de calcul.

Pour afficher les ressources de calcul connectées à un

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis le nom du de RDSbases de données.
3. Dans l'onglet Connectivity & security (Connectivité et sécurité), affichez les ressources de calcul dans Connected compute resources (Ressources de calcul connectées).



Connexion à une instance de base de données qui exécute un moteur de base de données spécifique

Pour plus d'informations sur la connexion à une instance de base de données qui exécute un moteur de base de données spécifique, suivez les instructions relatives à votre moteur de base de données :

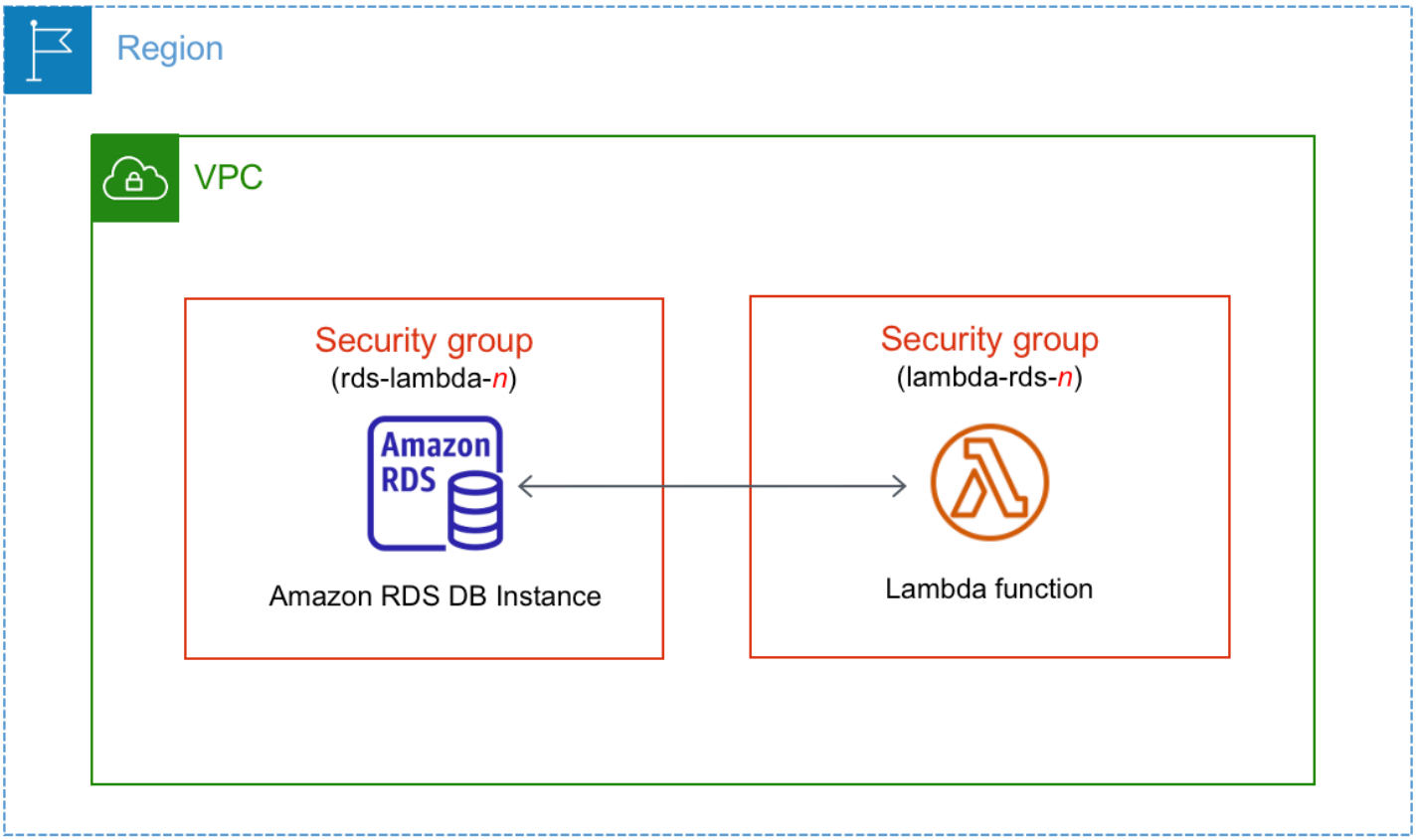
- Connexion à votre instance de base de données MariaDB
- Connexion à votre instance de base de données Microsoft SQL Server
- Connexion à votre instance de base de données MySQL
- Connexion à votre instance de base de données Oracle
- Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL

Connexion automatique d'une fonction Lambda et d'une instance de base de données

Vous pouvez utiliser la RDS console Amazon pour simplifier la configuration d'une connexion entre une fonction Lambda et une instance de base de données (un cluster de base de données). Souvent, le de base de données se trouve dans un sous-réseau privé au sein d'unVPC. La fonction Lambda peut être utilisée par les applications pour accéder à votre instance de base de données privée.

Pour obtenir des instructions sur la configuration d'une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ, consultez the section called “Connexion d'une fonction Lambda et d'un cluster de bases de données multi-AZ”.

L'image suivante montre une connexion directe entre votre instance de base de données et votre fonction Lambda.

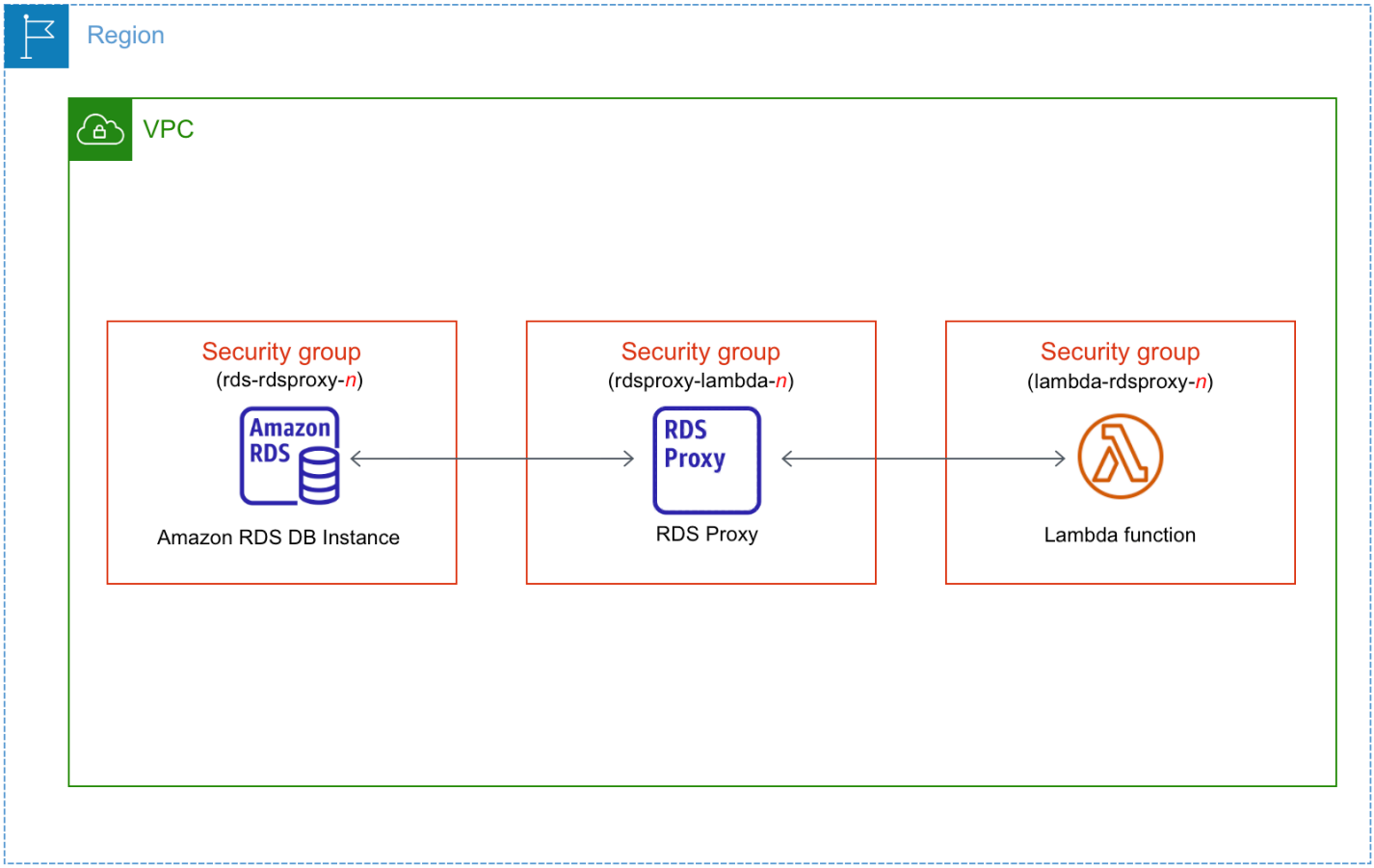


Vous pouvez configurer la connexion entre votre fonction Lambda et le de base de données via un RDS proxy afin d'améliorer les performances et la résilience de votre base de données. Les fonctions Lambda établissent souvent des connexions de base de données fréquentes et courtes

qui bénéficient du regroupement de connexions proposé RDS par Proxy. Vous pouvez tirer parti de n'importe quelle authentification AWS Identity and Access Management (IAM) dont vous disposez déjà pour les fonctions Lambda, au lieu de gérer les informations d'identification de base de données dans le code de votre application Lambda. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .

Lorsque vous utilisez la console pour vous connecter à un proxy existant, Amazon RDS met à jour le groupe de sécurité du proxy pour autoriser les connexions depuis votre instance de base de données, votre de base de données et la fonction Lambda.

Vous pouvez également créer un nouveau proxy à partir de la même page de console. Lorsque vous créez un proxy dans la console, pour accéder au de base de données, vous devez saisir vos informations d'identification de base de données ou sélectionner un AWS Secrets Manager secret.



> **Tip**
>
> Pour connecter rapidement une fonction Lambda à une instance de base de données (un cluster), vous pouvez également utiliser l'assistant guidé intégré à la console. Pour ouvrir l'assistant, procédez comme suit :
>
> 1. Ouvrez la page Functions (Fonctions) de la console Lambda.
> 2. Sélectionnez la fonction à laquelle vous souhaitez connecter une base de données.
> 3. Dans l'onglet Configuration, sélectionnez les RDSbases de données.
> 4. Choisissez Connect to RDS database.
>
> Après avoir connecté votre fonction à une base de données, vous pouvez créer un proxy en choisissant Ajouter un proxy.

Rubriques

- Vue d'ensemble de la connectivité automatique avec une fonction Lambda
- Connexion automatique d'une fonction Lambda à un cluster de base de RDSdonnées Aurora
- Affichage des ressources de calcul connectées

Vue d'ensemble de la connectivité automatique avec une fonction Lambda

Les conditions requises pour connecter une fonction Lambda à une instance de base de données (cluster de base de RDSdonnées) sont les suivantes :

- La fonction Lambda doit exister au même endroit que le cluster de base de données VPC de l' de base de données.
- L'utilisateur qui configure la connectivité doit être autorisé à effectuer les opérations AmazonRDS, Amazon, LambdaEC2, Secrets Manager et IAM autres suivantes :
  - Amazon RDS
    - `rds:CreateDBProxies`
    - `rds:DescribeDBInstances`
    - `rds:DescribeDBProxies`
    - `rds:ModifyDBInstance`

    - `rds:ModifyDBProxy`
    - `rds:RegisterProxyTargets`
- Amazon EC2
    - `ec2:AuthorizeSecurityGroupEgress`
    - `ec2:AuthorizeSecurityGroupIngress`
    - `ec2:CreateSecurityGroup`
    - `ec2:DeleteSecurityGroup`
    - `ec2:DescribeSecurityGroups`
    - `ec2:RevokeSecurityGroupEgress`
    - `ec2:RevokeSecurityGroupIngress`
- Lambda
    - `lambda:CreateFunctions`
    - `lambda:ListFunctions`
    - `lambda:UpdateFunctionConfiguration`
- Secrets Manager
    - `secretsmanager:CreateSecret`
    - `secretsmanager:DescribeSecret`
- IAM
    - `iam:AttachPolicy`
    - `iam:CreateRole`
    - `iam:CreatePolicy`
- AWS KMS
    - `kms:describeKey`

> **Note**
>
> Si l'instance de base de données et la fonction Lambda se trouvent dans des zones de disponibilité différentes, votre compte peut être confronté à des coûts croisés entre zones de disponibilité.

Si vous utilisez RDS un proxy, Amazon configure RDS également le groupe VPC de sécurité pour le proxy. Amazon RDS agit conformément à la configuration actuelle des groupes de sécurité associés à l'instance de base de données, au de base de données, à la fonction Lambda et au proxy, comme décrit dans le tableau suivant.

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés au de base de données dont le nom correspond au modèle `rds-lambda-` *n* ou, si un proxy est déjà connecté à votre de base `TargetHealth` de données, RDS vérifie si le proxy associé l'est`AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source.	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n* (où *n* est un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité n'a qu'une seule règle sortante avec le groupe de VPC sécurité du de base de données ou le proxy comme destination.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n* (où *n* est un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié.	Amazon ne RDS prend aucune mesure. Une connexion a déjà été configuré e automatiquement entre la fonction Lambda, le proxy (facultatif) et l'instanc e de base de données. Comme une connexion existe déjà entre la fonction, le proxy et la base de données, les groupes de sécurité ne sont pas modifiés.
L'une des conditions suivantes s'applique :	L'une des conditions suivantes s'applique :	L'une des conditions suivantes s'applique :	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
• Aucun groupe de sécurité n'est associé à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. • Un ou plusieurs groupes de sécurité sont associés à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. Toutefois, aucun de ces groupes de sécurité ne peut être utilisé pour la connexion à la fonction Lambda.	• Aucun groupe de sécurité n'est associé à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. • Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Cependant, Amazon ne RDS peut utiliser aucun de ces groupes de sécurité pour la connexion avec le de base de données. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle sortante avec le groupe de	• Aucun groupe de sécurité n'est associé au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. • Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond à `rdsproxy-lambda-` *n*. Amazon ne RDS peut toutefois utiliser aucun de ces groupes de sécurité pour la connexion avec l'instance de base de données, le de base de données ou la fonction Lambda. Amazon ne RDS peut pas utiliser un groupe de sécurité dépourvu de règles entrantes et sortantes avec le groupe de	

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié. Des exemples de modifications incluent l'ajout d'une règle ou la modification du port d'une règle existante.	VPC sécurité de l'instance de base de données, du de base de données ou du proxy comme destination. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	VPC sécurité du de base de données et la fonction Lambda. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source.	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Cependant, Amazon ne RDS peut utiliser aucun de ces groupes de sécurité pour la connexion avec le de base de données. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle sortante avec le groupe de VPC sécurité de l'instance de base de données, du de base de données ou du proxy comme destination. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. Amazon ne RDS peut toutefois utiliser aucun de ces groupes de sécurité pour la connexion avec l'instance de base de données, le de base de données ou la fonction Lambda. Amazon ne RDS peut pas utiliser un groupe de sécurité dépourvu de règles entrantes et sortantes avec le groupe de VPC sécurité du de base de données et la fonction Lambda. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité ne possède qu'une seule règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source.	Il existe un groupe de sécurité Lambda valide pour la connexion, mais il n'est pas associé à la fonction Lambda. Le nom de ce groupe de sécurité correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Il n'a pas été modifié. Il n'a qu'une seule règle sortante avec le groupe de VPC sécurité de l'instance de base de données, du de base de données ou du proxy comme destination.	Il existe un groupe de sécurité de proxy valide pour la connexion, mais il n'est pas associé au proxy. Le nom de ce groupe de sécurité correspond au modèle `rdsproxy-lambda-` *n*. Il n'a pas été modifié. Il possède des règles entrantes et sortantes avec le groupe de VPC sécurité du de base de données et la fonction Lambda.	RDS action: associate Lambda security group

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. • Un ou plusieurs groupes de sécurité sont associés à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. Amazon ne RDS peut toutefois utiliser aucun de	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité n'a qu'une seule règle sortante avec le groupe de VPC sécurité de l'instance de base de données, de l' de base de données ou du proxy comme destination.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité possède des règles entrantes et sortantes avec le groupe de VPC sécurité du de base de données et la fonction Lambda.	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
ces groupes de sécurité pour la connexion avec la fonction Lambda ou le proxy. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.			

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. • Un ou plusieurs groupes de sécurité sont associés à l'instance de base de données avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. Amazon ne RDS peut toutefois utiliser aucun de	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. • Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Cependant, Amazon ne RDS peut utiliser aucun de ces groupes de sécurité pour la connexion avec le de base de données. Amazon ne RDS peut pas utiliser un groupe	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. • Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond à `rdsproxy-lambda-` *n*. Amazon ne RDS peut toutefois utiliser aucun de ces groupes de sécurité pour la connexion avec l'instance de base de données, le de base de données ou la fonction Lambda. Amazon ne RDS peut pas utiliser un groupe de sécurité	RDS action: create new security groups

Configuration actuelle RDS du groupe de sécurité	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
ces groupes de sécurité pour la connexion avec la fonction Lambda ou le proxy. Amazon ne RDS peut pas utiliser un groupe de sécurité qui n'a pas de règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	de sécurité qui n'a pas de règle sortante avec le groupe de VPC sécurité de l'instance de base de données, du de base de données ou du proxy comme source. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	dépourvu de règles entrantes et sortantes avec le groupe de VPC sécurité du de base de données et la fonction Lambda. Amazon ne peut pas non RDS plus utiliser un groupe de sécurité qui a été modifié.	

RDSaction : créer de nouveaux groupes de sécurité

Amazon RDS prend les mesures suivantes :

- Crée un nouveau groupe de sécurité qui correspond au modèle `rds-lambda-`*n* ou `rds-rdsproxy-`*n* (si vous choisissez d'utiliser un RDS proxy). Ce groupe de sécurité possède une règle entrante avec le groupe de VPC sécurité de la fonction Lambda ou du proxy comme source. Ce groupe de sécurité est associé à l'instance de base de données et permet à la fonction ou au proxy d'accéder à l'instance de base de données.
- Crée un nouveau groupe de sécurité qui correspond au modèle `lambda-rds-`*n* ou `lambda-rdsproxy-`*n*. Ce groupe de sécurité dispose d'une règle sortante avec le groupe de VPC sécurité de l'instance de base de données, du de base de données ou du proxy comme destination. Ce

groupe de sécurité est associé à la fonction Lambda et permet à cette dernière d'envoyer du trafic vers l'instance de base de données ou d'envoyer du trafic via un proxy.

- Crée un nouveau groupe de sécurité qui correspond au modèle `rdsproxy-lambda-`*n*. Ce groupe de sécurité possède des règles entrantes et sortantes avec le groupe de VPC sécurité du de base de données et la fonction Lambda.

RDSaction : associer un groupe de sécurité Lambda

Amazon RDS associe le groupe de sécurité Lambda valide et existant à la fonction Lambda. Ce groupe de sécurité permet à la fonction Lambda d'envoyer du trafic vers l'instance de base de données ou d'envoyer du trafic via un proxy.

Connexion automatique d'une fonction Lambda à un cluster de base de RDSdonnées Aurora

Vous pouvez utiliser la RDS console Amazon pour connecter automatiquement une fonction Lambda à votre cluster de base de données d' de base de données. Cela simplifie le processus de configuration d'une connexion entre ces ressources.

Vous pouvez également utiliser RDS Proxy pour inclure un proxy dans votre connexion. Les fonctions Lambda établissent fréquemment de courtes connexions à la base de données qui bénéficient du regroupement de connexions proposé RDS par Proxy. Vous pouvez également utiliser n'importe quelle IAM authentification que vous avez déjà configurée pour votre fonction Lambda, au lieu de gérer les informations d'identification de base de données dans le code de votre application Lambda.

Vous pouvez connecter une instance de base de données existante aux fonctions Lambda nouvelles et existantes à l'aide de la page Configurer une connexion Lambda. Le processus de configuration configure automatiquement les groupes de sécurité requis pour vous.

Avant de configurer une connexion entre une fonction Lambda et une instance de base de données, assurez-vous que :

- Votre fonction Lambda et votre cluster de base de données d'instance . VPC
- Vous disposez des autorisations appropriées pour votre compte d'utilisateur. Pour plus d'informations sur les exigences, consultez Vue d'ensemble de la connectivité automatique avec une fonction Lambda.

Si vous modifiez les groupes de sécurité après avoir configuré la connectivité, ces modifications peuvent affecter la connexion entre la fonction Lambda et l'instance de base de données.

> **Note**
>
> Vous pouvez configurer automatiquement une connexion entre un de base de données et une fonction Lambda uniquement dans le. AWS Management Console Pour connecter une fonction Lambda, l'instance de base de données doit être dans l'état Disponible.

Pour connecter automatiquement une fonction Lambda et une instance de base de données

<result>

Après avoir confirmé la configuration, Amazon RDS commence le processus de connexion de votre fonction Lambda, de votre RDS proxy (si vous avez utilisé un proxy) et de votre cluster de base de données d' de base de données. La console affiche la boîte de dialogue Détails de connexion, qui répertorie les modifications de groupe de sécurité qui permettent les connexions entre vos ressources.
</result>

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis choisissez l'instance de base de données que vous voulez connecter à une fonction Lambda.
3. Pour Actions, choisissez Configurer la connexion Lambda.
4. Sur la page Configurer la connexion Lambda, sous Sélectionner une fonction Lambda, effectuez l'une des opérations suivantes :
   - Si vous avez une fonction Lambda existante VPC identique à votre de base de données, choisissez Choisir une fonction existante, puis choisissez la fonction.
   - Si vous n'avez pas de fonction Lambda dans la même fenêtreVPC, choisissez Create new function, puis entrez un nom de fonction. L'environnement d'exécution par défaut est défini sur Nodejs.18. Vous pouvez modifier les paramètres de votre nouvelle fonction Lambda dans la console Lambda après avoir terminé la configuration de la connexion.
5. (Facultatif) Sous RDSProxy, sélectionnez Se connecter à l'aide d'un RDS proxy, puis effectuez l'une des opérations suivantes :

- Si vous souhaitez utiliser un proxy existant, choisissez Choisir un proxy existant, puis choisissez le proxy.
- Si vous n'avez pas de proxy et que vous souhaitez qu'Amazon RDS en crée un automatiquement pour vous, choisissez Create new proxy. Ensuite, pour Informations d'identification de la base de données, effectuez l'une des opérations suivantes :

  a. Choisissez Nom d'utilisateur et mot de passe de base de données, puis saisissez le Nom d'utilisateur et le Mot de passe de votre instance de base de données.

  b. Choisissez Secret Secrets Manager. Ensuite, pour Sélectionner un secret, choisissez un AWS Secrets Manager secret. Si vous n'avez pas de secret Secrets Manager, choisissez Créer un nouveau secret Secrets Manager pour créer un nouveau secret. Après avoir créé le secret, pour Sélectionner un secret, choisissez le nouveau secret.

  Après avoir créé le nouveau proxy, choisissez Choisir un proxy existant, puis choisissez le proxy. Notez qu'il peut s'écouler un certain temps avant que votre proxy soit disponible pour la connexion.

6. (Facultatif) Développez Récapitulatif de la connexion et vérifiez les mises à jour en surbrillance pour vos ressources.
7. Choisissez Set up (Configurer).

Affichage des ressources de calcul connectées

Vous pouvez utiliser le AWS Management Console pour afficher les fonctions Lambda connectées à votre cluster de base de données d' de base de données. Les ressources présentées incluent les connexions aux ressources de calcul qu'Amazon RDS met en place automatiquement.

Les ressources de calcul répertoriées n'incluent pas celles qui sont connectées manuellement à l'instance de base de données. Par exemple, vous pouvez autoriser une ressource de calcul à accéder manuellement au de base de données en ajoutant une règle à votre groupe VPC de sécurité associé à la base de données.

Pour que la console répertorie une fonction Lambda, les conditions suivantes doivent s'appliquer :

- Le nom du groupe de sécurité associé à la ressource de calcul correspond au modèle `lambda-rds-`*n* ou `lambda-rdsproxy-`*n* (où *n* est un nombre).

- Le groupe de sécurité associé à la ressource de calcul possède une règle sortante avec la plage de ports définie sur le port de l'instance de base de données ou d'un proxy associé. La destination de la règle sortante doit être définie sur un groupe de sécurité associé à l'instance de base de données ou un proxy associé.
- Si la configuration inclut un proxy, le nom du groupe de sécurité attaché au proxy associé à votre base de données correspond au modèle `rdsproxy-lambda-`*n* (où *n* est un nombre).
- Le groupe de sécurité associé à la fonction possède une règle sortante avec le port défini sur le port utilisé par l'instance de base de données ou le proxy associé. La destination doit être définie sur un groupe de sécurité associé à l'instance de base de données ou au proxy associé.

Pour afficher les ressources de calcul automatiquement connectées à une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis choisissez l'instance de base de données.
3. Dans l'onglet Connectivité et sécurité, examinez les ressources de calcul sous Ressources de calcul connectées.

Modification d'une RDS instance de base de données Amazon

Vous pouvez modifier les paramètres d'une instance de base de données pour accomplir des tâches telles que l'ajout de stockage supplémentaire ou la modification de la classe d'instance. Dans cette rubrique, vous découvrirez comment modifier une RDS instance de base de données Amazon et en savoir plus sur les paramètres des instances de base de données.

Nous vous recommandons de tester toutes les modifications apportées à une instance test avant de modifier une instance de production. Cela vous permet de bien comprendre l'impact de chaque changement. Les tests sont particulièrement importants lors de la mise à niveau des versions de base de données.

La plupart des modifications apportées à une instance de base de données peuvent être appliquées immédiatement ou différées jusqu'à la fenêtre de maintenance suivante. Certaines modifications, telles que les modifications d'un groupe de paramètres, nécessitent que vous redémarriez manuellement votre instance de base de données pour que la modification entre en vigueur.

> **Important**
>
> Certaines modifications entraînent des interruptions de service car Amazon RDS doit redémarrer votre instance de base de données pour que la modification soit prise en compte. Analysez l'impact sur votre base de données et les applications avant de modifier vos paramètres d'instance de base de données.

Console

Pour modifier une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier). La page Modifier l'instance de base de données s'affiche.
4. Modifiez les paramètres de votre choix. Pour plus d'informations sur chaque paramètre, consultez Paramètres des instances de base de données.
5. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez le résumé des modifications.

6. (Facultatif) Choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. La sélection de cette option peut entraîner des temps d'arrêt dans certains cas. Pour plus d'informations, consultez Utilisation du paramètre de modification du calendrier.
7. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modify DB instance (Modifier l'instance de base de données) pour enregistrer vos modifications.

   Ou choisissez Retour pour revoir vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour modifier une instance de base de données à l'aide de AWS CLI, appelez la modify-db-instancecommande. Spécifiez l'identifiant d'instance de base de données et les valeurs des options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter Paramètres des instances de base de données.

Example

Le code suivant modifie `mydbinstance` en définissant la période de rétention des sauvegardes sur 1 semaine (7 jours). Ce code active la protection contre la suppression en utilisant `--deletion-protection`. Pour désactiver la protection contre la suppression, utilisez `--no-deletion-protection`. Les modifications sont appliquées pendant le créneau de maintenance suivant à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --backup-retention-period 7 \
    --deletion-protection \
    --no-apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --backup-retention-period 7 ^
    --deletion-protection ^
```

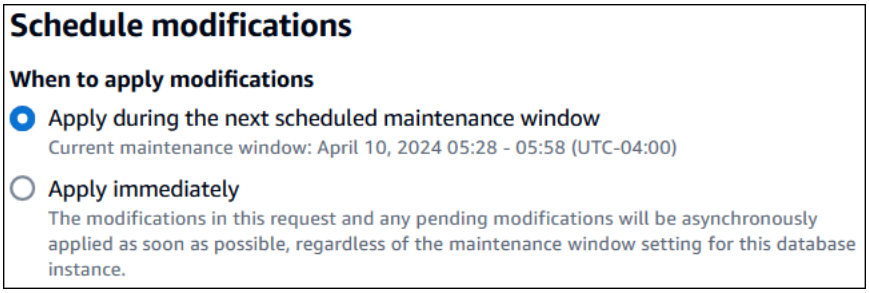
```
    --no-apply-immediately
```

RDS API

Pour modifier une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'odifyDBInstanceopération M. Spécifiez l'identifiant d'instance de base de données et les paramètres que vous souhaitez modifier. Pour plus d'informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Utilisation du paramètre de modification du calendrier

Lorsque vous modifiez votre instance de base de données, vous décidez quand vous souhaitez que les modifications se produisent.



Pour appliquer les modifications immédiatement plutôt que dans la fenêtre de maintenance suivante, choisissez l'option Appliquer immédiatement dans le AWS Management Console. Vous pouvez également utiliser le `--apply-immediately` paramètre lorsque vous appelez le AWS CLI ou définissez le `ApplyImmediately` paramètre sur `true` lorsque vous utilisez Amazon RDSAPI.

Si vous ne choisissez pas d'appliquer les modifications immédiatement, RDS place les modifications dans la file d'attente des modifications. Au cours de la fenêtre de maintenance suivante, RDS applique toutes les modifications en attente dans la file d'attente. Si vous choisissez d'appliquer les modifications immédiatement, vos nouvelles modifications et les modifications placées dans la file d'attente des modifications en attente sont appliquées.

Pour voir les modifications en attente pour la prochaine fenêtre de maintenance, utilisez la describe-db-instances AWS CLI commande et cochez le `PendingModifiedValues` champ.

> **Important**
>
> Si des modifications en attente nécessitent une indisponibilité temporaire de l'instance de base de données (temps d'arrêt), le choix de l'option Appliquer immédiatement peut entraîner une interruption imprévue.

Si vous choisissez d'appliquer une modification immédiatement, les modifications en attente sont également appliquées immédiatement, au lieu d'attendre la fenêtre de maintenance suivante.
Si vous ne souhaitez pas qu'une modification en attente soit appliquée lors de la fenêtre de maintenance suivante, vous pouvez modifier l'instance de base de données de façon à inverser la modification. Vous pouvez le faire en utilisant l'option AWS CLI et en spécifiant l'`--apply-immediately`option.

Les modifications de certains paramètres de base de données sont appliquées immédiatement, même si vous choisissez de reporter vos modifications. Pour savoir comment les différents paramètres de base de données interagissent avec le paramètre Appliquer immédiatement, veuillez consulter Paramètres des instances de base de données.

Paramètres des instances de base de données

Le tableau suivant contient des détails sur les paramètres que vous pouvez et ne pouvez pas modifier. Vous pouvez également savoir quand les modifications peuvent être appliquées et si les modifications entraînent des temps d'arrêt pour votre instance de base de données. En utilisant les fonctions Amazon RDS telles que multi-AZ, vous pouvez réduire les temps d'arrêt si vous modifiez ultérieurement l'instance de base de données. Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Vous pouvez modifier une instance de base de données à l'aide de la console, de la commande de CLI `modify-db-instance` ou de l'opération d'API RDS `ModifyDBInstance`.

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Stockage alloué	Option de l'interface CLI :	Si vous choisissez d'appliquer	Aucun temps d'arrêt n'a lieu pendant cette	Tous les moteurs

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Volume de stockage en gibioctets que vous voulez allouer pour votre instance de base de données. Vous pouvez uniquement augmenter le stockage alloué. Vous ne pouvez pas le réduire.
 	
Vous ne pouvez pas modifier le stockage de certaines instances de base de données plus anciennes ou les instances de base de données restaurées à partir d'instantanés de bases de données. Le paramètre Stockage alloué est désactivé dans la console si votre instance de base de données n'est pas éligible. Vous pouvez vérifier si vous pouvez allouer plus de stockage à l'aide de la commande CLI describe-valid-db-instance-modifications. Cette commande renvoie les options de stockage valides pour votre instance de base de données.
	`--allocated-storage`
	
	Paramètre de l'API RDS :
	
	`AllocatedStorage`
		la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			modification. Les performances peuvent se dégrader pendant la modification.	de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Vous ne pouvez pas modifier le stockage alloué si le statut de l'instance de base de données est storage-optimization. Vous ne pouvez pas non plus modifier le stockage alloué pour une instance de base de données s'il a été modifié au cours des six dernières heures. Le stockage maximal autorisé dépend de votre moteur de base de données et du type de stockage. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS.				

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Configuration de l'architecture Configuration qui permet à plusieurs bases de données locataire de résider dans votre instance de base de données. Actuellement, seul RDS pour les bases de données de conteneurs Oracle (CDBs) prend en charge ce paramètre. Si votre CDB est dans la configuration à locataire unique, vous pouvez la modifier pour utiliser la configuration à locataires multiples. Dans cette configuration, vous pouvez utiliser RDS APIs pour créer 1 à 30 bases de données mutualisées, en fonction de l'édition de la base de données et des licences optionnelles requises. L'application PDBs et le proxy PDBs ne sont pas pris en charge. La configuration à locataires multiples est définitive, ce qui signifie que vous ne pourrez pas reconvertir ultérieur	Option de l'interface CLI : `--multi-tenant` (configuration à locataires multiples de l'architecture CDB) `--no-multi-tenant` (configuration à locataire unique de l'architecture CDB) Paramètre de l'API : `MultiTenant`	La modification a lieu immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Oracle

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
ement votre CDB en configuration à locataire unique. **Note** La configuration Amazon RDS est appelée « multi-tenant » plutôt que « multitenant » car il s'agit d'une fonctionnalité d'Amazon RDS, et pas seulement du moteur de base de données Oracle. De même, le terme « locataire » du RDS fait référence à tout locataire d'une configuration RDS, et pas seulement à Oracle. PDBs Dans la documentation RDS, le terme sans trait d'union « Oracle multitenant » fait exclusivement référence à l'architecture CDB de la base de données Oracle, qui est compatible avec				

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
les déploiements sur site et RDS. Pour de plus amples informations, veuillez consulter Présentation de RDS pour Oracle CDBs.				

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Paramètres de l'architecture L'architecture de la base de données : CDB ou non CDB. Si vous choisissez Architecture multilocataire Oracle, RDS for Oracle convertit votre base de données non CDB en une CDB qui utilise la configuration à locataire unique. Ce paramètre n'est pris en charge que si votre base de données est une base de données non-CDB exécutant Oracle Database 19c avec une RU d'avril 2021 ou ultérieure. Après la conversion, votre CDB contient une base de données initiale enfichable (PDB). Le changement d'architecture est définitif, ce qui signifie que vous ne pouvez pas reconvertir votre CDB en base de données non CDB.	Option de l'interface CLI : `--engine oracle-ee-cdb` (architecture multilocataire Oracle) `--engine oracle-se2-cdb` (architecture multilocataire Oracle) Paramètre de l'API : `Engine`	Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement. Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.	Un temps d'arrêt a lieu pendant cette modification.	Oracle

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Note Pour convertir une CDB dans la configuration à locataire unique en configuration à locataires multiples, modifiez à nouveau votre instance de CDB et choisissez Configuration à locataires multiples pour l'option Configuration de l'architecture. Pour de plus amples informations, veuillez consulter Configuration à locataire unique de l'architecture CDB.				

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Mise à niveau automatique de versions mineures
 	
Choisissez Activer la mise à niveau automatique des versions mineures pour permettre à votre instance de base de données de recevoir automatiquement les mises à niveau des versions mineures préférées du moteur de base de données lorsqu'el les sont disponibles. Il s'agit du comportement de par défaut. Amazon RDS effectue les mises à niveau automatiques des versions mineures dans la fenêtre de maintenance. Si vous ne sélectionnez pas Activer la mise à niveau automatique des versions mineures, votre instance de base de données n'est pas mise à niveau automatiquement lorsque de nouvelles versions mineures sont disponibles.
 	
Pour de plus amples informati ons, veuillez consulter Mise à
	Option de l'interface CLI :
	
	`--auto-minor-version-upgrade | --no-auto-minor-version-upgrade`
	
	Paramètre de l'API RDS :
	
	`AutoMinorVersionUpgrade`
		La modificat ion a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiate ment.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
niveau automatique de la version mineure du moteur.				
Réplication des sauvegardes Choisissez Activer la réplication vers une autre AWS région pour créer des sauvegardes dans une région supplémentaire à des fins de reprise après sinistre. Sélectionnez ensuite la Région de destination des sauvegardes supplémentaires.	Non disponible lors de la modification d'une instance de base de données. Pour plus d'informations sur l'activation des sauvegardes entre régions à l'aide de l'API AWS CLI ou RDS, consultez . Activation des sauvegardes automatisées entre régions pour Amazon RDS	La modification est appliquée de manière asynchrone, dès que possible.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Période de rétention des sauvegardes
 	
Le nombre de jours de conservation des sauvegardes automatiques. Pour désactiver les sauvegardes automatiques, définissez la période de conservation des sauvegardes sur 0.
 	
Pour de plus amples informations, veuillez consulter Présentation des sauvegardes.
 	
Note
Si vous avez l' AWS Backup habitude de gérer vos sauvegardes, cette option ne s'applique pas. Pour plus d'informations à ce sujet AWS Backup, consultez le AWS Backup Developer Guide.
	Option de l'interface CLI :
	
	`--backup-retention-period`
	
	Paramètre de l'API RDS :
	
	`BackupRetentionPeriod`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous choisissez de ne pas appliquer la modification immédiatement et que vous remplacez la valeur non nulle du paramètre par une autre valeur non nulle, la modification est appliquée de manière asynchron
			Un temps d'arrêt se produit si vous passez de 0 à une valeur non nulle, ou d'une valeur non nulle à 0.
			
			Cela s'applique aux instances de base de données Mono-AZ et Multi-AZ.
				Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
		e dès que possible. Sinon, la modification est appliquée pendant la fenêtre de maintenance suivante.		

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Fenêtre de sauvegarde
 	
L'intervalle de temps pendant lequel des sauvegardes automatiques de vos bases de données se produisent. Le créneau de sauvegarde correspond à une heure de début en heure UTC (Universal Coordinated Time) et une durée en heures.
 	
Pour de plus amples informations, veuillez consulter Présentation des sauvegardes.
 	
Note
Si vous avez l' AWS Backup habitude de gérer vos sauvegardes, cette option n'apparaît pas. Pour plus d'informations à ce sujet AWS Backup, consultez le guide du AWS Backup développeur.
	Option de l'interface CLI :
	
	`--preferred-backup-window`
	
	Paramètre de l'API RDS :
	
	`PreferredBackupWindow`
		La modification est appliquée de manière asynchrone, dès que possible.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Autorité de certification
 	
L'autorité de certification (CA) pour le certificat de serveur utilisé par l'instance de base de données.
 	
Pour de plus amples informations, veuillez consulter .
	Option de l'interface CLI :
	
	`--ca-certificate-identifier`
	
	Paramètre de l'API RDS :
	
	`CACertificateIdentifier`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt survient uniquement si le moteur de base de données ne prend pas en charge la rotation sans redémarrage. Vous pouvez utiliser la describe-db-engine-versions AWS CLI commande pour déterminer si le moteur de base de données prend en charge la rotation sans redémarrage.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Copier les balises aux instantanés
 	
Si vous avez des balises d'instance de base de données, activez cette option pour les copier lorsque vous créez un instantané de bases de données.
 	
Pour plus d'informations, consultez Marquage d' Amazon RDS.
	Option de l'interface CLI :
	
	`--copy-tags-to-snapshot` ou `--no-copy-tags-to-snapshot`
	
	Paramètre de l'API RDS :
	
	`CopyTagsToSnapshot`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Port de la base de données
 	
Port que vous souhaitez utiliser pour accéder à l'instance de bases de données.
 	
La valeur du port ne doit correspondre à aucune des valeurs de port spécifiées pour les options du groupe d'options associé à l'instance de base de données.
 	
Pour plus d'informations, consultez Connexion à une RDS instance de base de données Amazon.
	Option de l'interface CLI :
	
	`--db-port-number`
	
	Paramètre de l'API RDS :
	
	`DBPortNumber`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	L'instance de base de données est redémarrée immédiatement.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Version du moteur de base de données
 	
Version du moteur de base de données que vous souhaitez utiliser. Avant de mettre à niveau votre instance de bases de données de production, nous vous recommandons de tester le processus de mise à niveau sur une instance de base de données de test. Cela permet de vérifier sa durée et de valider vos applications.
 	
Pour de plus amples informations, veuillez consulter Mise à niveau de la version du moteur d'une instance de base de données.
	Option de l'interface CLI :
	
	`--engine-version`
	
	Paramètre de l'API RDS :
	
	`EngineVersion`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Classe d'instances de base de données
 	
La classe d'instance de base de données que vous souhaitez utiliser.
 	
Pour plus d'informations, consultez Classes d'instances de base de données .
	Option de l'interface CLI :
	
	`--db-instance-class`
	
	Paramètre de l'API RDS :
	
	`DBInstanceClass`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Identifiant d'instance de base de données
 	
Nouvel identifiant de l'instance de base de données. Cette valeur est stockée sous la forme d'une chaîne en minuscules.
 	
Pour plus d'informations sur les effets du changement de nom d'une instance de base de données, consultez Affectation d'un nouveau nom à une instance DB.
	Option de l'interface CLI :
	
	`--new-db-instance-identifier`
	
	Paramètre de l'API RDS :
	
	`NewDBInstanceIdentifier`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Des temps d'arrêt se produisent pendant cette modification, sauf si la version de votre moteur de base de données prend en charge le chargement SSL dynamique. Pour déterminer si votre version nécessite un redémarrage, exécutez la AWS CLI commande suivante :
			
			`aws rds describe-db-engine`
				Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
			`-versions \ --default-only \ --engine your-db-engine \ --query 'DBEngineVersions[*].SupportsCertificateRotationWithoutRestart'`	

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Groupe de paramètres de base de données
 	
Groupe de paramètres de base de données que vous souhaitez associer à l'instance de base de données.
 	
Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).
	Option de l'interface CLI :
	
	`--db-parameter-group-name`
	
	Paramètre de l'API RDS :
	
	`DBParameterGroupName`
		L'association du nouveau groupe de paramètres de base de données à l'instance de base de données a lieu immédiatement.	Aucune interruption de service ne se produit lorsque vous associez un nouveau groupe de paramètres de base de données à votre instance de base de données.
			
			L'association d'un groupe de paramètres de base de données est différente de l'application de modifications de paramètres au sein d'un groupe de paramètres. RDS applique les paramètre
				Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
			s statiques et dynamiques modifiés dans le nouveau groupe associé uniquemen t après le redémarra ge manuel de l'instanc e de base de données. Toutefois, si vous modifiez les paramètre s dynamique s du groupe de paramètre s de base de données après l'avoir associé à l'instance de base de données, ces paramètres sont appliqués immédiatement sans nécessite	

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
			r de redémarrage. Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS) et Redémarrage d'une instance de base de données.	

Volumes dédiés aux journaux
 	
Utilisez un volume dédié aux journaux (DLV) pour stocker les journaux de transactions de base de données sur un volume de stockage distinct du volume contenant les tables de base de données.
 	
Pour de plus amples informations, veuillez consulter Utilisation d'un volume dédié aux journaux (DLV).
	Option de l'interface CLI :
	
	`-dedicated-log-volume`
	
	Paramètre de l'API RDS :
	
	`DedicatedLogVolume`
		La modification est appliquée lorsque l'instance de base de données est redémarrée.	Une interruption de service se produit lors du redémarrage de l'instance de base de données.	MariaDB, MySQL, PostgreSQL

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Deletion protection (Protection contre la suppression)
 	
Enable deletion protection (Activer la protection contre la suppression) vise à empêcher la suppression de votre instance de base de données.
 	
Pour plus d'informations, consultez Suppression d'une instance DB.
	Option de l'interface CLI :
	
	`--deletion-protection|--no-deletion-protection`
	
	Paramètre de l'API RDS :
	
	`DeletionProtection`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Surveillance améliorée
 	
Activer la surveillance améliorée permet d'activer la collecte des métriques en temps réel pour le système d'exploitation sur lequel votre instance de base de données s'exécute.
 	
Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.
	Option de l'interface CLI :
	
	`--monitoring-interval` and `--monitoring-role-arn`
	
	Paramètre de l'API RDS :
	
	`MonitoringInterval` and `MonitoringRoleArn`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Authentification de base de données IAM
 	
Enable IAM DB authentication (Activer l'authentification de base de données IAM) pour authentifier les utilisateurs de base de données via les utilisateurs et les rôles.
 	
Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.
	Option de l'interface CLI :
	
	`--enable-iam-database-authentication |--no-enable-iam-database-authentication`
	
	Paramètre de l'API RDS :
	
	`EnableIAMDatabaseAuthentication`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Aucun temps d'arrêt n'a lieu pendant cette modification.	MariaDB, MySQL et PostgreSQL uniquement

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Authentification Kerberos
 	
Choisissez le répertoire Active Directory vers lequel déplacer l'instance de base de données. Le répertoire doit exister avant cette opération. Si un répertoire est déjà sélectionné, vous pouvez spécifier Aucun pour supprimer l'instance de base de données de son répertoire actuel.
 	
Pour plus d'informations, consultez Authentification Kerberos.
	Option de l'interface CLI :
	
	`--domain` and `--domain-iam-role-name`
	
	Paramètre de l'API RDS :
	
	`Domain` and `DomainIAMRoleName`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un bref temps d'arrêt a lieu au cours de cette modification.	Uniquement Microsoft SQL Server, MySQL, Oracle et PostgreSQL

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Modèle de licence
 	
Choisissez bring-your-own-licensed'utiliser votre licence pour Db2 et Oracle.
 	
Choisissez licence-incluse pour utiliser le contrat de licence général pour Microsoft SQL Server ou Oracle.
 	
Pour plus d'informations, consultez Options de licence Amazon RDS pour DB2, Gestion des licences Microsoft SQL Server sur Amazon RDS et RDSpour les options de licence Oracle.
	Option de l'interface CLI :
	
	`--license-model`
	
	Paramètre de l'API RDS :
	
	`LicenseModel`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt a lieu pendant cette modification.	Uniquement Microsoft SQL Server et Oracle

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Exportations des journaux
 	
Les types de fichiers journaux de base de données à publier sur Amazon CloudWatch Logs.
 	
Pour de plus amples informations, veuillez consulter Publication des journaux de base de données sur Amazon CloudWatch Logs.
	Option de l'interface CLI :
	
	`--cloudwatch-logs-export-configuration`
	
	Paramètre de l'API RDS :
	
	`CloudwatchLogsExportConfiguration`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Fenêtre de maintenance
 	
L'intervalle de temps pendant lequel la maintenance du système a lieu. La maintenance du système inclut les mises à niveau, le cas échéant. Le fenêtre de maintenance correspond à une heure de début en heure UTC (Universal Coordinated Time) et une durée en heures.
 	
Si vous définissez la fenêtre sur l'heure actuelle, il doit y avoir au moins 30 minutes entre l'heure actuelle et la fin du créneau. Ce calendrier permet de s'assurer que toutes les modifications en attente sont appliquées.
 	
Pour de plus amples informations, veuillez consulter Fenêtre de maintenance Amazon RDS.
	Option de l'interface CLI :
	
	`--preferred-maintenance-window`
	
	Paramètre de l'API RDS :
	
	`PreferredMaintenanceWindow`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Si une ou plusieurs actions en attente entraînent un temps d'arrêt et que la fenêtre de maintenance est modifiée pour inclure l'heure actuelle, les actions en attente sont appliquées immédiatement et un temps d'arrêt se produit.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Gérez les informations d'identification principales dans AWS Secrets Manager
 	
Sélectionnez Gérer les informations d'identification principales dans AWS Secrets Manager pour gérer le mot de passe d'utilisateur principal dans un secret, dans Secrets Manager.
 	
Vous pouvez éventuellement choisir une clé KMS à utiliser pour protéger le secret. Choisissez l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
 	
Si RDS gère déjà le mot de passe d'utilisateur principal pour l'instance de base de données, vous pouvez effectuer la rotation du mot de passe d'utilisateur principal en choisissant Rotate secret immediately (Effectuer immédiatement une rotation du secret).
	Option de l'interface CLI :
	
	`--manage-master-user-password | --no-manage-master-user-password`
	
	`--master-user-secret-kms-key-id`
	
	`--rotate-master-user-password | --no-rotate-master-user-password`
	
	Paramètre de l'API RDS :
		Si vous activez ou désactivez la gestion automatique des mots de passe d'utilisateur principal, la modification se produit immédiatement. Cette modification ignore le paramètre d'application immédiate.
		
		Si vous effectuez la rotation du mot de passe de l'utilisateur principal, vous devez spécifier que la modification doit s'appliqu
			Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.	`ManageMasterUserPassword`
	
	`MasterUserSecretKmsKeyId`
	
	`RotateMasterUserPassword`
		er immédiatement.		

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

déploiement multi-AZ
 	
Oui pour déployer votre instance de base de données dans plusieurs zones de disponibilité. Dans le cas contraire, Non.
 	
Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
	Option de l'interface CLI :
	
	`--multi-az|--no-multi-az`
	
	Paramètre de l'API RDS :
	
	`MultiAZ`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Aucun temps d'arrêt n'a lieu pendant cette modification. Toutefois, il existe un impact possible sur les performances. Pour de plus amples informations, veuillez consulter Conversion d'une instance de base de données en déploiement multi-AZ pour Amazon RDS.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Network type (Type de réseau)
 	
Les protocoles d'adressage IP pris en charge par l'instance de la base de données.
 	
IPv4pour spécifier que les ressources peuvent communiquer avec l'instance de base de données uniquement via le protocole d'adressage Internet Protocol version 4 (IPv4).
 	
Mode double pile pour spécifier que les ressources peuvent communiquer avec l'instance de base de données via IPv4 le protocole Internet version 6 (IPv6) ou les deux. Utilisez le mode double pile si vous avez des ressources qui doivent communiquer avec votre instance de base de données via le protocole d' IPv6adressage. Assurez-vous également d'associer un bloc IPv6 CIDR à tous les sous-réseaux du groupe
	Option de l'interface CLI :
	
	`--network-type`
	
	Paramètre de l'API RDS :
	
	`NetworkType`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt est possible pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
de sous-réseaux de base de données que vous spécifiez. Pour de plus amples informations, veuillez consulter Adressage IP Amazon RDS.				

New master password
 	
Le mot de passe de votre utilisateur principal. Le mot de passe doit contenir entre 8 et 41 caractères alphanumériques.
	Option de l'interface CLI :
	
	`--master-user-password`
	
	Paramètre de l'API RDS :
	
	`MasterUserPassword`
		La modification est appliquée de manière asynchrone, dès que possible. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Groupe d'options
 	
Le groupe d'options que vous voulez associer à l'instance de base de données.
 	
Pour plus d'informations, consultez Utilisation de groupes d'options.
	Option de l'interface CLI :
	
	`--option-group-name`
	
	Paramètre de l'API RDS :
	
	`OptionGroupName`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Aucun temps d'arrêt n'a lieu pendant cette modification. Seul l'ajout du plug-in d'audit MariaDB à une instance de base de données RDS for MariaDB ou RDS for MySQL est susceptible de provoquer une panne.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Performance Insights
 	
Activer Performance Insights permet de surveiller la charge de votre instance de base de données et ainsi d'analyser les performances de votre base de données et de résoudre les problèmes associés.
 	
Performance Insights n'est pas disponible pour certaines versions de moteurs de base de données et classes d'instance de base de données. La section Performance Insights n'apparaît pas dans la console si elle n'est pas disponible pour votre instance de base de données.
 	
Pour plus d'informations, consultez Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS et Prise en charge de la classe d'instances, de la région et du moteur de base
	Option de l'interface CLI :
	
	`--enable-performance-insights| --no-enable-performance-insights`
	
	Paramètre de l'API RDS :
	
	`EnablePerformanceInsights`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous sauf Db2

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
de données Amazon RDS pour Performance Insights.				

Performance Insights AWS KMS key
 	
Identifiant AWS KMS clé AWS KMS key pour le chiffrement des données Performance Insights. L'identifiant de clé est le Amazon Resource Name (ARN), AWS KMS l'identifiant de clé ou l'alias de clé KMS.
 	
Pour de plus amples informations, veuillez consulter Activer et désactiver Performance Insights pour Amazon RDS.
	Option de l'interface CLI :
	
	`--performance-insights-kms-key-id`
	
	Paramètre de l'API RDS :
	
	`PerformanceInsightsKMSKeyId`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous sauf Db2

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Performance Insights Retention period (Période de rétention des analyses de performances)
 	
Durée de conservation, en jours, des données de Performance Insights. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.
 	
Pour de plus amples informations, veuillez consulter Activer et désactiver Performance Insights pour Amazon RDS.
	Option de l'interface CLI :
	
	`--performance-insights-retention-period`
	
	Paramètre de l'API RDS :
	
	`PerformanceInsightsRetentionPeriod`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous sauf Db2

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Processor features (Caractéristiques du processeur)
 	
Nombre de cœurs d'UC et nombre de threads par cœur pour la classe à laquelle appartient l'instance de base de données.
 	
Pour plus d'informations, consultez Configuration du processeur pour une classe d'instances de base de données dans RDS for Oracle.
	Option de l'interface CLI :
	
	`--processor-features` and `--use-default-processor-features` | `--no-use-default-processor-features`
	
	Paramètre de l'API RDS :
	
	`ProcessorFeatures` and `UseDefaultProcessorFeatures`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt a lieu pendant cette modification.	Uniquement Oracle

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

IOPS provisionnés
 	
Valeur d'IOPS provisionnés (opérations d'I/O par seconde) pour l'instance de base de données. Ce paramètre n'est disponible que si vous choisissez l'une des options suivantes pour Storage type (Type de stockage) :
 	
• General purpose SSD (gp3) (SSD à usage général (gp3))
• Provisioned IOPS SSD (io1) (SSD à IOPS provisionnés (io1))
• SSD IOPS provisionné (io2)
 	
Pour plus d'informations, consultez the section called "Stockage sur volumes IOPS provisionnés" et the section called "Stockage GP3 (recommandé)".
	Option de l'interface CLI :
	
	`--iops`
	
	Paramètre de l'API RDS :
	
	`Iops`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Accès public
 	
Publicly accessible (Accessible publiquement) dote l'instance de base de données d'une adresse IP publique, ce qui signifie qu'elle est accessible en dehors du VPC. Pour être accessible au public, l'instance de base de données doit aussi se trouver dans un sous-réseau public du VPC.
 	
Not publicly accessible (Non accessible publiquement) rend l'instance de base de données accessible uniquement à partir de l'intérieur du VPC.
 	
Pour de plus amples informations, veuillez consulter Masquer un(e) instance de base de données dans un VPC depuis Internet.
 	
Pour se connecter à une instance de base de données depuis l'extérieur de son VPC, l'instance de base de données doit être accessible publiquement. En
	Option de l'interface CLI :
	
	`--publicly-accessible | --no-publicly-accessible`
	
	Paramètre de l'API RDS :
	
	`PubliclyAccessible`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
outre, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité de l'instance de base de données. En outre, d'autres exigences doivent être respectées. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS. Si votre instance de base de données n'est pas accessible au public, vous pouvez également utiliser une connexion AWS Site-to-Site VPN ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour de plus amples informati ons, veuillez consulter Confident ialité du trafic inter-réseau.				

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Groupe de sécurité
 	
Groupe de sécurité de VPC que vous voulez associer à l'instance de base de données.
 	
Pour plus d'informations, consultez Contrôle d'accès par groupe de sécurité.
	Option de l'interface CLI :
	
	`--vpc-security-group-ids`
	
	Paramètre de l'API RDS :
	
	`VpcSecurityGroupIds`
		La modification est appliquée de manière asynchrone, dès que possible. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Dimensionnement automatique du stockage
 	
Enable storage autoscaling (Activer le dimensionnement automatique du stockage) permet à Amazon RDS d'augmenter automatiquement l'espace de stockage quand cela est nécessaire pour éviter que votre instance de base de données en manque.
 	
Utilisez Maximum storage threshold (Seuil de stockage maximum) pour définir la limite supérieure au-delà de laquelle Amazon RDS augmente automatiquement l'espace de stockage pour votre instance de base de données. La valeur par défaut est 1 000 GiO.
 	
Pour plus d'informations, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.
	Option de l'interface CLI :
	
	`--max-allocated-storage`
	
	Paramètre de l'API RDS :
	
	`MaxAllocatedStorage`
		La modification a lieu immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Storage throughput (Débit de stockage)
 	
La nouvelle valeur de débit de stockage de l'instance de base de données. Ce paramètre n'est disponible que si vous choisissez General purpose SSD (gp3) (SSD à usage général (gp3)) pour Storage type (Type de stockage).
 	
Pour de plus amples informations, veuillez consulter the section called “Stockage GP3 (recommandé)”.
	Option de l'interface CLI :
	
	`--storage-throughput`
	
	Paramètre de l'API RDS :
	
	`StorageThroughput`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Aucun temps d'arrêt n'a lieu pendant cette modification.	Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Type de stockage
 	
Type de stockage que vous souhaitez utiliser.
 	
Si vous choisissez General Purpose SSD (gp3) (SSD à usage général (gp3)), vous pouvez allouer des Provisioned IOPS (IOPS provisionnés) et un Storage throughput (Débit de stockage) supplémentaires sous Advanced settings (Paramètres avancés).
 	
Si vous choisissez Provisioned IOPS SSD (io1) ou Provisioned IOPS SSD (io2), entrez la valeur Provisioned IOPS.
 	
Une fois qu'Amazon RDS a commencé à modifier votre instance de base de données pour modifier la taille ou le type de stockage, vous ne pouvez pas envoyer d'autre demande de modification de la taille de stockage, des performances ou du type pendant six heures.
	Option de l'interface CLI :
	
	`--storage-type`
	
	Paramètre de l'API RDS :
	
	`StorageType`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Toutes les modifications suivantes entraînent un bref temps d'arrêt au démarrage du processus. Après cette interruption, vous pouvez utiliser votre base de données normalement pendant les modifications.
			
			• De General Purpose (SSD) (Usage général (SSD)) ou Provisioned IOPS (SSD) (IOPS provision
				Tous les moteurs de base de données

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge
Pour de plus amples informations, veuillez consulter Types de stockage Amazon RDS.			nés (SSD)) à Magnetic (Magnétique). • De Magnetic (Magnétique) à General Purpose (SSD) (Usage général (SSD)) ou Provisioned IOPS (SSD) (IOPS provisionnés (SSD)).	

Configuration et description de la console	Option de CLI et paramètre de l'API RDS	Lorsque la modification a lieu	Remarques sur les temps d'arrêt	Moteurs de base de données pris en charge

Groupe de sous-réseaux de base de données
 	
Le groupe de sous-réseaux de base de données pour l'instance de base de données. Vous pouvez utiliser ce paramètre pour déplacer votre instance de base de données vers un autre VPC.
 	
Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.
	Option de l'interface CLI :
	
	`--db-subnet-group-name`
	
	Paramètre de l'API RDS :
	
	`DBSubnetGroupName`
		Si vous choisissez d'appliquer la modification immédiatement, elle se produit immédiatement.
		
		Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
			Un temps d'arrêt a lieu pendant cette modification.	Tous les moteurs de base de données

Entretien d'une instance de base de données

Amazon RDS effectue régulièrement la maintenance des ressources Amazon RDS. Les rubriques suivantes décrivent ces actions de maintenance et expliquent comment les appliquer.

Vue d'ensemble des mises à jour relatives à la maintenance de bases de données

La maintenance implique le plus souvent la mise à jour des ressources suivantes dans votre instance de base de données :

- Matériel sous-jacent
- Système d'exploitation (SE) sous-jacent
- Version du moteur de base de données

Les mises à jour du système d'exploitation se produisent le plus souvent pour des raisons de sécurité. Nous vous recommandons de les faire le plus rapidement possible. Pour plus d'informations sur les mises à jour du système d'exploitation, consultez the section called “Application des mises à jour”.

Rubriques

- Ressources hors ligne lors des mises à jour de maintenance
- Modifications différées d'une
- Cohérence éventuelle de l' DescribePendingMaintenanceActions API

Ressources hors ligne lors des mises à jour de maintenance

Certains éléments de maintenance exigent qu'Amazon RDS mette votre instance de base de données hors ligne pendant un court moment. Parmi les éléments de maintenance qui nécessitent qu'une ressource soit hors ligne figure l'application obligatoire de correctifs au système d'exploitation ou à la base de données. Les mises à jour correctives obligatoires sont planifiées automatiquement uniquement pour les correctifs associés à la sécurité et à la fiabilité de l'instance. Ce type d'application de correctifs est peu fréquent, généralement une fois tous les quelques mois. Cela nécessite rarement plus d'une fraction de votre fenêtre de maintenance.

Modifications différées d'une

Les modifications différées de l'instance de base de données que vous avez choisi de ne pas appliquer immédiatement le sont pendant la fenêtre de maintenance. Par exemple, vous pouvez choisir de modifier la classe ou le groupe de paramètres d'une instance de base de données pendant le créneau de maintenance. Les modifications que vous spécifiez à l'aide du paramètre de redémarrage en attente n'apparaissent pas dans la liste Maintenance en attente. Pour plus d'informations sur la modification d'une instance de base de données , consultez Modification d'une RDS instance de base de données Amazon.

Pour voir les modifications en attente pour la prochaine fenêtre de maintenance, utilisez la describe-db-instances AWS CLI commande et cochez le `PendingModifiedValues` champ.

Cohérence éventuelle de l' DescribePendingMaintenanceActions API

L'`DescribePendingMaintenanceActions`API Amazon RDS suit un modèle de cohérence final. Cela signifie que le résultat de la `DescribePendingMaintenanceActions` commande peut ne pas être immédiatement visible pour toutes les commandes RDS suivantes. Gardez cela à l'esprit lorsque vous l'utilisez `DescribePendingMaintenanceActions` immédiatement après avoir utilisé une commande d'API précédente.

La cohérence éventuelle peut affecter la façon dont vous avez géré vos mises à jour de maintenance. Par exemple, si vous exécutez la `ApplyPendingMaintenanceActions` commande pour mettre à jour la version du moteur de base de données pour un de base de données, elle sera finalement visible par`DescribePendingMaintenanceActions`. Dans ce scénario, cela `DescribePendingMaintenanceActions` peut indiquer que l'action de maintenance n'a pas été appliquée alors qu'elle l'était.

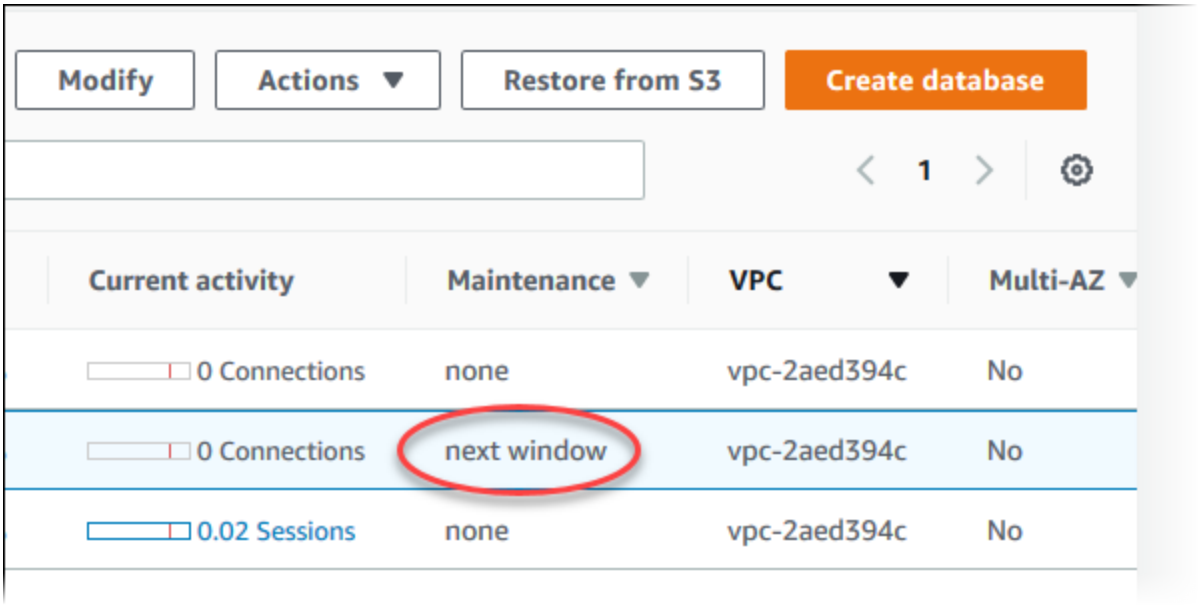
Pour gérer une éventuelle cohérence, vous pouvez effectuer les opérations suivantes :

- Vérifiez l'état du de base de données avant d'exécuter une commande pour le modifier. Exécutez la `DescribePendingMaintenanceActions` commande appropriée à l'aide d'un algorithme de réduction exponentielle afin de laisser suffisamment de temps à la commande précédente pour se propager dans le système. Pour ce faire, exécutez la `DescribePendingMaintenanceActions` commande à plusieurs reprises, en commençant par quelques secondes d'attente, puis en augmentant progressivement jusqu'à cinq minutes.
- Ajoutez un temps d'attente entre les commandes suivantes, même si une `DescribePendingMaintenanceActions` commande renvoie une réponse précise. Appliquez

un algorithme de réduction exponentielle en commençant par quelques secondes d'attente, puis augmentez progressivement jusqu'à environ cinq minutes.

Afficher les mises à jour de maintenance en attente

Vérifiez si une mise à jour de maintenance est disponible pour votre d'instances de base de données à l'aide de la console RDS, de l' AWS CLI API RDS ou de l'API RDS. Si une mise à jour est disponible, elle est indiquée dans la colonne Maintenance du d'instances de base de données sur la console Amazon RDS, comme le montre cette figure.



Si aucune mise à jour de maintenance n'est disponible pour une instance de base de données, la valeur de la colonne est none.

Si une mise à jour de maintenance est disponible pour une instance de base de données, les valeurs de colonne suivantes sont possibles :

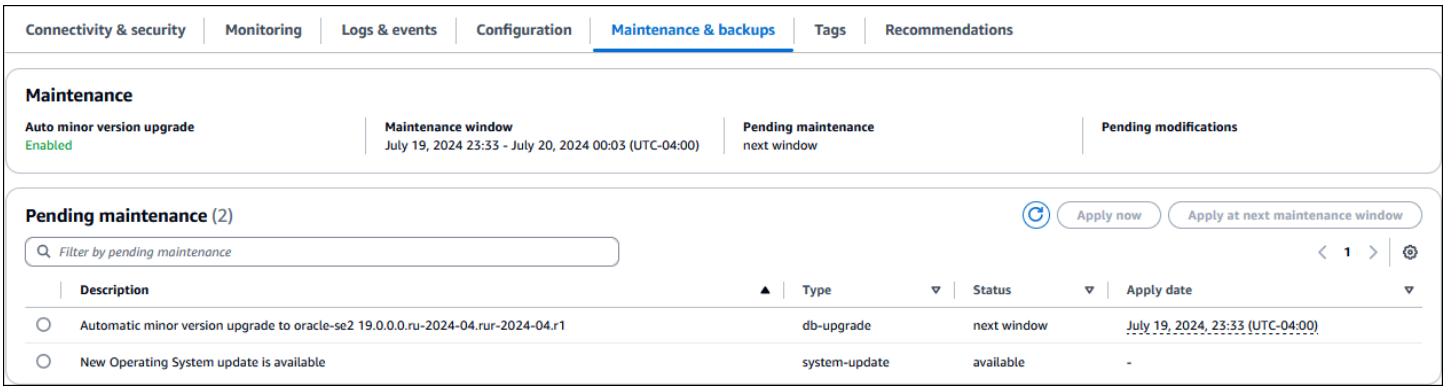
- required (obligatoire) – L'action de maintenance sera appliquée à la ressource et ne peut pas être reportée indéfiniment.
- available – L'action de maintenance est disponible, mais ne sera pas appliquée automatiquement à la ressource. Vous pouvez l'appliquer manuellement.
- next window – L'action de maintenance sera appliquée à la ressource lors de la prochaine fenêtre de maintenance.
- En cours — L'action de maintenance est appliquée à la ressource.

Si une mise à jour est disponible, vous pouvez procéder de l'une des manières suivantes :

- Si la valeur de maintenance correspond à la fenêtre suivante, reportez les actions de maintenance en choisissant Différer la mise à niveau dans Actions. Vous ne pouvez pas différer une action de maintenance qui a déjà commencé.
- Appliquer immédiatement les actions de maintenance.
- Appliquez les actions de maintenance lors de votre prochaine fenêtre de maintenance.
- Ne rien faire.

Pour effectuer une action à l'aide du AWS Management Console

1. Choisissez l'instance de base de données pour afficher ses détails.
2. Choisissez Maintenance et sauvegardes. Les actions de maintenance en attente apparaissent.
3. Choisissez l'action à effectuer, puis choisissez quand l'appliquer.



La fenêtre de maintenance détermine quand les opérations en attente démarrent et ne limite pas la durée d'exécution totale de ces opérations. Il n'est pas garanti que les opérations de maintenance seront terminées avant la fin de la fenêtre de maintenance ; elles peuvent continuer au-delà de l'heure de fin spécifiée. Pour de plus amples informations, veuillez consulter Fenêtre de maintenance Amazon RDS.

Vous pouvez également voir si une mise à jour de maintenance est disponible pour votre d'instances de base de données en exécutant la describe-pending-maintenance-actions AWS CLI commande.

Pour plus d'informations sur l'application des mises à jour de maintenance, consultezAppliquer des mises à jour à un d'instances de base de données.

Actions de maintenance pour Amazon RDS

Les actions de maintenance suivantes s'appliquent aux instances de base de données RDS :

- `ca-certificate-rotation`— Mettez à jour le certificat de l'autorité de certification Amazon RDS pour l'instance de base de données.
- `db-upgrade`— Mettez à niveau la version du moteur de base de données pour l'instance de base de données.
- `hardware-maintenance`— Effectue la maintenance du matériel sous-jacent pour l'instance de base de données.
- `system-update`— Mettez à jour le système d'exploitation de l'instance de base de données.

Maintenance pour les déploiements multi-AZ

L'exécution d'une instance de base de données en tant que déploiement multi-AZ peut encore réduire l'impact d'un événement de maintenance. Il en résulte qu'Amazon RDS applique les mises à jour du système d'exploitation en suivant les étapes suivantes :

1. Réalisation de la maintenance sur l'instance de secours.
2. Promotion de l'instance de secours comme instance principale.
3. Réalisation de la maintenance sur l'ancienne instance principale, qui devient la nouvelle instance de secours.

Si vous mettez à niveau le moteur de base de données pour votre instance de base de données dans un déploiement Multi-AZ, Amazon RDS modifie simultanément les instances de base de données principale et secondaire. Dans ce cas, les deux instances de base de données principale et secondaire dans le déploiement Multi-AZ ne sont pas disponibles pendant la mise à niveau. Cette opération entraîne des temps d'arrêt tant que la mise à niveau n'est pas terminée. La durée du temps d'arrêt varie en fonction de la taille de votre instance de base de données.

Si des correctifs de système d'exploitation sous-jacents doivent être appliqués, un court basculement multi-AZ est nécessaire pour appliquer les correctifs à l'instance de base de données principale. Ce basculement dure généralement moins d'une minute.

Si votre instance de base de données exécute RDS pour MySQL, RDS pour PostgreSQL ou RDS pour MariaDB, vous pouvez minimiser le temps d'arrêt requis pour une mise à niveau en utilisant

un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données. Si vous mettez à niveau une instance de base de données RDS pour SQL Server ou RDS Custom pour SQL Server dans le cadre d'un déploiement multi-AZ, Amazon RDS effectue des mises à niveau progressives, de sorte que vous ne subissez une panne que pendant la durée d'un basculement. Pour de plus amples informations, veuillez consulter Considérations multi-AZ.

Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Fenêtre de maintenance Amazon RDS

La fenêtre de maintenance est un intervalle de temps hebdomadaire pendant lequel les modifications du système sont appliquées. Chaque d'instances de base de données dispose d'une fenêtre de maintenance hebdomadaire. La fenêtre de maintenance est l'occasion de contrôler le moment où les modifications et les correctifs logiciels sont appliqués. Pour plus d'informations sur le réglage de la fenêtre de maintenance, consultez Ajustement du créneau de maintenance préféré pour une instance de base de données.

RDS consomme certaines des ressources de votre instance données pendant les opérations de maintenance. Vous remarquerez peut-être un effet minimal sur les performances. Dans le cas d'une instance de base de données, en de rares occasions, un basculement Multi-AZ peut être requis pour terminer une mise à jour de maintenance.

Si un événement de maintenance est planifié pour une semaine donnée, il est déclenché pendant le créneau de maintenance de 30 minutes que vous identifiez. La plupart des événements de maintenance se terminent également au cours du créneau de maintenance de 30 minutes, mais des événements de maintenance plus importants peuvent prendre plus de 30 minutes. La fenêtre de maintenance est suspendue lorsque le d'instances de base de données est arrêté.

Ce créneau de maintenance de 30 minutes est sélectionné de manière aléatoire sur un bloc horaire de 8 heures par région. Si vous ne spécifiez pas de créneau de maintenance lors de la création de l'instance de base de données, RDS attribue un créneau de maintenance de 30 minutes un jour de semaine aléatoire.

Le tableau suivant indique les intervalles de temps pour chacun Région AWS desquels les fenêtres de maintenance par défaut sont attribuées.

Nom de la région	Région	Bloc chronologique
USA Est (Virginie du Nord)	us-east-1	03:00–11:00 UTC
USA Est (Ohio)	us-east-2	03:00–11:00 UTC
USA Ouest (Californie du Nord)	us-west-1	06:00–14:00 UTC
US West (Oregon)	us-west-2	06:00–14:00 UTC
Africa (Cape Town)	af-south-1	03:00–11:00 UTC
Asie-Pacifique (Hong Kong)	ap us-east-1	06:00–14:00 UTC
Asie-Pacifique (Hyderabad)	ap-south-2	6h30–14h30 UTC
Asie-Pacifique (Jakarta)	ap-southeast-3	08:00–16:00 UTC
Asie-Pacifique (Malaisie)	ap-southeast-5	09H00 — 17H00 UTC
Asie-Pacifique (Melbourne)	ap-southeast-4	11:00–19:00 UTC
Asie-Pacifique (Mumbai)	ap-south-1	06:00–14:00 UTC
Asia Pacific (Osaka)	ap-northeast-3	22:00–23:59 UTC
Asia Pacific (Seoul)	ap-northeast-2	13:00–21:00 UTC
Asia Pacific (Singapore)	ap-southeast-1	14:00–22:00 UTC
Asia Pacific (Sydney)	ap-southeast-2	12:00–20:00 UTC

Nom de la région	Région	Bloc chronologique
Asie-Pacifique (Taipei)	ap-est-2	9H00 À 17H00 UTC
Asie-Pacifique (Thaïlande)	ap-southeast-7	8 H 00 — 16 H UTC
Asie-Pacifique (Tokyo)	ap-northeast-1	13:00–21:00 UTC
Canada (Central)	ca-central-1	03:00–11:00 UTC
Canada Ouest (Calgary)	ca-west-1	18:00–02:00 UTC
Chine (Beijing)	cn-north-1	06:00–14:00 UTC
China (Ningxia)	cn-northwest-1	06:00–14:00 UTC
Europe (Frankfurt)	eu-central-1	21:00–05:00 UTC
Europe (Ireland)	eu-west-1	22:00–06:00 UTC
Europe (London)	eu-west-2	22:00–06:00 UTC
Europe (Milan)	eu-south-1	02:00–10:00 UTC
Europe (Paris)	eu-west-3	23:59–07:29 UTC
Europe (Espagne)	eu-south-2	02:00–10:00 UTC
Europe (Stockholm)	eu-north-1	23:00–07:00 UTC
Europe (Zurich)	eu-central-2	02:00–10:00 UTC
Israël (Tel Aviv)	il-central-1	03:00–11:00 UTC
Mexique (centre)	mx-central-1	19 H 00 — 3 H UTC

Nom de la région	Région	Bloc chronologique
Moyen-Orient (Bahreïn)	me-south-1	06:00–14:00 UTC
Moyen-Orient (EAU)	me-central-1	05:00–13:00 UTC
Amérique du Sud (São Paulo)	sa-east-1	00:00–08:00 UTC
AWS GovCloud (USA Est)	us-gov-east-1	17:00–01:00 UTC
AWS GovCloud (US-Ouest)	us-gov-west-1	06:00–14:00 UTC

Rubriques

- Ajustement du créneau de maintenance préféré pour une instance de base de données

Ajustement du créneau de maintenance préféré pour une instance de base de données

Le créneau de maintenance doit intervenir au moment où l'utilisation est la plus faible et peut donc nécessiter d'être modifié de temps en temps. Votre instance de base de données est indisponible pendant cette période uniquement si des modifications système, telles qu'un changement de classe d'instance de base de données, sont appliquées et nécessitent une interruption de service. Votre instance de base de données n'est pas disponible uniquement pendant le délai minimum requis pour apporter les modifications nécessaires.

Dans l'exemple suivant, vous ajustez le créneau de maintenance préféré pour une instance de base de données.

Pour cet exemple, nous supposons qu'une instance de base données nommée mydbinstance existe et est associée à un créneau de maintenance préféré comme suit : « Sun:05:00-Sun:06:00 » UTC.

console

Pour ajuster le créneau de maintenance préféré

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis sélectionnez l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify. La page Modifier l'instance de base de données s'affiche.
4. Dans la section Maintenance, mettez à jour la fenêtre de maintenance.

   > **Note**
   >
   > Le créneau de maintenance et le créneau de sauvegarde de l'instance de base de données ne peuvent pas se chevaucher. Si la valeur que vous entrez pour le créneau de maintenance chevauche le créneau de sauvegarde, un message d'erreur s'affiche.

5. Choisissez Continuer.

   Sur la page de confirmation, examinez vos modifications.
6. Pour appliquer immédiatement les modifications à la fenêtre de maintenance, sélectionnez Appliquer immédiatement.
7. Choisissez Modifier l'instance de base de données pour enregistrer vos modifications.

   Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour ajuster la fenêtre de maintenance préférée, utilisez la AWS CLI `modify-db-instance`commande avec les paramètres suivants :

- `--db-instance-identifier`
- `--preferred-maintenance-window`

Example

L'exemple de code suivant définit le créneau de maintenance sur Tuesdays from 4:00-4:30AM UTC.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
--db-instance-identifier mydbinstance \
--preferred-maintenance-window Tue:04:00-Tue:04:30
```

Dans Windows :

```
aws rds modify-db-instance ^
--db-instance-identifier mydbinstance ^
--preferred-maintenance-window Tue:04:00-Tue:04:30
```

API RDS

Pour ajuster le créneau de maintenance préféré, utilisez l'opération `ModifyDBInstance` de l'API Amazon RDS avec les paramètres suivants :

- `DBInstanceIdentifier`
- `PreferredMaintenanceWindow`

Appliquer des mises à jour à un d'instances de base de données

Amazon RDS vous permet de choisir le moment d'application des opérations de maintenance. Vous pouvez décider quand Amazon RDS applique les mises à jour en utilisant l' AWS Management Console API AWS CLI, ou RDS.

console

Pour gérer une mise à jour du système d'exploitation pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données pour laquelle une mise à jour obligatoire est disponible.
4. Sous Actions, choisissez une des options suivantes :

   - Corrigez maintenant
   - Patch dans la fenêtre suivante

   > **Note**
   >
   > Si vous choisissez Patch dans la fenêtre suivante et que vous souhaitez ultérieurement retarder la mise à jour, vous pouvez choisir Différer la mise à niveau.
   > Vous ne pouvez pas reporter une action de maintenance en cours.
   > Pour annuler une action de maintenance, modifiez l'instance de base de données et désactivez Mise à niveau automatique des versions mineures.

AWS CLI

Pour appliquer une mise à jour en attente à un d'instances de base de données, utilisez la apply-pending-maintenance-action AWS CLI commande.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds apply-pending-maintenance-action \
```

```
    --resource-identifier arn:aws:rds:us-west-2:001234567890:db:mysql-db \
    --apply-action system-update \
    --opt-in-type immediate
```

Dans Windows :

```
aws rds apply-pending-maintenance-action ^
    --resource-identifier arn:aws:rds:us-west-2:001234567890:db:mysql-db ^
    --apply-action system-update ^
    --opt-in-type immediate
```

> **Note**
>
> Pour différer une action de maintenance, spécifiez `undo-opt-in` pour `--opt-in-type`. Vous ne pouvez pas indiquer`undo-opt-in` pour `--opt-in-type` si l'action de maintenance est en cours.
> Pour annuler une action de maintenance, exécutez la modify-db-instance AWS CLI commande et spécifiez`--no-auto-minor-version-upgrade`.

Pour renvoyer une liste des ressources dont au moins une mise à jour est en attente, utilisez la describe-pending-maintenance-actions AWS CLI commande.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-pending-maintenance-actions \
    --resource-identifier arn:aws:rds:us-west-2:001234567890:db:mysql-db
```

Dans Windows :

```
aws rds describe-pending-maintenance-actions ^
    --resource-identifier arn:aws:rds:us-west-2:001234567890:db:mysql-db
```

Vous pouvez également renvoyer une liste de ressources pour un d'instances de base de données en spécifiant le `--filters` paramètre de la `describe-pending-maintenance-actions` AWS CLI commande. Le format de la commande `--filters` est `Name=`*`filter-name`*`,Value=`*`resource-id`*`,...`.

Les valeurs suivantes sont acceptées pour le paramètre `Name` d'un filtre :

- `db-instance-id`— Accepte une liste d'identifiants d'instance de base de données ou de noms de ressources Amazon (ARNs). La liste renvoyée inclut uniquement les actions de maintenance en attente pour les instances de base de données identifiées par ces identifiants ou ARNs.
- `db-cluster-id`— Accepte une liste d'identifiants de clusters de bases de données ou ARNs pour Amazon Aurora. La liste renvoyée inclut uniquement les actions de maintenance en attente pour les clusters de base de données identifiés par ces identifiants ou ARNs.

Par exemple, l'exemple suivant renvoie les actions de maintenance en attente pour les instances de base de données `sample-instance1` et `sample-instance2`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-pending-maintenance-actions \
 --filters Name=db-instance-id,Values=sample-instance1,sample-instance2
```

Dans Windows :

```
aws rds describe-pending-maintenance-actions ^
 --filters Name=db-instance-id,Values=sample-instance1,sample-instance2
```

API RDS

Pour appliquer une mise à jour à une instance de base de données, appelez l'opération `ApplyPendingMaintenanceAction` de l'API Amazon RDS.

Pour renvoyer la liste des ressources qui possèdent au moins une mise à jour en attente, appelez l'opération d'API Amazon RDS `DescribePendingMaintenanceActions`.

Mises à jour du système d'exploitation pour les instances de base de données RDS

Les instances RDS pour DB2, RDS pour MariaDB, RDS pour MySQL, RDS pour PostgreSQL, RDS pour SQL Server et RDS pour Oracle DB nécessitent parfois des mises à jour du système d'exploitation. Amazon RDS met à niveau le système d'exploitation vers une version plus récente afin

d'améliorer les performances de la base de données et la posture de sécurité globale des clients. En général, les mises à jour prennent environ 10 minutes. Les mises à jour du système d'exploitation ne modifient pas la version du moteur de base de données ou la classe d'instance de base de données d'une instance de base de données.

Les mises à jour du système d'exploitation peuvent être facultatives ou obligatoires :

- Une mise à jour facultative peut être appliquée à tout moment. Bien que ces mises à jour soient facultatives, nous vous recommandons de les appliquer régulièrement pour que votre flotte RDS reste à jour. RDS n'applique pas ces mises à jour automatiquement.

  Pour être averti de la disponibilité d'un nouveau correctif du système d'exploitation facultatif, vous pouvez vous inscrire à RDS-EVENT-0230 dans la catégorie des événements d'application de correctifs de sécurité. Pour obtenir des informations sur l'abonnement à des événements RDS, consultez S'abonner aux notifications d'RDSévénements Amazon.

  > **Note**
  >
  > `RDS-EVENT-0230` ne s'applique pas aux mises à niveau de distribution du système d'exploitation.

- Une mise à jour obligatoire est requise et une date d'application est définie. Prévoyez de planifier la mise à jour avant cette date d'application. Après la date d'application spécifiée, Amazon RDS met automatiquement à niveau le système d'exploitation de l'instance de base de données vers la dernière version au cours de l'une de vos fenêtres de maintenance attribuées.

> **Note**
>
> Vous devrez peut-être appliquer toutes les mises à jour facultatives et obligatoires afin de respecter diverses obligations de conformité. Nous vous recommandons d'appliquer systématiquement toutes les mises à jour qui sont mises à disposition par RDS pendant vos fenêtres de maintenance.

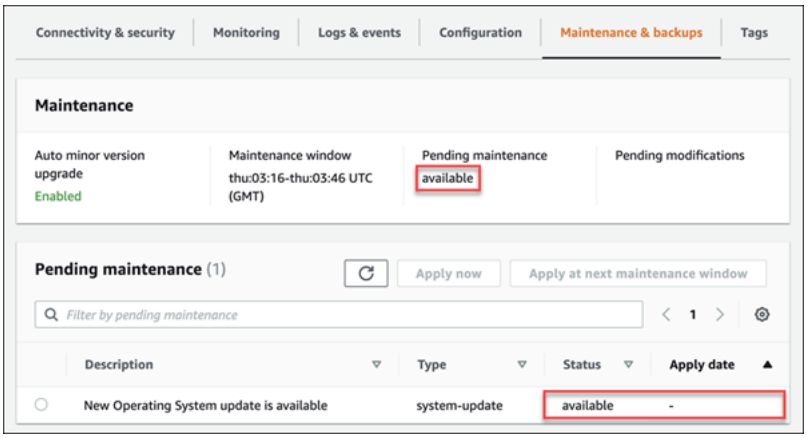
Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour obtenir des informations sur le type de mise à niveau du système d'exploitation.

console

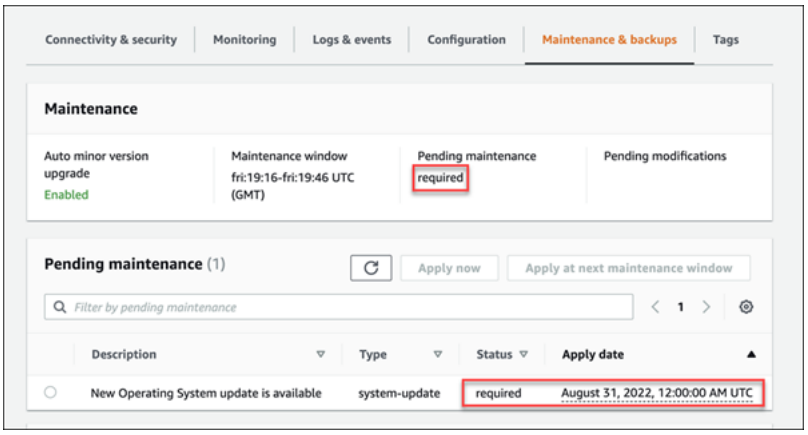
Pour obtenir des informations de mise à jour à l'aide du AWS Management Console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis sélectionnez l'instance de base de données.
3. Choisissez Maintenance et sauvegardes.
4. Dans la section Pending maintenance (Maintenance en attente), recherchez la mise à jour du système d'exploitation et sélectionnez Status (Statut).

Dans le AWS Management Console, le statut de maintenance d'une mise à jour facultative est défini sur disponible et n'a pas de date d'application, comme le montre l'image suivante.



La valeur de maintenance d'une mise à jour obligatoire définie dans Status (Statut) est required (obligatoire) et une date est définie dans Apply date (Date d'application), comme l'illustre l'image suivante.



AWS CLI

Pour obtenir des informations de mise à jour à partir du AWS CLI, utilisez la describe-pending-maintenance-actionscommande.

```
aws rds describe-pending-maintenance-actions
```

Une mise à jour obligatoire du système d'exploitation inclut les valeurs `AutoAppliedAfterDate` et `CurrentApplyDate`. Une mise à jour facultative du système d'exploitation n'inclut pas ces valeurs.

La sortie suivante indique une mise à jour obligatoire du système d'exploitation.

```
{
  "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:db:mydb1",
  "PendingMaintenanceActionDetails": [
    {
      "Action": "system-update",
      "AutoAppliedAfterDate": "2022-08-31T00:00:00+00:00",
      "CurrentApplyDate": "2022-08-31T00:00:00+00:00",
      "Description": "New Operating System update is available"
    }
  ]
}
```

La sortie suivante indique une mise à jour facultative du système d'exploitation.

```
{
  "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:db:mydb2",
  "PendingMaintenanceActionDetails": [
    {
      "Action": "system-update",
      "Description": "New Operating System update is available"
    }
  ]
}
```

Disponibilité des mises à jour du système d'exploitation

Les mises à jour du système d'exploitation sont spécifiques à la version du moteur de base de données et à la classe d'instance de base de données. Par conséquent, les instances de base de données reçoivent ou requièrent des mises à jour à différents moments. Lorsqu'une mise à jour

du système d'exploitation est disponible pour votre instance de base de données en fonction de sa version de moteur et de sa classe d'instance, la mise à jour apparaît dans la console. Il peut également être consulté en exécutant la describe-pending-maintenance-actions AWS CLI commande ou en appelant l'opération de l'API DescribePendingMaintenanceActionsRDS. Si une mise à jour est disponible pour votre instance, vous pouvez mettre à jour le système d'exploitation en suivant les instructions de la section Appliquer des mises à jour à un d'instances de base de données.

Mise à niveau de la version du moteur d'une instance de base de données

Amazon RDS fournit des versions plus récentes de chaque moteur de base de données pris en charge afin que vous puissiez conserver votre up-to-date d'instances de base de données. Ces versions plus récentes peuvent contenir des correctifs de bogues, des améliorations de sécurité et d'autres améliorations pour le moteur de base de données. Lorsque Amazon RDS prend en charge une nouvelle version d'un moteur de base de données, vous pouvez choisir comment et quand mettre à niveau vos instances de base de données.

Il existe deux types de mises à niveau : les mises à niveau de versions majeures et les mises à niveau de versions mineures. En général, une mise à niveau d'une version majeure du moteur peut introduire des modifications non compatibles avec les applications existantes. En revanche, une mise à niveau de version mineure contient uniquement des modifications rétrocompatibles avec les applications existantes.

Pour les clusters de bases de données multi-AZ, les mises à niveau de version majeure sont prises en charge uniquement pour RDS for PostgreSQL. Les mises à niveau de version mineure sont prises en charge pour tous les moteurs qui prennent en charge les clusters de bases de données multi-AZ. Pour de plus amples informations, veuillez consulter the section called “Mise à niveau d'un cluster de base de données multi-AZ”.

La séquence de numérotation des versions est spécifique à chaque moteur de base de données. Par exemple, RDS for MySQL 5.7 et 8.0 sont des versions majeures du moteur et la mise à niveau de la version 5.7 vers la version 8.0 constitue une mise à niveau de version majeure. RDS for MySQL version 5.7.22 et 5.7.23 sont des versions mineures et la mise à niveau de la version 5.7.22 vers la version 5.7.23 constitue une mise à niveau de version mineure.

> **Important**
>
> Vous ne pouvez pas modifier une instance DB lorsqu'elle est en cours de mise à niveau. Lors d'une mise à niveau, le statut de l'instance de base de données est `upgrading`.

Pour plus d'informations sur les mises à niveau de versions majeures et mineures pour un moteur de base de données spécifique, consultez la documentation ci-après correspondant à votre moteur de base de données :

- Mises à niveau du moteur de base de données MariaDB
- Mises à niveau du moteur de base de données Microsoft SQL Server
- Mises à niveau du moteur de base de données RDS pour MySQL
- Mise à niveau du moteur de base de données RDS for Oracle
- Mises à niveau du moteur de base de données RDS pour PostgreSQL

Pour les mises à niveau des versions majeures, vous devez modifier manuellement la version du moteur de base de données via l'API AWS Management Console AWS CLI, ou RDS. Pour les mises à niveau de version mineure, vous pouvez modifier manuellement la version du moteur ou choisir d'activer l'option Mise à niveau automatique des versions mineures.

> **Note**
>
> Les mises à niveau du moteur de base de données nécessitent un temps d'arrêt. Vous pouvez minimiser le temps d'arrêt nécessaire à la mise à niveau de l'instance de base de données en utilisant un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Rubriques

- Mise à niveau manuelle de la version du moteur
- Mise à niveau automatique de la version mineure du moteur

Mise à niveau manuelle de la version du moteur

Pour mettre à niveau manuellement la version du moteur d'une instance de base de données, vous pouvez utiliser l' AWS Management Console API AWS CLI, ou l'API RDS.

console

Pour mettre à niveau la version du moteur d'une instance de base de données à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez mettre à niveau.
3. Sélectionnez Modify. La page Modifier l'instance de base de données s'affiche.
4. Dans le champ Version du moteur de base de données, sélectionnez la nouvelle version.
5. Choisissez Continuer et vérifiez le récapitulatif des modifications.
6. Décidez quand planifier votre mise à niveau :

   - Pour placer les modifications dans la file d'attente des modifications, choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée. Au cours de la fenêtre de maintenance suivante, RDS applique toutes les modifications en attente dans la file d'attente.
   - Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement. La sélection de cette option peut entraîner une interruption de service dans certains cas. Pour plus d'informations, consultez Utilisation du paramètre de modification du calendrier.
7. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modify DB instance (Modifier l'instance de base de données) pour enregistrer vos modifications.

   Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour mettre à niveau la version du moteur d'une instance de base de données, utilisez la modify-db-instancecommande CLI. Spécifiez les paramètres suivants :

- `--db-instance-identifier` – le nom de l'instance de base de données.
- `--engine-version` – numéro de version du moteur de base de données vers lequel effectuer la mise à niveau.

  Pour plus d'informations sur les versions valides du moteur, utilisez la AWS CLI describe-db-engine-versionscommande.
- `--allow-major-version-upgrade` – pour mettre à niveau la version majeure.
- `--no-apply-immediately` – pour appliquer les modifications au cours de la fenêtre de maintenance suivante. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --engine-version new_version \
    --allow-major-version-upgrade \
    --no-apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --engine-version new_version ^
    --allow-major-version-upgrade ^
    --no-apply-immediately
```

API RDS

Pour mettre à niveau la version du moteur d'une instance de base de données, utilisez l'DBInstanceaction Modifier. Spécifiez les paramètres suivants :

- `DBInstanceIdentifier` – nom de l'instance de base de données, par exemple *`mydbinstance`*.
- `EngineVersion` – numéro de version du moteur de base de données vers lequel effectuer la mise à niveau. Pour plus d'informations sur les versions valides du moteur, utilisez l'opération DBEngineDescribe les versions.
- `AllowMajorVersionUpgrade` – pour autoriser une mise à niveau de version majeure. Pour ce faire, définissez la valeur sur `true`.
- `ApplyImmediately` – si des modifications doivent être appliquées immédiatement ou au cours du prochain créneau de maintenance. Pour appliquer les modifications immédiatement, définissez la valeur sur `true`. Pour appliquer les modifications pendant le créneau de maintenance suivant, définissez la valeur sur `false`.

Mise à niveau automatique de la version mineure du moteur

Les mises à niveau automatiques des versions mineures mettent régulièrement à jour votre base de données avec les versions récentes du moteur de base de données. Toutefois, la mise à niveau

n'inclut pas toujours la dernière version du moteur de base de données. Si vous devez conserver des versions spécifiques de vos bases de données à un moment donné, nous vous recommandons de procéder à une mise à niveau manuelle vers les versions de base de données dont vous avez besoin conformément au calendrier requis. En cas de problèmes de sécurité critiques ou lorsqu'une version atteint sa end-of-support date limite, Amazon RDS Amazon peut appliquer une mise à niveau de version mineure même si vous n'avez pas activé l'option de mise à niveau automatique de la version mineure. Pour plus d'informations, consultez la documentation de mise à niveau de votre moteur de base de données spécifique.

- Mises à niveau automatiques des versions mineures RDS pour Postgre SQL
- Mises à niveau automatiques des versions mineures pour RDS for My SQL
- Mises à niveau automatiques des versions mineures RDS pour MariaDB
- Mises à niveau des versions mineures d'Oracle
- Mises à niveau du moteur de base de données Microsoft SQL Server
- Versions de DB2 sur Amazon RDS

Rubriques

- Fonctionnement des mises à niveau automatiques de version mineures
- Activation des mises à niveau automatiques des versions mineures
- Détermination de la disponibilité des mises à niveau de maintenance
- Résultat de cibles de mise à niveau de la version mineure automatiques

Fonctionnement des mises à niveau automatiques de version mineures

La cible de mise à niveau est la version du moteur de base de données vers laquelle Amazon RDS () met à niveau votre base de données. Une version mineure du moteur est désignée comme cible de mise à niveau lorsque les conditions suivantes sont remplies :

- La base de données exécute une version mineure du moteur de base de données inférieure à la version du moteur secondaire cible.

  Vous pouvez trouver la version actuelle du moteur pour votre instance de base de données en consultant l'onglet Configuration de la page de détails de la base de données ou en exécutant la commande CLI`describe-db-instances`.
- La mise à niveau automatique des versions mineures est activée dans la base de données.

RDS planifie la mise à niveau pour qu'elle s'exécute automatiquement dans la fenêtre de maintenance. Au cours de la mise à niveau, RDS effectue les opérations suivantes :

1. Exécute une prévérification du système pour s'assurer que la base de données est saine et prête à être mise à niveau
2. Met à niveau le moteur de base de données vers la version mineure du moteur cible
3. Exécute les contrôles après la mise à niveau
4. Marque la mise à niveau de la base de données comme terminée

Les mises à niveau automatiques entraînent des temps d'arrêt. La durée du temps d'arrêt dépend de différents facteurs, notamment du type de moteur de base de données et de la taille de la base de données.

Activation des mises à niveau automatiques des versions mineures

Vous pouvez vérifier si la mise à niveau automatique des versions mineures est activée pour une instance de base de données lorsque vous effectuez les tâches suivantes :

- Création d'une instance de base de données
- Modification d'une instance de base de données
- Création d'un réplica en lecture
- Restauration d'une instance de base de données à partir d'un instantané
- Restauration d'un instance de base de données à une date déterminée
- Importation d'une instance de base de données à partir de Amazon S3 (pour une sauvegarde MySQL sur Amazon S3)

Lorsque vous effectuez ces tâches, vous pouvez vérifier si la mise à niveau automatique des versions mineures est activée pour l'instance de base de données comme suit :

- À l'aide de la console, définissez l'option Mise à niveau automatique des versions mineures.
- À l'aide AWS CLI de, définissez l'`--auto-minor-version-upgrade|--no-auto-minor-version-upgrade`option.
- À l'aide de l'API RDS, définissez le paramètre `AutoMinorVersionUpgrade`.

Détermination de la disponibilité des mises à niveau de maintenance

Pour déterminer si une mise à jour de maintenance, telle qu'une mise à niveau de version du moteur de base de données, est disponible pour votre d'instances de base de données AWS CLI, vous pouvez utiliser la console ou l'API RDS. Vous pouvez également mettre à niveau la version du moteur de base de données manuellement et ajustez la fenêtre de maintenance. Pour de plus amples informations, veuillez consulter Entretien d'une instance de base de données.

Résultat de cibles de mise à niveau de la version mineure automatiques

Vous pouvez utiliser la AWS CLI commande suivante pour déterminer la version cible de mise à niveau mineure automatique actuelle pour une version mineure du moteur de base de données spécifiée dans un domaine spécifique Région AWS. Vous trouverez les `--engine` valeurs possibles pour cette commande dans la description du `Engine` paramètre dans Create DBInstance.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine engine \
--engine-version minor-version \
--region region \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine engine ^
--engine-version minor-version ^
--region region ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output text
```

Par exemple, la AWS CLI commande suivante détermine la cible de mise à niveau mineure automatique pour la version mineure 8.0.11 de MySQL dans la AWS région USA Est (Ohio) (us-east-2).

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine mysql \
--engine-version 8.0.11 \
--region us-east-2 \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output table
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine mysql ^
--engine-version 8.0.11 ^
--region us-east-2 ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output table
```

Votre sortie est similaire à ce qui suit.

```
---------------------------------
|    DescribeDBEngineVersions    |
+--------------+-----------------+
|  AutoUpgrade |  EngineVersion  |
+--------------+-----------------+
|  False       |  8.0.15         |
|  False       |  8.0.16         |
|  False       |  8.0.17         |
|  False       |  8.0.19         |
|  False       |  8.0.20         |
|  False       |  8.0.21         |
|  True        |  8.0.23         |
|  False       |  8.0.25         |
+--------------+-----------------+
```

Dans cet exemple, la valeur de `AutoUpgrade` est `True` pour MySQL version 8.0.23. Ainsi, la cible de mise à niveau mineure automatique est la version 8.0.23 de MySQL, comme indiqué dans la sortie.

Important

Si vous prévoyez de migrer une instance de base de données RDS for PostgreSQL vers un cluster de bases de données Aurora PostgreSQL dans un avenir proche, nous vous recommandons vivement de désactiver les mises à niveau automatiques mineures de version pour l'instance de base de données tôt dans la phase de planification. La migration vers Aurora PostgreSQL peut être retardée si la version de RDS pour PostgreSQL n'est pas encore prise en charge par Aurora PostgreSQL. Pour plus d'informations sur Aurora PostgreSQL les versions, consultez Versions du moteur pour Amazon Aurora PostgreSQL.

Affectation d'un nouveau nom à une instance DB

Vous pouvez renommer une instance de base de données à l' AWS Management Console aide de la AWS CLI `modify-db-instance` commande ou de l'action d'API `ModifyDBInstance` Amazon RDS. Renommer une instance de base de données peut avoir des effets à grande portée. Vous trouverez ci-dessous une liste de considérations à prendre en compte avant de renommer une instance de base de données.

- Lorsque vous renommez une instance de base de données, le point de terminaison de l'instance de base de données change, parce que l'URL inclut le nom que vous avez attribué à l'instance de base de données. Vous devez toujours rediriger le trafic de l'ancienne URL vers la nouvelle.
- Lorsque vous renommez une instance de base de données, l'ancien nom DNS qui a été utilisé par l'instance de base de données est immédiatement supprimé, même s'il peut demeurer dans le cache quelques minutes. Le nouveau nom DNS de l'instance de base de données renommée devient effectif au bout de 10 minutes environ. L'instance de base de données renommée n'est pas disponible jusqu'à ce que le nouveau nom ne devienne effectif.
- Vous ne pouvez pas utiliser un nom d'instance de base de données existant lorsque vous renommez une instance.
- Tous les réplicas en lecture associés à une instance de base de données demeurent associés à cette instance une fois qu'elle a été renommée. Par exemple, supposons que vous ayez une instance de base de données qui traite votre base de données de production et que l'instance ait plusieurs réplicas en lecture associés. Si vous renommez l'instance de base de données, puis la remplacez dans l'environnement de production par un instantané de base de données, l'instance de base de données que vous avez renommée continue de conserver les réplicas en lecture qui lui sont associés.
- Les métriques et les événements associés au nom d'une instance de base de données sont conservés si vous réutilisez un nom d'instance de base de données. Par exemple, si vous effectuez la promotion d'un réplica en lecture et le renommez avec le nom de l'instance de base de données principale précédente, les événements et les métriques associés à l'instance de base de données principale sont associés à l'instance renommée.
- Les balises de l'instance de base de données demeurent avec l'instance de base de données, quel que soit le changement de nom.
- Les snapshots DB sont conservés pour une instance de base de données renommée.

> **Note**
>
> Une instance de bases de données est un environnement de base de données isolé s'exécutant dans le cloud. Une instance de base de données peut héberger plusieurs bases de données ou une seule base de données Oracle avec plusieurs schémas. Pour plus d'informations sur le changement de nom d'une base de données, consultez la documentation de votre moteur de base de données.

Renommer pour remplacer une instance de base de données existante

Les raisons les plus courantes pour renommer une instance de base de données sont que vous promouvez une réplique en lecture ou que vous restaurez des données à partir d'un instantané ou point-in-time d'une restauration de base de données (PITR). En renommant la base de données, vous pouvez remplacer l'instance de base de données sans avoir à changer un quelconque code d'application qui référence l'instance de base de données. Dans ces cas-là, vous effectuerez les opérations suivantes :

1. Arrêter tout le trafic en direction de l'instance de base de données principale. Cela peut impliquer la redirection du trafic à partir de l'accès aux bases de données sur l'instance de base de données ou toute autre solution que vous voudriez utiliser afin d'empêcher le trafic d'accéder à vos bases de données sur l'instance de base de données.
2. Renommez l'instance de base de données principale avec un nom indiquant qu'elle n'est plus l'instance de base de données principale comme décrit plus loin dans cette rubrique.
3. Créer une instance de base de données principale en la restaurant à partir d'un instantané de base de données ou en promouvant un réplica en lecture, puis attribuer à la nouvelle instance le nom de l'instance de base de données principale précédente.
4. Associer les réplicas en lecture à la nouvelle instance de base de données principale.

Si vous supprimez l'ancienne instance de base de données principale, vous êtes responsable de la suppression des instantanés de base de données indésirables de l'ancienne instance principale.

Pour de plus amples informations sur la promotion d'un réplica en lecture, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.

> **Important**
>
> L'instance de base de données est redémarrée lorsqu'elle est renommée. Pour les instances RDS for SQL Server Multi-AZ dont l'option de mise en miroir est activée AlwaysOn ou dont l'option de mise en miroir est activée, un basculement est attendu lorsque l'instance est redémarrée après l'opération de renommage.

console

Pour renommer une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez renommer.
4. Sélectionnez Modify.
5. Dans Paramètres, saisissez un nouveau nom pour Identifiant de l'instance DB.
6. Choisissez Continuer.
7. Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement. La sélection de cette option peut entraîner une interruption de service dans certains cas. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
8. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modification d'une instance de base de données pour enregistrer vos modifications.

   Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour renommer une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`. Fournissez la valeur `--db-instance-identifier` actuelle et le paramètre `--new-db-instance-identifier` avec le nouveau nom de l'instance de base de données.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier DBInstanceIdentifier \
    --new-db-instance-identifier NewDBInstanceIdentifier
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier DBInstanceIdentifier ^
    --new-db-instance-identifier NewDBInstanceIdentifier
```

API RDS

Pour renommer une instance de base de données, appelez l'opération de l'API Amazon RDS `ModifyDBInstance` avec les paramètres suivants :

- `DBInstanceIdentifier` : nom existant de l'instance
- `NewDBInstanceIdentifier` : nouveau nom de l'instance

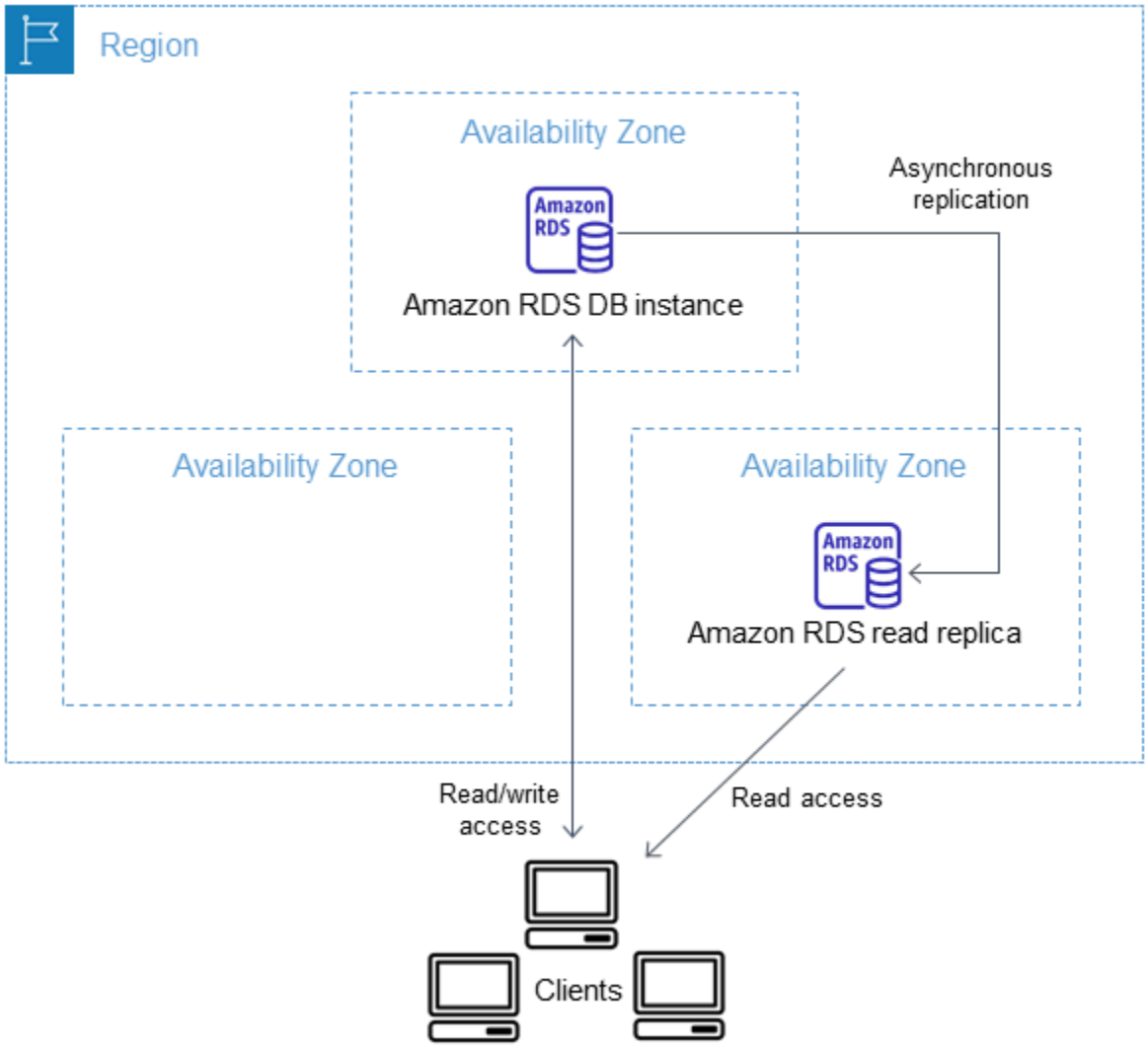
Utilisation des réplicas en lecture d'instance de base de données

Un réplica en lecture est une copie en lecture seule d'une instance de base de données. Vous pouvez réduire la charge sur votre instance de base de données principale en acheminant les requêtes depuis vos applications vers le réplica en lecture. Ainsi, vous pouvez effectuer une montée en puissance élastique au-delà des contraintes de capacité d'une seule instance de base de données dans le cas de charges de travail de base de données à lecture intensive.

Pour créer un réplica en lecture à partir d'une instance de base de données source, Amazon RDS utilise les fonctions de réplication intégrées du moteur de base de données. Pour plus d'informations sur l'utilisation de réplicas en lecture avec un moteur spécifique, veuillez consulter les sections suivantes :

- Utilisation de répliques pour Amazon RDS pour DB2
- Utilisation de réplicas en lecture MariaDB
- Utilisation des réplicas en lecture pour Microsoft SQL Server dans Amazon RDS
- Travailler avec les répliques de My SQL read
- Utilisation de réplicas en lecture pour Amazon RDS for Oracle
- Utilisation de réplicas en lecture pour Amazon RDS for PostgreSQL

Après que vous avez créé un réplica en lecture à partir d'une instance de base de données source, la source devient l'instance de base de données principale. Lorsque vous apportez des mises à jour à l'instance de base de données principale, Amazon RDS les copie de manière asynchrone vers le réplica en lecture. Le schéma suivant montre une instance de base de données source effectuant la réplication vers un réplica en lecture dans une zone de disponibilité (AZ) différente. Les clients ont read/write accès à l'instance de base de données principale et ont un accès en lecture seule à la réplique.



Les répliques de lecture sont facturées en tant qu'instances de base de données standard aux mêmes taux que la classe d'instance de base de données utilisée pour la réplique. Le transfert de données effectué lors de la réplication de données entre l'instance de base de données source et une réplique en lecture au sein de celle-ci Région AWS ne vous est pas facturé. Pour plus d'informations, consultez Coûts de la réplication entre régions et Facturation d'instance de base de données pour Amazon.

Rubriques

- Présentation des réplicas en lecture Amazon RDS
- Différences entre les répliques de lecture pour les moteurs de base de données
- Création d'un réplica en lecture
- Promotion d'un réplica en lecture en instance de bases de données autonome
- Supervision de la réplication en lecture

- Création d'un réplica en lecture dans une autre Région AWS

Présentation des réplicas en lecture Amazon RDS

Les sections ci-dessous traitent des réplicas en lecture d'une instance de base de données. Pour plus d'informations sur les réplicas en lecture d'un cluster de bases de données multi-AZ, consultez the section called “Utilisation des réplicas en lecture d'un cluster de base de données multi-AZ”.

Rubriques

- Cas d'utilisation pour les réplicas en lecture
- Fonctionnement des réplicas en lecture
- Réplicas en lecture dans un déploiement multi-AZ
- Réplicas en lecture entre Régions
- Types de stockage de réplica en lecture
- Restrictions relatives à la création d'un réplica à partir d'un réplica
- Considérations relatives à la suppression de réplicas

Cas d'utilisation pour les réplicas en lecture

Le déploiement d'un ou de plusieurs réplicas en lecture pour une instance de bases de données source donnée peut être judicieux dans divers scénarios, notamment dans les suivants :

- Extensibilité au-delà du calcul ou de la I/O capacité d'une seule instance de base de données pour les charges de travail de base de données gourmandes en lecture. Vous pouvez diriger ce trafic en lecture excessif vers un ou plusieurs réplicas en lecture.
- Service du trafic en lecture alors que l'instance de bases de données source est indisponible. Dans certains cas, votre instance de base de données source peut ne pas être en mesure de répondre aux I/O demandes, par exemple en raison d'une suspension des E/S pour les sauvegardes ou d'une maintenance planifiée. Vous pouvez alors diriger le trafic de lecture vers vos réplicas en lecture. Dans ce cas d'utilisation, gardez à l'esprit que les données sur le réplica en lecture peuvent être « périmées » car l'instance de bases de données source est indisponible.
- Scénarios de création de rapports commerciaux ou d'entreposage de données, dans lesquels vous pouvez souhaiter que les requêtes de rapports commerciaux s'exécutent sur un réplica en lecture, plutôt que sur votre instance de bases de données de production.

- Mise en œuvre de la reprise après sinistre. Vous pouvez effectuez la promotion d'un réplica en lecture en instance autonome comme plan de reprise après sinistre en cas de défaillance de l'instance de base de données principale.

Fonctionnement des réplicas en lecture

Lorsque vous créez une réplique en lecture, vous spécifiez une instance de base de données existante comme source. Ensuite, Amazon RDS prend un instantané de l'instance source et crée une instance en lecture seule à partir de celui-ci. Amazon RDS utilise la méthode de réplication asynchrone pour le moteur de base de données afin de mettre à jour le réplica en lecture chaque fois qu'une modification est apportée à l'instance de base de données principale.

La réplique en lecture fonctionne comme une instance de base de données qui autorise uniquement les connexions en lecture seule. Les exceptions sont le RDS pour DB2 et le RDS pour les moteurs de base de données Oracle, qui prennent en charge les répliques de bases de données en mode veille et en mode monté, respectivement. Une réplique en veille et une réplique montée n'acceptent pas les connexions utilisateur et ne peuvent donc pas prendre en charge les charges de travail en lecture seule. Les répliques de secours et les répliques montées sont principalement utilisées pour la reprise après sinistre entre régions. Pour plus d’informations, consultez Utilisation de répliques pour Amazon RDS pour DB2 et Utilisation de réplicas en lecture pour Amazon RDS for Oracle.

Les applications se connectent à un réplica en lecture de la même façon qu'à toute instance de base de données. Amazon RDS réplique toutes les bases de données à partir de l'instance de base de données source.

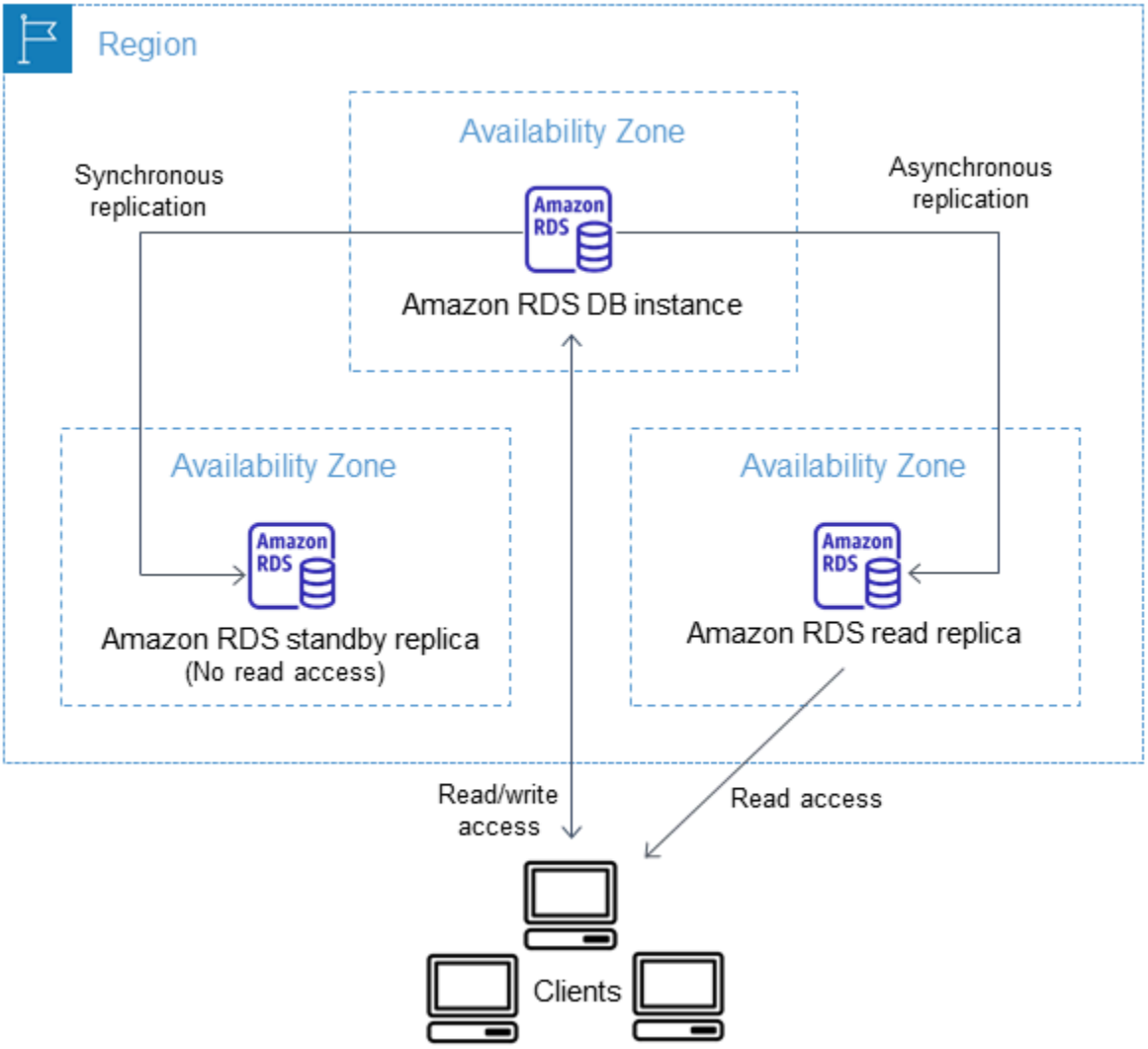
Vous devez créer manuellement des répliques de lecture. RDS ne prend pas en charge le dimensionnement automatique des répliques de lecture, qui consiste à ajouter ou à supprimer automatiquement des répliques de lecture lorsque la demande de lecture change.

Réplicas en lecture dans un déploiement multi-AZ

Vous pouvez configurer un réplica en lecture pour une instance de base de données disposant également d'un réplica de secours configuré pour la haute disponibilité dans un déploiement multi-AZ. La réplication avec le réplica de secours est synchrone. Contrairement à un réplica en lecture, un réplica de secours ne peut pas servir au trafic de lecture.

Dans le scénario suivant, les clients ont read/write accès à une instance de base de données principale dans une zone de disponibilité. L'instance principale copie les mises à jour de manière asynchrone vers un réplica en lecture dans une deuxième zone de disponibilité et les copie

également de manière synchrone vers un réplica de secours dans une troisième zone de disponibilité. Les clients possèdent un accès en lecture uniquement au réplica en lecture.



Pour plus d'informations sur les réplicas de secours configurés pour une haute disponibilité, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Réplicas en lecture entre Régions

Dans certains cas, une réplique en lecture réside dans une instance de base de données Région AWS différente de son instance de base de données principale. Dans ces cas, Amazon RDS configure un canal de communication sécurisé entre l'instance de base de données principale et le réplica en lecture. Amazon RDS établit toutes les configurations AWS de sécurité nécessaires pour activer le canal sécurisé, telles que l'ajout d'entrées de groupe de sécurité. Pour plus d'informations sur les réplicas en lecture entre régions, veuillez consulter Création d'un réplica en lecture dans une autre Région AWS.

Les informations de ce chapitre s'appliquent à la création de répliques de lecture Amazon RDS, soit dans la même Région AWS instance de base de données source, soit dans une instance séparée. Région AWS Les informations suivantes ne s'appliquent pas à la configuration de la réplication avec une instance exécutée sur une EC2 instance Amazon ou sur site.

Types de stockage de réplica en lecture

Par défaut, un réplica en lecture est créé avec le même type de stockage que l'instance de bases de données source. Toutefois, vous pouvez créer un réplica en lecture disposant d'un autre type de stockage que l'instance de bases de données source, en fonction des options répertoriées dans le tableau suivant.

Type de stockage de l'instance de bases de données source	Allocation de stockage d'instance de bases de données source	Options de type de stockage du réplica en lecture
IOPS provisionnés	100 Gio–64 Tio	IOPS provisionnés, usage général, magnétique
Usage général	100 Gio–64 Tio	IOPS provisionnés, usage général, magnétique
Usage général	<100 Gio	Usage général, magnétique
Magnétique	100 Gio - 6 Tio	IOPS provisionnés, usage général, magnétique
Magnétique	<100 Gio	Usage général, magnétique

> **Note**
>
> Lorsque vous augmentez le stockage alloué d'un réplica en lecture, il doit être d'au moins 10 %. Si vous tentez d'augmenter la valeur de moins de 10 %, une erreur s'affiche.

Restrictions relatives à la création d'un réplica à partir d'un réplica

Amazon RDS ne prend pas en charge la réplication circulaire. Vous ne pouvez pas configurer une instance de base de données afin de l'utiliser comme source de réplication pour une instance de base de données existante. Vous pouvez uniquement créer un nouveau réplica en lecture à partir d'une instance de base de données existante. Par exemple, si **`MySourceDBInstance`** est répliqué sur **`ReadReplica1`**, vous ne pouvez pas configurer **`ReadReplica1`** de façon à ce qu'il soit répliqué sur **`MySourceDBInstance`**.

Pour RDS for MariaDB et RDS for MySQL, et pour certaines versions de RDS for PostgreSQL, vous pouvez créer un réplica en lecture à partir d'un réplica en lecture existant. Par exemple, vous pouvez créer un nouveau réplica en lecture **`ReadReplica2`** à partir d'un réplica **`ReadReplica1`** existant. Pour RDS pour DB2, RDS pour Oracle et RDS pour SQL Server, vous ne pouvez pas créer de réplique en lecture à partir d'une réplique en lecture existante.

Considérations relatives à la suppression de réplicas

RDS ne prend pas en charge le dimensionnement automatique des répliques de lecture. Ainsi, RDS n'augmentera pas le nombre de répliques lorsque la demande augmente ou ne diminuera pas le nombre de répliques lorsque la demande diminue. Si vous n'avez plus besoin de répliques de lecture, supprimez-les manuellement en utilisant les mêmes mécanismes que ceux utilisés pour supprimer une instance de base de données. Si vous supprimez une instance de base de données source sans supprimer ses répliques de lecture Région AWS, chaque réplique est promue en instance de base de données autonome.

Pour plus d'informations sur la création d'une instance de base de données, veuillez consulter Suppression d'une instance DB. Pour plus d'informations sur la promotion d'un réplica en lecture, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome. Pour plus d'informations sur la suppression de l'instance de base de données source pour une réplique de lecture entre régions, consultezConsidérations liées à la réplication entre régions.

Différences entre les répliques de lecture pour les moteurs de base de données

Étant donné que les moteurs de base de données Amazon RDS implémentent la réplication différemment, il existe plusieurs différences importantes que vous devez connaître.

Db2

Les répliques pour RDS pour Db2 présentent les fonctionnalités et comportements suivants :

- Méthode de réplication — Réplication physique.
- Purge des journaux de transactions — RDS pour DB2 purge les journaux de l'instance de base de données principale lorsque les conditions suivantes sont remplies :
  - Les journaux datent d'au moins deux heures.
  - Le paramètre relatif aux heures de conservation des journaux d'archivage est dépassé.
  - RDS for Db2 a correctement répliqué les journaux sur toutes les instances de base de données répliquées.

  Cela s'applique à la fois aux mêmes Région AWS instances de base de données et aux instances de base de données interrégionales. Pour plus d'informations sur la définition des heures de conservation des journaux d'archivage, veuillez consulter rdsadmin.set_archive_log_retention.
- Répliques inscriptibles — Une réplique Db2 est une copie physique, et Db2 n'autorise pas les écritures dans une réplique. Vous pouvez promouvoir la réplique pour la rendre inscriptible. La réplique promue contient les données répliquées au moment où la demande de promotion a été faite.
- Sauvegardes — Les sauvegardes automatiques et les instantanés manuels sont pris en charge sur RDS pour les répliques DB2.
- Réplication parallèle : les données du journal d'archivage sont toujours transmises en parallèle de la base de données principale à toutes ses répliques.
- État de veille : les répliques de secours sont principalement utilisées pour la reprise après sinistre entre régions. Pour plus d'informations, consultez Utilisation de répliques pour Amazon RDS pour DB2.

MariaDB et MySQL

Les répliques de lecture pour RDS pour MariaDB et RDS pour MySQL présentent les caractéristiques et comportements suivants :

- Méthode de réplication — Réplication logique.
- Purge des journaux de transactions — RDS pour MariaDB et RDS pour MySQL conservent tous les journaux binaires qui n'ont pas été appliqués.

- Répliques inscriptibles — Vous pouvez activer la réplique en lecture MariaDB ou MySQL pour qu'elle soit accessible en écriture.
- Sauvegardes — Les sauvegardes automatiques et les instantanés manuels sont pris en charge sur les répliques de lecture RDS pour MariaDB ou RDS pour MySQL.
- Réplication parallèle — Toutes les versions de MariaDB et MySQL prises en charge autorisent les threads de réplication parallèles.
- État monté : non pris en charge.

Oracle

Les répliques de lecture pour RDS pour Oracle présentent les fonctionnalités et comportements suivants :

- Méthode de réplication — Réplication physique.
- Purge des journaux de transactions : si une instance de base de données principale ne possède pas de répliques de lecture interrégionales, Amazon RDS for Oracle conserve un minimum de deux heures de journaux de transactions sur l'instance de base de données source. Les journaux sont purgés de l'instance de base de données source au bout de deux heures ou une fois que le paramètre d'heures de conservation du journal d'archive est passé, le plus long des deux. Les journaux sont purgés du réplica en lecture une fois que le paramètre d'heures de conservation du journal d'archive est passé, uniquement s'ils ont été appliqués correctement à la base de données.

  Dans certains cas, une instance de base de données principale peut avoir un ou plusieurs réplicas en lecture entre régions. Dans ce cas, Amazon RDS for Oracle conserve les journaux de transaction sur l'instance de base de données source jusqu'à ce qu'ils aient été transmis et appliqués à tous les réplicas en lecture entre régions.

  Pour plus d'informations sur la définition des heures de conservation des journaux d'archivage, veuillez consulter Conservation des journaux redo archivés.
- Répliques inscriptibles : une réplique en lecture Oracle est une copie physique, et Oracle n'autorise pas les écritures dans une réplique en lecture. Vous pouvez promouvoir un réplica en lecture afin de le rendre inscriptible. Le réplica en lecture promu a les données répliquées jusqu'au moment où la demande a été faite pour le promouvoir.
- Sauvegardes : les sauvegardes automatiques et les instantanés manuels sont pris en charge sur RDS pour les répliques de lecture Oracle.

- Réplication parallèle — Les données du Redo Log sont toujours transmises en parallèle de la base de données principale à toutes ses répliques en lecture.
- État monté : les répliques montées sont principalement utilisées pour la reprise après sinistre entre régions. Une licence Active Data Guard n'est pas requise pour les réplicas montés. Pour de plus amples informations, veuillez consulter Utilisation de réplicas en lecture pour Amazon RDS for Oracle.

PostgreSQL

Les répliques de lecture pour RDS pour PostgreSQL présentent les fonctionnalités et comportements suivants :

- Méthode de réplication — Réplication physique.
- Purge des journaux de transactions : PostgreSQL dispose d'un `wal_keep_segments` paramètre qui détermine le nombre de fichiers journaux d'écriture anticipée (WAL) à conserver pour fournir des données aux répliques en lecture. La valeur de ce paramètre spécifie le nombre de journaux à conserver.
- Répliques inscriptibles : une réplique en lecture de PostgreSQL est une copie physique, et PostgreSQL ne permet pas de rendre une réplique en lecture accessible en écriture.
- Sauvegardes : les instantanés manuels sont pris en charge pour les répliques de lecture RDS pour PostgreSQL. Les sauvegardes automatiques pour les réplicas en lecture sont prises en charge pour RDS for PostgreSQL 14.1 et versions ultérieures uniquement. Vous ne pouvez pas activer les sauvegardes automatiques pour les réplicas en lecture PostgreSQL pour les versions de RDS for PostgreSQL antérieures à 14.1. Pour RDS for PostgreSQL 13 et versions antérieures, créez un instantané à partir d'un réplica en lecture si vous souhaitez en obtenir une sauvegarde.
- Réplication parallèle : PostgreSQL utilise un seul processus pour gérer la réplication.
- État monté : non pris en charge.

SQL Server

Les répliques de lecture pour RDS pour SQL Server présentent les fonctionnalités et comportements suivants :

- Méthode de réplication — Réplication physique.

- Purge des journaux de transactions — Le fichier journal virtuel (VLF) du journal des transactions du réplica principal peut être tronqué une fois qu'il n'est plus nécessaire pour les répliques secondaires.

  Le VLF ne peut être marqué comme inactif que lorsque les enregistrements de journaux ont été sécurisés dans les réplicas. Quelle que soit la rapidité avec laquelle les sous-systèmes de disque se trouvent dans la réplique principale, le journal des transactions le conservera VLFs jusqu'à ce que la réplique la plus lente l'ait durcie.
- Répliques inscriptibles : une réplique en lecture de SQL Server est une copie physique et n'autorise pas non plus les écritures. Vous pouvez promouvoir un réplica en lecture afin de le rendre inscriptible. Le réplica en lecture promu a les données répliquées jusqu'au moment où la demande a été faite pour le promouvoir.
- Sauvegardes : les sauvegardes automatiques et les instantanés manuels ne sont pas pris en charge sur les répliques en lecture de RDS pour SQL Server.
- Réplication parallèle — Les données du Redo Log sont toujours transmises en parallèle de la base de données principale à toutes ses répliques en lecture.
- État monté : non pris en charge.

Création d'un réplica en lecture

Vous pouvez créer une réplique de lecture à partir d'une instance de base de données existante à l'aide de l'API AWS Management Console AWS CLI, ou RDS. Vous créez un réplica en lecture en spécifiant `SourceDBInstanceIdentifier`, qui est l'identifiant de l'instance de base de données source à partir de laquelle vous souhaitez répliquer les données.

Lorsque vous créez un réplica en lecture, Amazon RDS prend un instantané de votre instance de base de données source et commence la réplication. L'instance de base de données source subit une très brève I/O suspension lorsque l'opération de capture instantanée de base de données commence. La I/O suspension dure généralement une seconde environ. Vous pouvez éviter la I/O suspension si l'instance de base de données source est un déploiement multi-AZ, car dans ce cas, le snapshot est pris à partir de l'instance de base de données secondaire.

Une transaction de longue durée active peut ralentir le processus de création du réplica en lecture. Nous vous recommandons d'attendre que les transactions de longue durée se terminent pour créer un réplica en lecture. Si vous créez plusieurs réplicas en lecture en parallèle à partir de la même instance de base de données source, Amazon RDS prend un seul instantané au début de la première action de création.

Lors de la création d'un réplica en lecture, il convient de prendre en considération plusieurs éléments. Tout d'abord, vous devez activer les sauvegardes automatiques sur l'instance de bases de données source en affectant à la période de rétention des sauvegardes une valeur différente de 0. Cette exigence s'applique également à un réplica en lecture qui serait l'instance de base de données source d'un autre réplica en lecture. Pour activer les sauvegardes automatiques sur un réplica en lecture RDS for MySQL, commencez par créer le réplica en lecture, puis modifiez-le pour activer les sauvegardes automatiques.

> **Note**
>
> Dans un Région AWS, nous vous recommandons vivement de créer toutes les répliques de lecture dans le même cloud privé virtuel (VPC) basé sur Amazon VPC en tant qu'instance de base de données source. Si vous créez un réplica en lecture dans un VPC différent de l'instance de base de données source, les plages d'adresses CIDR (classless inter-domain routing) peuvent se chevaucher entre le réplica et le système RDS. Le chevauchement CIDR rend le réplica instable, ce qui peut avoir un impact négatif sur les applications qui s'y connectent. Si vous recevez une erreur lors de la création du réplica en lecture, choisissez un autre groupe de sous-réseaux de base de données de destination. Pour de plus amples informations, veuillez consulter Utilisation d'un(e) instance de base de données dans un VPC. Il n'existe aucun moyen direct de créer une réplique de lecture dans un autre à Compte AWS l'aide de la console ou AWS CLI.

console

Pour créer un réplica en lecture à partir d'une instance de base de données source

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez l'instance de base de données que vous voulez utiliser comme source pour votre réplica en lecture.
4. Sous Actions, choisissez Créer des réplicas en lecture.
5. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture.
6. Choisissez la configuration de votre instance. Nous vous recommandons d'utiliser un type de stockage et une classe d'instances de base de données identiques ou supérieurs à l'instance de base de données source pour le réplica en lecture.

7. Pour Région AWS, spécifiez la région de destination du réplica en lecture.
8. Pour Stockage, spécifiez la taille de stockage allouée et si vous souhaitez utiliser la mise à l'échelle automatique du stockage.

   Si votre instance de base de données source n'utilise pas la dernière configuration de stockage, l'option Mettre à niveau la configuration du système de fichiers de stockage est disponible. Vous pouvez activer ce paramètre pour mettre à niveau le système de fichiers de stockage du réplica en lecture vers la configuration préférée. Pour de plus amples informations, veuillez consulter the section called “Mise à niveau du système de fichiers de stockage”.
9. Pour Disponibilité, choisissez si vous voulez créer une instance de secours de votre réplica dans une autre zone de disponibilité pour prendre en charge le basculement pour ce réplica.

   > **Note**
   >
   > La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.
10. Spécifiez d'autres paramètres d'instance de base de données. Pour obtenir des informations sur chaque paramètre disponible, consultez Paramètres des instances de base de données.
11. Pour créer un réplica en lecture chiffré, développez Configuration supplémentaire et spécifiez les paramètres suivants :

    a. Choisissez Activer le chiffrement.

    b. Pour AWS KMS key, choisissez l'identificateur AWS KMS key de la clé KMS.

    > **Note**
    >
    > L'instance de base de données source doit être chiffrée. Pour en savoir plus sur le chiffrement de l'instance de bases de données source, consultez Chiffrement des ressources Amazon RDS.
12. Choisissez Créer un réplica en lecture.

Une fois le réplica en lecture créé, vous pouvez le voir sur la page Bases de données de la console RDS. Il affiche le réplica dans la colonne Rôle .

AWS CLI

Pour créer une réplique en lecture à partir d'une instance de base de données source, utilisez la AWS CLI commande create-db-instance-read-replica. Cet exemple définit également la taille de stockage allouée, active la mise à l'échelle automatique du stockage et met à niveau le système de fichiers vers la configuration préférée.

Vous pouvez spécifier d'autres paramètres. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

> **Note**
>
> Pour créer une réplique de secours RDS pour DB2, incluez dans la AWS CLI commande suivante. `--replica-mode mounted`

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier myreadreplica \
    --source-db-instance-identifier mydbinstance \
    --allocated-storage 100 \
    --max-allocated-storage 1000 \
    --upgrade-storage-config
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier myreadreplica ^
    --source-db-instance-identifier mydbinstance ^
    --allocated-storage 100 ^
    --max-allocated-storage 1000 ^
    --upgrade-storage-config
```

API RDS

Pour créer une réplique en lecture à partir d'une instance de base de données MySQL, MariaDB, Oracle, PostgreSQL ou SQL Server source, appelez l'opération Amazon RDS API Create avec les paramètres requis suivants : DBInstance ReadReplica

- `DBInstanceIdentifier`
- `SourceDBInstanceIdentifier`

> **Note**
>
> Pour créer une réplique de secours RDS pour DB2, définissez l'`ReplicaMode`opération facultative sur. `mounted`

Promotion d'un réplica en lecture en instance de bases de données autonome

Vous pouvez promouvoir un réplica en lecture an tant qu'une instance de base de données autonome. Si une instance de base de données source possède plusieurs réplicas en lecture, la promotion d'un des réplicas en lecture en instance de base de données n'a aucun effet sur les autres réplicas.

Lorsque vous promouvez une réplique en lecture, RDS redémarre l'instance de base de données avant de la rendre disponible. Le processus de promotion peut prendre plusieurs minutes ou plus longtemps, selon la taille du réplica en lecture.



Cas d'utilisation pour promouvoir une réplique lue

Vous souhaiterez peut-être promouvoir une réplique en lecture vers une instance de base de données autonome pour l'une des raisons suivantes :

- Implémentation d'une récupération en cas de défaillance – Vous pouvez utiliser la promotion de réplica en lecture comme plan de récupération de données en cas de défaillance de l'instance de base de données principale. Cette approche complète la réplication synchrone, la détection automatique d'échec et le basculement.

Si vous êtes conscient des ramifications et des limitations de la réplication asynchrone et que vous souhaitez toujours utiliser la promotion de réplica en lecture pour la récupération des données, vous pouvez le faire. Pour cela, commencez par créer un réplica en lecture, puis surveillez l'instance de base de données principale pour détecter les pannes. En cas de panne, procédez comme suit :

1. Promouvez le réplica en lecture.
2. Dirigez le trafic de base de données vers l'instance de bases de données promue.
3. Créez un réplica en lecture de remplacement avec l'instance de base de données promue comme source.

- Mise à niveau de la configuration de stockage : si votre instance de base de données source ne correspond pas à la configuration de stockage préférée, vous pouvez créer un réplica en lecture de l'instance et mettre à niveau la configuration du système de fichiers de stockage. Cette option migre le système de fichiers du réplica en lecture vers la configuration préférée. Vous pouvez ensuite promouvoir le réplica en lecture en tant qu'instance autonome.

  Vous pouvez utiliser cette option pour surmonter les limitations de mise à l'échelle en matière de stockage et de taille de fichier pour les anciens systèmes de fichiers 32 bits. Pour de plus amples informations, veuillez consulter the section called “Mise à niveau du système de fichiers de stockage”.

  Cette option n'est disponible que si votre instance de base de données source ne possède pas la dernière configuration de stockage ou si vous modifiez la classe d'instance de base de données dans la même demande.

- Partitionnement – Le partitionnement incarne l'architecture « ne rien partager » et implique essentiellement la décomposition d'une grande base de données en plusieurs bases de données plus petites. Pour fractionner une base de données, l'une des méthodes consiste à fractionner les tables qui ne sont pas jointes dans la même requête sur différents hôtes. L'autre méthode consiste à dupliquer une table entre plusieurs hôtes, puis à utiliser un algorithme de hachage pour déterminer l'hôte qui reçoit une mise à jour donnée. Vous pouvez créer des réplicas en lecture correspondant à chacune de vos partitions (bases de données plus petites) et les promouvoir quand vous décidez de les convertir en partitions autonomes. Vous pouvez alors extraire l'espace clé (si vous fractionnez les lignes) ou la distribution des tables pour chaque partitionnement selon vos besoins.
- Exécution d'opérations DDL (MySQL et MariaDB uniquement) – Les opérations DDL, telles que la création ou la reconstruction d'index, peuvent prendre du temps et imposer une pénalité importante

de performances à votre instance de base de données. Vous pouvez exécuter ces opérations sur un réplica en lecture MySQL ou MariaDB une fois que ce réplica en lecture est synchronisé avec son instance de bases de données principale. Ensuite, vous pouvez promouvoir le réplica en lecture et indiquer à vos applications d'utiliser l'instance promue.

> **Note**
>
> Si votre réplique de lecture est une instance de base de données RDS pour Oracle, vous pouvez effectuer un changement au lieu d'une promotion. Lors d'un basculement, l'instance de base de données source devient la nouvelle réplique, et la réplique devient la nouvelle instance de base de données source. Pour de plus amples informations, veuillez consulter Exécution d'un basculement d'Oracle Data Guard.

Caractéristiques d'une réplique de lecture sponsorisée

Une fois que vous avez promu la réplique en lecture, elle cesse de fonctionner en tant que réplique en lecture et devient une instance de base de données autonome. La nouvelle instance de base de données autonome présente les caractéristiques suivantes :

- L'instance de base de données autonome conserve le groupe d'options et le groupe de paramètres de la réplique de lecture avant la promotion.
- Vous pouvez créer des répliques de lecture à partir de l'instance de base de données autonome et effectuer des opérations de point-in-time restauration.
- Vous ne pouvez pas utiliser l'instance de base de données comme cible de réplication car il ne s'agit plus d'une réplique en lecture.

Conditions préalables à la promotion d'une réplique lue

Avant de promouvoir une réplique lue, procédez comme suit :

- Passez en revue votre stratégie de sauvegarde :
  - Nous vous recommandons d'activer les sauvegardes et d'effectuer au moins une sauvegarde. La durée de la sauvegarde dépend du nombre de modifications apportées à la base de données depuis la dernière sauvegarde.

  - Si vous avez activé des sauvegardes sur votre réplica en lecture, configurez la fenêtre de sauvegarde automatique afin que les sauvegardes quotidiennes n'interfèrent pas avec la promotion du réplica en lecture.
  - Assurez-vous que votre réplique lue n'a pas le même `backing-up` statut. Vous ne pouvez pas promouvoir une réplique lue lorsqu'elle est dans cet état.
- Arrêtez l'écriture de transactions sur l'instance de base de données principale, puis attendez que RDS applique toutes les mises à jour à la réplique lue.

  Les mises à jour de la base de données ont lieu sur le réplica en lecture après avoir eu lieu sur l'instance de base de données principale. Le délai de réplication peut varier considérablement. Utilisez la métrique `Replica Lag` pour déterminer à quel moment toutes les mises à jour ont été effectuées sur le réplica en lecture.
- (MySQL et MariaDB uniquement) Pour apporter des modifications à une réplique de lecture MySQL ou MariaDB avant de la promouvoir, définissez le paramètre sur dans le groupe de paramètres de base de données `0` pour `read_only` la réplique en lecture. Vous pouvez alors effectuer toutes les opérations DDL requises, telles que la création d'index, sur le réplica en lecture. Les actions entreprises sur le réplica en lecture n'affectent pas la performance de l'instance de base de données principale.

Promouvoir une réplique lue : étapes de base

Les étapes suivantes montrent le processus général de promotion d'un réplica en lecture en instance de base de données :

1. Promouvez la réplique lue à l'aide de l'option Promote de la console Amazon RDS, de la AWS CLI commande **`promote-read-replica`**ou de l'opération d'API **`PromoteReadReplica`**Amazon RDS.

   > **Note**
   >
   > Le processus de promotion dure quelques minutes. Lorsque vous promouvez une réplique en lecture, RDS arrête la réplication et redémarre la réplique en lecture. Une fois le redémarrage terminé, le réplica en lecture est disponible en tant que nouvelle instance de base de données.

2. (Facultatif) Modifiez la nouvelle instance de base de données pour en faire un déploiement multi-AZ. Pour plus d'informations, veuillez consulter Modification d'une RDS instance de base de données Amazon et Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

console

Pour promouvoir un réplica en lecture en tant qu'instance de base de données autonome

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la console Amazon RDS, choisissez Bases de données.

   Le volet Bases de données s'affiche. Chaque réplica en lecture affiche Réplica dans la colonne Rôle.
3. Choisissez le réplica en lecture que vous voulez promouvoir.
4. Pour Actions, choisissez Promote (Promouvoir).
5. Dans la page Promouvoir le réplica en lecture, saisissez la période de rétention des sauvegardes et la fenêtre de sauvegarde pour l'instance de base de données nouvellement promue.
6. Lorsque les paramètres vous conviennent, choisissez Continue.
7. Dans la page de confirmation, choisissez Promouvoir le réplica en lecture.

AWS CLI

Pour promouvoir une réplique en lecture vers une instance de base de données autonome, utilisez la AWS CLI `promote-read-replica`commande.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds promote-read-replica \
    --db-instance-identifier myreadreplica
```

Dans Windows :

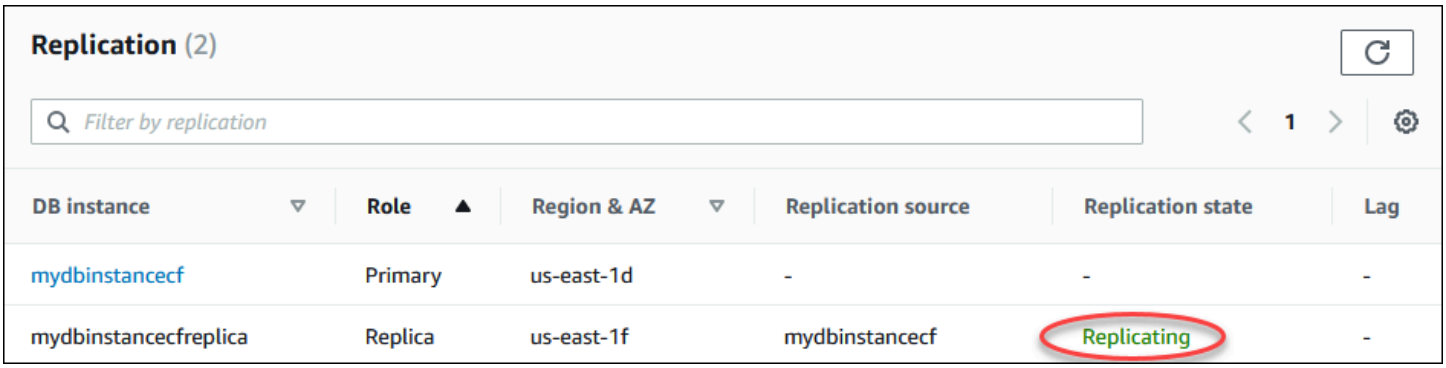
```
aws rds promote-read-replica ^
    --db-instance-identifier myreadreplica
```

API RDS

Pour promouvoir un réplica en lecture en tant qu'instance de base de données autonome, appelez l'opération `PromoteReadReplica` de l'API Amazon RDS avec le paramètre requis `DBInstanceIdentifier`.

Supervision de la réplication en lecture

Vous pouvez superviser le statut d'un réplica en lecture de différentes manières. La console Amazon RDS affiche le statut d'un réplica en lecture dans la section Replication (Réplication) de l'onglet Connectivity & security (Connectivité et sécurité), dans les détails du réplica en lecture. Pour consulter les détails d'un réplica en lecture, cliquez sur son nom dans la liste des instances de base de données de la console Amazon RDS.



Vous pouvez également consulter l'état d'une réplique lue à l'aide de la AWS CLI `describe-db-instances` commande ou de l'`DescribeDBInstances`opération d'API Amazon RDS.

Le statut d'un réplica en lecture peut avoir les valeurs suivantes :

- replicating (réplication en cours) – Le réplica en lecture réplique correctement.
- réplication dégradée (SQL Server et PostgreSQL uniquement) : les réplicas reçoivent des données de l'instance principale, mais une ou plusieurs bases de données peuvent ne pas recevoir de mises à jour. Cela peut se produire, par exemple, lorsqu'un réplica est en train de configurer des bases de données nouvellement créées. Cela peut également se produire lorsque des modifications de DDL non prises en charge ou d'objets volumineux sont apportées dans l'environnement bleu d'un blue/green déploiement.

  L'état ne passe pas de `replication degraded` à `error`, à moins qu'une erreur ne se produise pendant l'état dégradé.
- error (erreur) – Une erreur s'est produite dans le cadre de la réplication. Examinez le champ Replication Error (Erreur de réplication) dans la console Amazon RDS ou le journal des

événements pour déterminer l'erreur exacte. Pour plus d'informations sur la résolution d'une erreur de réplication, consultez Résolution d'un problème lié à SQL la réplique My read.

- terminated (arrêté) (MariaDB, MySQL ou PostgreSQL uniquement) – La réplication est arrêtée. Cela se produit si la réplication est arrêtée pendant plus de trente jours consécutifs, manuellement ou en raison d'une erreur de réplication. Dans ce cas, Amazon RDS met fin à la réplication entre l'instance de base de données principale et tous les réplicas en lecture. Amazon RDS fait cela pour éviter l'augmentation des besoins en stockage sur l'instance de base de données source et de longs délais de basculement.

  Une réplication interrompue peut affecter le stockage, car les journaux peuvent croître en taille et en nombre en raison du volume élevé des messages d'erreur consignés dans le journal. Une réplication interrompue peut également affecter la récupération en cas de défaillance en raison du temps dont a besoin Amazon RDS pour conserver et traiter le grand nombre de journaux au cours de la récupération.
- terminated (arrêté) (Oracle uniquement) – La réplication est arrêtée. Cela se produit si la réplication est arrêtée pendant plus de 8 heures car l'espace de stockage restant sur le réplica en lecture est insuffisant. Dans ce cas, Amazon RDS met fin à la réplication entre l'instance de base de données principale et les réplicas en lecture affectés. Ce statut est un état terminal et le réplica en lecture doit être recréé.
- stopped (interrompue) (MariaDB ou MySQL uniquement) – La réplication s'est interrompue en raison d'une demande initiée par le client.
- replication stop point set (point d'arrêt de réplication réglé) (MySQL uniquement) – Un point d'arrêt de réplication a été réglé par le client à l'aide de la procédure stockée et la réplication est en cours.
- replication stop point reached (point d'arrêt de réplication atteint) (MySQL uniquement) – Un point d'arrêt de réplication a été réglé par le client à l'aide de la procédure stockée et la réplication est arrêtée, car le point d'arrêt est atteint.

Vous pouvez voir où une instance de base de données est répliquée et, le cas échéant, vérifier son état de réplication. Sur la page Bases de données de la console RDS, elle affiche Primaire dans la colonne Rôle . Choisissez son nom d'instance de base de données. Sur sa page détaillée, dans l'onglet Connectivité et sécurité, son état de réplication se trouve sous Réplication.

Surveillance du retard de réplication

Vous pouvez surveiller le délai de réplication dans Amazon CloudWatch en consultant la `ReplicaLag` métrique Amazon RDS.

Pour Db2, la `ReplicaLag` métrique est le décalage maximal des bases de données qui ont pris du retard, en secondes. Par exemple, si deux bases de données sont décalées de 5 secondes et 10 secondes, le décalage `ReplicaLag` est de 10 secondes. Les bases de données dont le statut HADR (High Availability Disaster Recovery) n'est pas disponible ne sont pas incluses dans le calcul.

Pour MySQL et MariaDB, la métrique `ReplicaLag` contient la valeur du champ `Seconds_Behind_Master` de la commande `SHOW REPLICA STATUS`. Les causes courantes du retard de réplication pour MySQL et MariaDB sont les suivantes :

- Une indisponibilité du réseau.
- L'écriture dans des tables avec des index sur un réplica en lecture. Si le paramètre `read_only` n'a pas pour valeur 0 sur le réplica en lecture, il peut interrompre la réplication.
- Utilisation d'un moteur de stockage non transactionnel tel que MyISAM. La réplication est prise en charge uniquement pour le moteur de stockage InnoDB sur MySQL et pour le moteur de stockage XtraDB sur MariaDB.

> **Note**
>
> Les versions précédentes de MariaDB utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MariaDB inférieure à 10.5, utilisez. `SHOW SLAVE STATUS`

Lorsque la métrique `ReplicaLag` atteint 0, le réplica a rattrapé l'instance de bases de données principale. Si la métrique `ReplicaLag` retourne `-1`, la réplication n'est actuellement pas active. `ReplicaLag = -1` est équivalent à `Seconds_Behind_Master = NULL`.

Pour Oracle, la métrique `ReplicaLag` correspond à la somme de la valeur `Apply Lag` et à la différence entre la durée actuelle et la valeur `DATUM_TIME` du retard appliqué. La valeur `DATUM_TIME` correspond à la dernière heure à laquelle le réplica en lecture a reçu des données de son instance de base de données source. Pour plus d'informations, veuillez consulter V $DATAGUARD_STATS dans la documentation d'Oracle.

Pour SQL Server, la métrique `ReplicaLag` correspond au décalage maximal des bases de données qui ont pris du retard, en secondes. Par exemple, si vous avez deux bases de données qui accusent respectivement un retard de 5 secondes et 10 secondes, alors `ReplicaLag` a pour valeur 10 secondes. La métrique `ReplicaLag` renvoie la valeur de la requête suivante.

```
SELECT MAX(secondary_lag_seconds) max_lag FROM sys.dm_hadr_database_replica_states;
```

Pour plus d'informations, veuillez consulter secondary_lag_seconds dans la documentation Microsoft.

`ReplicaLag` renvoie `-1` si RDS ne peut pas déterminer le retard, par exemple lors de la configuration du réplica, ou lorsque le réplica en lecture est à l'état `error`.

> **Note**
>
> Les nouvelles bases de données ne sont pas incluses dans le calcul du retard tant qu'elles ne sont pas accessibles sur le réplica en lecture.

Pour PostgreSQL, la métrique `ReplicaLag` renvoie la valeur de la requête suivante.

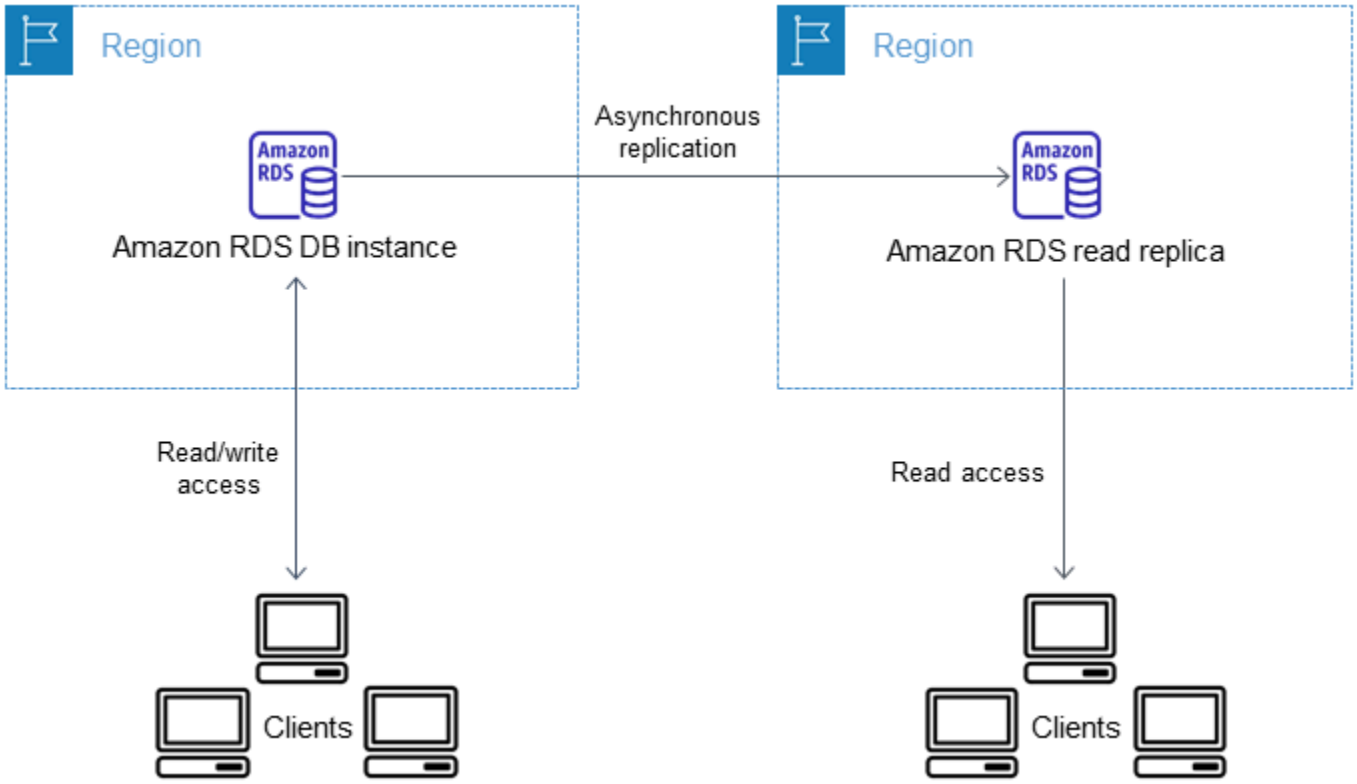
```
SELECT extract(epoch from now() - pg_last_xact_replay_timestamp()) AS reader_lag
```

PostgreSQL versions 9.5.2 et ultérieures utilise des emplacements physiques de réplication pour gérer la rétention WAL (Write Ahead Log) sur l'instance source. Pour chaque instance de réplica en lecture entre régions, Amazon RDS crée un emplacement de réplication physique et l'associe à l'instance. Deux CloudWatch indicateurs Amazon, `Oldest Replication Slot Lag` et`Transaction Logs Disk Usage`, montrent à quel point la réplique la plus en retard se situe en termes de données WAL reçues et de quantité de stockage utilisée pour les données WAL. La valeur `Transaction Logs Disk Usage` peut considérablement augmenter lorsqu'un réplica en lecture entre régions présente un retard important.

Pour plus d'informations sur la surveillance d'une instance de base de données avec CloudWatch, consultezSurveillance des métriques Amazon RDS () avec Amazon CloudWatch.

Création d'un réplica en lecture dans une autre Région AWS

Avec Amazon RDS, vous pouvez créer une réplique en lecture dans une instance de base de données différente Région AWS de l'instance de base de données source.



Vous créez une réplique de lecture dans un autre Région AWS pour effectuer les opérations suivantes :

- Améliorer vos capacités de reprise après sinistre.
- Adaptez les opérations de lecture au Région AWS plus près de vos utilisateurs.
- Facilitez la migration d'un centre de données de l'un Région AWS vers un centre de données d'un autre Région AWS.

La création d'une réplique en lecture dans une instance Région AWS différente de l'instance source est similaire à la création d'une réplique dans la même instance Région AWS. Vous pouvez utiliser AWS Management Console, exécuter la `create-db-instance-read-replica`commande ou appeler l'opération `CreateDBInstanceReadReplica`API.

> **Note**
>
> Pour créer une réplique de lecture chiffrée dans une instance de base de données Région AWS différente de l'instance de base de données source, l'instance de base de données source doit être chiffrée.

Rubriques

- Disponibilité des régions et des versions
- Création d'un réplica en lecture entre régions
- Processus de réplication entre régions au sein d'Amazon RDS
- Considérations liées à la réplication entre régions
- Coûts de la réplication entre régions

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions avec la réplication entre régions, consultez Régions et moteurs de base de données pris en charge pour les répliques de lecture entre régions dans Amazon RDS.

Création d'un réplica en lecture entre régions

Les procédures suivantes expliquent comment créer un réplica en lecture à partir d'une instance de base de données MariaDB, Microsoft SQL Server, MySQL, Oracle ou PostgreSQL source dans une autre Région AWS.

console

Vous pouvez créer une réplique en lecture à Régions AWS travers le AWS Management Console.

Pour créer une réplique en lecture Régions AWS avec la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données MariaDB, Microsoft SQL Server, MySQL, Oracle ou PostgreSQL que vous souhaitez utiliser comme source pour un réplica en lecture.
4. Sous Actions, choisissez Créer des réplicas en lecture.
5. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture.
6. Choisissez la Région de destination.

7. Choisissez les spécifications de l'instance que vous souhaitez utiliser. Nous vous recommandons d'utiliser un type de stockage et une classe d'instances de base de données identiques ou supérieurs pour le réplica en lecture.
8. Pour créer une réplique de lecture cryptée dans un autre Région AWS :

   a. Choisissez Activer le chiffrement.

   b. Pour AWS KMS key, choisissez l'identifiant AWS KMS key de la clé KMS dans la Région AWS de destination.

   > **Note**
   >
   > L'instance de base de données source doit être chiffrée pour que vous puissiez créer un réplica en lecture chiffré. Pour en savoir plus sur le chiffrement de l'instance de bases de données source, consultez Chiffrement des ressources Amazon RDS.

9. Choisissez d'autres options, telles que la scalabilité automatique du stockage.
10. Choisissez Créer un réplica en lecture.

AWS CLI

Pour créer une réplique en lecture à partir d'une instance de base de données MariaDB, Microsoft SQL Server, MySQL, Oracle ou PostgreSQL source dans Région AWS une autre instance de base de données, vous pouvez utiliser la commande. `create-db-instance-read-replica` Dans ce cas, vous utilisez `create-db-instance-read-replica` à partir de la Région AWS où vous voulez placer le réplica en lecture (région de destination) et vous spécifiez l'Amazon Resource Name (ARN) pour l'instance de base de données source. Un ARN identifie de façon unique une ressource créée dans Amazon Web Services.

Par exemple, si votre instance de base de données source se trouve dans la région US East (N. Virginia), l'ARN ressemble à l'exemple suivant :

```
arn:aws:rds:us-east-1:123456789012:db:mydbinstance
```

Pour plus d'informations sur ARNs, voirNoms des ressources Amazon (ARNs) dans Amazon RDS.

Pour créer une réplique de lecture dans une instance de base de données Région AWS différente de l'instance de base de données source, vous pouvez utiliser la AWS CLI `create-db-instance-`

read-replicacommande depuis la destination Région AWS. Les paramètres suivants sont nécessaires pour créer un réplica en lecture dans une autre Région AWS :

- `--region`— Destination Région AWS où la réplique lue est créée.
- `--source-db-instance-identifier` – Identifiant d'instance de base de données de l'instance de base de données source. Cet identifiant doit être au format ARN pour la Région AWS source.
- `--db-instance-identifier` – Identifiant du réplica en lecture dans la Région AWS de destination.

Example d'un réplica en lecture entre régions

Le code suivant crée un réplica en lecture dans la région USA Ouest (Oregon) à partir d'une instance de base de données source dans la région US East (N. Virginia).

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier myreadreplica \
    --region us-west-2 \
    --source-db-instance-identifier arn:aws:rds:us-east-1:123456789012:db:mydbinstance
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier myreadreplica ^
    --region us-west-2 ^
    --source-db-instance-identifier arn:aws:rds:us-east-1:123456789012:db:mydbinstance
```

Le paramètre suivant est également nécessaire pour créer un réplica en lecture chiffré dans une autre Région AWS :

- `--kms-key-id`— L' AWS KMS key identifiant de la clé KMS à utiliser pour chiffrer la réplique lue dans la destination Région AWS.

Example d'un réplica en lecture entre régions chiffré

Le code suivant crée un réplica en lecture chiffré dans la région USA Ouest (Oregon) à partir d'une instance de base de données source dans la région US East (N. Virginia).

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier myreadreplica \
    --region us-west-2 \
    --source-db-instance-identifier arn:aws:rds:us-east-1:123456789012:db:mydbinstance
 \
    --kms-key-id my-us-west-2-key
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier myreadreplica ^
    --region us-west-2 ^
    --source-db-instance-identifier arn:aws:rds:us-east-1:123456789012:db:mydbinstance
 ^
    --kms-key-id my-us-west-2-key
```

`--source-region`Cette option est requise lorsque vous créez une réplique de lecture cryptée entre les régions AWS GovCloud (USA Est) et AWS GovCloud (USA Ouest). Pour `--source-region`, spécifiez la Région AWS de l'instance de base de données source.

Si `--source-region` n'est pas spécifié, spécifiez une valeur `--pre-signed-url`. Une URL présignée est une URL qui contient une demande signée via Signature Version 4 pour la commande `create-db-instance-read-replica` qui est appelée dans la Région AWS source. Pour en savoir plus sur `pre-signed-url` cette option, consultez create-db-instance-read-replica dans le manuel de référence des AWS CLI commandes.

API RDS

Pour créer une réplique en lecture à partir d'une instance de base de données MariaDB, MySQL, Microsoft SQL Server, Oracle ou PostgreSQL source dans une Région AWS autre instance de base de données, vous pouvez appeler l'opération Create de l'API Amazon RDS. DBInstance ReadReplica Dans ce cas, vous appelez Create DBInstance ReadReplica à partir de l' Région AWS endroit où vous souhaitez lire la réplique (région de destination) et vous spécifiez le nom de ressource Amazon (ARN) pour l'instance de base de données source. Un ARN identifie de façon unique une ressource créée dans Amazon Web Services.

Pour créer une réplique de lecture chiffrée dans une instance de base de données Région AWS différente de l'instance de base de données source, vous pouvez utiliser l'`CreateDBInstanceReadReplica`opération d'API Amazon RDS depuis la destination Région AWS. Pour créer une réplique de lecture cryptée dans un autre Région AWS, vous devez spécifier une valeur pour`PreSignedURL`. `PreSignedURL`doit contenir une demande pour que l'`CreateDBInstanceReadReplica`opération appelle la source dans Région AWS laquelle la réplique de lecture est créée. Pour en savoir plus sur `PreSignedUrl`, consultez `CreateDBInstanceReadReplica`.

Par exemple, si votre instance de bases de données source se trouve dans la région US East (N. Virginia), l'ARN ressemble à ce qui suit.

```
arn:aws:rds:us-east-1:123456789012:db:mydbinstance
```

Pour plus d'informations sur ARNs, voirNoms des ressources Amazon (ARNs) dans Amazon RDS.

Example

```
https://us-west-2.rds.amazonaws.com/
    ?Action=CreateDBInstanceReadReplica
    &KmsKeyId=my-us-east-1-key
    &PreSignedUrl=https%253A%252F%252Frds.us-west-2.amazonaws.com%252F
         %253FAction%253DCreateDBInstanceReadReplica
         %2526DestinationRegion%253Dus-east-1
         %2526KmsKeyId%253Dmy-us-east-1-key
         %2526SourceDBInstanceIdentifier%253Darn%25253Aaws%25253Ards%25253Aus-
west-2%123456789012%25253Adb%25253Amydbinstance
         %2526SignatureMethod%253DHmacSHA256
         %2526SignatureVersion%253D4%2526SourceDBInstanceIdentifier%253Darn%25253Aaws
%25253Ards%25253Aus-west-2%25253A123456789012%25253Ainstance%25253Amydbinstance
         %2526Version%253D2014-10-31
         %2526X-Amz-Algorithm%253DAWS4-HMAC-SHA256
         %2526X-Amz-Credential%253DAKIADQKE4SARGYLE%252F20161117%252Fus-west-2%252Frds
%252Faws4_request
         %2526X-Amz-Date%253D20161117T215409Z
         %2526X-Amz-Expires%253D3600
         %2526X-Amz-SignedHeaders%253Dcontent-type%253Bhost%253Buser-agent%253Bx-amz-
content-sha256%253Bx-amz-date
         %2526X-Amz-Signature
%253D255a0f17b4e717d3b67fad163c3ec26573b882c03a65523522cf890a67fca613
    &DBInstanceIdentifier=myreadreplica
    &SourceDBInstanceIdentifier=&region-arn;rds:us-east-1:123456789012:db:mydbinstance
```

```
    &Version=2012-01-15
    &SignatureVersion=2
    &SignatureMethod=HmacSHA256
    &Timestamp=2012-01-20T22%3A06%3A23.624Z
    &AWSAccessKeyId=<&AWS; Access Key ID>
    &Signature=<Signature>
```

Processus de réplication entre régions au sein d'Amazon RDS

Amazon RDS utilise le processus ci-dessous pour créer un réplica en lecture entre régions. Ce processus peut prendre des Régions AWS heures en fonction de l'enjeu et de la quantité de données contenues dans les bases de données. Vous pouvez utiliser ces informations pour déterminer l'avancement du processus lorsque vous créez un réplica en lecture entre régions :

1. Amazon RDS commence par configurer l'instance DB source comme source de réplication et définit le statut sur modifying (modification).
2. Amazon RDS commence à configurer le réplica en lecture spécifié dans la Région AWS de destination et définit l'état sur creating (création en cours).
3. Amazon RDS crée un instantané de base de données automatisé de l'instance de base de données source dans la Région AWS source. Le format du nom de l'instantané de base de données est : `rds:<InstanceID>-<timestamp>`, où `<InstanceID>` est l'identifiant de l'instance source, et `<timestamp>` est la date et l'heure du début de la copie. Par exemple, `rds:mysourceinstance-2013-11-14-09-24` a été créé à partir de l'instance `mysourceinstance` le `2013-11-14-09-24`. Pendant la création de l'instantané de base de données automatique, le statut de l'instance de base de données source reste modifying (modification), le statut du réplica en lecture reste creating (création) et le statut de l'instantané de base de données est creating (création). La colonne d'avancement de la page de l'instantané de base de données dans la console indique le niveau d'avancement de la création de l'instantané de base de données. Une fois l'instantané de base de données terminé, le statut de l'instantané de base de données et celui de l'instance de base de données source sont définis sur available (disponible).
4. Amazon RDS commence une copie de l'instantané entre régions pour le transfert de données initial. La copie de l'instantané est répertoriée en tant qu'instantané automatisé dans la Région AWS de destination avec un statut creating (création en cours). Elle porte le même nom que l'instantané de base de données source. La colonne d'avancement de l'affichage de l'instantané de base de données indique le niveau d'avancement de la copie. Une fois la copie terminée, le statut de la copie de l'instantané de base de données est défini sur available (disponible).

5. Amazon RDS utilise alors l'instantané de base de données copié pour le chargement initial des données sur le réplica en lecture. Au cours de cette phase, le réplica en lecture figure dans la liste des instances de bases de données de la destination, avec le statut creating (création). Une fois le chargement terminé, le statut du réplica en lecture est défini sur available (disponible) et la copie de l'instantané de base de données est supprimée.
6. Lorsque le réplica en lecture atteint le statut disponible, Amazon RDS commence par répliquer les modifications apportées à l'instance source depuis le début de l'opération de création du réplica en lecture. Durant cette phase, la durée du retard de réplication pour le réplica en lecture est supérieure à 0.

   Pour plus d'informations sur la durée du retard de réplication, veuillez consulter Supervision de la réplication en lecture.

Considérations liées à la réplication entre régions

Toutes les considérations relatives à la réalisation de la réplication au sein d'un Région AWS s'appliquent à la réplication entre régions. Les considérations supplémentaires suivantes s'appliquent lors d'une réplication entre Régions AWS :

- Une instance de base de données source peut avoir des réplicas en lecture entre régions dans plusieurs Régions AWS. En raison de la limite du nombre d'entrées de liste de contrôle d'accès (ACL) pour le VPC source, RDS ne peut garantir plus de cinq instances de base de données de répliques en lecture interrégionales.
- Vous pouvez effectuer une réplication entre les régions GovCloud (USA Est) et GovCloud (USA Ouest), mais pas à destination ou en provenance des GovCloud États-Unis.
- Pour les moteurs RDS suivants, vous pouvez créer une réplique de lecture Amazon RDS interrégionale à partir d'une instance de base de données Amazon RDS source uniquement lorsqu'il ne s'agit pas d'une réplique en lecture d'une autre instance de base de données Amazon RDS :
  - RDS for SQL Server
  - RDS for Oracle
  - RDS pour les versions de PostgreSQL inférieures à 14.1

  Cette limitation ne s'applique pas aux instances de base de données exécutant RDS pour PostgreSQL version 14.1 ou ultérieure, RDS pour MariaDB et RDS pour MySQL.

- Vous pouvez vous attendre à un temps de latence plus élevé pour toute réplique en lecture située dans une instance différente Région AWS de celle de la source. Cette latence vient des canaux de réseau plus longs entre les centres de données régionaux.
- Pour ces réplicas en lecture entre régions, toutes les commandes de création de réplica en lecture qui spécifient le paramètre `--db-subnet-group-name` doivent spécifier un groupe de sous-réseaux DB du même VPC.
- Dans la plupart des cas, le réplica en lecture utilise le groupe de paramètres de base de données par défaut et le groupe d'options de base de données par défaut pour le moteur de base de données spécifié.

  Pour le moteur de base de données DB2, vous devez spécifier un groupe de paramètres personnalisé pour la réplique. Pour les répliques qui utilisent le modèle BYOL, ce groupe de paramètres personnalisés doit inclure votre IBM Site ID et. IBM Customer ID Vous pouvez spécifier ce groupe de paramètres personnalisés pour la réplique à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Pour de plus amples informations, veuillez consulter IBM IDs pour apporter votre propre licence (BYOL) pour Db2.

  Pour les moteurs de base de données MySQL et Oracle, vous pouvez spécifier un groupe de paramètres personnalisé pour la réplique en lecture dans l'`--db-parameter-group-name`option de la AWS CLI commande create-db-instance-read-replica. Vous ne pouvez pas spécifier un groupe de paramètres personnalisé lorsque vous utilisez AWS Management Console.
- Le réplica en lecture utilise le groupe de sécurité par défaut.
- Pour les instances de base de données MariaDB, Microsoft SQL Server, MySQL et Oracle, lorsque l'instance de base de données source pour un réplica en lecture entre régions est supprimée, le réplica en lecture est promu.
- Pour les instances de base de données PostgreSQL, lorsque l'instance de base de données source d'un réplica en lecture entre régions est supprimée, l'état de réplication du réplica en lecture est défini sur `terminated`. Le réplica en lecture n'est pas promu.

  Vous devez promouvoir le réplica en lecture ou le supprimer manuellement.

Demande d'un réplica en lecture entre régions

Pour communiquer avec la région source afin de demander la création d'un réplica en lecture entre régions, le demandeur (rôle IAM ou utilisateur IAM) doit avoir accès à l'instance de base de données source et à la région source.

Certaines conditions de la politique IAM du demandeur peuvent occasionner l'échec de la demande. Les exemples suivants supposent que l'instance de base de données source se trouve dans la USA Est (Ohio) et que le réplica en lecture est créé dans la US East (N. Virginia). Ces exemples illustrent les conditions de la politique IAM du demandeur qui occasionnent l'échec de la demande :

- La stratégie du demandeur est assortie d'une condition pour la `aws:RequestedRegion`.

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "aws:RequestedRegion": "us-east-1"
    }
}
```

  La demande échoue car la politique interdit l'accès à la région source. Pour qu'une demande aboutisse, spécifiez les régions source et destination.

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "aws:RequestedRegion": [
            "us-east-1",
            "us-east-2"
        ]
    }
}
```

- La politique du demandeur interdit l'accès à l'instance de base de données source.

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": "arn:aws:rds:us-east-1:123456789012:db:myreadreplica"
...
```

Pour qu'une demande aboutisse, spécifiez à la fois l'instance source et le réplica.

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": [
    "arn:aws:rds:us-east-1:123456789012:db:myreadreplica",
    "arn:aws:rds:us-east-2:123456789012:db:mydbinstance"
]
...
```

- La stratégie du demandeur refuse `aws:ViaAWSService`.

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": "*",
"Condition": {
    "Bool": {"aws:ViaAWSService": "false"}
}
```

  La communication avec la région source est effectuée par RDS pour le compte du demandeur. Pour qu'une demande aboutisse, ne refusez pas les appels effectués par les services AWS .

- La stratégie du demandeur est assortie d'une condition pour `aws:SourceVpc` ou `aws:SourceVpce`.

  Ces demandes peuvent échouer car l'appel effectué par RDS à la région distante ne provient pas du point de terminaison VPC ou du VPC spécifié.

Si vous devez utiliser l'une des conditions précédentes, qui sont susceptibles d'occasionner l'échec d'une requête, vous pouvez inclure une deuxième instruction avec `aws:CalledVia` dans votre politique pour que la demande soit couronnée de succès. Par exemple, vous pouvez utiliser `aws:CalledVia` avec `aws:SourceVpce` comme indiqué ici :

```
...
"Effect": "Allow",
"Action": "rds:CreateDBInstanceReadReplica",
"Resource": "*",
"Condition": {
```

```
    "Condition" : {
        "ForAnyValue:StringEquals" : {
          "aws:SourceVpce": "vpce-1a2b3c4d"
        }
     }
},
{
    "Effect": "Allow",
    "Action": [
        "rds:CreateDBInstanceReadReplica"
    ],
    "Resource": "*",
    "Condition": {
        "ForAnyValue:StringEquals": {
            "aws:CalledVia": [
                "rds.amazonaws.com"
            ]
        }
    }
}
```

Pour de plus amples informations, veuillez consulter Policies and permissions in IAM (Stratégies et autorisations dans IAM) dans le IAM Guide de l'utilisateur.

Autorisation du réplica en lecture

Après qu'une demande de création d'un réplica en lecture de base de données inter-région a renvoyé `success`, RDS démarre la création du réplica en arrière-plan. Une autorisation devant permettre à RDS d'accéder à l'instance de base de données source est créée. Cette autorisation associe l'instance de base de données source au réplica en lecture et permet à RDS de ne copier que vers le réplica en lecture spécifié.

L'autorisation est vérifiée par RDS à l'aide de l'autorisation `rds:CrossRegionCommunication` dans le rôle IAM lié au service. Si le réplica est autorisé, RDS communique avec la région source et accomplit la création du réplica.

RDS n'a pas accès aux instances DB qui n'ont pas été autorisées auparavant par une demande `CreateDBInstanceReadReplica`. L'autorisation est révoquée lorsque la création du réplica en lecture est terminée.

RDS utilise le rôle lié au service afin de vérifier l'autorisation dans la région source. Si vous supprimez le rôle lié au service durant le processus de création de réplication, la création échoue.

Pour plus d'informations, consultez Utilisation des rôles liés à un service dans le IAM Guide de l'utilisateur.

Utilisation des AWS Security Token Service informations d'identification

Les jetons de session provenant du point de terminaison global AWS Security Token Service (AWS STS) ne sont valides Régions AWS que s'ils sont activés par défaut (régions commerciales). Si vous utilisez les informations d'identification issues de l'opération d'`assumeRole`API dans AWS STS, utilisez le point de terminaison régional si la région source est une région optionnelle. Sinon, la demande échoue. Cela se produit parce que vos informations d'identification doivent être valides dans les deux régions, ce qui n'est vrai pour les régions optionnelles que lorsque le point de AWS STS terminaison régional est utilisé.

Pour utiliser le point de terminaison global, assurez-vous qu'il est activé dans les opérations pour les deux régions. Définissez le point de terminaison global sur `Valid in all Régions AWS` dans les paramètres du AWS STS compte.

La même règle s'applique aux informations d'identification dans le paramètre URL pré-signé.

Pour plus d'informations, consultez la section Gestion AWS STS dans et Région AWS dans le guide de l'utilisateur IAM.

Coûts de la réplication entre régions

Les données transférées pour la réplication entre régions génèrent des frais de transfert de données Amazon RDS. Ces actions de réplication entre régions génèrent des frais pour les données transférées hors de la Région AWS source :

- Lorsque vous créez un réplica en lecture, Amazon RDS prend un instantané de l'instance source et transfère cet instantané vers la Région AWS du réplica en lecture.
- Pour chaque modification de données effectuée dans les bases de données sources, Amazon RDS transfère les données de la source Région AWS vers la réplique Région AWS lue.

Pour de plus amples informations sur la tarification du transfert des données, veuillez consulter la Tarification Amazon RDS.

Pour les instances MySQL et MariaDB, vous pouvez réduire les coûts de transfert de données en réduisant le nombre de réplicas en lecture entre régions que vous créez. Supposons, par exemple, que vous ayez une instance de base de données source dans l'une Région AWS et que

vous souhaitiez avoir trois répliques de lecture dans une autre Région AWS. Dans ce cas, vous créez uniquement l'un des réplicas en lecture à partir de l'instance de base de données source. Ensuite, vous créez les deux autres réplicas à partir du premier réplica en lecture plutôt qu'à partir de l'instance de base de données source.

Par exemple, si vous `source-instance-1` en avez un Région AWS, vous pouvez effectuer les opérations suivantes :

- Créez `read-replica-1` dans le nouveau Région AWS, en spécifiant `source-instance-1` comme source.
- Créez `read-replica-2` à partir de `read-replica-1`.
- Créez `read-replica-3` à partir de `read-replica-1`.

Dans cet exemple, seules les données transférées de `source-instance-1` vers `read-replica-1` vous sont facturées. Les données transférées de `read-replica-1` vers les deux autres réplicas ne vous sont pas facturées, car ils figurent tous dans la même Région AWS. Si vous créez les trois répliques directement `source-instance-1` à partir d'une autre Région AWS, les transferts de données vers les trois répliques vous sont facturés.

Marquage d' Amazon RDS

Une RDS balise Amazon est une paire nom-valeur que vous définissez et associez à une RDS ressource Amazon telle qu'une instance de base de données ou un instantané de base de données. Le nom s'appelle la clé. Vous pouvez éventuellement fournir une valeur pour la clé.

Vous pouvez utiliser le AWS Management Console AWS CLI, le ou Amazon RDS API pour ajouter, répertorier et supprimer des balises sur les RDS ressources Amazon. Lorsque vous utilisez le CLI ouAPI, assurez-vous de fournir le nom de la ressource Amazon (ARN) pour la RDS ressource à utiliser. Pour plus d'informations sur la construction d'unARN, consultezCréation d'un ARN pour Amazon RDS.

Vous pouvez utiliser des balises pour ajouter des métadonnées à vos RDS ressources Amazon RDS . Vous pouvez utiliser les balises pour ajouter vos propres notations sur les instances de base de données, les instantanés, les clusters Aurora, etc. Cela peut vous aider à documenter vos RDS ressources Amazon RDS . Vous pouvez également utiliser les balises avec des procédures de maintenance automatisées.

Vous pouvez notamment utiliser ces balises avec des IAM politiques. Vous pouvez les utiliser pour gérer l'accès aux RDS ressources Amazon RDS et pour contrôler les actions qui peuvent être appliquées à ces ressources. Vous pouvez également utiliser ces balises pour suivre les coûts en regroupant les dépenses pour des ressources balisées de la même façon.

Vous pouvez baliser les RDS ressources Amazon RDS suivantes :

- Instances DB
- Clusters DB
- Clusters globaux Aurora
- Points de terminaison du cluster de bases de
- Réplicas en lecture
- Instantanés de base de données
- Instantanés de cluster DB
- Instances DB réservées
- Abonnements aux événements
- Groupes d'options DB
- Groupes de paramètres DB

- Groupes de paramètres de cluster DB
- Groupes de sous-réseaux DB
- RDSProxies
- RDSPoints de terminaison proxy
- Déploiements bleu/vert
- Zéro ETL intégrations

> **Note**
>
> Actuellement, vous ne pouvez pas baliser RDS les proxys et les points de terminaison des RDS proxys à l'aide du. AWS Management Console

Rubriques

- Pourquoi utiliser les balises RDS de ressources Amazon ?
- Comment fonctionnent les balises de RDS ressources Amazon
- Bonnes pratiques pour le balisage des ressources Amazon RDS
- Copier des balises dans des instantanés de base de données
- Ajouter et supprimer des tags sur Amazon RDS
- Tutoriel : Spécifiez les instances de base de données à arrêter à l'aide de balises

Pourquoi utiliser les balises RDS de ressources Amazon ?

Vous pouvez utiliser les balises pour effectuer les opérations suivantes :

- Classez vos RDS ressources par application, projet, département, environnement, etc. Par exemple, vous pouvez utiliser une clé de balise pour définir une catégorie, la valeur de la balise étant un élément de cette catégorie. Vous pouvez créer le tag`environment=prod`. Vous pouvez également définir une clé de balise `project` et une valeur de balise de`Salix`, ce qui indique qu'une RDS ressource Amazon est affectée au projet Salix.
- Automatisez les tâches de gestion des ressources. Par exemple, vous pouvez créer une fenêtre de maintenance pour les instances `environment=prod` étiquetées différente de la fenêtre pour les instances étiquetées`environment=test`. Vous pouvez également configurer des instantanés de base de données automatiques pour les instances étiquetées`environment=prod`.

- Contrôlez l'accès aux RDS ressources dans le cadre d'une IAM politique. Pour cela, vous devez utiliser la clé de condition globale `aws:ResourceTag/tag-key`. Par exemple, une politique peut autoriser uniquement les utilisateurs du `DBAdmin` groupe à modifier les instances de base de données étiquetées avec`environment=prod`. Pour plus d'informations sur la gestion de l'accès aux ressources balisées à l'aide de IAM politiques, consultez Gestion des identités et des accès pour Amazon RDS la section Contrôle de l'accès aux AWS ressources dans le guide de l'utilisateur d'AWS Identity and Access Management.
- Surveillez les ressources en fonction d'un tag. Par exemple, vous pouvez créer un tableau de CloudWatch bord Amazon pour les instances de base de données étiquetées avec`environment=prod`.
- Suivez les coûts en regroupant les dépenses pour des ressources étiquetées de la même manière. Par exemple, si vous balisez les RDS ressources associées au projet Salix avec`project=Salix`, vous pouvez générer des rapports de coûts et allouer des dépenses à ce projet. Pour de plus amples informations, veuillez consulter Comment fonctionne AWS la facturation avec les tags sur Amazon RDS.

Comment fonctionnent les balises de RDS ressources Amazon

AWS n'applique aucune signification sémantique à vos balises. Les balises sont interprétées de façon stricte, en tant que chaîne de caractères.

Rubriques

- Ensembles de tags sur Amazon RDS
- Structure des balises sur Amazon RDS
- RDSRessources Amazon éligibles au balisage
- Comment fonctionne AWS la facturation avec les tags sur Amazon RDS

Ensembles de tags sur Amazon RDS

Chaque RDS ressource Amazon possède un conteneur appelé ensemble de balises. Le conteneur inclut toutes les balises attribuées à la ressource. Une ressource possède exactement un ensemble de balises.

Un jeu de balises contient de 0 à 50 balises. Si vous ajoutez une balise à une RDS ressource avec la même clé qu'une balise de ressource existante, la nouvelle valeur remplace l'ancienne.

Structure des balises sur Amazon RDS

La structure d'une RDS balise est la suivante :

Clé du tag

La clé est le nom obligatoire de la balise. La valeur de chaîne doit comporter de 1 à 128 caractères Unicode et ne peut pas être préfixée par `aws:` ou. `rds:` La chaîne ne peut contenir que l'ensemble des lettres Unicode, des chiffres, des espaces blancs `_.:/,=`, `+-`, et`@`. L'expression régulière Java est`"^([\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]*)$"`. Les clés de balises sont sensibles à la casse. Ainsi, les clés `project` et les touches `Project` sont distinctes.

Une clé est propre à un ensemble de balises. Par exemple, une paire de clés ne peut pas être définie dans une balise avec la même clé mais avec des valeurs différentes, telles que `project=Trinity` et. `project=Xanadu`

Valeur du tag

La valeur est une valeur de chaîne facultative de la balise. La valeur de la chaîne doit comporter de 1 à 256 caractères Unicode. La chaîne ne peut contenir que l'ensemble des lettres Unicode, des chiffres, des espaces blancs `_.:/,=`, `+-`, et`@`. L'expression régulière Java est`"^([\\p{L}\\p{Z}\\p{N}_.:/=+\\-@]*)$"`. Les valeurs de balises sont sensibles à la casse. Ainsi, les valeurs `prod` et les valeurs `Prod` sont distinctes.

Les valeurs n'ont pas besoin d'être uniques dans un jeu de balises et peuvent être nulles. Par exemple, vous pouvez avoir une paire clé-valeur dans un ensemble de balises `project=Trinity` et `cost-center=Trinity`.

RDSRessources Amazon éligibles au balisage

Vous pouvez baliser les RDS ressources Amazon suivantes :

- Instances DB
- Clusters DB
- Points de terminaison du cluster de bases de
- Réplicas en lecture
- Instantanés de base de données
- Instantanés de cluster DB
- Instances DB réservées

- Abonnements aux événements
- Groupes d'options DB
- Groupes de paramètres DB
- Groupes de paramètres de cluster DB
- Groupes de sous-réseaux DB
- RDSProxies
- RDSPoints de terminaison proxy

> **Note**
>
> Actuellement, vous ne pouvez pas baliser RDS les proxys et les points de terminaison des RDS proxys à l'aide du. AWS Management Console

- Déploiements bleu/vert
- Zéro ETL intégrations (aperçu)

Comment fonctionne AWS la facturation avec les tags sur Amazon RDS

Utilisez des balises pour organiser votre AWS facture afin de refléter votre propre structure de coûts. Pour ce faire, inscrivez-vous pour recevoir votre Compte AWS facture avec les valeurs clés du tag incluses. Ensuite, pour voir le coût de vos ressources combinées, organisez vos informations de facturation en fonction des ressources possédant les mêmes valeurs de clé de balise. Par exemple, vous pouvez baliser plusieurs ressources avec un nom d'application spécifique, puis organiser vos informations de facturation pour afficher le coût total de cette application dans plusieurs services. Pour de plus amples informations, veuillez consulter Utilisation des balises d'allocation des coûts dans le Guide de l'utilisateur AWS Billing .

Fonctionnement des balises de répartition des coûts avec les instantanés de de bases de données

Vous pouvez ajouter une balise à un instantané de de base de données. Toutefois, votre facture ne reflètera pas ce groupement. Pour que les balises de répartition des coûts s'appliquent aux instantanés de bases de données, les conditions suivantes doivent être remplies :

- Les balises doivent être attachées à l'instance de base de données parent.
- L'instance de base de données parent doit exister au même endroit Compte AWS que le snapshot du de base de données.

- L'instance de base de données parent doit exister au même endroit Région AWS que le snapshot du de base de données.

Les instantanés de de base de données sont considérés comme orphelins s'ils n'existent pas dans la même région que l'instance de base de données parent ou si l'instance de base de données parent est supprimée. Les instantanés de base de données orphelins ne prennent pas en charge les balises de répartition des coûts. Les coûts des instantanés orphelins sont agrégés dans une seule rubrique non balisée. Les instantanés de de bases de données entre comptes ne sont pas considérés comme orphelins lorsque les conditions suivantes sont remplies :

- Ils existent dans la même région que l'instance de base de données parent.
- L'instance de base de données parent appartient au compte source.

> **Note**
>
> Si l'instance de base de données parent appartient à un autre compte, les balises de répartition des coûts ne s'appliquent pas aux instantanés entre comptes du compte de destination.

Bonnes pratiques pour le balisage des ressources Amazon RDS

Lorsque vous utilisez des balises, nous vous recommandons de suivre les bonnes pratiques suivantes :

- Documentez les conventions relatives à l'utilisation des balises qui sont suivies par toutes les équipes de votre organisation. Assurez-vous en particulier que les noms sont à la fois descriptifs et cohérents. Par exemple, normalisez le format `environment:prod` plutôt que d'étiqueter certaines ressources avec. `env:production`

> **Important**
>
> Ne stockez pas d'informations personnellement identifiables (PII) ou d'autres informations confidentielles ou sensibles dans des balises.

- Automatisez le balisage pour garantir la cohérence. Par exemple, vous pouvez utiliser les techniques suivantes :

  - Incluez des balises dans un AWS CloudFormation modèle. Lorsque vous créez des ressources à l'aide du modèle, elles sont étiquetées automatiquement.
  - Définissez et appliquez des balises à l'aide de AWS Lambda fonctions.
  - Créez un SSM document qui inclut les étapes à suivre pour ajouter des balises à vos RDS ressources.
- N'utilisez des balises que lorsque cela est nécessaire. Vous pouvez ajouter jusqu'à 50 balises pour une seule RDS ressource, mais il est recommandé d'éviter la prolifération et la complexité inutiles des balises.
- Vérifiez régulièrement la pertinence et l'exactitude des balises. Supprimez ou modifiez les balises obsolètes selon vos besoins.
- Envisagez de créer des balises à l'aide de l'éditeur de AWS balises dans le AWS Management Console. Vous pouvez utiliser l'éditeur de balises pour ajouter des balises à plusieurs AWS ressources prises en charge, y compris RDS des ressources, en même temps. Pour plus d'informations, consultez la section Tag Editor dans le Guide de l'utilisateur d'AWS Resource Groups.

Copier des balises dans des instantanés de base de données

Lorsque vous créez ou restaurez une instance de base de données, vous pouvez spécifier que les balises de l'instance de base de données soient copiées vers les snapshots de l'instance de base de données. La copie des balises garantit que les métadonnées pour les instantanés de base de données correspondent à l'instance de base de données source. Elle garantit également que toutes les stratégies d'accès pour les instantanés de base de données correspondent également à l'instance de base de données source.

Vous pouvez spécifier que les balises soient copiées vers des snapshots DB pour les actions suivantes :

- Création d'une instance de base de données
- Restauration d'une instance de base de données
- Création d'un réplica en lecture
- Copie d'un instantané de base de données

Pour copier des balises pour les actions précédentes, choisissez Copier les balises dans les instantanés dans le AWS Management Console, ou spécifiez-les `--copy-tags-to-snapshot` dans le AWS CLI.

Dans la plupart des cas, les identifications ne sont pas copiées par défaut. Toutefois, lorsque vous restaurez une instance de base de données à partir d'un instantané de base de données, RDS vérifie si vous spécifiez de nouvelles balises. Si oui, les nouvelles identifications sont ajoutées à l'instance de base de données restaurée. S'il n'y a pas de nouvelles balises, RDS ajoute les balises de l'instance de base de données source au moment de la création du snapshot à l'instance de base de données restaurée.

Pour empêcher l'ajout d'identifications provenant d'instances de base de données sources à des instances de base de données restaurées, nous vous recommandons de spécifier de nouvelles identifications lors de la restauration d'une instance de base de données.

> **Note**
>
> Dans certains cas, vous pouvez inclure une valeur pour le `--tags` paramètre de la create-db-snapshot AWS CLI commande. Vous pouvez également fournir au moins une balise à l'reateDBSnapshotAPIopération C. Dans ces cas, RDS ne copie pas les balises de l'instance de base de données source vers le nouveau snapshot de base de données. Cette fonctionnalité s'applique même si l'option `--copy-tags-to-snapshot` (`CopyTagsToSnapshot`) est activée sur l'instance de base de données source.
> Si vous optez pour cette approche, vous pouvez créer une copie d'une instance de base de données à partir d'un instantané de base de données. Cette approche évite d'ajouter des balises qui ne s'appliquent pas à la nouvelle instance de base de données. Vous créez votre instantané de base de données à l'aide de la AWS CLI `create-db-snapshot` commande (ou de l'`CreateDBSnapshot`RDSAPIopération). Après avoir créé votre instantané de base de données, vous pouvez ajouter des balises comme décrit plus loin dans cette rubrique.

Ajouter et supprimer des tags sur Amazon RDS

Vous pouvez effectuer les actions suivantes :

- Créez des balises lorsque vous créez une ressource, par exemple lorsque vous exécutez la AWS CLI commande`create-db-instance`.
- Ajoutez des balises à une ressource existante à l'aide de la commande`add-tags-to-resource`.

- Répertoriez les balises associées à une ressource spécifique à l'aide de la commande`list-tags-for-resource`.
- Mettez à jour les balises à l'aide de la commande`add-tags-to-resource`.
- Supprimez les balises d'une ressource à l'aide de la commande`remove-tags-from-resource`.

Les procédures suivantes montrent comment effectuer des opérations de balisage classiques sur les ressources associées aux instances de base de données . Notez que les balises sont mises en cache à des fins d'autorisation. C'est pourquoi, lorsque vous ajoutez ou mettez à jour des balises sur les RDS ressources Amazon, plusieurs minutes peuvent s'écouler avant que les modifications ne soient disponibles.

Console

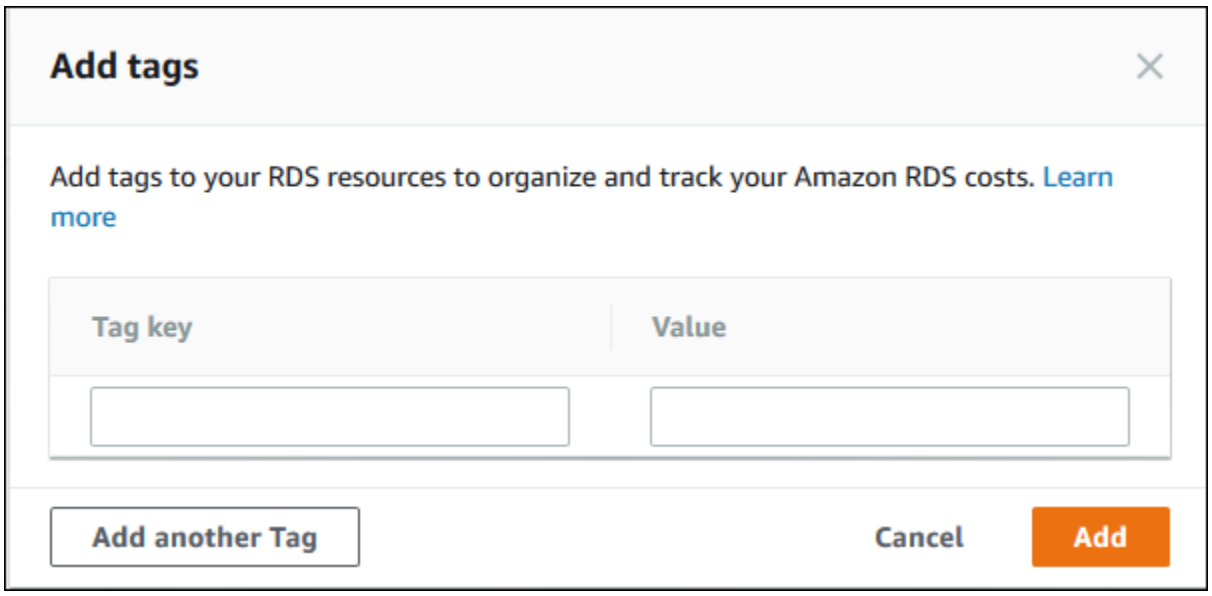
Le processus de balisage d'une RDS ressource Amazon est similaire pour toutes les ressources. La procédure suivante montre comment baliser une RDS instance de base de données Amazon.

Pour ajouter une balise à une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

   > **Note**
   >
   > Pour filtrer la liste des instances de base de données dans le volet Bases de données, saisissez une chaîne de texte dans Filter databases (Filtrer les bases de données). Seules les instances de base de données qui contiennent la chaîne apparaissent.

3. Sélectionnez le nom de l'instance de base de données que vous souhaitez baliser pour afficher ses détails.
4. Dans la section des détails, faites défiler jusqu'à la section Balises.
5. Choisissez Ajouter. La fenêtre Ajouter des balises s'affiche.



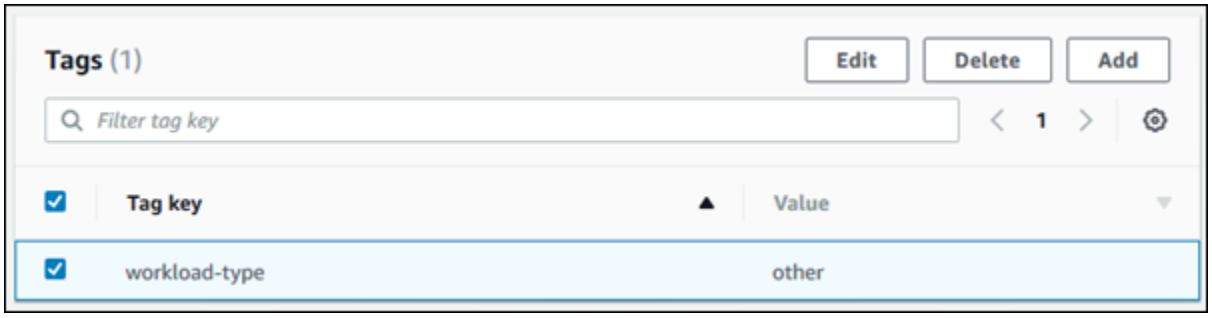
6. Saisissez une valeur pour Tag key (Clé de balise) et Valeur.
7. Pour ajouter une autre balise, vous pouvez choisir Ajouter une autre balise et saisir une valeur pour Tag key (Clé de balise) et Valeur.

   Répétez cette étape autant de fois que nécessaire.
8. Choisissez Ajouter.

Pour supprimer une balise d'une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

   > **Note**
   >
   > Pour filtrer la liste des instances de base de données dans le volet Bases de données, saisissez une chaîne de texte dans la zone Filter databases (Filtrer les bases de données). Seules les instances de base de données qui contiennent la chaîne apparaissent.
3. Sélectionnez le nom de l'instance de base de données pour afficher ses détails.
4. Dans la section des détails, faites défiler jusqu'à la section Balises.
5. Choisissez la balise que vous souhaitez supprimer.



6. Choisissez Supprimer, puis Supprimer dans la fenêtre Supprimer les balises.

AWS CLI

Vous pouvez ajouter, répertorier ou supprimer des balises pour une instance de base de données à l'aide de l AWS CLI.

- Pour ajouter une ou plusieurs balises à une RDS ressource Amazon, utilisez la AWS CLI commande `add-tags-to-resource`.
- Pour répertorier les tags d'une RDS ressource Amazon, utilisez la AWS CLI commande `list-tags-for-resource`.
- Pour supprimer une ou plusieurs balises d'une RDS ressource Amazon, utilisez la AWS CLI commande `remove-tags-from-resource`.

Pour en savoir plus sur la façon de construire le requisARN, voirCréation d'un ARN pour Amazon RDS.

RDS API

Vous pouvez ajouter, répertorier ou supprimer des balises pour une instance de base de données à l'aide d'Amazon RDSAPI.

- Pour ajouter un tag à une RDS ressource Amazon, utilisez l'`AddTagsToResource`opération.
- Pour répertorier les balises attribuées à une RDS ressource Amazon, utilisez le `ListTagsForResource`.
- Pour supprimer des balises d'une RDS ressource Amazon, utilisez l'`RemoveTagsFromResource`opération.

Pour en savoir plus sur la façon de construire le requisARN, voirCréation d'un ARN pour Amazon RDS.

Lorsque vous XML utilisez Amazon RDSAPI, les balises utilisent le schéma suivant :

```
<Tagging>
      <TagSet>
          <Tag>
              <Key>Project</Key>
              <Value>Trinity</Value>
          </Tag>
          <Tag>
              <Key>User</Key>
              <Value>Jones</Value>
          </Tag>
      </TagSet>
  </Tagging>
```

Le tableau suivant fournit une liste des XML balises autorisées et de leurs caractéristiques. Les valeurs pour `Key` et `Value` distinguent les majuscules et minuscules. Par exemple, `project=Trinity` et `PROJECT=Trinity` sont des balises distinctes.

Elément de balisage	Description
TagSet	Un ensemble de balises est un conteneur pour toutes les balises attribuées à une RDS ressource Amazon. Il ne peut y avoir qu'un ensemble de balises par ressource. Vous travaillez avec un TagSet uniquement via Amazon RDSAPI.
Tag	Une balise est une paire clé-valeur définie par l'utilisateur. Il peut y avoir de 1 à 50 balises dans un ensemble de balises.
Key	Une clé est le nom obligatoire de la balise. Pour les restrictions, voirStructure des balises sur Amazon RDS. La valeur de la chaîne peut comporter de 1 à 128 caractères Unicode et elle ne peut pas être précédée de `aws:` ou de `rds:`. La chaîne peut uniquement contenir l'ensemble de lettres Unicode, de chiffres, d'espaces, « _ », « . », « / », « = », « + », « - », (expression Java : "^([\\p{L}\\p{Z}\\p{N}_.:/=+\\-]*)$"). Les clés doivent être propres à un ensemble de balises. Par exemple, une paire de clés ne peut pas être définie dans une balise avec la même clé mais avec des valeurs différentes, telles que. project/Trinity and project/Xanadu

Elément de balisage	Description
Valeur	Une valeur est la valeur facultative de la balise. Pour les restrictions, voirStructure des balises sur Amazon RDS. La valeur de la chaîne peut comporter de 1 à 256 caractères Unicode et elle ne peut pas être précédée de `aws:` ou de `rds:`. La chaîne peut uniquement contenir l'ensemble de lettres Unicode, de chiffres, d'espaces, « _ », « . », « / », « = », « + », « - », (expression Java : "^([\\p{L}\\p{Z}\\p{N}_.:/=+\\-]*)$"). Les valeurs comprises dans un ensemble de balises ne doivent pas nécessair ement être uniques et peuvent être null. Par exemple, vous pouvez avoir une paire clé-valeur dans un ensemble de balises de. project/Trinity and cost-center/Trinity

Tutoriel : Spécifiez les instances de base de données à arrêter à l'aide de balises

Ce didacticiel part du principe que vous disposez de plusieurs instances de base de données dans un environnement de développement ou de test. Vous devez conserver ces instances de base de données pendant plusieurs jours. Certaines instances de base de données exécutent des tests pendant la nuit, tandis que d'autres peuvent être arrêtées pendant la nuit et redémarrées le lendemain.

Le didacticiel suivant montre comment attribuer une balise aux instances de base de données susceptibles de s'arrêter du jour au lendemain. Le didacticiel montre comment un script peut détecter quelles instances de base de données possèdent la balise, puis arrêter les instances de base de données étiquetées. Dans cet exemple, la partie valeur de la paire clé-valeur n'a pas d'importance. La présence de la balise `stoppable` signifie que l'instance de base de données possède cette propriété définie par l'utilisateur.

Dans le didacticiel suivant, les commandes et APIs le balisage fonctionnent avecARNs, ce qui permet de RDS travailler de manière fluide entre AWS les régions, les AWS comptes et les différents types de ressources susceptibles de porter des noms abrégés identiques. Vous pouvez spécifier l'ID d'instance de base de données ARN au lieu de l'ID dans CLI les commandes qui fonctionnent sur les instances de base de données.

Spécifier les instances de bases de données à arrêter

1. Déterminez l'instance ARN de base de données que vous souhaitez désigner comme arrêtable.

   Dans l'exemple suivant, remplacez le nom de vos propres instances de base de données par *dev-test-db-instance*. Dans les commandes suivantes qui utilisent ARN des paramètres, remplacez votre propre instance ARN de base de données. ARNCela inclut votre propre identifiant de AWS compte et le nom de la AWS région dans laquelle se trouve votre instance de base de données.

   ```
   $ aws rds describe-db-instances --db-instance-identifier dev-test-db-instance \
     --query "*[].{DBInstance:DBInstanceArn}" --output text
   arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance
   ```

2. Ajoutez la balise `stoppable` à cette instance de base de données.

   Vous choisissez le nom de cette balise. Étant donné que cet exemple traite la balise comme un attribut présent ou absent, il omet la partie `Value=` du paramètre `--tags`. Cette approche signifie que vous pouvez éviter de concevoir une convention de dénomination qui encode toutes les informations pertinentes dans les noms. Dans une telle convention, vous pouvez encoder des informations dans le nom de l'instance de base de données ou les noms d'autres ressources.

   ```
   $ aws rds add-tags-to-resource \
     --resource-name arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance \
     --tags Key=stoppable
   ```

3. Confirmez que la balise est présente dans l'instance de base de données.

   Les commandes suivantes récupèrent les informations de balise pour l'instance de base de données JSON au format et en texte brut séparé par des tabulations.

   ```
   $ aws rds list-tags-for-resource \
     --resource-name arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance
   {
       "TagList": [
           {
               "Key": "stoppable",
               "Value": ""

           }
       ]
   ```

```
}
aws rds list-tags-for-resource \
  --resource-name arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance --
output text
TAGLIST stoppable
```

4. Arrêtez toutes les instances de base de données désignées comme`stoppable`.

   L'exemple suivant crée un fichier texte répertoriant toutes vos instances de base de données. La commande shell parcourt la liste et vérifie si chaque instance de base de données est étiquetée avec l'attribut approprié et exécute la commande `aws rds stop-db-instance` pour chaque instance de base de données.

```
$ aws rds describe-db-instances --query "*[].[DBInstanceArn]" --output text >/tmp/
db_instance_arns.lst
$ for arn in $(cat /tmp/db_instance_arns.lst)
do
  match="$(aws rds list-tags-for-resource --resource-name $arn --output text | grep
 stoppable)"
  if [[ ! -z "$match" ]]
  then
      echo "DB instance $arn is tagged as stoppable. Stopping it now."
# Note that you need to get the DB instance identifier from the ARN.
      dbid=$(echo $arn | sed -e 's/.*://')
      aws rds stop-db-instance --db-instance-identifier $dbid
  fi
done

DB instance arn:arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance is
 tagged as stoppable. Stopping it now.
{
    "DBInstance": {
        "DBInstanceIdentifier": "dev-test-db-instance",
        "DBInstanceClass": "db.t3.medium",
        ...
```

Vous pouvez exécuter un script comme le précédent à la fin de chaque journée pour vous assurer que les instances de base de données non essentielles sont arrêtées. Vous pouvez également planifier une tâche à l'aide d'un utilitaire tel que `cron` pour effectuer une telle vérification chaque nuit. Par exemple, vous pouvez le faire si certaines instances de base de données restaient en

cours d'exécution par erreur. Dans ce cas, vous pouvez affiner la commande qui prépare la liste des instances de base de données à vérifier.

La commande suivante crée une liste de vos instances de base de données, mais uniquement celles au statut `available`. Le script peut ignorer les instances de base de données qui sont déjà arrêtées, car elles auront des valeurs de statut différentes telles que `stopped` ou `stopping`.

```
$ aws rds describe-db-instances \
  --query '*[].{DBInstanceArn:DBInstanceArn,DBInstanceStatus:DBInstanceStatus}|[?
DBInstanceStatus == `available`]|[].{DBInstanceArn:DBInstanceArn}' \
  --output text
arn:aws:rds:us-east-1:123456789102:db:db-instance-2447
arn:aws:rds:us-east-1:123456789102:db:db-instance-3395
arn:aws:rds:us-east-1:123456789102:db:dev-test-db-instance
arn:aws:rds:us-east-1:123456789102:db:pg2-db-instance
```

Tip

Vous pouvez utiliser l'attribution de balises et la recherche d'instances de base de données grâce à ces balises pour réduire les coûts par d'autres moyens. Par exemple, prenez ce scénario avec des instances de base de données utilisées pour le développement et les tests. Dans ce cas, vous pouvez désigner certaines instances de base de données à supprimer à la fin de chaque journée. Vous pouvez également les désigner pour que leurs instances de base de données soient remplacées par de petites classes d'instances de base de données pendant les périodes de faible utilisation prévues.

Noms des ressources Amazon (ARNs) dans Amazon RDS

Les ressources créées dans Amazon Web Services sont chacune identifiées de manière unique par un nom de ressource Amazon (ARN). Pour certaines RDS opérations Amazon, vous devez identifier de manière unique une RDS ressource Amazon en la spécifiantARN. Par exemple, lorsque vous créez une réplique de lecture d'RDSinstance de base de données, vous devez fournir le ARN pour l'instance de base de données source.

Pour plus d'informations sur la création ARN et l'obtention d'un existantARN, consultez les rubriques suivantes.

Rubriques

- Création d'un ARN pour Amazon RDS
- Obtenir un compte existant ARN pour Amazon RDS

Création d'un ARN pour Amazon RDS

Les ressources créées dans Amazon Web Services sont chacune identifiées de façon unique par un Amazon Resource Name (ARN). Vous pouvez construire un ARN pour une ressource Amazon RDS en utilisant la syntaxe suivante.

`arn:aws:rds:<region>:<account number>:<resourcetype>:<name>`

Nom de la région	Région	Point de terminaison	Protocole
US East (Ohio)	us-east-2	rds.us-east-2.amazonaws.com	HTTPS
		rds-fips.us-east-2.api.aws	HTTPS
		rds.us-east-2.api.aws	HTTPS
		rds-fips.us-east-2.amazonaws.com	HTTPS
US East (N. Virginia)	us-east-1	rds.us-east-1.amazonaws.com	HTTPS
		rds-fips.us-east-1.api.aws	HTTPS
		rds-fips.us-east-1.amazonaws.com	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
		rds.us-east-1.api.aws	HTTPS
USA Ouest (Californie du Nord)	us-west-1	rds.us-west-1.amazonaws.com	HTTPS
		rds.us-west-1.api.aws	HTTPS
		rds-fips.us-west-1.amazonaws.com	HTTPS
		rds-fips.us-west-1.api.aws	HTTPS
US West (Oregon)	us-west-2	rds.us-west-2.amazonaws.com	HTTPS
		rds-fips.us-west-2.amazonaws.com	HTTPS
		rds.us-west-2.api.aws	HTTPS
		rds-fips.us-west-2.api.aws	HTTPS
Afrique (Le Cap)	af-south-1	rds.af-south-1.amazonaws.com	HTTPS
		rds.af-south-1.api.aws	HTTPS
Asie-Pacifique (Hong Kong)	ap-east-1	rds.ap-east-1.amazonaws.com	HTTPS
		rds.ap-east-1.api.aws	HTTPS
Asie-Pacifique (Hyderabad)	ap-south-2	rds.ap-south-2.amazonaws.com	HTTPS
		rds.ap-south-2.api.aws	HTTPS
Asie-Pacifique (Jakarta)	ap-southeast-3	rds.ap-southeast-3.amazonaws.com	HTTPS
		rds.ap-southeast-3.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Malaisie)	ap-southeast-5	rds.ap-southeast-5.amazonaws.com	HTTPS
Asie-Pacifique (Melbourne)	ap-southeast-4	rds.ap-southeast-4.amazonaws.com rds.ap-southeast-4.api.aws	HTTPS HTTPS
Asia Pacific (Mumbai)	ap-south-1	rds.ap-south-1.amazonaws.com rds.ap-south-1.api.aws	HTTPS HTTPS
Asie-Pacifique (Osaka)	ap-northeast-3	rds.ap-northeast-3.amazonaws.com rds.ap-northeast-3.api.aws	HTTPS HTTPS
Asia Pacific (Seoul)	ap-northeast-2	rds.ap-northeast-2.amazonaws.com rds.ap-northeast-2.api.aws	HTTPS HTTPS
Asie-Pacifique (Singapour)	ap-southeast-1	rds.ap-southeast-1.amazonaws.com rds.ap-southeast-1.api.aws	HTTPS HTTPS
Asia Pacific (Sydney)	ap-southeast-2	rds.ap-southeast-2.amazonaws.com rds.ap-southeast-2.api.aws	HTTPS HTTPS
Asie-Pacifique (Taipei)	ap-est-2	rds.ap-east-2.amazonaws.com	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Asie-Pacifique (Thaïlande)	ap-southeast-7	rds.ap-southeast-7.amazonaws.com	HTTPS
Asie-Pacifique (Tokyo)	ap-northeast-1	rds.ap-northeast-1.amazonaws.com	HTTPS
		rds.ap-northeast-1.api.aws	HTTPS
Canada (Central)	ca-central-1	rds.ca-central-1.amazonaws.com	HTTPS
		rds.ca-central-1.api.aws	HTTPS
		rds-fips.ca-central-1.api.aws	HTTPS
		rds-fips.ca-central-1.amazonaws.com	HTTPS
Canada Ouest (Calgary)	ca-west-1	rds.ca-west-1.amazonaws.com	HTTPS
		rds-fips.ca-west-1.amazonaws.com	HTTPS
Europe (Francfort)	eu-central-1	rds.eu-central-1.amazonaws.com	HTTPS
		rds.eu-central-1.api.aws	HTTPS
Europe (Irlande)	eu-west-1	rds.eu-west-1.amazonaws.com	HTTPS
		rds.eu-west-1.api.aws	HTTPS
Europe (Londres)	eu-west-2	rds.eu-west-2.amazonaws.com	HTTPS
		rds.eu-west-2.api.aws	HTTPS
Europe (Milan)	eu-south-1	rds.eu-south-1.amazonaws.com	HTTPS
		rds.eu-south-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
Europe (Paris)	eu-west-3	rds.eu-west-3.amazonaws.com	HTTPS
		rds.eu-west-3.api.aws	HTTPS
Europe (Espagne)	eu-south-2	rds.eu-south-2.amazonaws.com	HTTPS
		rds.eu-south-2.api.aws	HTTPS
Europe (Stockholm)	eu-north-1	rds.eu-north-1.amazonaws.com	HTTPS
		rds.eu-north-1.api.aws	HTTPS
Europe (Zurich)	eu-central-2	rds.eu-central-2.amazonaws.com	HTTPS
		rds.eu-central-2.api.aws	HTTPS
Israël (Tel Aviv)	il-central-1	rds.il-central-1.amazonaws.com	HTTPS
		rds.il-central-1.api.aws	HTTPS
Mexique (centre)	mx-central-1	rds.mx-central-1.amazonaws.com	HTTPS
Moyen-Orient (Bahreïn)	me-south-1	rds.me-south-1.amazonaws.com	HTTPS
		rds.me-south-1.api.aws	HTTPS
Moyen-Orient (EAU)	me-central-1	rds.me-central-1.amazonaws.com	HTTPS
		rds.me-central-1.api.aws	HTTPS
Amérique du Sud (São Paulo)	sa-east-1	rds.sa-east-1.amazonaws.com	HTTPS
		rds.sa-east-1.api.aws	HTTPS

Nom de la région	Région	Point de terminaison	Protocole
AWS GovCloud (USA Est)	us-gov-east-1	rds.us-gov-east-1.amazonaws.com rds.us-gov-east-1.api.aws	HTTPS HTTPS
AWS GovCloud (US-Ouest)	us-gov-west-1	rds.us-gov-west-1.amazonaws.com rds.us-gov-west-1.api.aws	HTTPS HTTPS

Le tableau suivant indique le format à utiliser lors de la création d'un ARN pour un type de ressource Amazon RDS particulier.

Type de ressource	Format ARN
instance de base de données	arn:aws:rds : ::db : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :db:my-mysql-instance-1`
Cluster DB	arn:aws:rds : ::cluster : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :cluster: my-aurora-cluster-1`
Abonnement aux événements	arn:aws:rds : ::es : *<region>* *<account>* *<name>* Par exemple :

Type de ressource	Format ARN
	`arn:aws:rds: us-east-2 :123456789012 :es:my-subscription`
Groupe d'options DB	arn:aws:rds : ::og : *<region> <account> <name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :og:my-og`
Groupe de paramètres de base de données	arn:aws:rds : ::pg : *<region> <account> <name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :pg:my-param-enable-logs`
Groupe de paramètres de cluster DB	arn:aws:rds : ::cluster-pg : *<region> <account> <name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :cluster-pg: my-cluster-param-timezone`
instance de base de données réservée	arn:aws:rds : ::ri : *<region> <account> <name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :ri:my-reserved-postgresql`
Groupe de sécurité de base de données	arn:aws:rds : ::secgrp : *<region> <account> <name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :secgrp:my-public`

Type de ressource	Format ARN
Instantané de base de données automatique	arn:aws:rds : :snapshot:rds : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :snapshot:rds: my-mysql-db-2019-07-22-07-23`
Instantané de cluster de base de données automatique	arn:aws:rds : ::cluster-snapshot:rds : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :cluster-snapshot:rds: my-aurora-cluster-2019-07-22-16-16`
Instantané de base de données manuel	arn:aws:rds : ::snapshot : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :snapshot: my-mysql-db-snap`
Instantané de cluster de base de données manuel	arn:aws:rds : ::cluster-snapshot : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :cluster-snapshot: my-aurora-cluster-snap`
Groupe de sous-réseaux DB	arn:aws:rds : ::subgrp : *<region>* *<account>* *<name>* Par exemple : `arn:aws:rds: us-east-2 :123456789012 :subgrp:my-subnet-10`

Obtenir un compte existant ARN pour Amazon RDS

Vous pouvez obtenir le ARN nom d'une RDS ressource en utilisant le AWS Management Console, AWS Command Line Interface (AWS CLI) ou RDSAPI.

Console

Pour en obtenir un ARN AWS Management Console, naviguez jusqu'à la ressource ARN pour laquelle vous souhaitez obtenir un nom et consultez les détails de cette ressource.

Par exemple, vous pouvez obtenir le ARN pour une instance de base de données dans l'onglet Configuration des détails de l'instance de base de données.

AWS CLI

Pour obtenir un ARN depuis une RDS ressource donnée, vous devez utiliser la `describe` commande correspondant à cette ressource. AWS CLI Le tableau suivant présente chaque AWS CLI commande, ainsi que la ARN propriété utilisée avec la commande pour obtenir unARN.

AWS CLI commande	ARNpropriété
describe-event-subscriptions	EventSubscriptionArn
describe-certificates	CertificateArn
describe-db-parameter-groups	DBParameterGroupArn
describe-db-cluster-parameter-groupes	DBClusterParameterGroupArn
describe-db-instances	DBInstanceArn
describe-db-security-groups	DBSecurityGroupArn
describe-db-snapshots	DBSnapshotArn
describe-events	SourceArn
describe-reserved-db-instances	eservedDBInstanceArne R
describe-db-subnet-groups	DBSubnetGroupArn

AWS CLI commande	ARNpropriété
describe-option-groups	OptionGroupArn
describe-db-clusters	DBClusterArn
describe-db-cluster-snapshots	DBClusterSnapshotArn

Par exemple, la AWS CLI commande suivante obtient le ARN pour une instance de base de données.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-instances \
--db-instance-identifier DBInstanceIdentifier \
--region us-west-2 \
--query "*[].{DBInstanceIdentifier:DBInstanceIdentifier,DBInstanceArn:DBInstanceArn}"
```

Dans Windows :

```
aws rds describe-db-instances ^
--db-instance-identifier DBInstanceIdentifier ^
--region us-west-2 ^
--query "*[].{DBInstanceIdentifier:DBInstanceIdentifier,DBInstanceArn:DBInstanceArn}"
```

La sortie de cette commande se présente comme suit :

```
[
    {
        "DBInstanceArn": "arn:aws:rds:us-west-2:account_id:db:instance_id",
        "DBInstanceIdentifier": "instance_id"
    }
]
```

RDS API

Pour obtenir un ARN pour une RDS ressource particulière, vous pouvez effectuer les RDS API opérations suivantes et utiliser les ARN propriétés indiquées ci-dessous.

RDSAPIopération	ARNpropriété
DescribeEventSubscriptions	EventSubscriptionArn
DescribeCertificates	CertificateArn
escribeDBParameterGroupes D	DBParameterGroupArn
D escribeDBCluster Parameter Groups	DBClusterParameterGroupArn
D escribeDBInstances	DBInstanceArn
escribeDBSecurityGroupes D	DBSecurityGroupArn
D escribeDBSnapshots	DBSnapshotArn
DescribeEvents	SourceArn
DescribeReservedDBInstances	eservedDBInstanceArne R
escribeDBSubnetGroupes D	DBSubnetGroupArn
DescribeOptionGroups	OptionGroupArn
D escribeDBClusters	DBClusterArn
D escribeDBCluster Instantanés	DBClusterSnapshotArn

Utilisation du stockage pour les RDS instances de base de données Amazon

Pour spécifier la manière dont vous souhaitez que vos données soient stockées sur AmazonRDS, choisissez un type de stockage et indiquez une taille de stockage lorsque vous créez ou modifiez une instance de base de données. Ensuite, vous pouvez augmenter le volume ou modifier le type de stockage en changeant l'instance de base de données. Pour plus d'informations sur le type de stockage à utiliser pour votre charge de travail, consultez Types de stockage Amazon RDS.

Rubriques

- Augmentation de la capacité de stockage d'une instance de base de données
- Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS
- Mise à niveau du système de fichiers de stockage d'une instance de base de données
- Modification des paramètres du stockage provisionné IOPS SSD
- Modifications du stockage à forte intensité d'I/O
- Modification des paramètres de stockage SSD à usage général (gp3)
- Utilisation d'un volume dédié aux journaux (DLV)

Augmentation de la capacité de stockage d'une instance de base de données

Si vous avez besoin d'espace pour des données supplémentaires, vous pouvez augmenter l'espace de stockage d'une instance de base de données existante. Pour cela, vous pouvez utiliser la console de gestion Amazon RDS, l'API Amazon RDS ou l' AWS Command Line Interface (AWS CLI). Pour de plus amples informations sur les limites de stockage, veuillez consulter Stockage d'instance de base de données Amazon RDS.

> **Note**
>
> Vous ne pouvez pas réduire le volume de stockage d'une instance de base de données une fois qu'il a été alloué. Lorsque vous augmentez la valeur du stockage alloué, vous devez le faire d'au moins 10 %. Si vous tentez d'augmenter la valeur de moins de 10 %, une erreur s'affiche.

Le dimensionnement du stockage pour les instances de base de données RDS for SQL Server n'est pris en charge que pour les types de stockage SSD à usage général et SSD à IOPS provisionnées.

Pour surveiller la quantité de stockage disponible pour votre instance de base de données afin de pouvoir réagir en cas de besoin, nous vous recommandons de créer une CloudWatch alarme Amazon. Pour plus d'informations sur le réglage des CloudWatch alarmes, consultez la section Utilisation des CloudWatch alarmes.

La mise à l'échelle du stockage ne provoque généralement aucune panne ou dégradation des performances de l'instance de base de données. Après la modification de la taille de stockage d'une instance de base de données, l'instance passe à l'état storage-optimization.

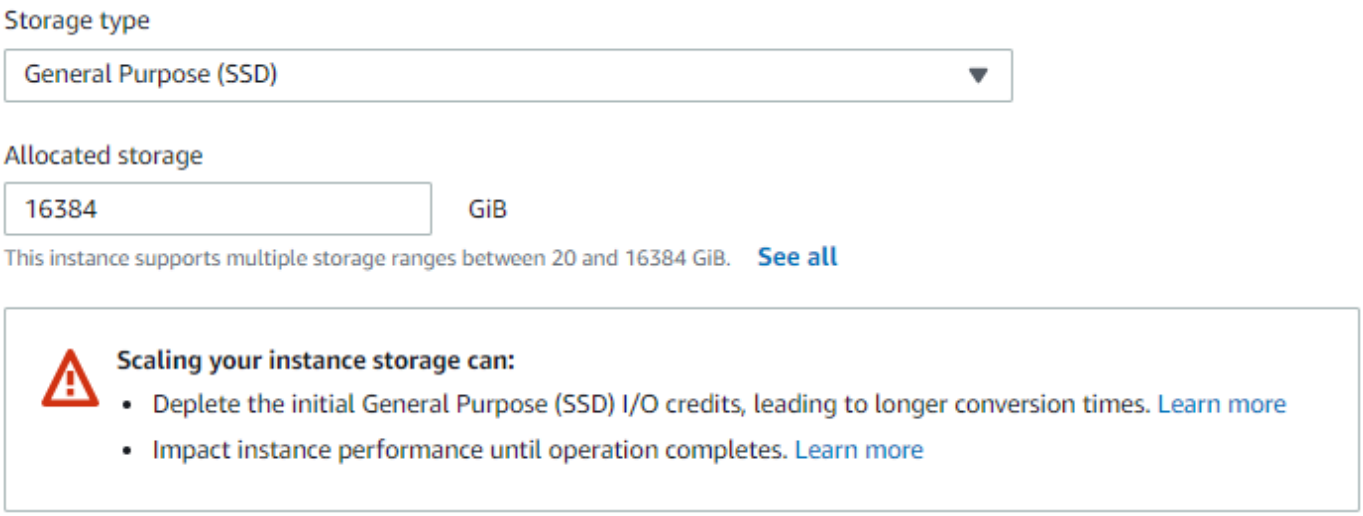
Note

L'optimisation du stockage peut prendre plusieurs heures. Vous ne pouvez pas apporter d'autres modifications au stockage avant six (6) heures ou avant la fin de l'optimisation du stockage sur l'instance, le délai le plus long prévalant. Vous pouvez consulter la progression de l'optimisation du stockage dans le AWS Management Console ou à l'aide de la describe-db-instances AWS CLI commande.

console

Pour augmenter le stockage d'une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de bases de données que vous souhaitez modifier.
4. Sélectionnez Modify.
5. Saisissez une nouvelle valeur pour Stockage alloué. Elle doit être supérieure à la valeur actuelle.



6. Choisissez Continuer pour passer à l'écran suivant.
7. Choisissez Appliquer immédiatement dans la section Planification des modifications pour appliquer immédiatement les modifications du stockage à l'instance de base de données.

   Ou choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée pour appliquer les modifications pendant la prochaine fenêtre de maintenance.
8. Lorsque les paramètres vous conviennent, choisissez Modifier l'instance de base de données.

AWS CLI

Pour augmenter le stockage d'une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`. Définissez les paramètres suivants :

- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `--apply-immediately` – Utilisez `--apply-immediately` pour appliquer immédiatement les modifications apportées au stockage.

  Vous pouvez également utiliser `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance. Une interruption immédiate a lieu lorsque les modifications sont appliquées.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

API RDS

Pour accroître l'espace de stockage d'une instance de base de données, utilisez l'opération d'API Amazon RDS `ModifyDBInstance`. Définissez les paramètres suivants :

- `AllocatedStorage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `ApplyImmediately` – Définissez cette option sur `True` pour appliquer les modifications de stockage immédiatement. Définissez cette option sur `False` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance. Une interruption immédiate a lieu lorsque les modifications sont appliquées.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS

Si votre charge de travail est imprévisible, vous pouvez activer la scalabilité automatique du stockage pour une instance de base de données Amazon RDS. Pour cela, vous pouvez utiliser la console Amazon RDS, l'API Amazon RDS ou l' AWS CLI.

Par exemple, vous pouvez utiliser cette fonctionnalité pour une nouvelle application mobile de jeu que les utilisateurs adoptent rapidement. Dans ce cas, une charge de travail qui augmente rapidement peut dépasser le stockage de base de données disponible. Pour éviter d'avoir à augmenter manuellement la capacité de stockage d'une base de données, vous pouvez utiliser le dimensionnement automatique du stockage Amazon RDS.

Si le dimensionnement automatique du stockage est activé, quand Amazon RDS détecte que vous êtes à court d'espace disponible dans la base de données, il augmente automatiquement l'échelle de votre stockage. Amazon RDS commence à modifier le stockage pour une instance de base de données pour laquelle la scalabilité automatique est activée si les conditions suivantes sont réunies :

- L'espace disponible est inférieur ou égal à 10 % du stockage alloué.
- La condition de stockage faible dure au moins cinq minutes.
- Au moins six heures se sont écoulées depuis la dernière modification du stockage ou l'optimisation du stockage s'est terminée sur l'instance, selon le délai le plus long.

Le stockage supplémentaire s'effectue par incréments de la valeur la plus élevée parmi les suivantes :

- 10 Gio
- 10 % du stockage actuellement alloué
- Croissance de stockage prévue dépassant la taille de stockage actuellement allouée au cours des 7 prochaines heures sur la base des métriques `FreeStorageSpace` de l'heure précédente. Pour plus d'informations sur les métriques, consultez la section Surveillance avec Amazon CloudWatch.

Le seuil de stockage maximal correspond à la limite que vous définissez pour la mise à l'échelle automatique de l'instance de base de données. Il présente les contraintes suivantes :

- Vous devez définir le seuil de stockage maximum à une valeur au moins 10 % supérieure au stockage actuellement alloué. Nous vous recommandons de le fixer à au moins 26 % de plus pour éviter de recevoir une notification d'événement indiquant que la taille de stockage approche du seuil de stockage maximal.

  Par exemple, si vous avez une instance de base de données avec 1 000 Go de stockage alloué, alors définissez le seuil de stockage maximum à au moins 1 100 Go. Si ce n'est pas le cas, vous obtenez un message d'erreur tel que Taille maximale de stockage non valide pour *`engine_name`*. Cependant, nous vous recommandons de définir le seuil de stockage maximum à au moins 1 260 Gio pour éviter la notification d'événement.
- Pour une instance de base de données qui utilise le stockage IOPS provisionné (io1 ou io2 Block Express), le rapport entre le nombre d'IOPS et le seuil de stockage maximal (en GiB) doit être compris dans une certaine plage. Pour de plus amples informations, veuillez consulter Stockage SSD d'IOPS par seconde provisionnées.
- Vous ne pouvez pas définir le seuil de stockage maximum pour les instances à mise à l'échelle automatique à une valeur supérieure au stockage maximum alloué pour le moteur de base de données et la classe d'instance de base de données.

  Par exemple, SQL Server Standard Edition on db.m5.xlarge possède un stockage alloué par défaut pour l'instance de 20 GiB (le minimum) et un stockage alloué maximum de 16 384 GiB. Le seuil de stockage maximum par défaut pour la scalabilité automatique est de 1 000 GiB. Si vous utilisez ce paramètre par défaut, l'instance ne se met pas automatiquement à l'échelle au-dessus de 1 000 GiB. Ceci est vrai même si le stockage alloué maximum pour l'instance est de 16 384 GiB.

> **Note**
>
> Nous vous recommandons de choisir soigneusement le seuil de stockage maximal en fonction des habitudes d'utilisation et des besoins des clients. En cas d'aberrations au niveau des habitudes d'utilisation, le seuil de stockage maximal peut empêcher la mise à l'échelle du stockage à une valeur trop élevée lorsque la mise à l'échelle automatique prédit un seuil très élevé. Une fois qu'une instance de base de données a été mise à l'échelle automatique, son stockage alloué ne peut pas être réduit.

Rubriques

- Limites de la mise à l'échelle automatique du stockage
- Activation du dimensionnement automatique du stockage pour une nouvelle instance de base de données
- Modification du paramètre de dimensionnement automatique du stockage pour une instance de base de données
- Dimensionnement manuel de votre instance de base de données vers le bas ou vers
- Désactivation du dimensionnement automatique du stockage pour une instance de base de données

Limites de la mise à l'échelle automatique du stockage

Les limitations suivantes s'appliquent à la mise à l'échelle automatique du stockage :

- Le dimensionnement automatique ne se produit pas si le seuil de stockage maximum peut être dépassé par l'incrémentation du stockage.
- Lors de la mise à l'échelle automatique, RDS prévoit la taille du stockage pour les opérations de mise à l'échelle automatique ultérieures. S'il prévoit qu'une opération ultérieure dépassera le seuil de stockage maximal, RDS met automatiquement à l'échelle sur le seuil de stockage maximal.
- La scalabilité automatique ne peut pas complètement empêcher les situations de stockage plein pour les charges de données volumineuses. En effet, d'autres modifications au stockage ne peuvent pas être effectuées avant six (6) heures ou avant la fin de l'optimisation du stockage sur l'instance, selon le délai le plus long.

Si vous effectuez un chargement de données volumineux et que le dimensionnement automatique ne fournit pas suffisamment d'espace, la base de données peut rester à l'état de stockage plein pendant plusieurs heures. Cela peut nuire à la base de données.

- Si vous lancez une opération de dimensionnement du stockage en même temps qu'Amazon RDS, votre modification du stockage est prioritaire. L'opération de dimensionnement automatique est annulée.
- La mise à l'échelle automatique ne peut pas diminuer le stockage alloué. Vous ne pouvez pas réduire le volume de stockage d'une instance de base de données une fois qu'il a été alloué.
- La mise à l'échelle automatique ne peut pas être utilisée avec le stockage magnétique.
- La mise à l'échelle automatique ne peut pas être utilisée avec les classes d'instance de génération précédente suivantes qui ont moins de 6 Tio de stockage ordonnable : db.m3.large, db.m3.xlarge et db.m3.2xlarge.
- Les opérations de mise à l'échelle automatique ne sont pas enregistrées. AWS CloudTrail Pour plus d'informations sur CloudTrail, voirSurveillance (Amazon RDS API appelle) AWS CloudTrail.

Bien que le dimensionnement automatique vous permette d'accroître l'espace de stockage de votre instance de base de données Amazon RDS de façon dynamique, vous devez quand même attribuer à votre instance de base de données une taille de stockage initiale adaptée à votre charge de travail habituelle.

Activation du dimensionnement automatique du stockage pour une nouvelle instance de base de données

Lorsque vous créez une nouvelle instance de base de données Amazon RDS, vous pouvez choisir d'activer ou non le dimensionnement automatique du stockage. Vous pouvez également définir une limite supérieure sur le stockage qu'Amazon RDS peut allouer pour l'instance de base de données.

L'activation de la mise à l'échelle automatique du stockage ne nécessite pas de redémarrage de la base de données et n'entraîne aucune interruption de service. La fonctionnalité prend effet immédiatement sans perturber les opérations de base de données.

> **Note**
>
> Lorsque vous clonez une instance de base de données Amazon RDS pour laquelle le dimensionnement automatique du stockage est activé, ce paramètre n'est pas hérité

automatiquement par l'instance clonée. La nouvelle instance de base de données a la même quantité de stockage alloué que l'instance d'origine. Vous pouvez activer à nouveau le dimensionnement automatique pour la nouvelle instance si l'instance clonée continue à augmenter ses exigences de stockage.

console

Pour activer le dimensionnement automatique du stockage pour une nouvelle instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la AWS région dans laquelle vous souhaitez créer l'instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données). Sur la page Sélectionner un moteur, choisissez votre moteur de base de données et spécifiez les informations de votre instance de base de données comme décrit dans Mise en route avec Amazon RDS.
5. Dans la section Storage autoscaling (Dimensionnement automatique du stockage), définissez la valeur de Maximum storage threshold (Limite de stockage maximum) pour l'instance de base de données.
6. Spécifiez le reste des informations de l'instance de base de données comme décrit dans Mise en route avec Amazon RDS.

AWS CLI

Pour activer le dimensionnement automatique du stockage pour une nouvelle instance de base de données, utilisez la AWS CLI commande `create-db-instance`. Définissez le paramètre suivant :

- `--max-allocated-storage` – Active la scalabilité automatique du stockage et définit la limite supérieure de la taille du stockage, en gigaoctets.

Pour vérifier que le dimensionnement automatique du stockage Amazon RDS est disponible pour votre instance de base de données, utilisez la AWS CLI `describe-valid-db-instance-modifications`commande. Pour vérifier en fonction de la classe de l'instance avant sa création,

utilisez la commande `describe-orderable-db-instance-options`. Vérifiez le champ suivant dans la valeur de retour :

- `SupportsStorageAutoscaling` – Indique si l'instance de base de données ou la classe d'instance prend en charge la scalabilité automatique du stockage.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

API RDS

Pour activer la mise à l'échelle automatique du stockage pour une nouvelle instance de base de données, utilisez l'opération d'API Amazon RDS `CreateDBInstance`. Définissez le paramètre suivant :

- `MaxAllocatedStorage` – Active la scalabilité automatique du stockage d'Amazon RDS et définit la limite supérieure de la taille du stockage, en gigaoctets.

Pour vérifier que la mise à l'échelle automatique du stockage Amazon RDS est disponible pour votre instance de base de données, utilisez l'opération d'API Amazon RDS `DescribeValidDbInstanceModifications` pour une instance existante ou l'opération `DescribeOrderableDBInstanceOptions` avant de créer une instance. Vérifiez le champ suivant dans la valeur de retour :

- `SupportsStorageAutoscaling` – Indique si l'instance de base de données prend en charge la scalabilité automatique du stockage.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

Modification du paramètre de dimensionnement automatique du stockage pour une instance de base de données

Vous pouvez activer le dimensionnement automatique du stockage pour une instance de base de données Amazon RDS existante. Vous pouvez également modifier la limite de stockage supérieure qu'Amazon RDS peut allouer pour l'instance de base de données.

La modification des paramètres de dimensionnement automatique du stockage ne nécessite pas de redémarrage de la base de données et n'entraîne aucune interruption de service. Les modifications prennent effet immédiatement sans perturber les opérations de base de données.

console

Pour modifier les paramètres de dimensionnement automatique du stockage pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez modifier, puis sélectionnez Modifier. La page Modifier l'instance de base de données s'affiche.
4. Modifiez la limite de stockage dans la section Dimensionnement automatique. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
5. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez les modifications.
6. Sur la page de confirmation, examinez vos modifications. S'ils sont corrects, choisissez Modifier l'instance de base de données pour enregistrer vos modifications. Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

   La modification de la limite de scalabilité automatique du stockage prend effet immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.

AWS CLI

Pour modifier les paramètres de mise à l'échelle automatique du stockage pour une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`. Définissez le paramètre suivant :

- `--max-allocated-storage` – Définit la limite de stockage supérieure du stockage, en gigaoctets. Si la valeur est supérieure au paramètre `--allocated-storage`, le dimensionnement automatique du stockage est activé. Si la valeur est égale au paramètre `--allocated-storage`, le dimensionnement automatique du stockage est désactivé.

Pour vérifier que le dimensionnement automatique du stockage Amazon RDS est disponible pour votre instance de base de données, utilisez la AWS CLI `describe-valid-db-instance-modifications`commande. Pour vérifier en fonction de la classe de l'instance avant sa création, utilisez la commande `describe-orderable-db-instance-options`. Vérifiez le champ suivant dans la valeur de retour :

- `SupportsStorageAutoscaling` – Indique si l'instance de base de données prend en charge la scalabilité automatique du stockage.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

API RDS

Pour modifier les paramètres de mise à l'échelle automatique du stockage pour une instance de base de données, utilisez l'opération d'API Amazon RDS `ModifyDBInstance`. Définissez le paramètre suivant :

- `MaxAllocatedStorage` – Définit la limite de stockage supérieure du stockage, en gigaoctets.

Pour vérifier que la mise à l'échelle automatique du stockage Amazon RDS est disponible pour votre instance de base de données, utilisez l'opération d'API Amazon RDS `DescribeValidDbInstanceModifications` pour une instance existante ou l'opération `DescribeOrderableDBInstanceOptions` avant de créer une instance. Vérifiez le champ suivant dans la valeur de retour :

- `SupportsStorageAutoscaling` – Indique si l'instance de base de données prend en charge la scalabilité automatique du stockage.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

Dimensionnement manuel de votre instance de base de données vers le bas ou vers

Amazon RDS propose une mise à l'échelle automatique du stockage pour répondre à la demande croissante. Cependant, il existe des limites en ce qui concerne la réduction et dans :

- Stockage RDS : bien que RDS prenne en charge l'augmentation automatique du stockage à mesure que la demande augmente, il ne le réduit pas automatiquement.
- Répliques de lecture : RDS ne prend pas en charge le dimensionnement automatique (ajout) ou redimensionnement (suppression) des répliques de lecture. Vous devez ajouter ou supprimer manuellement des répliques de lecture en fonction de vos exigences de charge.

Pour réduire vos ressources RDS, effectuez les actions manuelles suivantes :

- Pour le stockage, vous ne pouvez pas réduire manuellement le stockage alloué d'une instance de base de données à l'aide de la `modify-db-instance` commande. Choisissez plutôt l'une des techniques suivantes :
  - Utilisez un blue/green déploiement si votre moteur de base de données le prend en charge. Créez une base de données verte avec une taille de stockage inférieure, puis faites de votre base de données verte votre base de données bleue. Pour de plus amples informations, veuillez consulter Modifier les paramètres de stockage et de performance.
  - Créez une nouvelle instance de base de données avec un espace de stockage alloué inférieur, migrez manuellement les données de votre base de données actuelle vers la nouvelle instance de base de données et changez de point de terminaison de base de données.
- Pour les répliques en lecture, supprimez manuellement les répliques inutilisées via la console RDS ou. AWS CLI

Désactivation du dimensionnement automatique du stockage pour une instance de base de données

Si vous n'avez plus besoin qu'Amazon RDS augmente automatiquement le capacité de stockage d'une instance de base de données Amazon RDS, vous pouvez désactiver le dimensionnement automatique du stockage. Après avoir désactivé le dimensionnement automatique, vous pouvez toujours augmenter manuellement le volume de stockage de votre instance de base de données.

console

Pour désactiver le dimensionnement automatique du stockage pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez modifier, puis sélectionnez Modifier. La page Modifier l'instance de base de données s'affiche.
4. Cochez la case Enable storage autoscaling (Activer le dimensionnement automatique du stockage) dans la section Storage autoscaling (Dimensionnement automatique du stockage. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
5. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez les modifications.
6. Sur la page de confirmation, examinez vos modifications. S'ils sont corrects, choisissez Modifier l'instance de base de données pour enregistrer vos modifications. Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

La modification de la limite de scalabilité automatique du stockage prend effet immédiatement. Ce paramètre ignore le paramètre Appliquer immédiatement.

AWS CLI

Pour désactiver le dimensionnement automatique du stockage pour une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`et le paramètre suivant :

- `--max-allocated-storage` – Spécifiez une valeur égale au paramètre `--allocated-storage` pour la scalabilité automatique du stockage Amazon RDS pour l'instance de base de données spécifiée.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

API RDS

Pour désactiver la mise à l'échelle automatique du stockage pour une instance de base de données, utilisez l'opération d'API Amazon RDS `ModifyDBInstance`. Définissez le paramètre suivant :

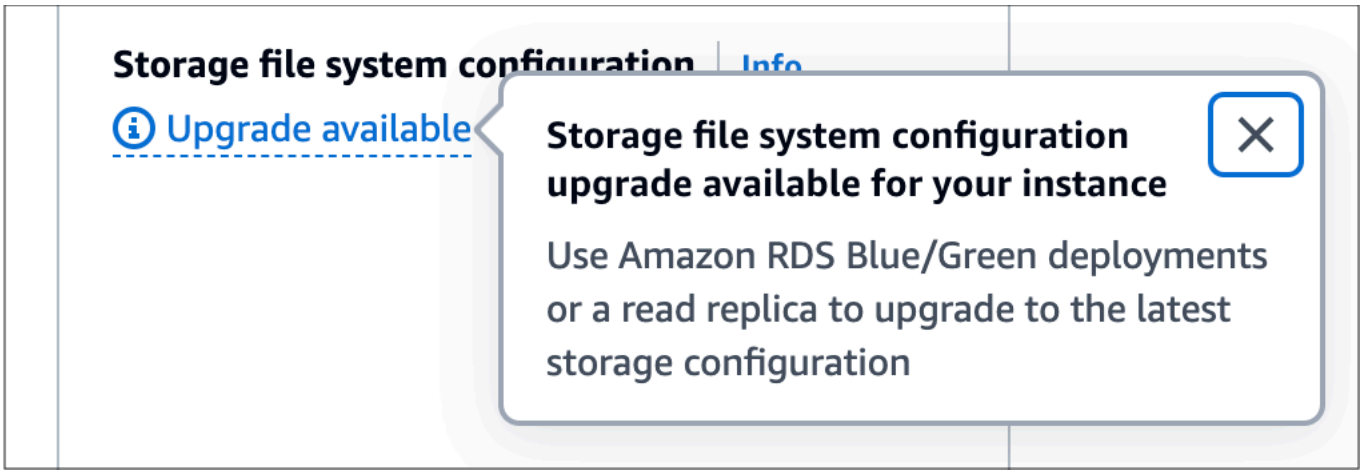
- `MaxAllocatedStorage` – Spécifiez une valeur égale au paramètre `AllocatedStorage` pour la scalabilité automatique du stockage Amazon RDS pour l'instance de base de données spécifiée.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

Mise à niveau du système de fichiers de stockage d'une instance de base de données

La plupart des instances de base de données RDS offrent une taille de stockage maximale de 64 TiB pour les bases de données RDS MariaDB, MySQL et PostgreSQL. Toutefois, sur certains systèmes de fichiers 32 bits anciens, les capacités de stockage peuvent être inférieures. Pour déterminer la capacité de stockage de votre instance de base de données, utilisez la AWS CLI commande describe-valid-db-instance-modifications.

RDS vérifie si votre système de stockage dispose d'une taille de stockage de 16 TiB, d'une limite de taille de fichier de 2 TiB ou d'écritures non optimisées. Si vos instances de base de données répondent à ces conditions, RDS vous avertit que la configuration de votre système de fichiers est éligible à une mise à niveau. Vous pouvez vérifier l'éligibilité à la mise à niveau d'une instance de base de données sur le panneau Stockage de la page de détails de l'instance de base de données.



Si votre instance de base de données est éligible à une mise à niveau du système de fichiers, utilisez l'une des techniques suivantes :

- Créez un déploiement bleu/vert et spécifiez l'option Mettre à niveau la configuration du système de fichiers de stockage.

  Cette option met à niveau le système de fichiers dans l'environnement vert vers la configuration préférée. Vous pouvez ensuite basculer le déploiement bleu/vert, qui favorise l'environnement vert comme nouvel environnement de production. Pour obtenir des instructions complètes, veuillez consulter the section called “Création d'un déploiement bleu/vert”.

- Créez un réplica en lecture d'instance de base de données et spécifiez l'option Mettre à niveau la configuration du système de fichiers de stockage.

  Cette option met à niveau le système de fichiers du réplica en lecture vers la configuration préférée. Vous pouvez ensuite promouvoir le réplica en lecture en tant qu'instance autonome. Pour obtenir des instructions complètes, veuillez consulter the section called “Création d'un réplica en lecture”.

Pendant la mise à niveau du stockage, votre moteur de base de données n'est pas disponible. La mise à niveau de la configuration du stockage est une opération à fort taux d'E/S et entraîne des délais de création plus longs pour les réplica en lecture et les déploiements bleu/vert. Le processus de mise à niveau du stockage est plus rapide lorsque les deux conditions suivantes sont réunies :

- L'instance de base de données source utilise un stockage SSD IOPS provisionné (io1 ou io2 Block Express).
- Vous avez provisionné l'environnement écologique ou lu une réplique avec une taille d'instance égale ou supérieure à 4 fois.

Les mises à niveau du stockage impliquant un stockage General Purpose SSD (gp2) peuvent épuiser votre solde de crédit d'E/S, ce qui entraîne des temps de mise à niveau plus longs. Pour de plus amples informations, veuillez consulter the section called “Stockage d'instance de base de données”.

Lors d'une mise à niveau du stockage, RDS augmente la taille de stockage allouée de 10 % pour l'instance verte ou pour la réplication en lecture si les deux conditions suivantes sont remplies :

- La consommation de stockage sur votre instance de base de données source est supérieure ou égale à 90 % de la taille de stockage allouée.
- Le dimensionnement automatique du stockage est activé.

RDS désactive le dimensionnement automatique lorsque la nouvelle taille de stockage est supérieure ou égale au stockage maximal alloué défini pour l'instance. Si le dimensionnement automatique du stockage est désactivé avant le début de la mise à niveau du stockage, la taille du stockage n'augmente pas pendant la mise à niveau.

Modification des paramètres du stockage provisionné IOPS SSD

Vous pouvez modifier les paramètres d'une instance de base de données qui utilise le IOPS SSD stockage provisionné à l'aide de la RDS console Amazon ou d'Amazon RDSAPI. AWS CLI Spécifiez le type de stockage, le stockage alloué et la quantité de stockage provisionné IOPS dont vous avez besoin. La plage dépend du moteur de base de données et du type d'instance

Bien que vous puissiez réduire le volume IOPS provisionné pour votre instance, vous ne pouvez pas réduire la taille du stockage.

Dans la plupart des cas, le dimensionnement du stockage ne requiert pas d'interruption et ne dégrade pas les performances du serveur. Une fois que vous avez modifié le stockage IOPS d'une instance de base de données, le statut de l'instance de base de données est Optimisation du stockage.

> **Note**
>
> L'optimisation du stockage peut prendre plusieurs heures. Vous ne pouvez pas apporter d'autres modifications au stockage avant six (6) heures ou avant la fin de l'optimisation du stockage sur l'instance, le délai le plus long prévalant.

Pour plus d'informations sur les plages de stockage allouées et provisionnées IOPS disponibles pour chaque moteur de base de données, consultezStockage SSD d'IOPS par seconde provisionnées.

Console

Pour modifier les IOPS paramètres provisionnés d'une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

   Pour filtrer la liste des instances de base de données, dans Filtrer les bases de données, entrez une chaîne de texte RDS à utiliser par Amazon pour filtrer les résultats. Seules les instances de bases de données dont les noms contiennent cette chaîne apparaissent.
3. Choisissez l'instance de base de données avec Provisioned IOPS que vous souhaitez modifier.
4. Sélectionnez Modifier.

5. Sur la page Modifier l'instance de base de données, choisissez Provisioned IOPS SSD (io1) ou Provisioned IOPS SSD (io2) pour le type de stockage.
6. Pour Provisioned IOPS, entrez une valeur.

   Si la valeur que vous spécifiez pour le stockage alloué ou provisionné IOPS dépasse les limites prises en charge par l'autre paramètre, un message d'avertissement s'affiche. Ce message indique la plage de valeurs requise pour l'autre paramètre.
7. Choisissez Continuer.
8. Choisissez Apply immediately (Appliquer immédiatement) dans la section Scheduling of modifications (Planification des modifications) pour appliquer immédiatement les modifications à l'instance de base de données. Ou choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée pour appliquer les modifications pendant la prochaine fenêtre de maintenance.
9. Passez en revue les paramètres à modifier et choisissez Modification d'une instance de base de données pour terminer la modification.

   La nouvelle valeur pour le stockage alloué ou pour le stockage provisionné IOPS apparaît dans la colonne État.

AWS CLI

Pour modifier le IOPS paramètre Provisioned d'une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`. Définissez les paramètres suivants :

- `--storage-type`— Défini sur `io1` ou `io2` pour Provisioned. IOPS
- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `--iops`— La nouvelle quantité de provisionnement IOPS pour l'instance de base de données, exprimée en opérations d'E/S par seconde.
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer les modifications immédiatement. Utilisez `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications pendant la prochaine fenêtre de maintenance.

RDS API

Pour modifier les IOPS paramètres provisionnés d'une instance de base de données, utilisez l'RDSAPIopération `ModifyDBInstance`Amazon. Définissez les paramètres suivants :

- `StorageType`— Défini sur `io1` ou `io2` pour Provisioned. IOPS
- `AllocatedStorage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `Iops`— Le nouveau IOPS taux pour l'instance de base de données, exprimé en opérations d'E/S par seconde.
- `ApplyImmediately` : définissez cette option sur `True` pour appliquer les modifications immédiatement. Définissez cette option sur `False` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

Modifications du stockage à forte intensité d'I/O

Les instances Amazon RDS DB utilisent les volumes Amazon Elastic Block Store (EBS) pour le stockage des bases de données et des journaux. En fonction de la quantité de stockage demandée, RDS (sauf RDS pour le SQL serveur) répartit automatiquement plusieurs EBS volumes Amazon afin d'améliorer les performances. RDSLes instances de base de données avec types de SSD stockage sont soutenues par un ou quatre EBS volumes Amazon répartis par bandes dans une configuration RAID 0. De par leur conception, les opérations de modification du stockage pour une RDS instance de base de données ont un impact minimal sur les opérations de base de données en cours.

Dans la plupart des cas, les modifications de dimensionnement du stockage sont entièrement transférées vers la EBS couche Amazon et sont transparentes pour la base de données. Ce processus s'effectue généralement en quelques minutes. Toutefois, certains volumes RDS de stockage plus anciens nécessitent un processus différent pour modifier la taille, le provisionnement IOPS ou le type de stockage. Cela implique de faire une copie complète des données en utilisant une opération potentiellement intensive en I/O.

La modification du stockage utilise une opération à forte intensité d'I/O si l'un des facteurs suivants s'applique :

- Le type de stockage source est magnétique. Le stockage magnétique ne prend pas en charge la modification élastique des volumes.

- L'RDSinstance de base de données ne se trouve pas sur une mise en EBS page Amazon en un ou quatre volumes. Vous pouvez consulter le nombre de EBS volumes Amazon utilisés sur vos RDS instances de base de données à l'aide de métriques de surveillance améliorées. Pour de plus amples informations, veuillez consulter Afficher les métriques du système d'exploitation dans la RDS console.
- La taille cible de la demande de modification augmente le stockage alloué au-dessus de 400 GiB pour les SQL instances RDS MariaDB, My et PostgreSQL, et de 200 GiB pour Oracle. RDS Les opérations de mise à l'échelle automatique du stockage ont le même effet lorsqu'elles augmentent la taille de stockage allouée de votre instance de base de données au-delà de ces seuils.

Si votre modification de stockage implique une opération à forte intensité d'I/O, elle consomme des ressources d'I/O et augmente la charge de votre instance de base de données. Les modifications de stockage associées à des opérations intensives en E/S impliquant un stockage à usage général SSD (gp2) peuvent épuiser votre solde créditeur d'E/S, ce qui entraîne des temps de conversion plus longs.

Nous recommandons, à titre de bonnes pratiques, de programmer ces requêtes de modification du stockage en dehors des heures de pointe afin de réduire le temps nécessaire à la réalisation de l'opération de modification du stockage. Vous pouvez également créer un réplica en lecture de l'instance de base de données et effectuer la modification du stockage sur le réplica en lecture. Ensuite, le réplica en lecture devient l'instance de base de données principale. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Pour plus d'informations, consultez Pourquoi une RDS instance de base de données Amazon est-elle bloquée dans l'état de modification lorsque j'essaie d'augmenter le stockage alloué ?

Modification des paramètres de stockage SSD à usage général (gp3)

Vous pouvez modifier les paramètres d'une instance de base de données qui utilise le stockage SSD à usage général (gp3) à l'aide de la console Amazon RDS ou de l' AWS CLI API Amazon RDS. Spécifiez le type de stockage, le stockage alloué, la quantité d'IOPS provisionnés et le débit de stockage dont vous avez besoin.

Bien que vous puissiez réduire la quantité d'IOPS provisionnées et le débit de stockage de votre instance de base de données, vous ne pouvez pas réduire la taille du stockage.

Dans la plupart des cas, la mise à l'échelle du stockage ne nécessite aucune interruption. Après la modification des IOPS de stockage d'une instance de base de données, l'instance passe à

l'état Optimisation du stockage. Vous pouvez vous attendre à des latences élevées, mais toujours inférieures à 10 millisecondes, lors de l'optimisation du stockage. L'instance de base de données est totalement opérationnelle après modification du stockage.

> **Note**
>
> Vous ne pouvez pas effectuer de modifications de la taille de stockage pendant six (6) heures après la fin de l'optimisation du stockage sur l'instance.

Pour obtenir des informations sur les plages de stockage alloué, les IOPS provisionnés et le débit de stockage disponibles pour chaque moteur de base de données, consultez Stockage GP3 (recommandé).

console

Pour modifier les paramètres de performance du stockage pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

   Pour filtrer la liste des instances de bases de données, pour Filter databases (Filtrer les bases de donnée), saisissez une chaîne de texte pour Amazon RDS à utiliser pour filtrer les résultats. Seules les instances de bases de données dont les noms contiennent cette chaîne apparaissent.
3. Choisissez l'instance de base de données avec le stockage gp3 que vous souhaitez modifier.
4. Sélectionnez Modifier.
5. Sur la page Modifier l'instance de base de données, choisissez le type de stockage SSD à usage général (gp3), puis procédez comme suit :

   a. Pour Provisioned IOPS (IOPS provisionnés), choisissez une valeur.

      Si la valeur que vous spécifiez pour Allocated Storage (Stockage alloué) ou Provisioned IOPS (IOPS provisionnés) sort des limites prises en charge par l'autre paramètre, un message d'avertissement apparaît. Ce message indique la plage de valeurs requise pour l'autre paramètre.

   b. Pour Storage throughput (Débit de stockage), choisissez une valeur.

Si la valeur que vous spécifiez pour Provisioned IOPS (IOPS provisionnés) ou Storage throughput (Débit de stockage) sort des limites prises en charge par l'autre paramètre, un message d'avertissement apparaît. Ce message indique la plage de valeurs requise pour l'autre paramètre.

6. Choisissez Continuer.
7. Choisissez Apply immediately (Appliquer immédiatement) dans la section Scheduling of modifications (Planification des modifications) pour appliquer immédiatement les modifications à l'instance de base de données. Ou choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée pour appliquer les modifications pendant la prochaine fenêtre de maintenance.
8. Passez en revue les paramètres à modifier et choisissez Modification d'une instance de base de données pour terminer la modification.

   La nouvelle valeur pour IOPS provisionnés apparaît dans la colonne Status (Statut).

AWS CLI

Pour modifier les paramètres de performance de stockage d'une instance de base de données, utilisez la AWS CLI commande `modify-db-instance`. Définissez les paramètres suivants :

- `--storage-type` – Définissez `gp3` pour SSD à usage général (gp3).
- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `--iops` – Nouveau volume de stockage des IOPS approvisionnées pour l'instance de base de données, exprimé en opérations d'I/O par seconde.
- `--storage-throughput`— Le nouveau débit de stockage pour l'instance de base de données, exprimé en MiBps.
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer les modifications immédiatement. Utilisez `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications pendant la prochaine fenêtre de maintenance.

API RDS

Pour modifier les paramètres de performances du stockage pour une instance de base de données, utilisez l'opération d'API Amazon RDS `ModifyDBInstance`. Définissez les paramètres suivants :

- `StorageType` – Définissez `gp3` pour SSD à usage général (gp3).
- `AllocatedStorage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `Iops` – Nouveau débit d'IOPS pour l'instance de bases de données, exprimé en opérations d'I/O par seconde.
- `StorageThroughput`— Le nouveau débit de stockage pour l'instance de base de données, exprimé en MiBps.
- `ApplyImmediately` : définissez cette option sur `True` pour appliquer les modifications immédiatement. Définissez cette option sur `False` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

Utilisation d'un volume dédié aux journaux (DLV)

Vous pouvez utiliser un volume de journal dédié (DLV) pour une instance de base de données qui utilise le stockage PIOPS (Provisioned IOPS). Un DLV déplace les journaux de transactions des bases de données PostgreSQL MySQL/MariaDB redo logs and binary logs to a storage volume that's separate from the volume containing the database tables. A DLV makes transaction write logging more efficient and consistent. DLVs are ideal for databases with large allocated storage, high I/O et les exigences par seconde (IOPS), ou les charges de travail sensibles à la latence.

DLVs sont pris en charge pour le stockage PIOPS (io1 et io2 Block Express) et sont créés avec une taille fixe de 1 024 GiB et 3 000 IOPS provisionnées.

Amazon RDS prend DLVs en charge toutes Régions AWS les versions suivantes :

- MariaDB 10.6.7 et versions 10 ultérieures
- MySQL 8.0.28 et versions 8.0 supérieures, MySQL 8.4.3 et versions supérieures 8.4
- PostgreSQL 13.10 et versions 13 ultérieures, 14.7 et versions 14 ultérieures, et 15.2 et versions 15 ultérieures

RDS prend en charge les DLVs déploiements multi-AZ. Lorsque vous modifiez ou créez une instance Multi-AZ, un DLV est créé à la fois pour l'instance principale et pour l'instance secondaire.

RDS prend en charge les DLVs répliques en lecture. Si un DLV est activé sur l'instance de base de données principale, tous les réplicas en lecture créés après l'activation du DLV auront également un DLV. Il ne sera pas activé sur les réplicas en lecture créés avant le passage au DLV, sauf s'il est

explicitement modifié à cet effet. Nous recommandons que tous les réplicas en lecture attachés à une instance principale avant l'activation du DLV soient également modifiés manuellement pour avoir un DLV.

> **Note**
>
> Nous recommandons DLVs les configurations de base de données de 5 TiB ou plus.

Pour plus d'informations sur les avantages de DLVs, consultez les articles de blog suivants :

- Améliorez les performances des bases de données grâce aux volumes de journaux dédiés d'Amazon RDS
- Évaluez Amazon RDS pour PostgreSQL avec des volumes de journaux dédiés
- Optimisation des performances de AWS RDS pour MySQL avec des volumes de log dédiés dans la documentation de Percona

Pour obtenir des informations sur les plages de stockage alloué, les IOPS provisionnés et le débit de stockage disponibles pour chaque moteur de base de données, consultez Stockage SSD d'IOPS par seconde provisionnées.

Rubriques

- Considérations relatives à l'activation et à la désactivation du DLV
- Activation du DLV lors de la création d'une instance de base de données
- Activation du DLV sur une instance de base de données existante
- Surveillance du stockage DLV

Considérations relatives à l'activation et à la désactivation du DLV

L'activation et la désactivation du DLV peuvent prendre beaucoup de temps et entraîner des interruptions de service. Le processus consiste à copier tous les journaux de transactions ou les journaux de rétablissement et les journaux binaires (selon le moteur de base de données) sur le nouveau volume lors de l'activation, ou à les remettre dans le stockage d'origine lors de la désactivation. La durée de cette opération est influencée par plusieurs facteurs :

- Nombre de journaux de transactions :

  - Les bases de données volumineuses comportant un plus grand nombre de transactions génèrent davantage de journaux, ce qui augmente le temps nécessaire à la copie.
  - Les journaux de transactions peuvent s'accumuler sur l'instance de base de données principale si les emplacements de réplication sont inactifs ou si la réplication prend du retard, ce qui augmente le temps nécessaire à la copie. Assurez-vous que la réplication est à jour et supprimez tous les emplacements inutiles.
- Configuration du stockage :
  - Bande passante EBS de l'instance de base de données : une bande passante plus élevée permet un transfert de données plus rapide.
  - Nombre d'IOPS provisionnées : un plus grand nombre d'opérations d'entrée/sortie par seconde (IOPS) peut accélérer le processus de copie.
- Activité de base de données : des niveaux élevés d'activité de la base de données pendant la configuration peuvent ralentir le processus.

Pour minimiser les temps d'arrêt, nous vous recommandons de planifier et de planifier pendant les périodes de faible activité ou pendant les périodes de maintenance.

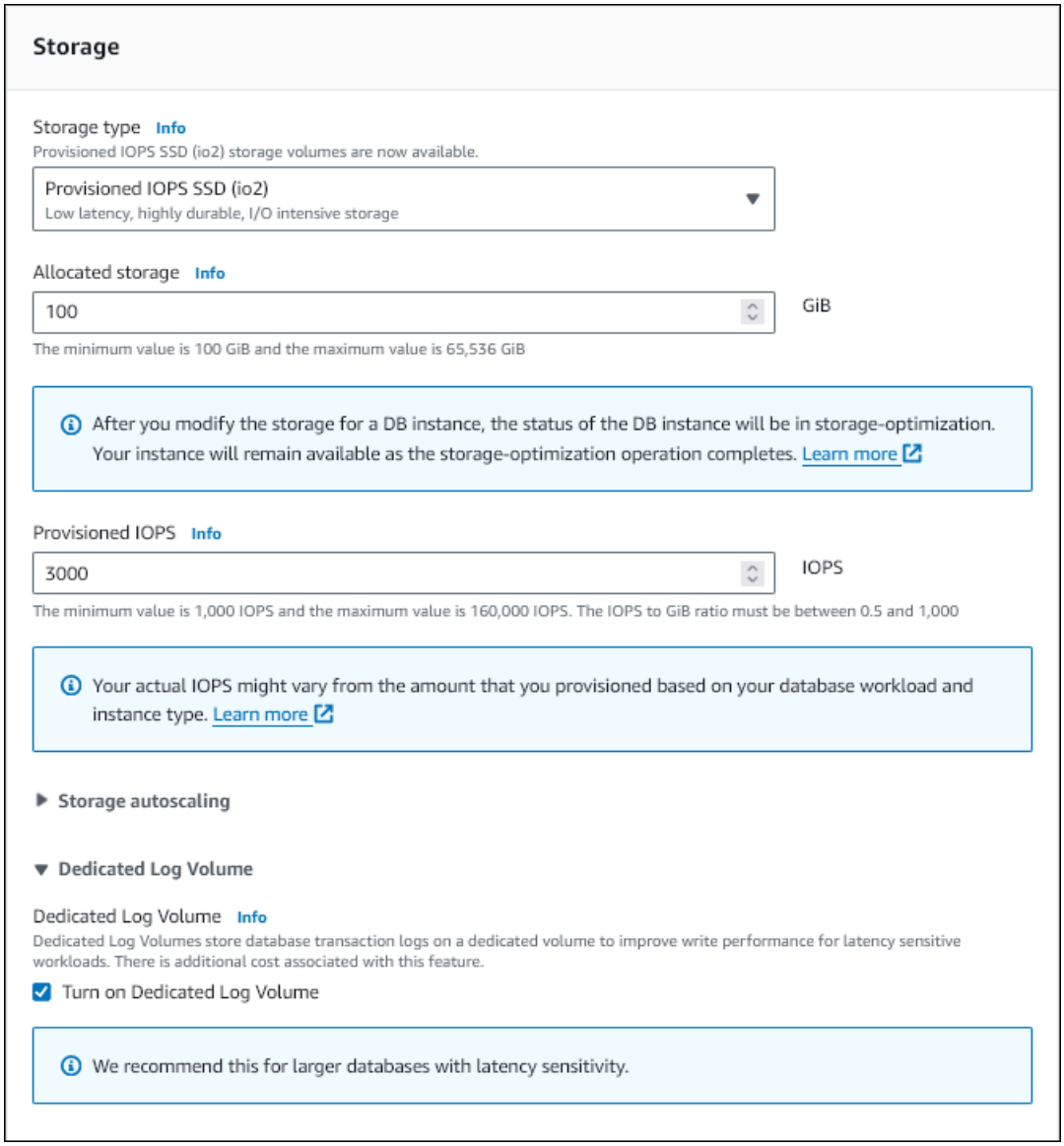
Activation du DLV lors de la création d'une instance de base de données

Vous pouvez utiliser l'API AWS Management Console AWS CLI, ou RDS pour créer une instance de base de données avec DLV activé.

console

Pour activer DLV sur une nouvelle instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Créer une base de données.
3. Sur la page Créer une instance de base de données, choisissez un moteur de base de données qui prend en charge le DLV.
4. Pour le stockage :
   a. Choisissez un SSD IOPS provisionné (io1) ou un SSD IOPS provisionné (io2).
   b. Entrez le stockage alloué et les IOPS provisionnées que vous souhaitez.
   c. Augmentez le volume de journal dédié, puis sélectionnez Activer le volume de journal dédié.



5. Choisissez d'autres paramètres selon vos besoins.
6. Choisissez Créer une base de données.

Une fois la base de données créée, la valeur de Dedicated Log Volume apparaît dans l'onglet Configuration de la page de détails de la base de données.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour activer DLV lorsque vous créez une instance de base de données à l'aide du stockage IOPS provisionné, utilisez la commande. AWS CLI create-db-instance Définissez les paramètres suivants :

- `--dedicated-log-volume`— Active un volume de journal dédié.
- `--storage-type`— Paramétré pour `io1` ou `io2` pour les IOPS provisionnées.
- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `--iops`— Le nombre d'IOPS provisionnées pour l'instance de base de données, exprimé en opérations d'E/S par seconde.

API RDS

Pour activer le DLV lorsque vous créez une instance de base de données à l'aide du stockage IOPS provisionné, utilisez l'opération Create de l'API Amazon RDS. DBInstance Définissez les paramètres suivants :

- `DedicatedLogVolume`— Réglé sur `true` pour activer un volume de journal dédié.
- `StorageType`— Paramétré pour `io1` ou `io2` pour les IOPS provisionnées.
- `AllocatedStorage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets.
- `Iops`— Le taux d'IOPS pour l'instance de base de données, exprimé en opérations d'E/S par seconde.

Activation du DLV sur une instance de base de données existante

Vous pouvez utiliser l'API AWS Management Console AWS CLI, ou RDS pour modifier une instance de base de données afin d'activer le DLV.

Après avoir modifié le paramètre DLV d'une instance de base de données, vous devez redémarrer l'instance de base de données.

console

Pour activer DLV sur une instance de base de données existante

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

Pour filtrer la liste des instances de bases de données, pour Filter databases (Filtrer les bases de donnée), saisissez une chaîne de texte pour Amazon RDS à utiliser pour filtrer les résultats. Seules les instances de bases de données dont les noms contiennent cette chaîne apparaissent.

3. Choisissez l'instance de base de données avec un stockage IOPS provisionné que vous souhaitez modifier.
4. Sélectionnez Modifier.
5. Sur la page Modifier une instance de base de données :
   - Pour le stockage, augmentez le volume de journal dédié, puis sélectionnez Activer le volume de journal dédié.
6. Choisissez Continuer.
7. Choisissez Appliquer immédiatement pour appliquer immédiatement les modifications à l'instance de base de données. Ou choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée pour appliquer les modifications pendant la prochaine fenêtre de maintenance.
8. Passez en revue les paramètres à modifier et choisissez Modification d'une instance de base de données pour terminer la modification.

La nouvelle valeur pour Dedicated Log Volume apparaît dans l'onglet Configuration de la page de détails de la base de données.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour activer ou désactiver le DLV sur une instance de base de données existante à l'aide du stockage IOPS provisionné, utilisez la commande. AWS CLI `modify-db-instance` Définissez les paramètres suivants :

- `--dedicated-log-volume`— Active un volume de journal dédié.

  Utilisez `--no-dedicated-log-volume` (valeur par défaut) pour désactiver un volume de journal dédié.
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer les modifications immédiatement.

  Utilisez `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications pendant la prochaine fenêtre de maintenance.

API RDS

Pour activer ou désactiver un DLV sur une instance de base de données existante à l'aide du stockage IOPS provisionnés, utilisez l'opération d'API Amazon RDS `ModifyDBInstance`. Définissez les paramètres suivants :

- `DedicatedLogVolume`— Définissez cette option pour `true` activer un volume de journal dédié.

  Définissez cette option sur `false` pour désactiver un volume de journal dédié. C'est la valeur par défaut.
- `ApplyImmediately` : définissez cette option sur `True` pour appliquer les modifications immédiatement.

  Définissez cette option sur `False` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

Surveillance du stockage DLV

Vous pouvez surveiller l'utilisation du stockage DLV à l'aide de la `FreeStorageSpaceLogVolume` métrique contenue dans CloudWatch.

Vous pouvez utiliser la requête suivante pour RDS pour PostgreSQL afin de déterminer la taille occupée par les journaux de transactions :

```
SELECT pg_size_pretty(COALESCE(sum(size), 0)) AS total_wal_generated_size
FROM pg_catalog.pg_ls_waldir();
```

Si le DLV n'a plus d'espace de stockage, l'instance de base de données passe dans `storage-full` cet état, ce qui entraîne une interruption de service.

Suppression d'une instance DB

Vous pouvez supprimer une instance de base de données à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Si vous souhaitez supprimer une instance de base de données d'un cluster de bases de données Aurora, consultez Suppression de clusters de bases de données Aurora et d'instances de bases de données.

Rubriques

- Conditions préalables pour la suppression d'une instance de base de données
- Considérations lors de la suppression d'une instance de base de données
- Suppression d'une instance DB

Conditions préalables pour la suppression d'une instance de base de données

Avant d'essayer de supprimer votre instance de base de données, veillez à ce que la protection contre la suppression soit désactivée. Par défaut, la protection contre la suppression est activée pour une instance de base de données créée avec la console.

Si la protection contre la suppression est activée sur votre instance de base de données, vous pouvez la désactiver en modifiant les paramètres de votre instance. Choisissez Modifier dans la page des détails de la base de données ou appelez la modify-db-instancecommande. Cette opération n'entraîne pas de panne. Pour plus d'informations, consultez Paramètres des instances de base de données.

Considérations lors de la suppression d'une instance de base de données

La suppression d'une instance de base de données a un effet sur la capacité de restauration de l'instance, la disponibilité des sauvegardes et le statut de lecture du réplica. Examinez les problèmes suivants :

- Vous pouvez choisir de créer un instantané de bases de données final. Vous avez les options suivantes :
  - Si vous prenez un instantané final, vous pouvez l'utiliser pour restaurer votre instance de base de données supprimée. RDS conserve à la fois l'instantané final et tous les instantanés manuels que vous avez pris précédemment. Vous ne pouvez pas créer un instantané de base de données final de votre instance de base de données si elle n'est pas dans l'état `Available`.

Pour plus d'informations, consultez Affichage de l'état de l'instance de base de données dans un cluster Aurora.

- Si vous ne prenez pas de capture finale, la suppression de votre instance est plus rapide. L'inconvénient est qu'il n'existe aucun instantané final que vous pourrez restaurer ultérieurement. Si vous décidez de restaurer votre instance de base de données supprimée, conservez les sauvegardes automatiques ou utilisez un instantané manuel antérieur pour restaurer votre instance de base de données à la même date que l'instantané précédent.

- Vous pouvez choisir de conserver les sauvegardes automatisées. Vous avez les options suivantes :
  - Si vous conservez les sauvegardes automatisées, RDS les garde pendant la période de conservation définie sur l'instance de base de données au moment où vous la supprimez. Vous pouvez utiliser les sauvegardes automatiques pour restaurer votre instance de base de données pendant la période de conservation, mais pas après. Cette période de conservation entre en vigueur, que vous choisissez de créer un instantané de base de données final ou non. Pour supprimer une sauvegarde automatisée conservée, consultez Suppression des sauvegardes automatisées conservées.
  - Les sauvegardes automatisées conservées et les instantanés manuels sont facturés tant qu'ils ne sont pas supprimés. Pour plus d'informations, consultez Coûts de conservation.
  - Si vous ne conservez pas les sauvegardes automatisées, RDS supprime les sauvegardes automatisées qui se trouvent dans le même emplacement Région AWS que votre instance de base de données. Il n'est pas possible de récupérer ces sauvegardes. Si vos sauvegardes automatisées ont été répliquées vers une autre Région AWS, RDS les conserve même si vous ne choisissez pas de conserver les sauvegardes automatisées. Pour plus d'informations, consultez Réplication des sauvegardes automatisées vers une autre Région AWS.

  > **Note**
  >
  > Vous n'avez généralement pas besoin de conserver les sauvegardes automatisées si vous créez un instantané de base de données final.

- Lorsque vous supprimez votre instance de base de données, RDS ne supprime pas les instantanés de base de données manuels. Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
- Si vous souhaitez supprimer toutes les ressources RDS, notez que les ressources suivantes sont facturées :

- Instances de base de données
- Instantanés de base de données
- Clusters de bases de données

Si vous avez acheté des instances réservées, elles sont facturées conformément au contrat que vous avez accepté lors de l'achat de l'instance. Pour plus d'informations, consultez Instances de base de données réservées pour Amazon RDS. Vous pouvez obtenir les informations de facturation pour toutes vos AWS ressources en utilisant le AWS Cost Explorer. Pour plus d'informations, consultez la section Analyse de vos coûts avec AWS Cost Explorer.

- Si vous supprimez une instance de base de données contenant des répliques de lecture Région AWS, chaque réplique de lecture est automatiquement promue en instance de base de données autonome. Pour plus d'informations, consultez Promotion d'un réplica en lecture en instance de bases de données autonome. Si votre instance de base de données possède des répliques de lecture différentes Régions AWS, consultez Considérations liées à la réplication entre régions les informations relatives à la suppression de l'instance de base de données source pour une réplique de lecture entre régions.
- Lorsque le statut d'une instance de base de données est`deleting`, la valeur de son certificat CA n'apparaît pas dans la console RDS ni dans les sorties des AWS CLI commandes ou des opérations de l'API RDS. Pour de plus amples informations sur les certificats d'autorité de certification, veuillez consulter .
- Le temps nécessaire à la suppression d'une instance de base de données varie en fonction de la période de conservation de la sauvegarde (c'est-à-dire du nombre de sauvegardes à supprimer), de la quantité de données supprimées et de la réalisation d'un instantané final.

Suppression d'une instance DB

Vous pouvez supprimer une instance de base de données à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Vous devez procéder comme suit :

- Indiquez le nom de l'instance de base de données.
- Activez ou désactivez l'option permettant de créer un instantané de base de données final de l'instance.
- Activez ou désactivez l'option de conservation des sauvegardes automatiques.

> **Note**
>
> Vous ne pouvez pas supprimer une instance de base de données lorsque la protection contre la suppression est activée. Pour plus d'informations, consultez Conditions préalables pour la suppression d'une instance de base de données.

Console

Pour supprimer une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez supprimer.
3. Pour Actions, choisissez Supprimer.
4. Pour créer un instantané de base de données final pour l'instance de base de données, choisissez Create final snapshot? (Créer un instantané final ?).
5. Si vous avez choisi de créer un instantané final, entrez le paramètre Final snapshot name (Nom de l'instantané final).
6. Pour conserver les sauvegardes automatiques, choisissez Conserver les sauvegardes automatiques.
7. Saisissez **`delete me`** dans la zone.
8. Sélectionnez Delete.

AWS CLI

Pour trouver les ID des instances de base de données de votre compte, appelez la describe-db-instancescommande :

```
aws rds describe-db-instances --query 'DBInstances[*].[DBInstanceIdentifier]' --output
 text
```

Pour supprimer une instance de base de données à l'aide de AWS CLI, appelez la delete-db-instancecommande avec les options suivantes :

- `--db-instance-identifier`

- `--final-db-snapshot-identifier` ou `--skip-final-snapshot`

Example Avec un instantané final et sans conservation des sauvegardes automatiques

Pour LinuxmacOS, ou Unix :

```
aws rds delete-db-instance \
    --db-instance-identifier mydbinstance \
    --final-db-snapshot-identifier mydbinstancefinalsnapshot \
    --delete-automated-backups
```

Dans Windows :

```
aws rds delete-db-instance ^
    --db-instance-identifier mydbinstance ^
    --final-db-snapshot-identifier mydbinstancefinalsnapshot ^
    --delete-automated-backups
```

Example Avec conservation des sauvegardes automatiques et pas d'instantané final

Pour LinuxmacOS, ou Unix :

```
aws rds delete-db-instance \
    --db-instance-identifier mydbinstance \
    --skip-final-snapshot \
    --no-delete-automated-backups
```

Dans Windows :

```
aws rds delete-db-instance ^
    --db-instance-identifier mydbinstance ^
    --skip-final-snapshot ^
    --no-delete-automated-backups
```

API RDS

Pour arrêter une instance de base de données à l'aide l'API Amazon RDS, appelez l'opération `DeleteDBInstance` avec les paramètres suivants :

- `DBInstanceIdentifier`

- `FinalDBSnapshotIdentifier` ou `SkipFinalSnapshot`

Tutoriel : Gestion d'un environnement d'instance My SQL DB, du développement à la production

Introduction

Lors de la gestion d'une RDS instance de base de données Amazon, il est courant de superviser son cycle de vie, du développement initial au déploiement en production. Ce didacticiel fournit des conseils pour gérer les tâches clés afin de garantir que votre base de données fonctionne de manière optimale et s'adapte à l'évolution de vos besoins opérationnels. En outre, il décrit les options permettant de synchroniser les modifications apportées entre vos environnements de développement et de production afin de garantir la cohérence et la fiabilité.

En suivant ces étapes, vous apprendrez à :

- Comment effectuer des tâches spécifiques avec les instances My SQL DB, telles que l'ajout et la mise à jour de RDS balises Amazon, l'extension du stockage, la création de répliques de lecture et la suppression de ressources.
- Comment synchroniser les mises à jour d'un environnement de production vers un environnement de développement pour des tests et une validation complets.

Pour terminer ce didacticiel, effectuez les tâches suivantes :

1. Créez une instance My SQL DB.
2. Ajoutez des RDS balises Amazon pour classer votre instance de base de données en tant qu'environnement de développement.
3. Augmentez la capacité de stockage de votre instance de base de données pour faire face à des charges de travail accrues.
4. Créez des répliques de lecture pour améliorer la résilience et la disponibilité de votre instance de base de données.
5. Mettez à jour les RDS balises Amazon pour classer votre instance de base de données en tant qu'environnement de production.
6. Supprimez les instances de base de données dont vous n'avez plus besoin afin qu'elles n'entraînent pas de frais supplémentaires.
7. Prochaines étapes : synchronisez votre instance de développement avec la production pour garantir la cohérence entre les environnements

Prérequis

Avant de commencer, suivez les étapes détaillées dans les sections suivantes :

- Inscrivez-vous pour un Compte AWS
- Création d’un utilisateur doté d’un accès administratif

Ajoutez des balises pour classer votre instance de base de données en tant qu'environnement de développement

Pour classer l'instance de base de données en tant qu'environnement de développement, ajoutez une RDS balise Amazon à l'instance que vous avez créée. Une RDS balise Amazon est une paire clé-valeur que vous définissez et associez à votre RDS instance. Le balisage de vos AWS ressources permet de faire la distinction entre vos AWS ressources de développement et de production. Pour plus d'informations sur les RDS tags Amazon, consultezMarquage d' Amazon RDS.

1. Dans la RDS console Amazon, sélectionnez Databases.
2. Sélectionnez l'instance de base de données que vous souhaitez étiqueter.
3. Dans la section des détails, accédez à la section Tags.
4. Choisissez Gérer les tags, puis sélectionnez Ajouter un nouveau tag.
5. Saisissez une valeur pour Tag key (Clé de balise) et Valeur. Par exemple, vous pouvez utiliser le tag key environment avec la valeur dev pour spécifier que l'instance de base de données fait partie de l'environnement de développement.
6. Choisissez Ajouter une nouvelle étiquette et Enregistrer les modifications.

Votre instance de base de données est désormais étiquetée en tant qu'environnement de développement. Cela facilite l'identification de l'instance de base de données et la gestion des coûts associés à cette ressource.

Augmenter la capacité de stockage d'une instance de base de données pour répondre aux besoins croissants en matière de données

Modifiez ensuite la capacité de stockage de l'instance My SQL DB pour accueillir des données supplémentaires. Au départ, la capacité de stockage de votre instance de base de données est définie pour répondre aux besoins immédiats de votre application. Toutefois, à mesure que les

volumes de données augmentent, il peut être nécessaire d'ajuster les paramètres de stockage pour garantir les performances et la stabilité continues de la base de données. Ce processus implique d'augmenter le stockage alloué à votre instance de base de données. Pour plus d'informations sur la modification de la capacité de stockage de votre instance de base de données, consultezUtilisation du stockage pour les RDS instances de base de données Amazon.

1. Dans la RDS console Amazon, sélectionnez Databases.
2. Sélectionnez l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modifier.
4. Dans Stockage, augmentez le stockage alloué. La valeur de stockage modifiée doit être supérieure à la valeur actuelle.
5. Choisissez Continuer.
6. Dans Planification des modifications, vous pouvez soit choisir Appliquer immédiatement pour appliquer immédiatement les modifications de stockage à l'instance de base de données, soit choisir Appliquer lors de la prochaine fenêtre de maintenance planifiée pour appliquer les modifications lors de la fenêtre de maintenance suivante.
7. Lorsque les paramètres vous conviennent, choisissez Modifier l'instance de base de données.

La capacité de stockage de votre instance de base de données est désormais augmentée. Cela lui permet de gérer efficacement des volumes de données plus importants et de garantir des performances et une stabilité continues à mesure que les besoins en données de votre application augmentent.

Créez des répliques de lecture pour améliorer la résilience et la disponibilité d'une instance de base de données

Créez une réplique en lecture de l'instance My SQL DB. Les répliques de lecture améliorent la résilience et la disponibilité de votre instance de base de données. Pour réduire le trafic de lecture sur votre instance de base de données principale, créez une réplique de lecture de votre instance de base de données. Cela achemine les requêtes vers la réplique lue, ce qui permet de répartir la charge et d'améliorer les performances globales de la base de données. Pour plus d'informations sur les répliques de lecture d'instances de base de données, consultezUtilisation des réplicas en lecture d'instance de base de données.

Avant qu'une instance My SQL DB puisse servir de source de réplication, les sauvegardes automatiques doivent être activées sur l'instance de base de données source. Cela peut être fait

en définissant la période de conservation des sauvegardes sur une valeur autre que 0. Pour plus d'informations sur My SQL read replicas, voirTravailler avec les répliques de My SQL read.

1. Dans la RDS console Amazon, sélectionnez Databases.
2. Sélectionnez l'instance de base de données que vous souhaitez utiliser comme source pour la réplique en lecture.
3. Dans Actions, sélectionnez Créer une réplique de lecture.
4. Pour l'identifiant de l'instance de base de données, entrez le nom de la réplique lue en lettres minuscules.
5. Choisissez la configuration de votre instance. Nous vous recommandons d'utiliser un type de stockage et une classe d'instances de base de données identiques ou supérieurs à l'instance de base de données source pour le réplica en lecture.
6. Pour Région AWS, spécifiez la région de destination du réplica en lecture.
7. Conservez les paramètres par défaut ou modifiez-les selon vos besoins.
8. Choisissez Créer un réplica en lecture.

La réplique lue apparaît sous votre instance de base de données source sur la page Bases de données de la RDS console. Il montre Replica dans la colonne Role.

Mettre à jour les balises pour classer une instance de base de données en tant qu'environnement de production

Lorsque votre instance de base de données est prête à passer de la phase de développement à la phase de production, il est important de mettre à jour ses balises pour refléter sa transition. Pour aligner votre instance de base de données sur vos stratégies opérationnelles et de surveillance, mettez à jour les balises initiales pour indiquer que l'instance de base de données fait désormais partie de l'environnement de production. Cela garantit une meilleure visibilité et une meilleure gestion de la base de données.

1. Dans la RDS console Amazon, sélectionnez Databases.
2. Sélectionnez l'instance de base de données que vous souhaitez mettre à jour
3. Dans la section des détails, accédez à la section Tags.
4. Sélectionnez Gérer les balises.
5. Supprimez votre balise initiale indiquant un environnement de développement.

6. Sélectionnez Ajouter un nouveau tag.
7. Entrez une nouvelle valeur pour Tag key et Value. Par exemple, vous pouvez utiliser l'environnement de clé de balise avec la valeur prod pour spécifier que l'instance de base de données fait partie de l'environnement de production.
8. Choisissez Ajouter une nouvelle étiquette et Enregistrer les modifications.

La balise de votre instance de base de données est mise à jour pour indiquer la transition de la base de données vers un environnement de production.

Supprimer une instance de base de données lorsqu'elle n'est plus nécessaire pour éviter d'encourir des coûts supplémentaires

Avant la fin de ce tutoriel, il est crucial d'aborder la gestion de vos ressources. Si certaines de vos ressources ne sont plus nécessaires, vous devez les supprimer pour éviter d'encourir des coûts supplémentaires et optimiser votre environnement cloud.

1. Dans la RDS console Amazon, sélectionnez Databases.
2. Sélectionnez l'instance de base de données que vous souhaitez supprimer
3. Dans Actions, sélectionnez Supprimer. La suppression d'une instance de base de données supprimera définitivement l'instance avec tout son contenu et les ressources associées.
4. Confirmez la suppression de l'instance de base de données et sélectionnez Supprimer.

Sinon, si vous choisissez de conserver votre instance de base de données pour une utilisation future, vous pouvez continuer à la gérer dans le cadre de votre environnement de production. Cela implique de maintenir un environnement de développement synchronisé afin de faciliter les tests et les validations complets. Pour de plus amples informations, veuillez consulter Prochaines étapes : synchroniser votre instance de développement avec la production pour garantir la cohérence entre les environnements.

Prochaines étapes : synchroniser votre instance de développement avec la production pour garantir la cohérence entre les environnements

Créer un environnement de développement

Pour gérer un environnement de production, il est important de maintenir un environnement de développement synchronisé pour des tests et des validations complets. Pour créer un nouvel

environnement de développement, créez d'abord un instantané de base de données de l'instance de base de données de production actuelle. Un instantané de base de données capture l'intégralité de l'instance de base de données en créant un instantané de volume de stockage. Pour obtenir des instructions sur la création d'un instantané de base de données sur la RDS console Amazon, consultezCréation d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Après avoir créé l'instantané de base de données de votre environnement de production, créez une nouvelle instance de base de données pour votre environnement de développement en restaurant un instantané de base de données. Les instances de base de données restaurées sont automatiquement associées aux groupes de paramètres et d'options de base de données par défaut. Cependant, vous pouvez appliquer un groupe de paramètres ou d'options personnalisé en le définissant au moment de la restauration. Pour obtenir des instructions sur la restauration d'un instantané de base de données, consultezTutoriel : Restaurer une RDS instance de base de données Amazon à partir d'un instantané de base de données.

Enfin, désignez la nouvelle instance de base de données comme nouvel environnement de développement en mettant à jour ses RDS balises Amazon. Pour obtenir des conseils sur la mise à jour des RDS balises Amazon afin de refléter cette modification, consultez la section précédenteMettre à jour les balises pour classer une instance de base de données en tant qu'environnement de production.

Vous disposez désormais d'un nouvel environnement de développement qui reflète la configuration de base de données de votre environnement de production.

Synchroniser un environnement de développement avec un environnement de production

Une fois votre nouvel environnement de développement établi, il est nécessaire de le synchroniser avec les modifications apportées à l'environnement de production. Cela garantit que votre environnement de développement reflète fidèlement l'état actuel de la production, ce qui est essentiel pour des tests, une validation et un dépannage efficaces. Amazon RDS propose différentes méthodes pour maintenir votre environnement de développement à jour par rapport à votre environnement de production. Pour plus d'informations sur ces options, consultez la section Orchestration des actualisations de base de données pour Amazon RDS et Amazon Aurora.

L'un des principaux moyens de synchroniser vos environnements de développement et de production consiste à créer et à restaurer des instantanés de base de données. Un instantané de base de données vous permet de créer un environnement de développement qui reflète la configuration de base de données de l'environnement de production au moment de la création de l'instantané. Pour

plus d'informations sur les instantanés de base de données, consultezGestion des sauvegardes manuelles. Pour plus d'informations sur la restauration d'une instance de base de données, consultezRestauration vers une instance de base de données.

Les instantanés de base de données sont particulièrement utiles dans les cas d'utilisation suivants.

- Configuration initiale d'un environnement de développement : les instantanés de base de données sont utiles pour créer l'environnement de développement initial à des fins de test, car ils fournissent une base de référence cohérente qui reflète l'état exact de l'environnement de production au moment de la capture instantanée.
- Applications à trafic élevé : dans les environnements de production où le fonctionnement continu est essentiel, l'utilisation de déploiements multi-AZ pour les instantanés évite la suspension des E/S sur la base de données principale, garantissant ainsi des performances ininterrompues et une haute disponibilité.
- Partage de données entre différents RDS comptes : les instantanés de base de données peuvent être partagés entre différents comptes Comptes AWS, ce qui facilite le transfert de données entre les comptes ou les régions. Cela est utile pour les projets collaboratifs ou les scénarios dans lesquels les données doivent être partagées à des fins diverses. Pour de plus amples informations, veuillez consulter Partage d'un instantané de base de données pour Amazon RDS.

Dans ce didacticiel, vous avez exploré les tâches essentielles pour gérer votre instance de base de données tout au long de son cycle de vie. Vous avez appris à créer une instance de base de données, à ajouter et à mettre à jour des RDS balises Amazon, à étendre le stockage et à créer des répliques de lecture. Vous avez également appris à tirer parti de ces opérations fondamentales et à gérer efficacement votre environnement de production. Cela comprenait la mise en place d'un environnement de développement pour les tests et sa synchronisation avec l'environnement de production pour des raisons de cohérence. Ces tâches contribuent à maintenir une infrastructure de base de données résiliente et évolutive, garantissant ainsi le fonctionnement efficace de votre RDS environnement Amazon.

Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS

Les déploiements multi-AZ peuvent comporter une ou deux instance de base de données de secours. Lorsque le déploiement comporte une instance de base de données de secours, il s'agit d'un déploiement d'instance de base de données multi-AZ. Un déploiement d'instance de base de données multi-AZ comporte une instance de base de données de secours qui prend en charge le basculement, mais qui ne traite pas le trafic en lecture. Lorsque le déploiement comporte deux instances de base de données de secours, il s'agit d'un déploiement de cluster de base de données multi-AZ. Un déploiement de cluster de base de données multi-AZ comporte des instances de base de données de secours qui prennent en charge le basculement et peuvent également traiter le trafic en lecture.

Vous pouvez utiliser le AWS Management Console pour déterminer si un déploiement multi-AZ est un déploiement d'instance de base de données multi-AZ ou un déploiement de cluster de bases de données multi-AZ. Dans le panneau de navigation, choisissez Databases (Bases de données), choisissez un DB identifier (Identifiant de base de données).

- Un déploiement d'instance de base de données Multi-AZ présente les caractéristiques suivantes :
  - Il n'y a qu'une seule ligne pour l'instance de base de données.
  - La valeur pour Role (Rôle) est Instance (Instance) ou Primary (Principal).
  - La valeur pour Multi-AZ est Yes (Oui).
- Un déploiement de cluster de base de données Multi-AZ présente les caractéristiques suivantes :
  - Il y a une ligne de niveau cluster avec trois lignes d'instance de base de données en-dessous.
  - Pour la ligne de niveau cluster, la valeur de Role (Rôle) est Multi-AZ DB cluster (Cluster de base de données Multi-AZ).
  - Pour chaque ligne de niveau instance, la valeur de Role (Rôle) est Writer instance (Instance de rédacteur) ou Reader instance (Instances de lecteur).
  - Pour chaque ligne de niveau instance, la valeur de Multi-AZ est 3 Zones (3 zones).

Rubriques

- Déploiements d'instances de base de données multi-AZ pour Amazon RDS
- Déploiements de clusters de bases de données multi-AZ pour Amazon RDS

En outre, les rubriques suivantes s'appliquent à la fois aux instances de base de données et aux clusters de base de données multi-AZ.

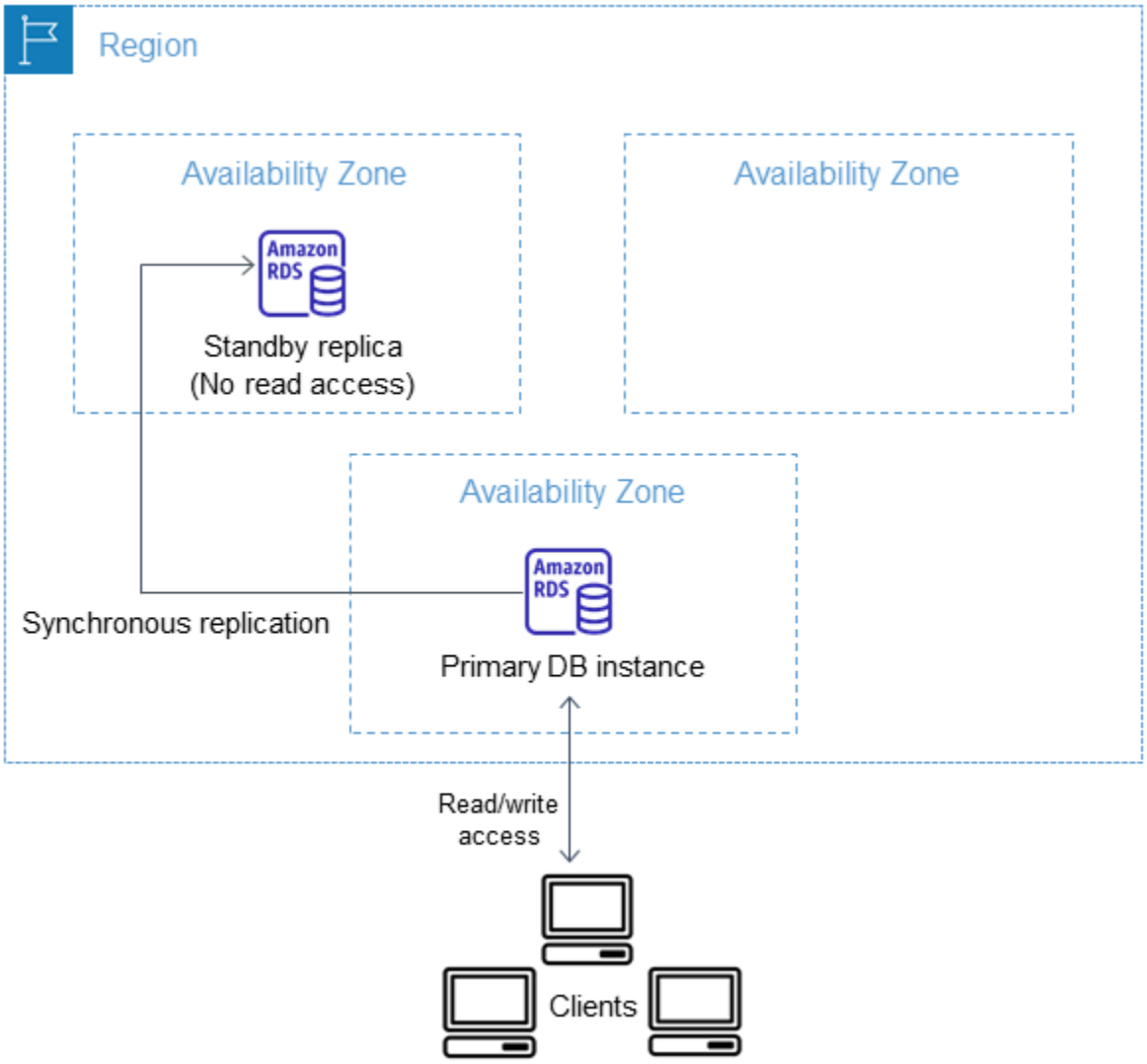
- the section called “Marquage d' RDS de ses ressources”
- the section called “ARNssur Amazon RDS”
- the section called “Utilisation du stockage”
- the section called “Entretien d'une instance de base de données”
- the section called “Mise à niveau de la version du moteur”

Déploiements d'instances de base de données multi-AZ pour Amazon RDS

Amazon RDS fournit une haute disponibilité et un support de basculement pour les instances de base de données utilisant des déploiements multi-AZ avec une seule instance de base de données de secours. Ce type de déploiement est appelé déploiement d'instance de base de données multi-AZ. Amazon RDS utilise plusieurs technologies différentes pour fournir cette assistance en cas de basculement. Les déploiements multi-AZ pour les instances de SQL base de données MariaDB, MySQL, Oracle, Postgre SQL et RDS Custom for Server utilisent la technologie de basculement Amazon. Les instances de base de données Microsoft SQL SQL Server utilisent la mise en miroir des bases de données du serveur (DBM) ou les groupes de disponibilité Always On (AGs). Pour plus d'informations sur la prise en charge des versions de SQL serveur pour Multi-AZ, consultezDéploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL. Pour plus d'informations sur l'utilisation de RDS Custom for SQL Server for Multi-AZ, consultezGestion d'un déploiement multi-AZ pour RDS Custom for Server SQL.

Dans le cadre d'un déploiement d'instance de base de données multi-AZ, Amazon provisionne et gère RDS automatiquement une réplique de secours synchrone dans une autre zone de disponibilité. L'instance de base de données primaire est répliquée de manière synchrone dans les zones de disponibilité sur un réplica de secours afin d'assurer une redondance des données et de limiter les pics de latence lors des sauvegardes système. L'exécution d'une instance de base de données en haute disponibilité peut améliorer la disponibilité pendant la maintenance planifiée du système. Elle peut également contribuer à protéger vos bases de données contre la défaillance d'une instance de base de données et la perturbation d'une zone de disponibilité. Pour plus d'informations sur les zones de disponibilité, consultez Régions, zones de disponibilité et zones locales.

> **Note**
>
> L'option de haute disponibilité n'est pas une solution de mise à l'échelle pour les scénarios de lecture seule. Vous ne pouvez pas utiliser un réplica de secours pour traiter le trafic en lecture. Pour traiter le trafic en lecture seule, utilisez plutôt un cluster de base de données multi-AZ ou un réplica en lecture. Pour plus d'informations sur les clusters de base de données multi-AZ, consultez Déploiements de clusters de bases de données multi-AZ pour Amazon RDS. Pour plus d'informations sur les réplicas en lecture, consultez Utilisation des réplicas en lecture d'instance de base de données.



À l'aide de la RDS console, vous pouvez créer un déploiement d'instance de base de données multi-AZ en spécifiant simplement Multi-AZ lors de la création d'une instance de base de données. Vous pouvez utiliser la console pour convertir des instances de base de données existantes en déploiements d'instance de base de données multi-AZ. Pour cela, vous devez modifier l'instance de base de données et spécifier l'option multi-AZ. Vous pouvez également spécifier un déploiement d'instance de base de données multi-AZ avec AWS CLI ou Amazon RDSAPI. Utilisez la modify-db-instanceCLIcommande create-db-instanceou, ou l'odifyDBInstanceAPIopération C reateDBInstance ou M.

La RDS console indique la zone de disponibilité de la réplique de secours (appelée AZ secondaire). Vous pouvez également utiliser la describe-db-instancesCLIcommande ou l'escribeDBInstancesAPIopération D pour trouver l'AZ secondaire.

Les instances de base de données qui utilisent des déploiements d'instance de base de données multi-AZ peuvent avoir une latence d'écriture et de validation accrue par rapport à un déploiement mono-AZ. Cela peut se produire en raison de la réplication de données synchrone qui se produit. La latence peut changer si votre déploiement bascule vers la réplique de secours, même si elle AWS est conçue avec une connectivité réseau à faible latence entre les zones de disponibilité. Pour les charges de travail de production, nous vous recommandons d'utiliser Provisioned IOPS (opérations d'entrée/sortie par seconde) pour des performances rapides et cohérentes. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .

Conversion d'une instance de base de données en déploiement multi-AZ pour Amazon RDS

La modification d'une instance de base de données vers un déploiement multi-AZ améliore la disponibilité en ajoutant une instance de secours dans une autre zone de disponibilité. Le processus implique des temps d'arrêt minimaux et nécessite une planification minutieuse des impacts sur le stockage et les performances. Cette modification améliore la tolérance aux pannes et réduit le temps de restauration en cas de panne, ce qui la rend idéale pour les environnements à haute disponibilité.

Si vous avez une instance de base de données dans un déploiement mono-AZ et que vous la modifiez en un déploiement d'instance de base de données multi-AZ, Amazon RDS effectue les actions suivantes :

1. Prend un instantané des volumes Amazon Elastic Block Store (EBS) de l'instance de base de données principale.
2. Crée de nouveaux volumes pour le réplica en attente à partir de l'instantané. Ces volumes s'initialisent en arrière-plan, et les performances maximales du volume sont atteintes après l'initialisation complète des données.
3. Active la réplication synchrone au niveau des blocs entre les volumes des réplicas principal et secondaire.

> **Important**
>
> La création d'une instance de base de données de secours à partir d'un instantané lors d'une conversion mono-AZ en multi-AZ permet d'éviter les temps d'arrêt mais peut avoir un impact sur les performances, en particulier pour les charges de travail sensibles à l'écriture. La réplication synchrone peut augmenter la latence des E/S, affectant ainsi les performances

de la base de données. Il est recommandé d'éviter de convertir une instance de base de données de production en une instance de base de données multi-AZ.
Créez plutôt une réplique en lecture, activez les sauvegardes sur celle-ci, convertissez-la au format Multi-AZ, chargez les données dans ses volumes, puis promouvez-la vers l'instance de base de données principale. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Il existe deux façons de modifier une instance de base de données en déploiement d'instance de base de données multi-AZ :

Rubriques

- Passez à un déploiement d'instance de base de données multi-AZ avec la console RDS
- Transformation d'une instance de base de données en déploiement d'instance de base de données multi-AZ

Passez à un déploiement d'instance de base de données multi-AZ avec la console RDS

Vous pouvez utiliser la RDS console pour convertir une instance de base de données en déploiement d'instance de base de données multi-AZ.

Vous ne pouvez utiliser la console que pour finaliser la conversion. Pour utiliser le AWS CLI ou RDSAPI, suivez les instructions figurant dansTransformation d'une instance de base de données en déploiement d'instance de base de données multi-AZ.

Pour passer à un déploiement d'instance de base de données multi-AZ avec la console RDS

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Dans Actions, choisissez Convert to Multi-AZ deployment (Convertir en déploiement multi-AZ).
4. Sur la page de confirmation, choisissez Apply immediately (Appliquer immédiatement) pour appliquer les modifications immédiatement. Le choix de cette option n'entraîne pas d'interruption de service, mais il existe un impact possible sur les performances. Vous pouvez également

choisir d'appliquer la mise à jour pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

5. Choisissez Convert to Multi-AZ (Convertir en multi-AZ).

Transformation d'une instance de base de données en déploiement d'instance de base de données multi-AZ

Vous pouvez modifier une instance de base de données pour en faire un déploiement d'instance de base de données multi-AZ de la manière suivante :

- À l'aide de la RDS console, modifiez l'instance de base de données et définissez le déploiement multi-AZ sur Oui.
- À l'aide de AWS CLI, appelez la modify-db-instancecommande et définissez l'`--multi-az`option.
- À l'aide de RDSAPI, appelez l'odifyDBInstanceopération M et définissez le `MultiAZ` paramètre sur`true`.

Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon. Une fois la modification terminée, Amazon RDS déclenche un événement (RDS- EVENT -0025) indiquant que le processus est terminé. Vous pouvez surveiller les RDS événements Amazon. Pour plus d'informations sur les événements, consultez Utilisation des notifications d'RDSévénements Amazon.

Défaillance d'une instance de base de données multi-AZ pour Amazon RDS

Si une interruption planifiée ou imprévue de votre instance de base de données Multi-AZ résulte d'un défaut d'infrastructure, Amazon passe RDS automatiquement à une réplique de secours dans une autre zone de disponibilité.

La durée du basculement dépend de l'activité de la base de données et d'autres conditions au moment où l'instance de base de données primaire est devenue indisponible. Les durées de basculement oscillent généralement entre 60 et 120 secondes. Cependant, les transactions importantes ou les processus de récupération longs peuvent augmenter le temps de basculement. Lorsque le basculement est terminé, la RDS console peut mettre plus de temps à refléter la nouvelle zone de disponibilité.

> **Note**
>
> Vous pouvez forcer un basculement manuellement lorsque vous redémarrez une instance de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Amazon RDS gère automatiquement les basculements afin que vous puissiez reprendre les opérations de base de données le plus rapidement possible sans intervention administrative. L'instance de base de données primaire bascule automatiquement vers le réplica de secours si l'une des conditions décrites dans le tableau suivant se produit : Vous pouvez consulter les raisons du basculement dans le journal des événements.

Raison du basculement	Description
Le système d'exploitation sous-jacent à l'instance RDS de base de données est corrigé lors d'une opération hors ligne.	Un basculement a été déclenché pendant la fenêtre de maintenance d'un correctif du système d'exploitation ou d'une mise à jour de sécurité. Pour de plus amples informations, veuillez consulter Entretien d'une instance de base de données.
L'hôte principal de l'instance RDS Multi-AZ est défectueux.	Le déploiement d'instance de base de données multi-AZ a détecté une instance de base de données primaire déficiente et a opéré un basculement.
L'hôte principal de l'instance RDS Multi-AZ est inaccessible en raison d'une perte de connectiv ité réseau.	RDSla surveillance a détecté une défaillan ce de l'accessibilité réseau de l'instance de base de données principale et a déclenché un basculement.
L'RDSinstance a été modifiée par le client.	Une modification d'RDSinstance de base de données a déclenché un basculement.

Raison du basculement	Description
	Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Raison du basculement	Description
L'instance principale RDS Multi-AZ est occupée et ne répond pas.	L'instance de base de données primaire ne répond pas. Nous vous recommandons d'effectuer les opérations suivantes : • Examinez l'événement et CloudWatch les journaux pour détecter toute utilisation excessive CPU de la mémoire ou de l'espace de swap. Pour plus d'informations, consultez Utilisation des notifications d'RDSévénements Amazon et Création d'une règle qui se déclenche sur un événement Amazon RDS. • Évaluez votre charge de travail pour déterminer si vous utilisez la classe d'instance de base de données appropriée. Pour plus d'informations, consultez Classes d'instances de base de données . • Utilisez la surveillance améliorée pour les métriques du système d'exploitation en temps réel. Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée. • Utilisez Performance Insights pour analyser les problèmes qui affectent les performances de votre instance de base de données. Pour plus d'informations, consultez Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

Raison du basculement	Description
	Pour plus d'informations sur ces recommandations, consultez la section Outils de surveillance pour Amazon RDS et Bonnes pratiques relatives à Amazon RDS..
Le volume de stockage sous-jacent à l'hôte principal de l'instance RDS Multi-AZ est tombé en panne.	Le déploiement d'instance de base de données multi-AZ a détecté un problème de stockage sur l'instance de base de données primaire et a opéré un basculement.
L'utilisateur a demandé un basculement de l'instance de base de données.	Vous avez redémarré l'instance de base de données et choisi l'option Redémarrer avec basculement. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Pour déterminer si votre instance de base de données Multi-AZ a basculé, voici ce que vous pouvez faire :

- Configurez des abonnements aux événements de base de données pour vous informer par e-mail ou SMS qu'un basculement a été initié. Pour plus d'informations sur les événements, consultez Utilisation des notifications d'RDSévénements Amazon.
- Affichez les événements de votre base de données à l'aide de la RDS console ou API des opérations.
- Consultez l'état actuel du déploiement de votre instance de base de données multi-AZ à l'aide de la RDS console ou API des opérations.

Pour plus d'informations sur la manière dont vous pouvez réagir en cas de basculement, réduire le temps de restauration et sur d'autres bonnes pratiques pour AmazonRDS, consultezBonnes pratiques relatives à Amazon RDS..

Configuration JVM TTL des recherches DNS de noms

Le mécanisme de basculement modifie automatiquement l'enregistrement du système de noms de domaine (DNS) de l'instance de base de données pour qu'il pointe vers l'instance de base de données de secours. Par conséquent, vous devez rétablir toutes les connexions existantes à votre instance de base de données. Dans un environnement de machine virtuelle Java (JVM), en raison du fonctionnement du mécanisme de DNS mise en cache Java, vous devrez peut-être reconfigurer JVM les paramètres.

Les recherches de DNS noms de JVM caches. Lorsque le JVM convertit un nom d'hôte en adresse IP, il met l'adresse IP en cache pendant une période spécifiée, connue sous le nom de time-to-live(TTL).

Étant donné que les AWS ressources utilisent des entrées de DNS nom qui changent parfois, nous vous recommandons de configurer votre nom JVM avec une TTL valeur ne dépassant pas 60 secondes. Cela garantit que lorsque l'adresse IP d'une ressource change, votre application peut recevoir et utiliser la nouvelle adresse IP de la ressource en demandant le. DNS

Sur certaines configurations Java, la JVM valeur par défaut TTL est définie de manière à ne jamais actualiser les DNS entrées avant JVM le redémarrage. Ainsi, si l'adresse IP d'une AWS ressource change alors que votre application est toujours en cours d'exécution, elle ne peut pas utiliser cette ressource tant que vous ne l'avez pas redémarrée manuellement JVM et que les informations IP mises en cache ne sont pas actualisées. Dans ce cas, il est essentiel de définir le JVM « s » TTL afin qu'il actualise régulièrement ses informations IP mises en cache.

Vous pouvez obtenir la JVM valeur TTL par défaut en récupérant la valeur de la `networkaddress.cache.ttl`propriété :

```
String ttl = java.security.Security.getProperty("networkaddress.cache.ttl");
```

> **Note**
>
> La valeur par défaut TTL peut varier en fonction de la version de votre ordinateur JVM et de l'installation ou non d'un gestionnaire de sécurité. Beaucoup JVMs fournissent une valeur par défaut TTL inférieure à 60 secondes. Si vous utilisez un tel gestionnaire JVM sans utiliser de gestionnaire de sécurité, vous pouvez ignorer le reste de cette rubrique. Pour de plus amples informations sur les responsables de la sécurité dans Oracle, veuillez consulter The Security Manager dans la documentation Oracle.

Pour modifier les « JVM s »TTL, définissez la valeur de la `networkaddress.cache.ttl` propriété. Utilisez l'une des méthodes suivantes selon vos besoins :

- Pour définir la valeur de propriété de manière globale pour toutes les applications utilisant leJVM, définissez `networkaddress.cache.ttl` dans le `$JAVA_HOME/jre/lib/security/java.security` fichier.

```
networkaddress.cache.ttl=60
```

- Pour définir la propriété localement pour votre application uniquement, définissez `networkaddress.cache.ttl` dans le code d'initialisation de votre application avant que les connexions réseau ne soient établies.

```
java.security.Security.setProperty("networkaddress.cache.ttl" , "60");
```

Déploiements de clusters de bases de données multi-AZ pour Amazon RDS

Un déploiement de cluster de bases de données multi-AZ est un mode de déploiement semi-synchrone à haute disponibilité d'Amazon RDS avec deux instances de base de données répliques lisibles. Un cluster de base de données multi-AZ possède une instance de base de données d'écriture et deux instances de base de données de lecture dans trois zones de disponibilité distinctes d'une même Région AWS. Les clusters de base de données multi-AZ offrent une haute disponibilité, une capacité accrue pour les charges de travail en lecture et une moindre latence en écriture par rapport aux déploiements d'instances de base de données multi-AZ.

Vous pouvez importer des données d'une base de données sur site vers un cluster de bases de données multi-AZ en suivant les instructions dans Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt.

Vous pouvez acheter des instances de base de données réservées pour un cluster de bases de données multi-AZ. Pour de plus amples informations, veuillez consulter Instances de base de données réservées pour un cluster de bases de données multi-AZ.

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour plus d'informations sur la disponibilité des versions et des régions d'Amazon RDS avec des clusters de base de données multi-AZ, consultezRégions et moteurs de base de données pris en charge pour les clusters de bases de données multi-AZ dans Amazon RDS.

Rubriques

- Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ
- Architecture de cluster de bases de données multi-AZ
- Groupes de paramètres pour les clusters de bases de données multi-AZ
- RDSProxy avec clusters de bases de données multi-AZ
- Retard de réplica et clusters de base de données multi-AZ
- Instantanés de clusters de bases de données multi-AZ
- Création d'un cluster de base de données multi-AZ pour Amazon RDS
- Connexion à un cluster de bases de données multi-AZ pour Amazon RDS
- Connexion automatique d'une ressource AWS de calcul et d'un cluster de base de données multi-AZ pour Amazon RDS

- Modification d'un cluster de base de données multi-AZ pour Amazon RDS
- Mise à niveau de la version du moteur d'un cluster de base de données multi-AZ pour Amazon RDS
- Modification du nom d'un cluster de bases de données multi-AZ pour Amazon RDS
- Redémarrage d'un cluster de base de données multi-AZ et d'instances de base de données de lecture pour Amazon RDS
- Défaillance d'un cluster de base de données multi-AZ pour Amazon RDS
- Configuration de la réplication SQL logique Postgre avec des clusters de bases de données multi-AZ pour Amazon RDS
- Utilisation de répliques de lecture de clusters de bases de données multi-AZ pour Amazon RDS
- Configuration de la réplication externe à partir de clusters de bases de données multi-AZ pour Amazon RDS
- Suppression d'un cluster de base de données multi-AZ pour Amazon RDS
- Limites des clusters de bases de données multi-AZ pour Amazon RDS

> **Important**
>
> Les clusters de base de données multi-AZ sont différents des clusters de base de données Aurora. Pour en savoir plus sur les clusters de base de données Aurora, consultez le Guide de l'utilisateur Amazon Aurora.

Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ

Les déploiements de clusters de bases de données multi-AZ sont pris en charge pour les classes d'instances de base de données suivantes : `db.m5d` `db.m6gd` `db.m6id` `db.m6idn`db.r5d,db.r6gd,db.x2iedn,db.r6id,db.r6idn, et. `db.c6gd`

> **Note**
>
> Les classes d'instance c6gd sont les seules à prendre en charge la taille de l'`medium`instance.

Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter the section called “Classes d'instances de base de données”.

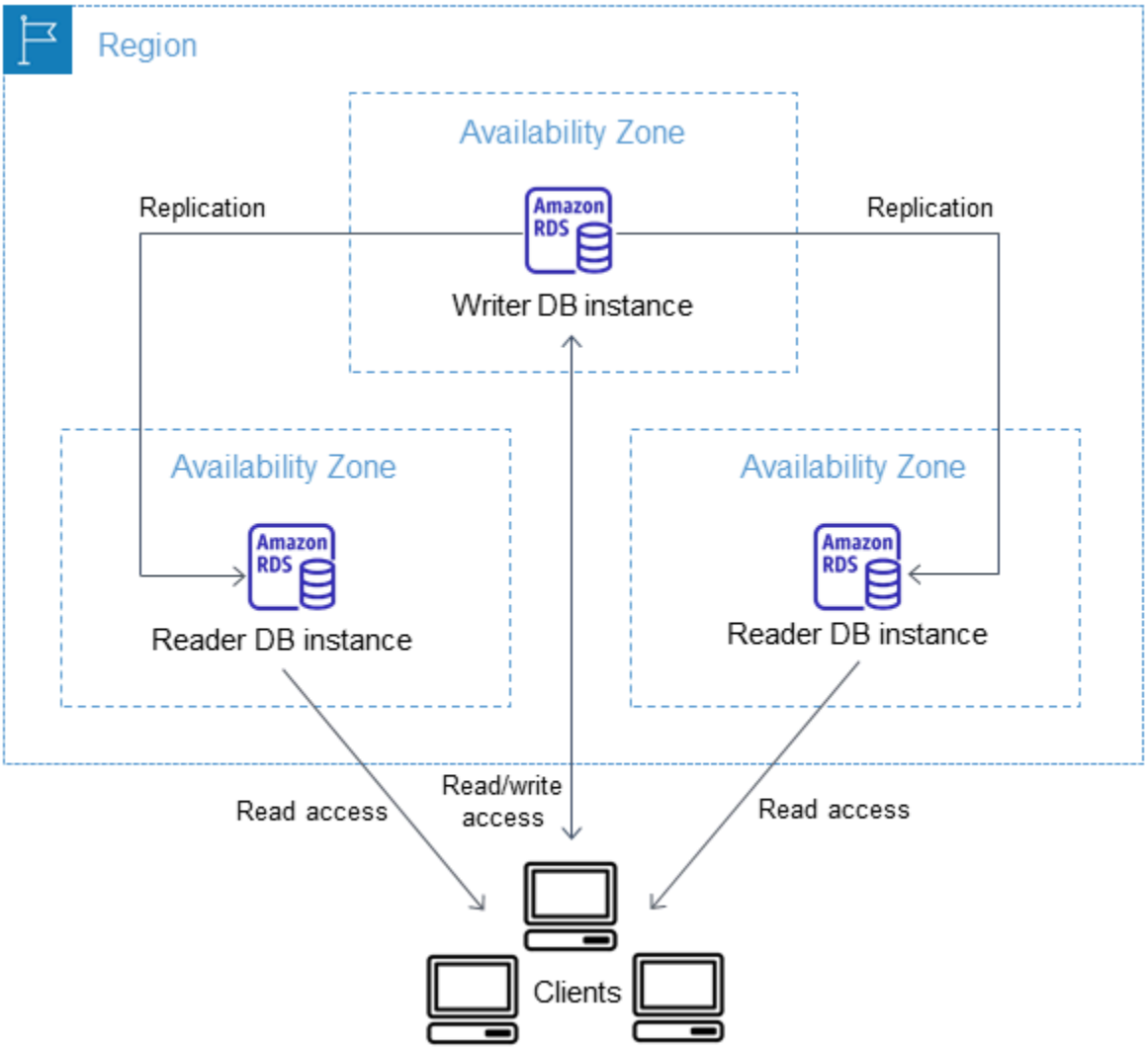
Architecture de cluster de bases de données multi-AZ

Avec un cluster de base de données multi-AZ, Amazon RDS réplique les données de l'instance de base de données du rédacteur vers les deux instances de base de données du lecteur en utilisant les capacités de réplication natives du moteur de base de données. Lorsqu'une modification est apportée à l'instance de base de données d'écriture, elle est transmise à chaque instance de base de données de lecture.

Les déploiements de clusters de bases de données multi-AZ utilisent une réplication semi-synchrone, qui nécessite un accusé de réception d'au moins une instance de base de données de lecture pour qu'une modification soit appliquée. Il n'est pas nécessaire de confirmer que les événements ont été entièrement exécutés et validés sur tous les réplicas.

Les instances de base de données d'écriture font office de cibles de basculement automatique et traitent également le trafic en lecture pour accroître le débit de lecture des applications. En cas de panne sur votre instance de base de données d'écriture, RDS gère le basculement vers l'une des instances de base de données de lecteur. RDSeffectue cette opération en fonction de l'instance de base de données du lecteur qui possède l'enregistrement de modification le plus récent.

Le schéma suivant illustre un cluster de base de données multi-AZ.



Les clusters de base de données multi-AZ ont généralement une latence d'écriture moindre par rapport aux déploiements d'instances de base de données multi-AZ. Ils permettent également d'exécuter des charges de travail en lecture seule sur des instances de base de données de lecteurs. La RDS console affiche la zone de disponibilité de l'instance de base de données du rédacteur et les zones de disponibilité des instances de base de données du lecteur. Vous pouvez également utiliser la describe-db-clustersCLIcommande ou l'escribeDBClustersAPIopération D pour trouver ces informations.

> **Important**
>
> Pour éviter les erreurs de réplication dans RDS les clusters My SQL Multi-AZ DB, nous recommandons vivement que toutes les tables possèdent une clé primaire.

Groupes de paramètres pour les clusters de bases de données multi-AZ

Dans un cluster de base de données multi-AZ, un groupe de paramètres de cluster de base de données sert de conteneur pour les valeurs de configuration du moteur qui sont appliquées à chaque instance de base de données contenue dans le cluster de base de données multi-AZ.

Dans un cluster de base de données multi-AZ, un groupe de paramètres de base de données est défini comme étant le groupe de paramètres de base de données par défaut pour le moteur et la version du moteur de base de données Les paramètres du groupe de paramètres de cluster de base de données s'appliquent à toutes les instances de base de données du cluster.

Pour plus d'informations sur les groupes de paramètres, veuillez consulter the section called “Groupes de paramètres de cluster DB”.

RDSProxy avec clusters de bases de données multi-AZ

Vous pouvez utiliser Amazon RDS Proxy pour créer un proxy pour vos clusters de bases de données multi-AZ. En utilisant le RDS proxy, vos applications peuvent regrouper et partager des connexions à des bases de données afin d'améliorer leur capacité à évoluer. Chaque proxy effectue le multiplexage des connexions, également appelé réutilisation des connexions. Avec le multiplexage, RDS Proxy exécute toutes les opérations d'une transaction en utilisant une connexion de base de données sous-jacente. RDSLe proxy peut également réduire à une seconde ou moins le temps d'arrêt lié à une mise à niveau de version mineure d'un cluster de base de données multi-AZ. Pour plus d'informations sur les avantages du RDS proxy, consultezProxy Amazon RDS.

Pour configurer un proxy pour un cluster de base de données multi-AZ, choisissez Create an RDS Proxy lors de la création du cluster. Pour obtenir des instructions sur la création et la gestion des points de terminaison du RDS proxy, consultezthe section called “Utilisation des points de terminaison du proxy RDS”.

Retard de réplica et clusters de base de données multi-AZ

Le retard de réplica est la différence de temps entre la dernière transaction au niveau de l'instance de base de données d'enregistreur et la dernière transaction appliquée sur une instance de base de données de lecteur. La CloudWatch métrique Amazon `ReplicaLag` représente ce décalage horaire. Pour plus d'informations sur CloudWatch les métriques, consultezSurveillance des métriques Amazon RDS () avec Amazon CloudWatch.

Bien que les clusters de base de données Multi-AZ permettent des performances d'écriture élevées, un retard de réplica peut toujours se produire en raison de la nature de la réplication basée sur

le moteur. Étant donné que tout basculement doit d'abord résoudre le retard du réplica avant de promouvoir une nouvelle instance de base de données d'enregistreur, la surveillance et la gestion de ce retard de réplica sont à prendre en compte.

RDSPour les clusters de base de données My SQL Multi-AZ, le temps de basculement dépend du délai de réplication des deux instances de base de données de lecteur restantes. Les deux instances de base de données de lecteur doivent appliquer des transactions non appliquées avant que l'une d'elles ne soit promue vers la nouvelle instance de base de données de rédacteur.

RDSPour les clusters de base de données SQL multi-AZ Postgre, le temps de basculement dépend du délai de réplication le plus faible des deux instances de base de données de lecteur restantes. L'instance de base de données de lecteur ayant le plus faible décalage de réplica doit appliquer les transactions non appliquées avant d'être promue en tant que nouvelle instance de base de données de rédacteur.

Pour un didacticiel expliquant comment créer une CloudWatch alarme lorsque le délai de réplication dépasse une durée définie, voirTutoriel : Création d'une CloudWatch alarme Amazon pour le retard de réplication d'un cluster de bases de données multi-AZ pour Amazon RDS.

Causes courantes du retard de réplica

En général, le retard de réplica se produit lorsque la charge de travail en écriture est trop élevée pour que les instances de base de données du lecteur puissent appliquer efficacement les transactions. Diverses charges de travail peuvent entraîner un retard de réplica temporaire ou continu. Voici quelques exemples de causes courantes :

- Une concurrence d'écriture élevée ou une mise à jour par lots lourde sur l'instance de base de données de l'enregistreur, ce qui entraîne un retard du processus d'application sur les instances de base de données du lecteur.
- Une charge de travail de lecture lourde qui utilise des ressources sur une ou plusieurs instances de base de données du lecteur. L'exécution de requêtes lentes ou volumineuses peut affecter le processus d'application et entraîner un retard de réplica.
- Les transactions qui modifient de grandes quantités de données ou d'DDLinstructions peuvent parfois entraîner une augmentation temporaire du délai de réplication, car la base de données doit préserver l'ordre de validation.

Atténuation du retard de réplica

Pour les clusters de bases de données multi-AZ RDS pour for My SQL et RDS pour PostgreSQL, vous pouvez atténuer le délai de réplication en réduisant la charge sur votre instance de base de données d'écriture. Vous pouvez également utiliser le contrôle de flux pour réduire le décalage de réplica. Le contrôle de flux fonctionne en limitant les écritures sur l'instance de base de données d'enregistreur, ce qui garantit que le retard de réplica ne continue pas à augmenter sans limite. La limitation des écritures est obtenue en ajoutant un délai à la fin d'une transaction, ce qui réduit le débit d'écriture sur l'instance de base de données d'enregistreur. Bien que le contrôle de flux ne garantit pas l'élimination du retard, il peut contribuer à réduire le retard global pour de nombreuses charges de travail. Les sections suivantes fournissent des informations sur l'utilisation du contrôle de flux avec RDS for My SQL et RDS pour PostgreSQL.

Atténuer le délai de réplication grâce au contrôle du flux pour RDS for My SQL

Lorsque vous utilisez RDS des clusters My SQL Multi-AZ DB, le contrôle de flux est activé par défaut à l'aide du paramètre `rpl_semi_sync_master_target_apply_lag` dynamique. Ce paramètre spécifie la limite supérieure souhaitée pour le décalage du réplica. Lorsque le délai de réplication approche de cette limite configurée, le contrôle de flux limite les transactions d'écriture sur l'instance de base de données du rédacteur pour essayer de contenir le décalage de réplication en dessous de la valeur spécifiée. Dans certains cas, le décalage de réplica peut dépasser la limite spécifiée. Par défaut, ce paramètre est défini à 120 secondes. Pour désactiver le contrôle du flux, réglez ce paramètre sur sa valeur maximale de 86 400 secondes (un jour).

Pour afficher le délai de courant injecté par le contrôle de flux, affichez le paramètre `Rpl_semi_sync_master_flow_control_current_delay` en exécutant la requête suivante.

```
SHOW GLOBAL STATUS like '%flow_control%';
```

Votre sortie doit ressembler à ce qui suit :

```
+-----------------------------------------------+-------+
| Variable_name                                 | Value |
+-----------------------------------------------+-------+
| Rpl_semi_sync_master_flow_control_current_delay | 2010  |
+-----------------------------------------------+-------+
1 row in set (0.00 sec)
```

> **Note**
>
> Le délai est affiché en microsecondes.

Lorsque Performance Insights est activé RDS pour un cluster de base de données My SQL Multi-AZ, vous pouvez surveiller l'événement d'attente correspondant à une SQL instruction indiquant que les requêtes ont été retardées par un contrôle de flux. Lorsqu'un retard a été introduit par un contrôle de flux, vous pouvez consulter l'événement d'attente `/wait/synch/cond/semisync/semi_sync_flow_control_delay_cond` correspondant à la SQL déclaration sur le tableau de bord Performance Insights. Pour afficher ces métriques, assurez-vous que le schéma de performances est activé. Pour plus d'informations sur Performance Insights, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

Atténuer le retard de réplication grâce au contrôle du flux RDS pour Postgre SQL

Lorsque vous utilisez des clusters de base RDS de données Postgre SQL Multi-AZ, le contrôle de flux est déployé sous forme d'extension. Il active un processus de travail en arrière-plan pour toutes les instances de base de données du cluster de base de données. Par défaut, les processus de travail en arrière-plan sur les instances de base de données de lecteur communiquent le retard actuel du réplica avec le processus de travail en arrière-plan sur l'instance de base de données d'enregistreur. Si le retard dépasse deux minutes sur n'importe quelle instance de base de données de lecteur, le processus de travail en arrière-plan de l'instance de base de données d'enregistreur ajoute un délai à la fin d'une transaction. Pour contrôler le seuil de retard, utilisez le paramètre `flow_control.target_standby_apply_lag`.

Lorsqu'un contrôle de flux limite un SQL processus Postgre, l'événement d'`Extension`attente et Performance `pg_stat_activity` Insights l'indiquent. La fonction `get_flow_control_stats` affiche des détails sur le délai actuellement ajouté.

Le contrôle du flux peut profiter à la plupart des charges de travail de traitement des transactions en ligne (OLTP) qui comportent des transactions courtes mais très simultanées. Si le retard est causé par des transactions de longue durée, telles que des opérations par lots, le contrôle de flux n'offre pas un avantage aussi important.

Vous pouvez désactiver le contrôle de flux en supprimant l'extension de `shared_preload_libraries` et en redémarrant votre instance de base de données.

Instantanés de clusters de bases de données multi-AZ

Amazon RDS crée et enregistre des sauvegardes automatisées de votre cluster de base de données multi-AZ pendant la fenêtre de sauvegarde configurée. RDScrée un instantané du volume de stockage de votre cluster de base de données, en sauvegardant l'intégralité du cluster et pas seulement les instances individuelles.

Vous pouvez également effectuer des sauvegardes manuelles de votre cluster de base de données multi-AZ. Pour les sauvegardes à très long terme, envisagez d'exporter les données des instantanés vers Amazon S3. Pour de plus amples informations, veuillez consulter the section called “Création d'un instantané de cluster de bases de données multi-AZ”.

Vous pouvez restaurer un cluster de base de données Multi-AZ à un moment précis dans le temps, en créant un nouveau cluster de base de données Multi-AZ. Pour obtenir des instructions, consultez the section called “Restauration d'un cluster de base de données multi-AZ à une date définie”.

Vous pouvez également restaurer un instantané de cluster de base de données multi-AZ vers un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ. Pour obtenir des instructions, consultez the section called “Restauration d'un instantané de cluster de bases de données multi-AZ dans une instance de base de données”.

Création d'un cluster de base de données multi-AZ pour Amazon RDS

Un cluster de base de données multi-AZ compte une instance de base de données d'écriture et deux instances de base de données de lecture dans trois zones de disponibilité distinctes. Les clusters de base de données multi-AZ offrent une haute disponibilité, une capacité accrue pour les charges de travail en lecture et une moindre latence par rapport aux déploiements multi-AZ. Pour de plus amples informations sur les clusters de base de données multi-AZ, consultez Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.

> **Note**
>
> Les clusters de base de données multi-AZ sont pris en charge uniquement pour les moteurs de base de données MySQL et PostgreSQL.

Prérequis des clusters de bases de données

> **Important**
>
> Avant de pouvoir créer un cluster de base de données Multi-AZ, vous devez effectuer les tâches indiquées dans Configuration de votre environnement Amazon RDS.

Voici les conditions préalables à remplir avant de créer un cluster de base de données Multi-AZ.

Rubriques

- Configurer le réseau pour la base de données
- Prérequis supplémentaires

Configurer le réseau pour la base de données

Vous ne pouvez créer un cluster de bases de données multi-AZ que dans un cloud privé virtuel (VPC) basé sur un service Amazon VPC. Il doit se trouver dans une zone Région AWS comportant au moins trois zones de disponibilité. Le groupe de sous-réseaux de base de données que vous choisissez pour le cluster de base de données doit couvrir au moins trois zones de disponibilité. Cette configuration garantit que chaque instance de base de données du cluster de base de données se trouve dans une zone de disponibilité différente.

Pour configurer la connectivité entre votre nouveau cluster de bases de données et une EC2 instance Amazon dans le même VPC, faites-le lors de la création du cluster de bases de données. Pour vous connecter à votre cluster de base de données à partir de ressources autres que des EC2 instances du même VPC, configurez les connexions réseau manuellement.

Rubriques

- Configuration de la connectivité réseau automatique avec une EC2 instance
- Configuration manuelle du réseau

Configuration de la connectivité réseau automatique avec une EC2 instance

Lorsque vous créez un cluster de base de données multi-AZ, vous pouvez utiliser le AWS Management Console pour configurer la connectivité entre une EC2 instance et le nouveau cluster de bases de données. Dans ce cas, RDS configure automatiquement votre VPC et vos paramètres réseau. Le cluster de base de données est créé dans le même VPC que l' EC2 instance afin que celle-ci puisse accéder au EC2 cluster de base de données.

Les conditions requises pour connecter une EC2 instance au cluster de base de données sont les suivantes :

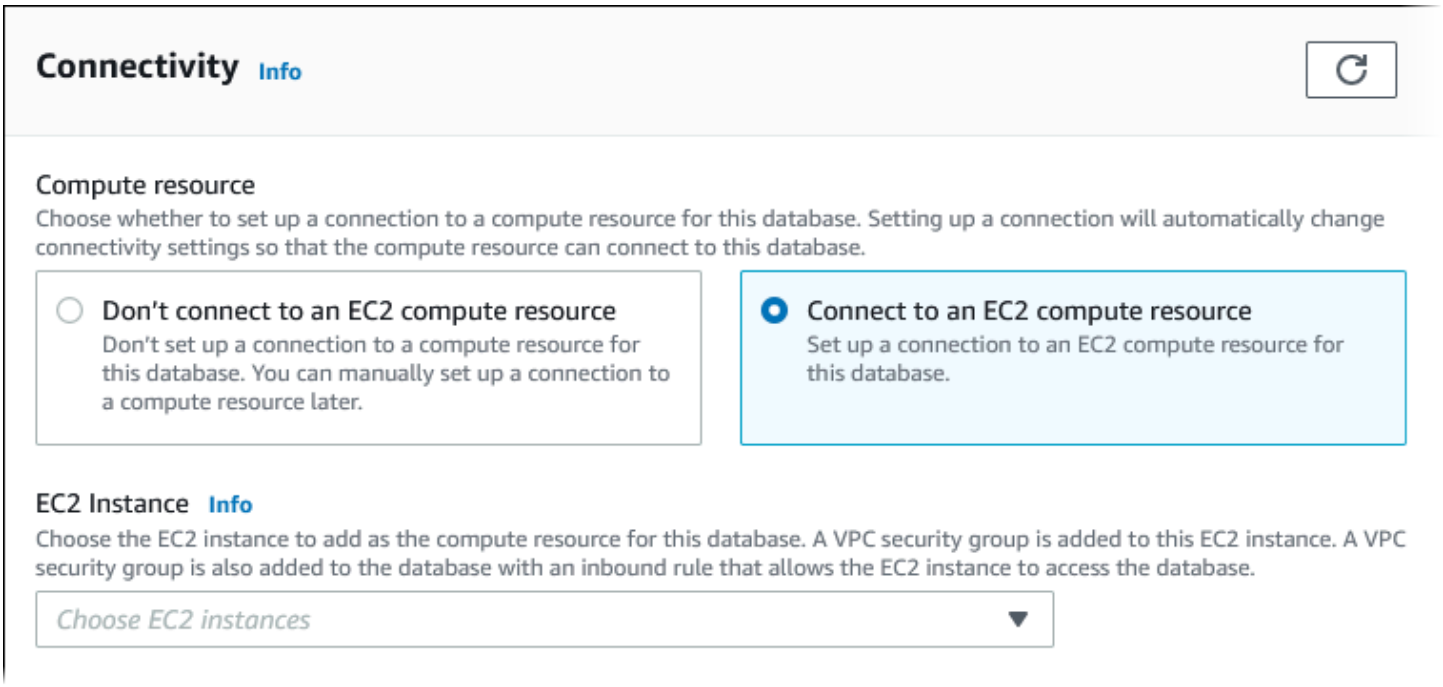
- L' EC2 instance doit exister dans le cluster de base de données Région AWS avant de créer le cluster de base de données.

  Si aucune EC2 instance n'existe dans le Région AWS, la console fournit un lien pour en créer une.
- L'utilisateur qui crée le cluster de base de données doit avoir les autorisations nécessaires pour effectuer les opérations suivantes :
  - `ec2:AssociateRouteTable`
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:AuthorizeSecurityGroupIngress`
  - `ec2:CreateRouteTable`
  - `ec2:CreateSubnet`
  - `ec2:CreateSecurityGroup`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeRouteTables`

- `ec2:DescribeSecurityGroups`
- `ec2:DescribeSubnets`
- `ec2:ModifyNetworkInterfaceAttribute`
- `ec2:RevokeSecurityGroupEgress`

Cette option permet de créer un cluster de base de données privé. Le cluster de base de données utilise un groupe de sous-réseaux de base de données avec uniquement des sous-réseaux privés pour restreindre l'accès aux ressources au sein du VPC.

Pour connecter une EC2 instance au cluster de base de données, choisissez Se connecter à une ressource de EC2 calcul dans la section Connectivité de la page Créer une base de données.



Lorsque vous choisissez Se connecter à une ressource de EC2 calcul, RDS définit automatiquement les options suivantes. Vous ne pouvez pas modifier ces paramètres sauf si vous choisissez de ne pas configurer la connectivité avec une EC2 instance en choisissant Ne pas vous connecter à une ressource de EC2 calcul.

Option console	Réglage automatique
Virtual Private Cloud (VPC)	RDS définit le VPC sur celui associé EC2 à l'instance.

Option console	Réglage automatique
Groupe de sous-réseaux de base de données	RDS nécessite un groupe de sous-réseaux de base de données avec un sous-réseau privé dans la même zone de disponibilité que l'instance. EC2 Si un groupe de sous-réseau de base de données répondant à cette exigence existe, RDS utilise alors le groupe de sous-réseau de base de données existant. Par défaut, cette option est définie sur Automatic setup (Configur ation automatique). Lorsque vous choisissez Automatic setup (Configuration automatique) et qu'aucun groupe de sous-réseaux de base de données ne répond à cette exigence, l'action suivante se produit. RDS utilise trois sous-réseaux privés disponibles dans trois zones de disponibilité où l'une des zones de disponibilité est identique à l'instance. EC2 Si un sous-réseau privé n'est pas disponible dans une zone de disponibilité, RDS crée un sous-réseau privé dans la zone de disponibilité. RDS crée ensuite le groupe de sous-réseau de base de données. Lorsqu'un sous-réseau privé est disponible, RDS utilise la table de routage qui lui est associée avec le sous-réseau et ajoute les sous-réseaux qu'il crée à cette table de routage. Lorsqu'au cun sous-réseau privé n'est disponible, RDS crée une table de routage sans accès à la passerelle Internet et ajoute les sous-réseaux qu'il crée à la table de routage. RDS vous permet également d'utiliser des groupes de sous-réseaux de base de données existants. Sélectionnez Choose existing (Choisir existants) si vous souhaitez utiliser un groupe de sous-réseaux de base de données existant de votre choix.

Option console	Réglage automatique
Accès public	RDS choisit No (Non) pour que le cluster de base de données ne soit pas publiquement accessible. Pour des raisons de sécurité, il est préférable de garder la base de données privée et de s'assurer qu'elle n'est pas accessible depuis Internet.
VPC security group (firewall) [Groupe de sécurité VPC (pare-feu)]	RDS crée un nouveau groupe de sécurité qui est employé avec le cluster de base de données. Le groupe de sécurité est nommé `rds-ec2-`*n*, où *n* est un nombre. Ce groupe de sécurité inclut une règle entrante dont le groupe de sécurité EC2 VPC (pare-feu) est la source. Ce groupe de sécurité associé au cluster de base de données permet à l' EC2 instance d'accéder au cluster de base de données. RDS crée également un nouveau groupe de sécurité associé à l' EC2 instance. Le groupe de sécurité est nommé `ec2-rds-`*n*, où *n* est un nombre. Ce groupe de sécurité comprend une règle de sortie avec le groupe de sécurité VPC du cluster de base de données comme source. Ce groupe de sécurité permet à l' EC2 instance d'envoyer du trafic vers le cluster de base de données. Vous pouvez ajouter un autre groupe de sécurité en sélection nant Create new (Créer nouveau) et en saisissant le nom du nouveau groupe de sécurité. Vous pouvez ajouter des groupes de sécurité existants en choisissant Choose existing (Choisir existant) et en sélection nant les groupes de sécurité à ajouter.

Option console	Réglage automatique
Zone de disponibilité	RDS choisit la zone de disponibilité de l' EC2 instance pour une instance de base de données dans le cadre du déploiement du cluster de base de données multi-AZ. RDS choisit de manière aléatoire une zone de disponibilité différente pour les deux autres instances de la base de données. L'instance de base de données Writer est créée dans la même zone de disponibi lité que l' EC2 instance. Il peut y avoir des coûts supplémen taires liés aux zones de disponibilité croisées si un basculement se produit et que l'instance de base de données en écriture se trouve dans une zone de disponibilité différente.

Pour plus d'informations sur ces paramètres, consultez la page Paramètres de création de clusters de base de données multi-AZ.

Si vous modifiez ces paramètres après la création du cluster de base de données, les modifications peuvent affecter la connexion entre l' EC2 instance et le cluster de base de données.

Configuration manuelle du réseau

Pour vous connecter à votre cluster de base de données à partir de ressources autres que des EC2 instances du même VPC, configurez les connexions réseau manuellement. Si vous utilisez le AWS Management Console pour créer votre cluster de base de données multi-AZ, Amazon RDS peut créer automatiquement un VPC pour vous. Une autre solution consiste à utiliser un VPC existant ou à en créer un pour votre cluster de base de données multi-AZ. Le VPC doit disposer d'au moins un sous-réseau dans au moins trois zones de disponibilité pour que vous puissiez l'utiliser avec un cluster de base de données multi-AZ. Pour plus d'informations sur VPCs, voirAmazon VPC et Amazon RDS.

Si vous n'avez pas de VPC par défaut ou que vous n'en avez pas créé et que vous n'avez pas prévu d'utiliser la console, procédez comme suit :

- Créez un VPC avec au moins un sous-réseau dans chacune des trois zones de disponibilité de la AWS région dans laquelle vous souhaitez déployer votre cluster de bases de données. Pour de plus amples informations, veuillez consulter Utilisation d'un(e) instance de base de données dans un VPC.

- Spécifiez un groupe de sécurité VPC qui autorise les connexions à votre cluster de bases de données. Pour plus d'informations, consultez Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC et Contrôle d'accès par groupe de sécurité.
- Spécifiez un groupe de sous-réseaux de base de données RDS avec au moins trois sous-réseaux définis dans le VPC qui peuvent être utilisés par le cluster de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.

Pour plus d'informations sur les limitations qui s'appliquent aux clusters de base de données Multi-AZ, consultez Limites des clusters de bases de données multi-AZ pour Amazon RDS.

Si vous souhaitez vous connecter à une ressource qui ne se trouve pas dans le même VPC que le cluster de bases de données multi-AZ, consultez les scénarios appropriés dans Scénarios d'accès à un d'instances de base de données dans un VPC.

Prérequis supplémentaires

Avant de créer votre cluster de base de données Multi-AZ, tenez compte des conditions préalables supplémentaires suivantes :

- Pour personnaliser les paramètres de configuration pour votre cluster de bases de données, spécifiez un groupe de paramètres de cluster de bases de données avec les valeurs de paramètres nécessaires. Pour en savoir plus sur la création ou la modification d'un groupe de paramètres de cluster de base de données, consultez Groupes de paramètres pour les clusters de bases de données multi-AZ.
- Déterminez le numéro de TCP/IP port à spécifier pour votre cluster de base de données. Dans certaines entreprises, les pare-feu bloquent les connexions à ces ports par défaut. Si le pare-feu de votre entreprise bloque le port par défaut, choisissez un autre port pour le cluster de bases de données. Toutes les instances de base de données d'un cluster de base de données utilisent le même port.
- Si la version principale du moteur de votre base de données a atteint la date de fin de support standard RDS, vous devez utiliser l'option Extended Support CLI ou le paramètre API RDS. Pour plus d'informations, consultez RDS Extended Support dansParamètres de création de clusters de base de données multi-AZ.

Création d'un cluster de base de données

Vous pouvez créer un cluster de base de données multi-AZ à l'aide de l' AWS Management Console API, de AWS CLI, ou de l'API RDS.

console

Vous pouvez créer un cluster de base de données Multi-AZ en choisissant Multi-AZ DB cluster (Cluster de base de données Multi-AZ) dans la section Availability and durability (Disponibilité et durabilité).

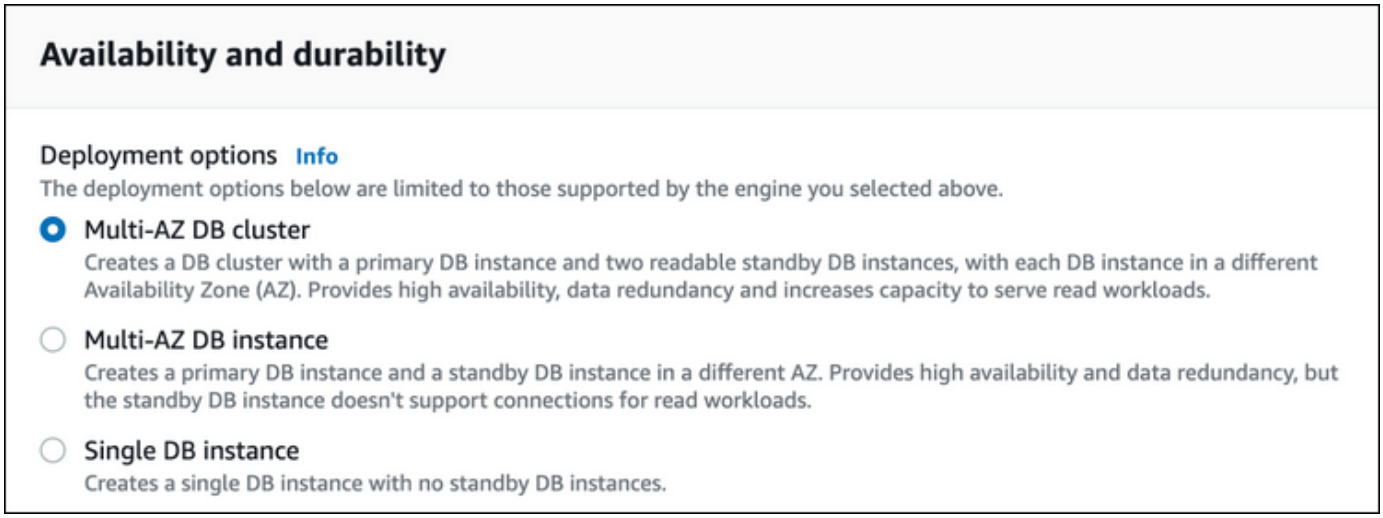
Pour créer un cluster de base de données multi-AZ à partir de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit du AWS Management Console, choisissez celui Région AWS dans lequel vous souhaitez créer le cluster de base de données.

   Pour plus d'informations sur ceux Régions AWS qui prennent en charge les clusters de bases de données multi-AZ, consultezLimites des clusters de bases de données multi-AZ pour Amazon RDS.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).

   Pour créer un cluster de base de données multi-AZ, vérifiez que Standard Create (Création standard) est sélectionné et que Easy Create (Création facile) ne l'est pas.
5. Dans Engine type (Type de moteur), choisissez MySQL ou PostgreSQL.
6. Dans Version, choisissez la version du moteur de base de données.

   Pour obtenir plus d'informations sur les versions du moteur de base de données qui prennent en charge les clusters de base de données Multi-AZ, consultez Limites des clusters de bases de données multi-AZ pour Amazon RDS.
7. Dans Templates (Modèles), choisissez le modèle approprié pour votre déploiement.
8. Dans Availability and durability (Disponibilité et durabilité), choisissez Multi-AZ DB cluster (Cluster de base de données Multi-AZ).



9. Dans DB cluster identifier (Identificateur du cluster de base de données), saisissez l'identifiant de votre cluster de base de données.
10. Dans Master username (Nom d'utilisateur principal), saisissez votre nom d'utilisateur principal ou conservez le paramètre par défaut.
11. Saisissez votre mot de passe principal :

    a. Dans la section Settings (Paramètres), ouvrez Credential Settings (Paramètres des informations d'identification).

    b. Si vous souhaitez spécifier un mot de passe, décochez la case Auto generate a password (Générer un mot de passe automatiquement) si elle est cochée.

    c. (Facultatif) Changez la valeur de Master username (Nom d'utilisateur principal).

    d. Saisissez le même mot de passe dans Master password (Mot de passe principal) et Confirm password (Confirmer le mot de passe).
12. Pour Classe d'instance de base de données, choisissez une classe d'instance de base de données. Pour obtenir la liste des classes d'instances de base de données prises en charge, consultez the section called “Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ”.
13. (Facultatif) Configurez une connexion à une ressource de calcul pour ce cluster de base de données.

    Vous pouvez configurer la connectivité entre une EC2 instance Amazon et le nouveau cluster de base de données lors de la création du cluster de base de données. Pour de plus amples informations, veuillez consulter Configuration de la connectivité réseau automatique avec une EC2 instance.

14. Dans la section Connectivité sous Groupe de sécurité VPC (pare-feu), si vous sélectionnez Créer, un groupe de sécurité VPC est créé avec une règle entrante qui autorise l'adresse IP de votre ordinateur local à accéder à la base de données.
15. Pour les sections restantes, spécifiez vos paramètres de cluster de base de données. Pour plus d'informations sur chaque paramètre, consultez Paramètres de création de clusters de base de données multi-AZ.
16. Choisissez Create database (Créer une base de données).

    Si vous choisissez de générer un mot de passe automatiquement, le bouton View credential details (Afficher les informations d'identification) apparaît sur la page Databases (Bases de données).

    Pour afficher l'identifiant principal et le mot de passe pour le cluster de base de données, choisissez View credential details (Afficher les informations d'identification).

    Pour vous connecter au cluster de base de données en tant qu'utilisateur principal, utilisez le nom d'utilisateur et le mot de passe affichés.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau.

17. Pour Databases (Bases de données), choisissez le nom du nouveau cluster de base de données.

Sur la console RDS, les détails du nouveau cluster de base de données s'affichent. Le cluster de base de données présente le statut Creating (Création en cours) tant qu'il n'est pas créé et prêt à l'emploi. Dès que l'état passe à Available (Disponible), vous pouvez vous connecter au cluster de base de données. Selon la classe de cluster de base de données et le stockage alloué, plusieurs minutes peuvent être nécessaires avant que le nouveau cluster de base de données soit disponible.

AWS CLI

Avant de créer un cluster de base de données multi-AZ à l'aide de AWS CLI, assurez-vous de remplir les conditions requises. Il s'agit notamment de créer un VPC et un groupe de sous-réseaux de base de données RDS. Pour de plus amples informations, veuillez consulter Prérequis des clusters de bases de données.

Pour créer un cluster de base de données multi-AZ à l'aide de AWS CLI, appelez la create-db-clustercommande. Spécifiez `--db-cluster-identifier`. Pour l'option `--engine`, spécifiez `mysql` ou `postgres`.

Pour plus d'informations sur chaque option, veuillez consulter Paramètres de création de clusters de base de données multi-AZ.

Pour plus d'informations sur les Régions AWS moteurs de base de données et les versions de moteurs de base de données qui prennent en charge les clusters de base de données multi-AZ, consultezLimites des clusters de bases de données multi-AZ pour Amazon RDS.

La commande `create-db-cluster` crée l'instance de base de données d'écriture pour votre cluster de base de données et deux instances de base de données de lecture. Chaque instance de base de données se trouve dans une zone de disponibilité différente.

Par exemple, la commande suivante crée un cluster de base de données multi-AZ MySQL 8.0 nommé `mysql-multi-az-db-cluster`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-cluster \
   --db-cluster-identifier mysql-multi-az-db-cluster \
   --engine mysql \
   --engine-version 8.0.32  \
   --master-username admin \
   --manage-master-user-password  \
   --port 3306 \
   --backup-retention-period 1  \
   --db-subnet-group-name default \
   --allocated-storage 4000 \
   --storage-type io1 \
   --iops 10000 \
   --db-cluster-instance-class db.m5d.xlarge
```

Dans Windows :

```
aws rds create-db-cluster ^
   --db-cluster-identifier mysql-multi-az-db-cluster ^
   --engine mysql ^
   --engine-version 8.0.32 ^
```

```
    --manage-master-user-password ^
    --master-username admin ^
    --port 3306 ^
    --backup-retention-period 1 ^
    --db-subnet-group-name default ^
    --allocated-storage 4000 ^
    --storage-type io1 ^
    --iops 10000 ^
    --db-cluster-instance-class db.m5d.xlarge
```

La commande suivante crée un cluster de base de données multi-AZ PostgreSQL 13.4 nommé `postgresql-multi-az-db-cluster`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-cluster \
    --db-cluster-identifier postgresql-multi-az-db-cluster \
    --engine postgres \
    --engine-version 13.4 \
    --manage-master-user-password \
    --master-username postgres \
    --port 5432 \
    --backup-retention-period 1  \
    --db-subnet-group-name default \
    --allocated-storage 4000 \
    --storage-type io1 \
    --iops 10000 \
    --db-cluster-instance-class db.m5d.xlarge
```

Dans Windows :

```
aws rds create-db-cluster ^
    --db-cluster-identifier postgresql-multi-az-db-cluster ^
    --engine postgres ^
    --engine-version 13.4 ^
    --manage-master-user-password ^
    --master-username postgres ^
    --port 5432 ^
    --backup-retention-period 1 ^
    --db-subnet-group-name default ^
    --allocated-storage 4000 ^
```

```
    --storage-type io1 ^
    --iops 10000 ^
    --db-cluster-instance-class db.m5d.xlarge
```

API RDS

Avant de pouvoir créer un cluster de base de données multi-AZ à l'aide de l'API RDS, veillez à remplir les différents prérequis en créant notamment un VPC et d'un groupe de sous-réseaux de base de données RDS. Pour de plus amples informations, veuillez consulter Prérequis des clusters de bases de données.

Pour créer un cluster de base de données multi-AZ à l'aide de l'API RDS, appelez l'opération Create DBCluster. Spécifiez `DBClusterIdentifier`. Pour le paramètre `Engine`, spécifiez `mysql` ou `postgresql`.

Pour plus d'informations sur chaque option, veuillez consulter Paramètres de création de clusters de base de données multi-AZ.

L'opération `CreateDBCluster` crée l'instance de base de données d'écriture pour votre cluster de base de données ainsi que deux instances de base de données de lecture. Chaque instance de base de données se trouve dans une zone de disponibilité différente.

Paramètres de création de clusters de base de données multi-AZ

Pour obtenir des détails sur les paramètres disponibles au moment de créer un cluster de base de données multi-AZ, consultez le tableau suivant. Pour plus d'informations sur les AWS CLI options, consultez create-db-cluster. Pour plus d'informations sur les paramètres de l'API RDS, consultez Create DBCluster.

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Stockage alloué	Capacité de stockage à allouer pour chaque instance de base de données du cluster de base de données (en gibioctets). Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.	Option de l'interface CLI :
		`--allocated-storage`
		Paramètre de l'API :
		`AllocatedStorage`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Mise à niveau automatique de versions mineures	Activez la mise à niveau automatique des versions mineures pour permettre au cluster de base de données de recevoir automatiquement les mises à niveau de la version mineure préférée du moteur de base de données lorsqu'elles sont disponibles. Amazon RDS effectue les mises à niveau automatiques des versions mineures dans la fenêtre de maintenance.	Option de l'interface CLI :
		`--auto-minor-version-upgrade`
		`--no-auto-minor-version-upgrade`
		Paramètre de l'API :
		`AutoMinorVersionUpgrade`

Période de rétention des sauvegardes	Nombre de jours durant lesquels les sauvegardes automatiques de votre cluster de base de données doivent être retenues. Pour un cluster de base de données multi-AZ, cette valeur doit être égale ou supérieure à **1**.
	Pour de plus amples informations, veuillez consulter Présentation des sauvegardes.
		Option de l'interface CLI :
		`--backup-retention-period`
		Paramètre de l'API :
		`BackupRetentionPeriod`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Fenêtre de sauvegarde	Période durant laquelle Amazon RDS effectue automatiquement une sauvegarde de votre cluster de base de données. Si vous n'avez pas besoin que votre base de données soit sauvegardée à un moment précis, utilisez la valeur par défaut No preference (Aucune préférence).
	
	Pour plus d'informations, consultez Présentation des sauvegardes.
		Option de l'interface CLI :
		
		`--preferred-backup-window`
		
		Paramètre de l'API :
		
		`PreferredBackupWindow`

Autorité de certification	L'autorité de certification (CA) pour le certificat de serveur utilisé par le cluster de base de données.
	
	Pour de plus amples informations, veuillez consulter .
		Option de l'interface CLI :
		
		`--ca-certificate-identifier`
		
		Paramètre de l'API RDS :
		
		`CACertificateIdentifier`

Copier les balises aux instantanés	Cette option permet de copier toutes les identifications de cluster de base de données dans un instantané de base de données lorsque vous créez un instantané.
	
	Pour de plus amples informations, veuillez consulter Marquage d' Amazon RDS.
		Option de l'interface CLI :
		
		`-copy-tags-to-snapshot`
		
		`-no-copy-tags-to-snapshot`
		
		Paramètre de l'API RDS :
		
		`CopyTagsToSnapshot`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Authentification de base de données	L'option d'authentification de base de données que vous souhaitez utiliser.
	
	Choisissez Authentification par mot de passe pour authentifier les utilisateurs de base de données avec des mots de passe de base de données uniquement.
	
	Choisissez Password and IAM DB authentication (Mot de passe et authentification de base de données IAM) pour authentifier les utilisateurs de base de données avec des mots de passe de base de données et des informations d'identification utilisateur via des utilisateurs et rôles. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.
		Option de l'interface CLI :
		
		`--enable-iam-database-authentication`
		
		`--no-enable-iam-database-authentication`
		
		Paramètre de l'API RDS :
		
		`EnableIAMDatabaseAuthentication`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Port de la base de données	Port par lequel vous souhaitez accéder au cluster de base de données. La valeur par défaut du port est indiquée.
	
	Le port ne peut pas être modifié après la création du cluster de base de données.
	
	Dans certaines entreprises, les pare-feu bloquent les connexion s aux ports par défaut. Si le pare-feu de votre entreprise bloque le port par défaut, saisissez un autre port pour votre cluster de base de données.
		Option de l'interface CLI :
		
		`--port`
		
		Paramètre de l'API RDS :
		
		`Port`

Identificateur du cluster DB	Nom de votre cluster de base de données. Nommez vos clusters de base de données de la même façon que vous nommez vos serveurs sur site. L'identifiant de votre cluster de base de données peut contenir jusqu'à 63 caractères alphanumé riques et doit être unique pour votre compte dans la AWS région que vous avez choisie.	Option de l'interface CLI :
		
		`--db-cluster-identifier`
		
		Paramètre de l'API RDS :
		
		`DBClusterIdentifier`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Classe d'instances de base de données	Capacité de calcul et de mémoire de chaque instance de base de données contenue dans le cluster de base de données multi-AZ, par exemple `db.m5d.xlarge` .
	
	Dans la mesure du possible, choisissez une classe d'instance de base de données suffisamment grande pour qu'un ensemble de travail de requête classique puisse tenir dans la mémoire. Lorsque les ensembles de travail sont en mémoire, le système peut éviter d'écrire sur le disque, ce qui améliore les performances.
	
	Pour obtenir la liste des classes d'instances de base de données prises en charge, consultez the section called “Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ”.
		Option de l'interface CLI :
		
		`--db-cluster-instance-class`
		
		Paramètre de l'API RDS :
		
		`DBClusterInstanceClass`

Groupe de paramètres de cluster de bases de données	Groupe de paramètres de cluster de bases de données que vous souhaitez associer au cluster de bases de données.
	
	Pour de plus amples informations, veuillez consulter Groupes de paramètres pour les clusters de bases de données multi-AZ.
		Option de l'interface CLI :
		
		`--db-cluster-parameter-group-name`
		
		Paramètre de l'API RDS :
		
		`DBClusterParameterGroupName`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Version du moteur de base de données	Version du moteur de base de données que vous souhaitez utiliser.	Option de l'interface CLI :
		`--engine-version`
		Paramètre de l'API RDS :
		`EngineVersion`

Groupe de paramètres de cluster de bases de données	Le groupe de paramètres de l'instance de base de données à associer au cluster de base de données. Pour de plus amples informati ons, veuillez consulter Groupes de paramètres pour les clusters de bases de données multi-AZ.	Option de l'interface CLI : `--db-cluster-parameter-grou p-name` Paramètre de l'API RDS : **`DBClusterParameterGroupName`**

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Groupe de sous-réseaux de base de données	Le groupe de sous-réseaux de base de données à utiliser pour le cluster de bases de données. Sélectionnez Choose existing (Choisir existants) pour utiliser un groupe de sous-réseaux de base de données. Choisissez ensuite le groupe de sous-réseaux requis dans la liste déroulante Existing DB subnet groups (Groupes de sous-réseaux de base de données existants).
	
	Choisissez Automatic setup (Configuration automatique) pour permettre à RDS de sélectionner un groupe de sous-réseaux de base de données compatible. S'il n'en existe aucun, RDS crée un nouveau groupe de sous-réseaux pour votre cluster.
	
	Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.
		Option de l'interface CLI :
		
		`--db-subnet-group-name`
		
		Paramètre de l'API RDS :
		
		`DBSubnetGroupName`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Deletion protection (Protection contre la suppression)	Sélectionnez Enable deletion protection (Activer la protection de la suppression) pour empêcher la suppression de votre cluster de bases de données. Si vous créez un cluster de base de données de production avec la console, la protection contre la suppression est activée par défaut.
	
	Pour de plus amples informations, veuillez consulter Suppression d'une instance DB.
		Option de l'interface CLI :
		
		`--deletion-protection`
		
		`--no-deletion-protection`
		
		Paramètre de l'API RDS :
		
		`DeletionProtection`

Chiffrement	Choisissez Encryption (Chiffrement) pour activer le chiffrement au repos pour ce cluster de base de données.
	
	Le chiffrement est activé par défaut pour les clusters de base de données multi-AZ.
	
	Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.
		Options d'interface de ligne de commande :
		
		`--kms-key-id`
		
		`--storage-encrypted`
		
		`--no-storage-encrypted`
		
		Paramètres de l'API RDS :
		
		`KmsKeyId`
		
		`StorageEncrypted`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Surveillance améliorée	Choisissez Enhanced monitoring (Surveillance améliorée) pour activer la collecte de métriques en temps réel pour le système d'exploitation sur lequel votre cluster de base de données s'exécute.
	
	Pour de plus amples informations, veuillez consulter Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.
		Options d'interface de ligne de commande :
		
		`--monitoring-interval`
		
		`--monitoring-role-arn`
		
		Paramètres de l'API RDS :
		
		`MonitoringInterval`
		
		`MonitoringRoleArn`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Nom de la base de données initiale	Nom de la base de données dans votre cluster de base de données. Si vous ne fournissez pas de nom, Amazon RDS ne crée pas de base de données dans ce cluster de bases de données pour MySQL. Cependant, il crée une base de données sur le cluster de bases de données pour PostgreSQL. Le nom ne peut pas être un mot réservé par le moteur de base de données. Il a d'autres contraintes en fonction du moteur de base de données.
	
	MySQL :
	
	• Il doit contenir entre 1 et 64 caractères alphanumériques.
	
	PostgreSQL :
	
	• Il doit contenir entre 1 et 63 caractères alphanumériques.
	• Il doit commencer par une lettre ou un trait de soulignem ent. Les caractères suivants peuvent être des lettres, des traits de soulignement ou des chiffres (0-9).
	• Le nom initial de la base de données est `postgres`.
		Option de l'interface CLI :
		
		`--database-name`
		
		Paramètre de l'API RDS :
		
		`DatabaseName`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Exportations des journaux	Les types de fichiers journaux de base de données à publier sur Amazon CloudWatch Logs.
	
	Pour de plus amples informations, veuillez consulter Publication des journaux de base de données sur Amazon CloudWatch Logs.
		Option de l'interface CLI :
		
		`-enable-cloudwatch-logs-exports`
		
		Paramètre de l'API RDS :
		
		`EnableCloudwatchLogsExports`

Fenêtre de maintenance	Fenêtre de 30 minutes durant laquelle les modifications en attente sont appliquées à votre cluster de base de données. Si la période n'a pas d'importance, choisissez No preference (Aucune préférence).
	
	Pour plus d'informations, consultez Fenêtre de maintenance Amazon RDS.
		Option de l'interface CLI :
		
		`--preferred-maintenance-window`
		
		Paramètre de l'API RDS :
		
		`PreferredMaintenanceWindow`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Gérez les informations d'identification principales dans AWS Secrets Manager	Sélectionnez Gérer les informations d'identification principales dans AWS Secrets Manager pour gérer le mot de passe d'utilisateur principal dans un secret, dans Secrets Manager.
	
	Vous pouvez éventuellement choisir une clé KMS à utiliser pour protéger le secret. Choisissez l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
	
	Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.
		Option de l'interface CLI :
		
		`--manage-master-user-password | --no-manage-master-user-password`
		
		`--master-user-secret-kms-key-id`
		
		Paramètre de l'API RDS :
		
		`ManageMasterUserPassword`
		
		`MasterUserSecretKmsKeyId`

Mot de passe principal	Mot de passe de votre compte utilisateur principal.	Option de l'interface CLI :
		
		`--master-user-password`
		
		Paramètre de l'API RDS :
		
		`MasterUserPassword`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Identifiant principal	Nom que vous utilisez comme nom d'utilisateur principal pour vous connecter au cluster de base de données avec tous les privilèges de base de données.
	• Il peut contenir entre 1 et 16 caractères alphanumériques et des traits de soulignement.
	• Son premier caractère doit être une lettre.
	• Il ne peut pas être un mot réservé par le moteur de base de données.
	
	Vous ne pouvez pas changer le nom de l'utilisateur principal après la création du cluster de base de données Multi-AZ.
	
	Pour en savoir plus sur les privilèges accordés à l'utilisateur principal, consultez Privilèges du compte utilisateur principal.
		Option de l'interface CLI :
		
		`--master-username`
		
		Paramètre de l'API RDS :
		
		`MasterUsername`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Performance Insights	Activez Performance Insights pour contrôler la charge de votre cluster de base de données, analyser les performances de la base de données et résoudre les problèmes éventuels.
	
	Choisissez une période de conservation pour déterminer l'historique des données de Performance Insights à conserver. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.
	
	Choisissez la clé KMS à utiliser pour protéger la clé servant à chiffrer ce volume de base de données. Choisissez l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
	
	Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.
		Options d'interface de ligne de commande :
		
		`--enable-performance-insights`
		
		`--no-enable-performance-insights`
		
		`--performance-insights-retention-period`
		
		`--performance-insights-kms-key-id`
		
		Paramètres de l'API RDS :
		
		`EnablePerformanceInsights`
		
		`PerformanceInsightsRetentionPeriod`
		
		`PerformanceInsightsKMSKeyId`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

IOPS provisionnés	Quantité d'IOPS provisionnés (opérations d'entrée/sortie par seconde) à allouer initialement pour le cluster de base de données.	Option de l'interface CLI :
		`--iops`
		Paramètre de l'API RDS :
		`Iops`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Accès public	Accessible publiquement pour doter le cluster de base de données d'une adresse IP publique, ce qui signifie qu'il est accessible en dehors du VPC. Pour être accessibl e publiquement, le cluster de base de données doit aussi se trouver dans un sous-réseau public du VPC.
	
	Non accessible publiquement pour rendre le cluster de base de données accessible uniquement à partir du VPC.
	
	Pour de plus amples informations, veuillez consulter Masquer un(e) instance de base de données dans un VPC depuis Internet.
	
	Pour pouvoir se connecter à un cluster de base de données en dehors de son VPC, il doit être accessible publiquement. De plus, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité du cluster de base de données, et d'autres conditions doivent être remplies. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.
		Option de l'interface CLI :
		
		`--publicly-accessible`
		
		`--no-publicly-accessible`
		
		Paramètre de l'API RDS :
		
		`PubliclyAccessible`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS
	Si votre cluster de base de données n'est pas accessible au public, vous pouvez utiliser une connexion AWS Site-to-Site VPN ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour de plus amples informations, veuillez consulter Confidentialité du trafic inter-réseau.	

Support étendu RDS	Sélectionnez Activer le support étendu RDS pour permettre aux versions principales du moteur prises en charge de continuer à fonctionner après la date de fin du support standard RDS.
	
	Lorsque vous créez un cluster de base de données, Amazon RDS utilise par défaut RDS Extended Support. Pour empêcher la création d'un nouveau cluster de base de données après la date de fin du support standard RDS et pour éviter les frais liés au support étendu RDS, désactivez ce paramètre. Vos clusters de base de données existants ne seront pas facturés avant la date de début des tarifs du Support étendu RDS.
	
	Pour de plus amples informations, veuillez consulter .
		Option de l'interface CLI :
		
		`--engine-lifecycle-support`
		
		Paramètre de l'API RDS :
		
		`EngineLifecycleSupport`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS
Storage throughput (Débit de stockage)	La valeur du débit de stockage pour le cluster de base de données. Ce paramètre n'est visible que si vous choisissez le type de stockage SSD à usage général (gp3). Ce paramètre n'est pas configurable et est défini automatiquement en fonction des IOPS que vous spécifiez. Pour de plus amples informations, veuillez consulter Stockage GP3 (recommandé).	Cette valeur est automatiquement calculée et ne comporte pas d'option CLI.
RDS Proxy (Proxy RDS)	Sélectionnez Create an RDS Proxy (Créer un proxy RDS) pour créer un proxy pour votre cluster de bases de données. Amazon RDS crée automatiquement un rôle IAM et un secret Secrets Manager pour le proxy.	Non disponible lors de la création d'un cluster de bases de données.

Type de stockage	Type de stockage pour votre cluster de base de données.
	
	Seul le stockage SSD à usage général (gp3), IOPS provisionné (io1) et SSD IOPS provisionné (io2) est pris en charge.
	
	Pour de plus amples informations, veuillez consulter Types de stockage Amazon RDS.
		Option de l'interface CLI :
		
		`--storage-type`
		
		Paramètre de l'API RDS :
		
		`StorageType`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS
Virtual Private Cloud (VPC)	Un VPC basé sur le service Amazon VPC à associer à ce cluster de bases de données. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.	Pour la CLI et l'API, vous spécifiez le groupe de sécurité VPC. IDs

VPC security group (firewall) [Groupe de sécurité VPC (pare-feu)]	Groupes de sécurité à associer au cluster de base de données.
	
	Pour de plus amples informations, veuillez consulter Présentation des groupes de sécurité VPC.
		Option de l'interface CLI :
		
		`--vpc-security-group-ids`
		
		Paramètre de l'API RDS :
		
		`VpcSecurityGroupIds`

Paramètres non applicables pendant la création de clusters de base de données multi-AZ

Les paramètres suivants de la AWS CLI commande `create-db-cluster`et du fonctionnement de l'API RDS ne s'appliquent `CreateDBCluster`pas aux clusters de bases de données multi-AZ.

Vous ne pouvez pas non plus spécifier ces paramètres pour les clusters de base de données multi-AZ dans la console.

AWS CLI réglage	Paramètre de l'API RDS
`--availability-zones`	`AvailabilityZones`
`--backtrack-window`	`BacktrackWindow`
`--character-set-name`	`CharacterSetName`
`--domain`	`Domain`
`--domain-iam-role-name`	`DomainIAMRoleName`

AWS CLI réglage	Paramètre de l'API RDS
`--enable-global-write-forwarding | --no-enable-global-write-forwarding`	`EnableGlobalWriteForwarding`
`--enable-http-endpoint | --no-enable-http-endpoint`	`EnableHttpEndpoint`
`--global-cluster-identifier`	`GlobalClusterIdentifier`
`--option-group-name`	`OptionGroupName`
`--pre-signed-url`	`PreSignedUrl`
`--replication-source-identifier`	`ReplicationSourceIdentifier`
`--scaling-configuration`	`ScalingConfiguration`

Connexion à un cluster de bases de données multi-AZ pour Amazon RDS

Un cluster de base de données multi-AZ compte trois instances de base de données (et non une). Chaque connexion est gérée par une instance de base de données spécifique. Lorsque vous vous connectez à un cluster de base de données multi-AZ, le nom d'hôte et le port que vous spécifiez pointent vers un nom de domaine complet appelé point de terminaison. Le cluster de base de données multi-AZ utilise le mécanisme du point de terminaison pour faire abstraction de ces connexions. Vous n'avez donc pas besoin de spécifier exactement à quelle instance de base de données du cluster de base de données vous souhaitez vous connecter. Ainsi, vous n'avez pas besoin de coder en dur tous les noms d'hôtes ou d'écrire votre propre logique de réacheminement des connexions lorsque certaines instances de base de données ne sont pas disponibles.

Le point de terminaison d'écriture se connecte à l'instance de base de données de rédacteur du cluster de base de données, qui prend en charge les opérations de lecture et d'écriture. Le point de terminaison du lecteur se connecte à l'une des deux instances de base de données de lecteur, qui ne prennent en charge que les opérations de lecture.

En utilisant des points de terminaison, vous pouvez mapper chaque connexion à l'instance ou groupe d'instances de base de données approprié, selon votre cas d'utilisation. Par exemple, pour exécuter DDL des DML instructions, vous pouvez vous connecter à l'instance de base de données qui est l'instance de base de données du rédacteur. Pour effectuer des requêtes, vous pouvez vous connecter au point de terminaison du lecteur, le cluster de base de données Multi-AZ gérant automatiquement les connexions entre les instances de base de données de lecteur. Pour le diagnostic et le réglage, vous pouvez vous connecter au point de terminaison d'une instance de base de données spécifique pour en examiner les détails.

Pour en savoir plus sur la connexion à une instance de base de données, consultez Connexion à une RDS instance de base de données Amazon.

Pour plus d'informations sur la connexion aux clusters de bases de données multi-AZ, consultez les rubriques suivantes.

Rubriques

- Points de terminaison de cluster
- Points de terminaison du lecteur
- Point de terminaison d'instance
- Connexions à haute disponibilité

- Connexion à des clusters de bases de données multi-AZ avec les AWS pilotes pour Amazon RDS

Types de points de terminaison de cluster de base de données multi-AZ

Un point de terminaison est représenté par un identificateur unique qui contient une adresse d'hôte. Voici types de points de terminaison accessibles à un cluster de base de données multi-AZ :

Point de terminaison de cluster

Dans le cas d'un cluster de base de données multi-AZ, un point de terminaison de cluster (ou point de terminaison d'écriture) se connecte à l'instance de base de données d'écriture active du cluster. Ce point de terminaison est le seul à pouvoir effectuer des opérations d'écriture telles que DML des instructions DDL et. Ce point de terminaison peut également effectuer des opérations de lecture.

Chaque cluster de base de données multi-AZ dispose d'un point de terminaison de cluster et d'une instance de base de données d'écriture.

Vous utilisez le point de terminaison du cluster pour toutes les opérations d'écriture sur le cluster de base de données, y compris les insertions, les mises à jour, les suppressions et les DDL modifications. Vous pouvez aussi utiliser le point de terminaison de cluster pour les opérations de lecture, par exemple les requêtes.

En cas de défaillance de l'instance de base de données d'écriture active d'un cluster de base de données, le cluster de base de données multi-AZ bascule automatiquement vers une nouvelle instance de base de données d'écriture. Pendant le basculement, le cluster de base de données continue de traiter les demandes de connexion au point de terminaison de cluster à partir de la nouvelle instance de base de données d'écriture, avec une interruption de service minime.

L'exemple suivant illustre un point de terminaison de cluster pour un cluster de base de données multi-AZ.

```
mydbcluster.cluster-123456789012.us-east-1.rds.amazonaws.com
```

Pour plus d'informations sur la connexion aux points de terminaison du cluster, consultezPoints de terminaison de cluster.

Point de terminaison du lecteur

Dans le cas d'un cluster de base de données Multi-AZ, un point de terminaison du lecteur prend en charge les connexions en lecture seule au cluster de base de données. Utilisez le point de

terminaison de lecteur pour les opérations de lecture, par exemple les requêtes `SELECT`. En traitant ces instructions sur les instances de base de données de lecture, ce point de terminaison réduit la surcharge au niveau de l'instance de base de données d'écriture. Il aide également le cluster à mettre à l'échelle la capacité à traiter simultanément les requêtes `SELECT`. Chaque cluster de base de données multi-AZ dispose d'un point de terminaison de lecteur.

Le point de terminaison du lecteur répartit la charge de chaque demande de connexion entre les instances de base de données de lecteur. Lorsque vous utilisez le point de terminaison du lecteur pour une session, vous pouvez uniquement exécuter des instructions en lecture seule, telles que `SELECT`, dans cette session.

L'exemple suivant illustre un point de terminaison de lecteur pour un cluster de base de données multi-AZ. L'intention de lecture seule d'un point de terminaison de lecteur est indiquée par le suffixe `-ro` qui figure dans le nom du point de terminaison du cluster.

```
mydbcluster.cluster-ro-123456789012.us-east-1.rds.amazonaws.com
```

Pour plus d'informations sur la connexion aux points de terminaison du lecteur, consultezPoints de terminaison du lecteur.

Point de terminaison d'instance

Un point de terminaison d'instance se connecte à une instance de base de données spécifique dans un cluster de base de données multi-AZ. Chaque instance de bases de données d'un cluster de bases de données a son propre point de terminaison d'instance unique. Par conséquent, il existe un point de terminaison d'instance pour l'instance de base de données d'écriture active du cluster de base de données, et un point de terminaison d'instance pour chaque instance de base de données de lecture du cluster de base de données.

Le point de terminaison d'instance permet de contrôler directement les connexions au cluster de base de données. Cela vous permet de gérer les cas où l'utilisation du point de terminaison de cluster ou du point de terminaison de lecteur n'est pas appropriée. Par exemple, votre application client peut exiger une répartition de charge plus précis en fonction de la charge de travail. Dans ce cas, vous pouvez configurer plusieurs clients pour qu'ils se connectent à différentes instances de base de données de lecture au sein d'un cluster de base de données afin de distribuer les charges de travail en lecture.

L'exemple suivant illustre un point de terminaison d'instance pour une instance de base de données au sein d'un cluster de base de donnes multi-AZ.

```
mydbinstance.123456789012.us-east-1.rds.amazonaws.com
```

Pour plus d'informations sur la connexion aux points de terminaison de l'instance, consultezPoint de terminaison d'instance.

Affichage des points de terminaison

Utilisez la console ou Amazon RDS API pour afficher les points de terminaison du cluster, du lecteur et de l'instance. AWS CLI

Console

Dans le AWS Management Console, vous pouvez voir le point de terminaison du cluster et le point de terminaison du lecteur sur la page de détails de chaque cluster de base de données multi-AZ. La page de détails de chaque instance de base de données présente le point de terminaison d'instance.

AWS CLI

Avec le AWS CLI, vous pouvez voir les points de terminaison du rédacteur et du lecteur dans la sortie de la describe-db-clusterscommande. Par exemple, la commande suivante affiche les attributs du point de terminaison pour tous les clusters de votre AWS région actuelle.

```
aws rds describe-db-cluster-endpoints
```

Amazon RDS API

Avec Amazon RDSAPI, vous pouvez récupérer les points de terminaison en appelant l'action DescribeDBCluster Endpoints. La sortie présente également les points de terminaison du cluster de base de données Amazon Aurora, le cas échéant.

Points de terminaison de cluster

Chaque cluster de base de données multi-AZ possède un seul point de terminaison de cluster intégré, dont le nom et les autres attributs sont gérés par AmazonRDS. Vous ne pouvez pas créer, supprimer ni modifier ce type de point de terminaison.

Vous utilisez le point de terminaison du cluster lorsque vous administrez votre cluster de base de données, effectuez des opérations d'extraction, de transformation, de chargement (ETL) ou

lorsque vous développez et testez des applications. Le point de terminaison de cluster se connecte à l'instance de base de données d'écriture du cluster. L'instance de base de données Writer est la seule instance de base de données dans laquelle vous pouvez créer des tables et des index, exécuter `INSERT` des instructions et effectuer d'autres DDL DML opérations.

L'adresse IP physique à laquelle renvoie le point de terminaison du cluster change lorsque le mécanisme de basculement promeut une nouvelle instance de base de données au rang d'instance principale de base de données d'écriture du cluster. Si vous utilisez une forme quelconque de regroupement de connexions ou de multiplexage, soyez prêt à vider ou à réduire le nombre d'informations time-to-live mises en cache. DNS Cette technique vous empêche d'essayer d'établir une connexion en lecture/écriture à une instance de base de données qui est devenue indisponible ou qui n'est plus qu'en lecture seule après un basculement.

Points de terminaison du lecteur

Le point de terminaison de lecteur vous sert pour les connexions en lecture seule au cluster de base de données multi-AZ. Ce point de terminaison aide votre cluster de base de données à gérer une charge de travail exigeante en requêtes. Le point de terminaison du lecteur est le point de terminaison que vous fournissez aux applications qui créent les rapports ou qui effectuent d'autres opérations en lecture seule sur le cluster. Le point de terminaison du lecteur envoie des connexions aux instances de base de données de lecteur disponibles dans un cluster de base de données Multi-AZ.

Chaque cluster multi-AZ possède un seul point de terminaison de lecture intégré, dont le nom et les autres attributs sont gérés par AmazonRDS. Vous ne pouvez pas créer, supprimer ni modifier ce type de point de terminaison.

Point de terminaison d'instance

Chaque instance de base de données d'un cluster de base de données multi-AZ possède son propre point de terminaison d'instance intégré, dont le nom et les autres attributs sont gérés par AmazonRDS. Vous ne pouvez pas créer, supprimer ni modifier ce type de point de terminaison. Avec un cluster de base de données multi-AZ, vous utilisez généralement plus souvent les points de terminaison d'écriture et de lecture que les points de terminaison d'instance.

Dans le day-to-day cadre des opérations, la principale méthode d'utilisation des points de terminaison d'instance consiste à diagnostiquer les problèmes de capacité ou de performance qui affectent une instance de base de données spécifique dans un cluster de base de données multi-AZ. Lorsque vous êtes connecté à une instance de base de données spécifique, vous pouvez examiner ses variables

de statut, ses métriques, etc. Cette approche vous permet de déterminer en quoi le comportement de cette instance de base de données se distingue de celui des autres instances de base de données du cluster.

Connexions à haute disponibilité

Dans le cas des clusters de base de données Multi-AZ où la haute disponibilité est importante, utilisez le point de terminaison d'écriture pour les opérations de lecture/écriture ou les connexions à usage général et le point de terminaison du lecteur pour les connexions en lecture seule. Les points de terminaison de l'enregistreur et du lecteur gèrent le basculement d'instance de base de données mieux que ne le font les points de terminaison d'instance. Contrairement aux points de terminaison d'instance, les points de terminaison de l'enregistreur et du lecteur modifient automatiquement l'instance de base de données à laquelle ils se connectent si une instance de base de données de votre cluster devient indisponible.

Si l'instance de base de données d'écriture d'un cluster de base de données échoue, Amazon RDS bascule automatiquement vers une nouvelle instance de base de données d'écriture. Une instance de base de données de lecture est alors promue au rang d'instance de base de données d'écriture. Si un basculement échoue, vous pouvez utiliser le point de terminaison d'écriture pour vous reconnecter à l'instance de base de données d'écriture nouvellement promue. Vous pouvez également utiliser le point de terminaison du lecteur pour vous reconnecter à l'une des instances de base de données de lecture du cluster de base de données. Pendant le basculement, le point de terminaison du lecteur peut brièvement diriger les connexions vers la nouvelle instance de base de données d'écriture d'un cluster de base de données après qu'une instance de base de données de lecture a été promue au rang de nouvelle instance de base de données d'écriture. Si vous concevez votre propre logique d'application pour gérer les connexions de point de terminaison d'instance, vous pouvez découvrir manuellement ou par programmation l'ensemble d'instances de base de données disponibles dans le cluster de base de données.

Connexion à des clusters de bases de données multi-AZ avec les AWS pilotes pour Amazon RDS

La AWS suite de pilotes a été conçue pour accélérer les temps de basculement et de basculement, ainsi que pour l'authentification avec AWS Secrets Manager, AWS Identity and Access Management (IAM) et Federated Identity. Les AWS pilotes s'appuient sur la surveillance de l'état du cluster de bases de données et sur la connaissance de la topologie du cluster pour déterminer le nouveau rédacteur. Cette approche réduit les temps de basculement et de basculement à un chiffre, contre des dizaines de secondes pour les pilotes open source.

À mesure que de nouvelles fonctionnalités de service sont introduites, l'objectif de la AWS suite de pilotes est de fournir un support intégré pour ces fonctionnalités de service.

Connexion à des clusters de bases de données multi-AZ à l'aide du pilote Amazon Web Services (AWS) JDBC

Le JDBC pilote Amazon Web Services (AWS) est conçu comme un JDBC wrapper avancé pour aider les applications à tirer parti des fonctionnalités des bases de données en cluster. Ce wrapper complète et étend les fonctionnalités d'un JDBC pilote existant. Le pilote est compatible avec les pilotes communautaires suivants :

- Mon SQL connecteur/J
- MariaDB Connector/J
- pg JDBC

Pour installer le AWS JDBC pilote, ajoutez le fichier .jar du AWS JDBC pilote (situé dans l'application`CLASSPATH`) et conservez les références au pilote communautaire correspondant. Mettez à jour le URL préfixe de connexion correspondant comme suit :

- `jdbc:mysql://` sur `jdbc:aws-wrapper:mysql://`
- `jdbc:mariadb://` sur `jdbc:aws-wrapper:mariadb://`
- `jdbc:postgresql://` sur `jdbc:aws-wrapper:postgresql://`

Pour plus d'informations sur le AWS JDBC pilote et des instructions complètes pour son utilisation, consultez le GitHub référentiel de JDBC pilotes Amazon Web Services (AWS).

Connexion à des clusters de bases de données multi-AZ avec le pilote Python Amazon Web Services (AWS)

Le pilote Python Amazon Web Services (AWS) est conçu comme un wrapper Python avancé. Ce wrapper complète et étend les fonctionnalités du pilote open source Psycopg. Le pilote AWS Python prend en charge les versions 3.8 et supérieures de Python. Vous pouvez installer le `aws-advanced-python-wrapper` package à l'aide de la `pip` commande, en même temps que les packages `psycopg` open source.

Pour plus d'informations sur le pilote AWS Python et des instructions complètes pour son utilisation, consultez le GitHub référentiel de pilotes Python Amazon Web Services (AWS).

Connexion automatique d'une ressource AWS de calcul et d'un cluster de base de données multi-AZ pour Amazon RDS

Vous pouvez connecter automatiquement un cluster de base de données multi-AZ et des ressources de AWS calcul telles que les instances et les AWS Lambda fonctions Amazon Elastic Compute Cloud (AmazonEC2).

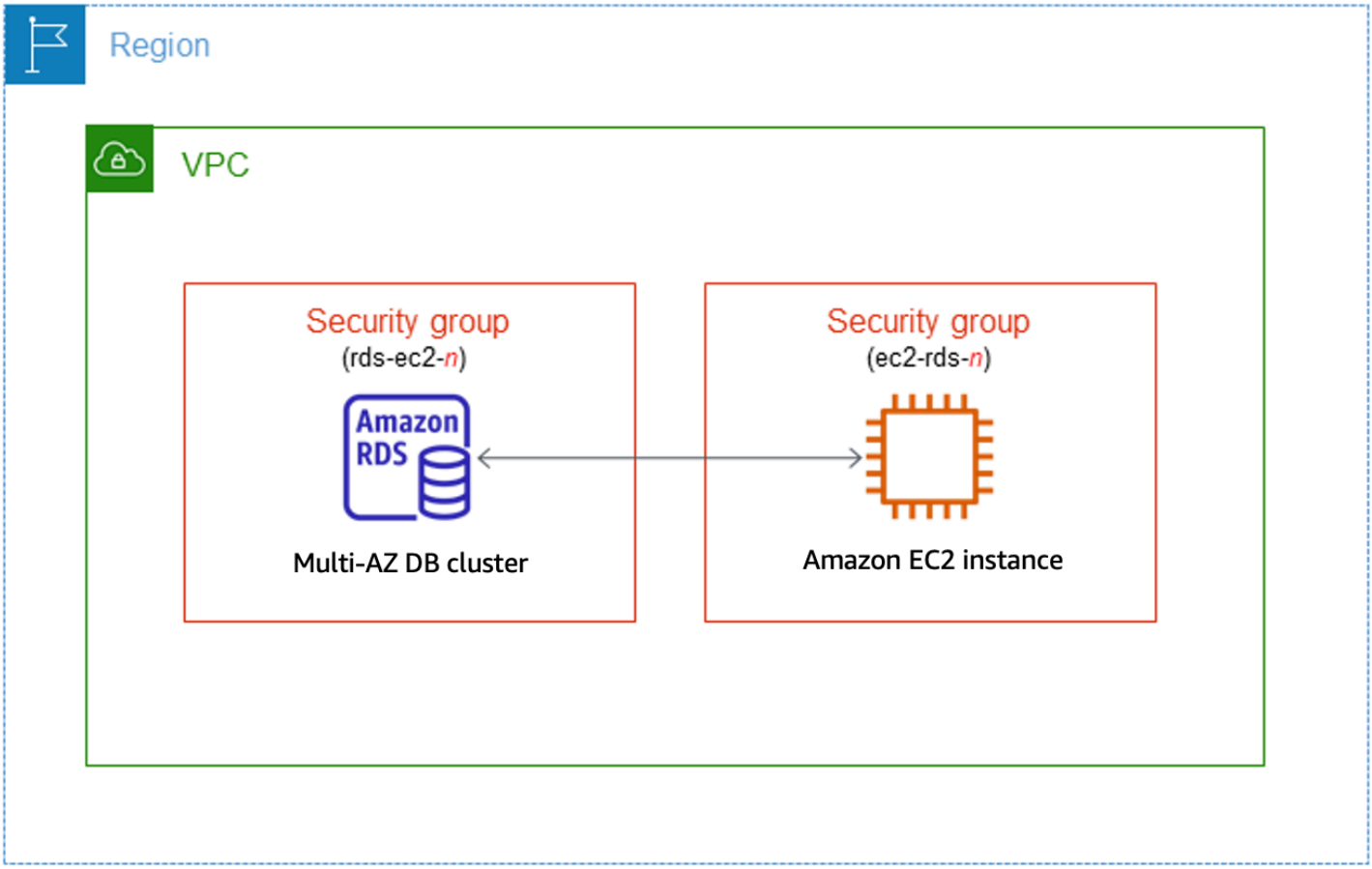
Les rubriques suivantes fournissent des instructions détaillées pour configurer les paramètres réseau, les groupes de sécurité et les paramètres de connexion afin d'établir des connexions fiables aux instances Amazon RDS DB dans le cadre d'un déploiement de cluster de base de données multi-AZ. Ils se concentrent sur l'optimisation de la connectivité réseau et des performances pour les applications interagissant avec un cluster de base de données multi-AZ, garantissant ainsi des opérations de données sécurisées et efficaces.

Rubriques

- Connexion automatique d'une instance EC2 et d'un cluster de bases de données multi-AZ
- Connexion automatique d'une fonction Lambda et d'un cluster de bases de données multi-AZ

Connexion automatique d'une instance EC2 et d'un cluster de bases de données multi-AZ

Vous pouvez utiliser la console Amazon RDS pour simplifier la configuration d'une connexion entre une instance Amazon Elastic Compute Cloud (Amazon EC2) et un cluster de bases de données multi-AZ. Souvent, votre cluster de bases de données multi-AZ se trouve dans un sous-réseau privé et votre instance EC2 se trouve dans un sous-réseau public au sein d'un VPC. Vous pouvez utiliser un client SQL sur votre instance EC2 pour vous connecter à votre cluster de bases de données multi-AZ. L'instance EC2 peut également exécuter des serveurs web ou des applications qui accèdent à votre cluster de bases de données multi-AZ.



Si vous souhaitez vous connecter à une instance EC2 qui ne se trouve pas dans le même VPC que le cluster de base de données multi-AZ, consultez les scénarios dans the section called “Scénarios d'accès à un d'instances de base de données dans un VPC”.

Rubriques

- Présentation de la connectivité automatique avec une instance EC2
- Connexion automatique d'une instance EC2 et d'un cluster de base de données multi-AZ
- Affichage des ressources de calcul connectées

Présentation de la connectivité automatique avec une instance EC2

Lorsque vous établissez automatiquement une connexion entre une instance EC2 et un cluster de bases de données multi-AZ, Amazon RDS configure le groupe de sécurité VPC pour votre instance EC2 et pour votre cluster de bases de données.

Voici les conditions requises pour connecter une instance EC2 au cluster de base de données multi-AZ :

- L'instance EC2 doit exister dans le même VPC que le cluster de base de données multi-AZ.

  S'il n'y a pas d'instances EC2 dans le même VPC, la console fournit un lien pour en créer une.
- L'utilisateur qui configure la connectivité doit avoir les autorisations nécessaires pour effectuer les opérations EC2 :
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:AuthorizeSecurityGroupIngress`
  - `ec2:CreateSecurityGroup`
  - `ec2:DescribeInstances`
  - `ec2:DescribeNetworkInterfaces`
  - `ec2:DescribeSecurityGroups`
  - `ec2:ModifyNetworkInterfaceAttribute`
  - `ec2:RevokeSecurityGroupEgress`

Lorsque vous configurez une connexion à une instance EC2, Amazon RDS agit en fonction de la configuration actuelle des groupes de sécurité associés au cluster de bases de données multi-AZ et à l'instance EC2, comme décrit dans le tableau suivant.

Configuration du groupe de sécurité RDS actuel	Configuration du groupe de sécurité EC2 actuel	Action RDS
Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n* (où *n* est un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source.	Un ou plusieurs groupes de sécurité sont associés à l'instance EC2 avec un nom qui correspond au modèle `rds-ec2-`*n* (où *n* est un nombre). Un groupe de sécurité qui correspon d au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle sortante avec le groupe de sécurité du VPC du cluster	Amazon RDS n'entreprend aucune action. Une connexion était déjà configurée automatiquement entre l'instance EC2 et le cluster de base de données multi-AZ. Comme une connexion existe déjà entre l'instance EC2 et la base de données RDS, les groupes de sécurité ne sont pas modifiés.

Configuration du groupe de sécurité RDS actuel	Configuration du groupe de sécurité EC2 actuel	Action RDS
	de base de données multi-AZ comme source.	

Configuration du groupe de sécurité RDS actuel	Configuration du groupe de sécurité EC2 actuel	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. • Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. Toutefois, aucun de ces groupes de sécurité ne peut être utilisé pour la connexion à l'instance EC2. Ce groupe de sécurité ne peut pas être utilisé s'il n'a pas de règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source. Un groupe de sécurité ne peut pas non plus être utilisé s'il a été modifié. Des exemples de modifications incluent l'ajout d'une règle ou la modification du port d'une règle existante.	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à l'instanc e EC2 avec un nom qui correspond au modèle `ec2-rds-`*n*. • Un ou plusieurs groupes de sécurité sont associés à l'instance EC2 avec un nom qui correspond au modèle `ec2-rds-`*n*. Toutefois , aucun de ces groupes de sécurité ne peut être utilisé pour la connexion au cluster de base de données multi-AZ. Un groupe de sécurité ne peut pas être utilisé s'il n'a pas de règle sortante avec le groupe de sécurité du VPC du cluster de base de données multi-AZ comme source. Un groupe de sécurité ne peut pas non plus être utilisé s'il a été modifié.	RDS action: create new security groups

Configuration du groupe de sécurité RDS actuel	Configuration du groupe de sécurité EC2 actuel	Action RDS
Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source.	Un ou plusieurs groupes de sécurité sont associés à l'instance EC2 avec un nom qui correspond au modèle `ec2-rds-`*n*. Toutefois, aucun de ces groupes de sécurité ne peut être utilisé pour la connexion au cluster de base de données multi-AZ. Un groupe de sécurité ne peut pas être utilisé s'il n'a pas de règle sortante avec le groupe de sécurité du VPC du cluster de base de données multi-AZ comme source. Un groupe de sécurité ne peut pas non plus être utilisé s'il a été modifié.	RDS action: create new security groups
Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source.	Il existe un groupe de sécurité EC2 valide pour la connexion , mais il n'est pas associé à l'instance EC2. Le nom de ce groupe de sécurité correspond au modèle `rds-ec2-`*n*. Il n'a pas été modifié. Il comprend une seule règle sortante avec le groupe de sécurité du VPC du cluster de base de données multi-AZ comme source.	RDS action: associate EC2 security group

Configuration du groupe de sécurité RDS actuel	Configuration du groupe de sécurité EC2 actuel	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. • Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-ec2-`*n*. Toutefois, aucun de ces groupes de sécurité ne peut être utilisé pour la connexion à l'instance EC2. Ce groupe de sécurité ne peut pas être utilisé s'il n'a pas de règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source. Un groupe de sécurité ne peut pas non plus être utilisé s'il a été modifié.	Un ou plusieurs groupes de sécurité sont associés à l'instance EC2 avec un nom qui correspond au modèle `rds-ec2-`*n*. Un groupe de sécurité qui correspon d au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle sortante avec le groupe de sécurité du VPC du cluster de base de données multi-AZ comme source.	RDS action: create new security groups

Action RDS : créer de nouveaux groupes de sécurité

Amazon RDS entreprend les actions suivantes :

- Crée un nouveau groupe de sécurité qui correspond au modèle `rds-ec2-`*n*. Ce groupe de sécurité comprend une règle entrante avec le groupe de sécurité du VPC de l'instance EC2 comme source. Ce groupe de sécurité est associé au cluster de base de données multi-AZ et permet à l'instance EC2 d'accéder au cluster de base de données multi-AZ.
- Crée un nouveau groupe de sécurité qui correspond au modèle `ec2-rds-`*n*. Ce groupe de sécurité comprend une règle sortante avec le groupe de sécurité du VPC du cluster de base de données multi-AZ comme source. Ce groupe de sécurité est associé à l'instance EC2 et permet à l'instance EC2 d'envoyer du trafic vers le cluster de base de données multi-AZ.

Action RDS : associer un groupe de sécurité EC2

Amazon RDS associe le groupe de sécurité EC2 existant valide à l'instance EC2. Ce groupe de sécurité permet à l'instance EC2 d'envoyer du trafic au cluster de base de données multi-AZ.

Connexion automatique d'une instance EC2 et d'un cluster de base de données multi-AZ

Avant de configurer une connexion entre une instance EC2 et une base de données RDS, assurez-vous de répondre aux exigences décrites dans Vue d'ensemble de la connectivité automatique avec une EC2 instance.

Si vous modifiez ces groupes de sécurité après avoir configuré la connectivité, cela peut affecter la connexion entre l'instance EC2 et la base de données RDS.

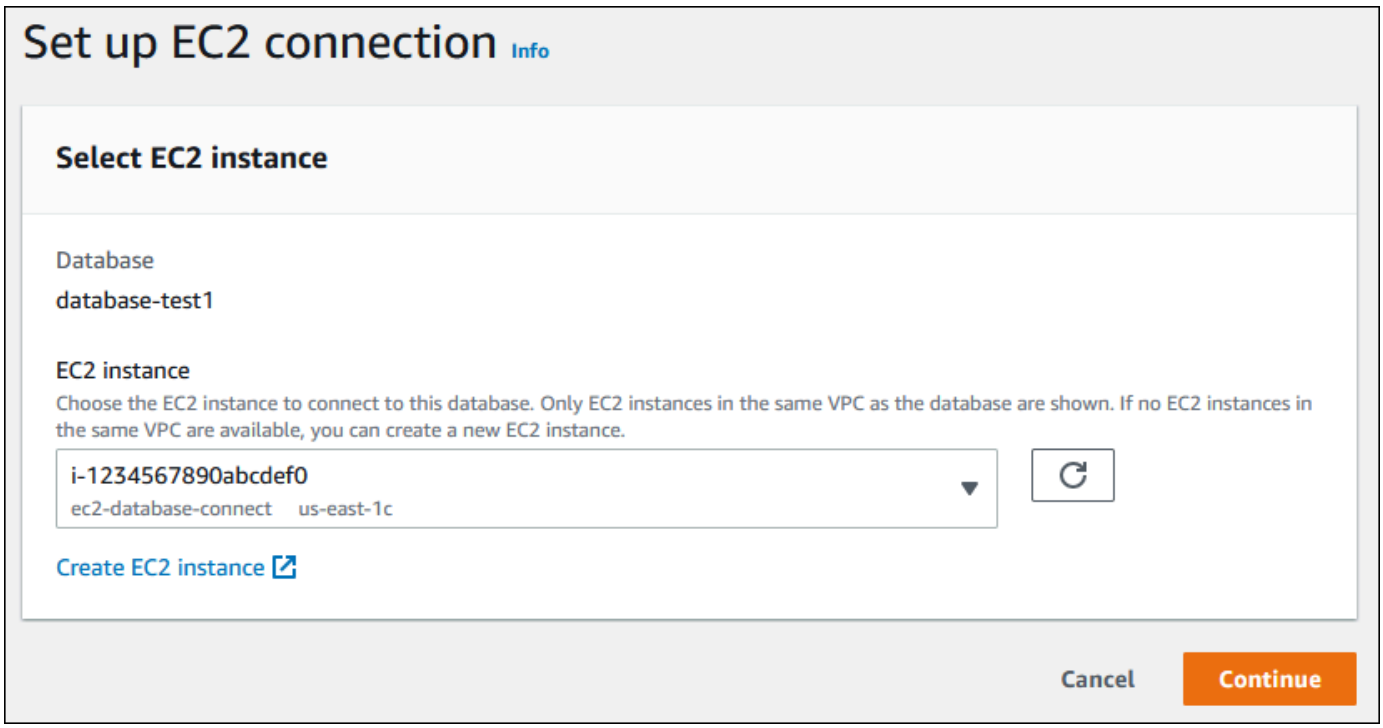
> **Note**
>
> Vous pouvez uniquement configurer automatiquement une connexion entre une instance EC2 et une base de données RDS à l'aide de la AWS Management Console. Vous ne pouvez pas configurer une connexion automatiquement avec l'API AWS CLI ou l'API RDS.

Connecter automatiquement une instance EC2 et une base de données RDS

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases (Bases de données), puis RDS database (Base de données RDS).
3. Pour Actions, choisissez Configurer la connexion EC2.

La page Set up EC2 connection (Configurer la connexion EC2) s'affiche.

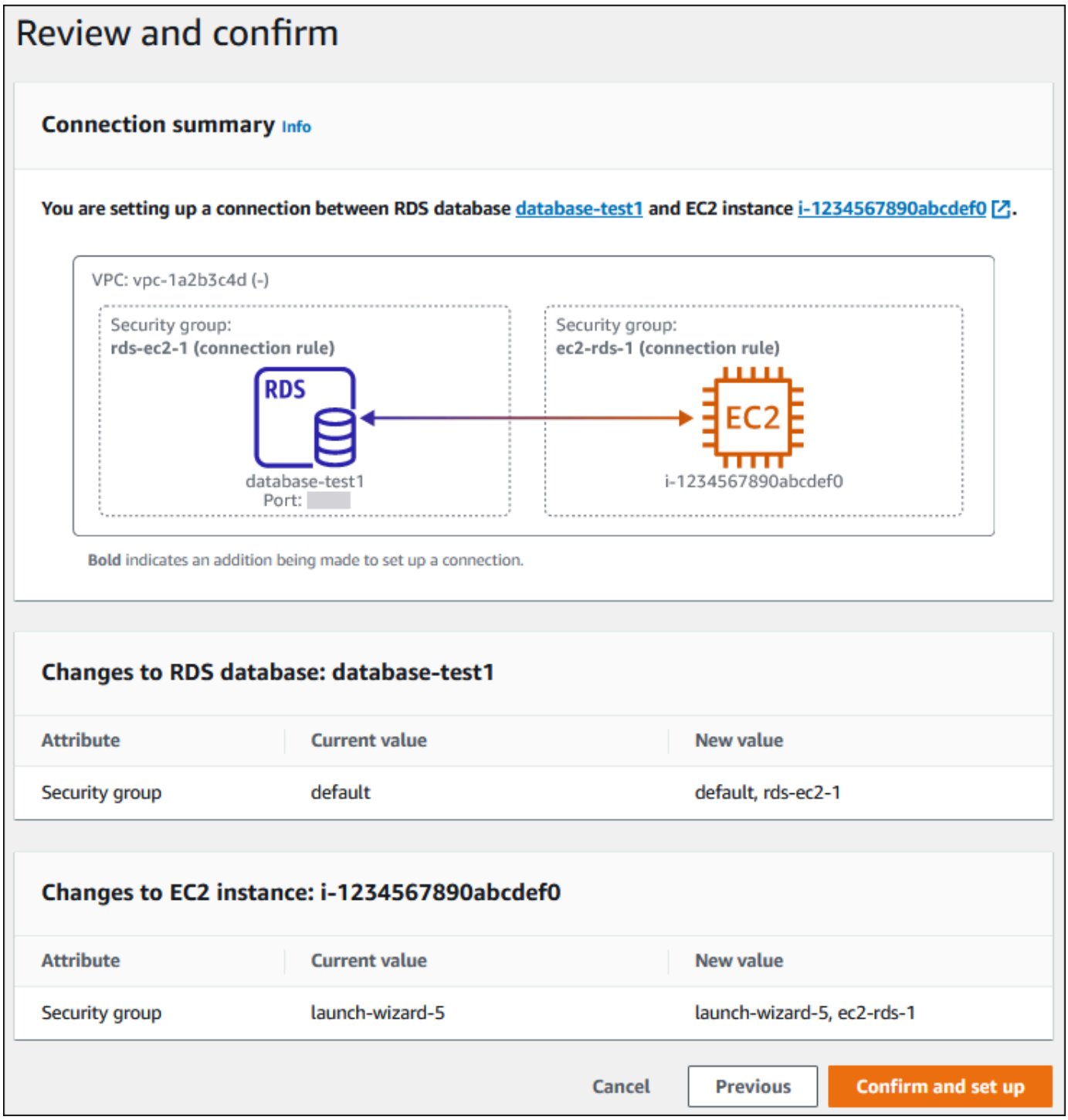
4. Sur la page Set up EC2 connection (Configurer la connexion EC2), choisissez l'instance EC2.



Si aucune instance EC2 n'existe dans le même VPC, choisissez Create EC2 instance (Créer une instance EC2) pour en créer une. Dans ce cas, assurez-vous que la nouvelle instance EC2 se trouve dans le même VPC que la base de données RDS.

5. Choisissez Continuer.

La page Review and confirm (Vérifier et confirmer) s'affiche.



6. Sur la page Review and confirm (Vérifier et confirmer), passez en revue les modifications que RDS apportera pour configurer la connectivité avec l'instance EC2.

   Si les modifications sont correctes, choisissez Confirmer et configurer.

   Si les modifications ne sont pas correctes, choisissez Previous (Précédent) ou Cancel (Annuler).

Affichage des ressources de calcul connectées

Vous pouvez utiliser le AWS Management Console pour afficher les ressources de calcul connectées à un . Les ressources affichées comprennent les connexions de ressources de calcul qui ont été configurées automatiquement. Vous pouvez configurer automatiquement la connectivité avec les ressources de calcul de la manière suivante :

- Vous pouvez sélectionner la ressource de calcul lorsque vous créez la base de données.

  Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS et Création d'un cluster de base de données multi-AZ pour Amazon RDS.
- Vous pouvez configurer la connectivité entre une base de données existante et une ressource de calcul.

  Pour plus d’informations, consultez Connexion automatique d'une EC2 instance et d'un .

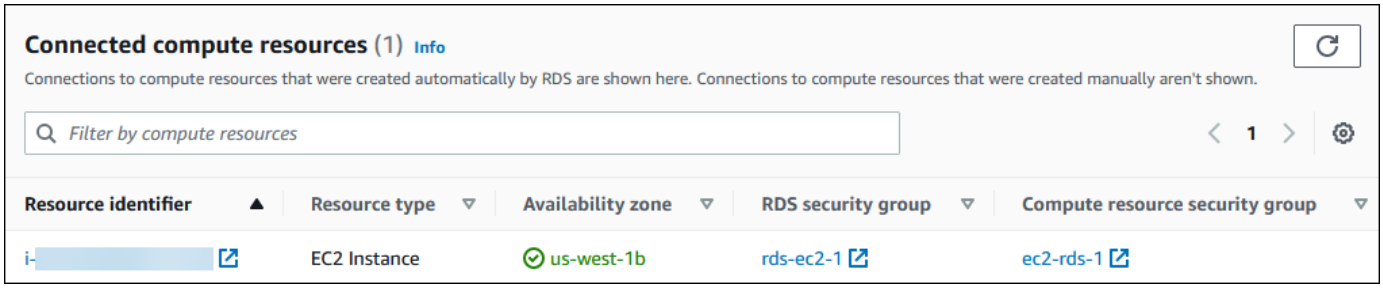
Les ressources de calcul répertoriées n'incluent pas celles qui ont été connectées manuellement à la base de données. Par exemple, vous pouvez autoriser une ressource de calcul à accéder manuellement à une base de données en ajoutant une règle au groupe de sécurité du VPC associé à la base de données.

Pour qu'une ressource de calcul soit répertoriée, les conditions suivantes doivent s'appliquer :

- Le nom du groupe de sécurité associé à la ressource de calcul correspond au modèle `ec2-rds-`*n* (où *n* est un nombre).
- Le groupe de sécurité associé à la ressource de calcul possède une règle sortante avec la plage de ports définie sur le port utilisé par la base de données RDS.
- Le groupe de sécurité associé à la ressource de calcul possède une règle de sortie dont la source est définie sur un groupe de sécurité associé à la base de données RDS.
- Le nom du groupe de sécurité associé à la base de données RDS correspond au modèle `rds-ec2-`*n* (où *n* est un nombre).
- Le groupe de sécurité associé à la base de données RDS possède une règle entrante avec la plage de ports définie sur le port utilisé par la base de données RDS.
- Le groupe de sécurité associé à la base de données RDS possède une règle d'entrée dont la source est définie sur un groupe de sécurité associé à la ressource informatique.

Pour visualiser les ressources de calcul connectées à une base de données RDS

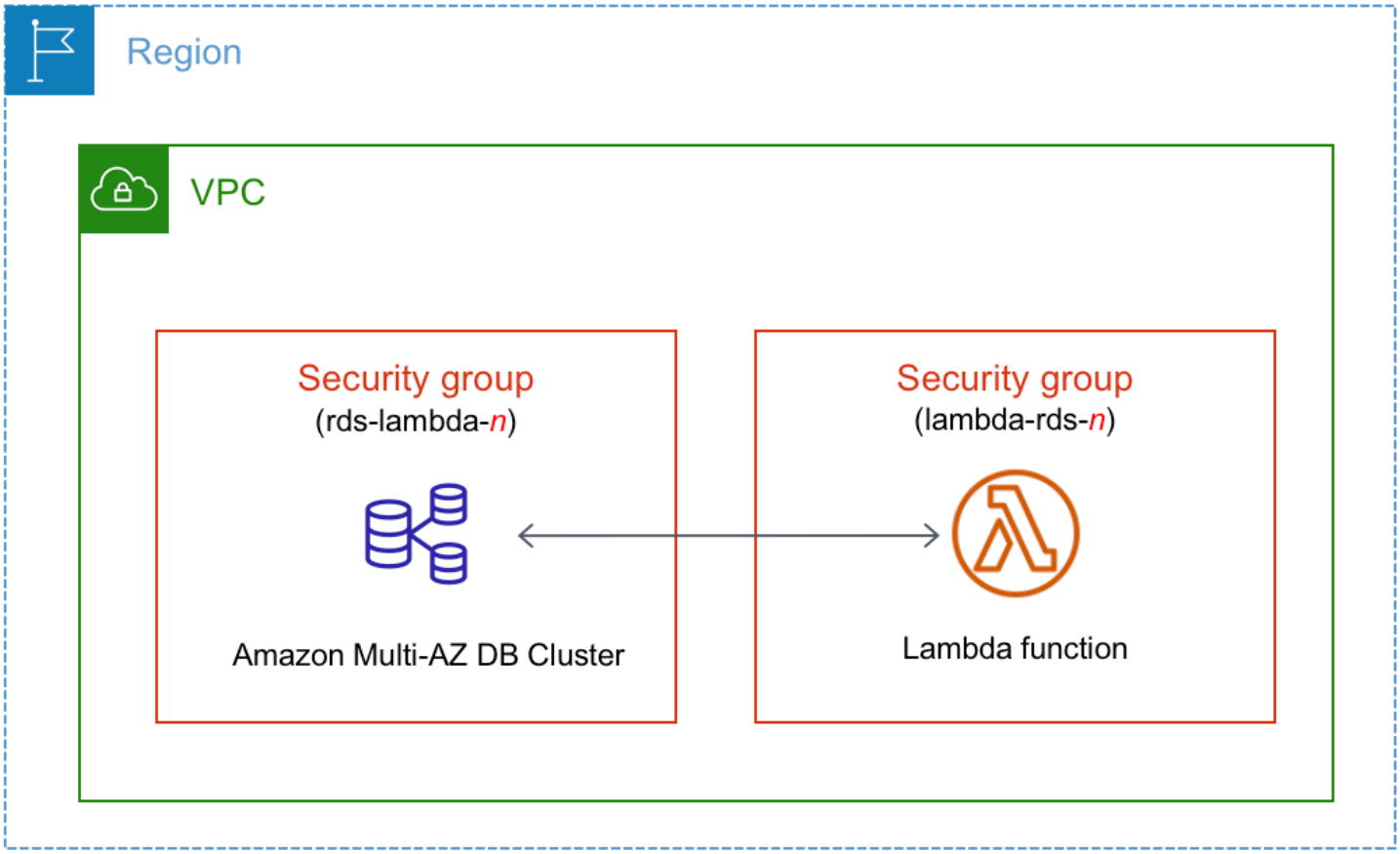
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases (Bases de données), puis le nom de la base de données RDS.
3. Dans l'onglet Connectivity & security (Connectivité et sécurité), affichez les ressources de calcul dans Connected compute resources (Ressources de calcul connectées).



Connexion automatique d'une fonction Lambda et d'un cluster de bases de données multi-AZ

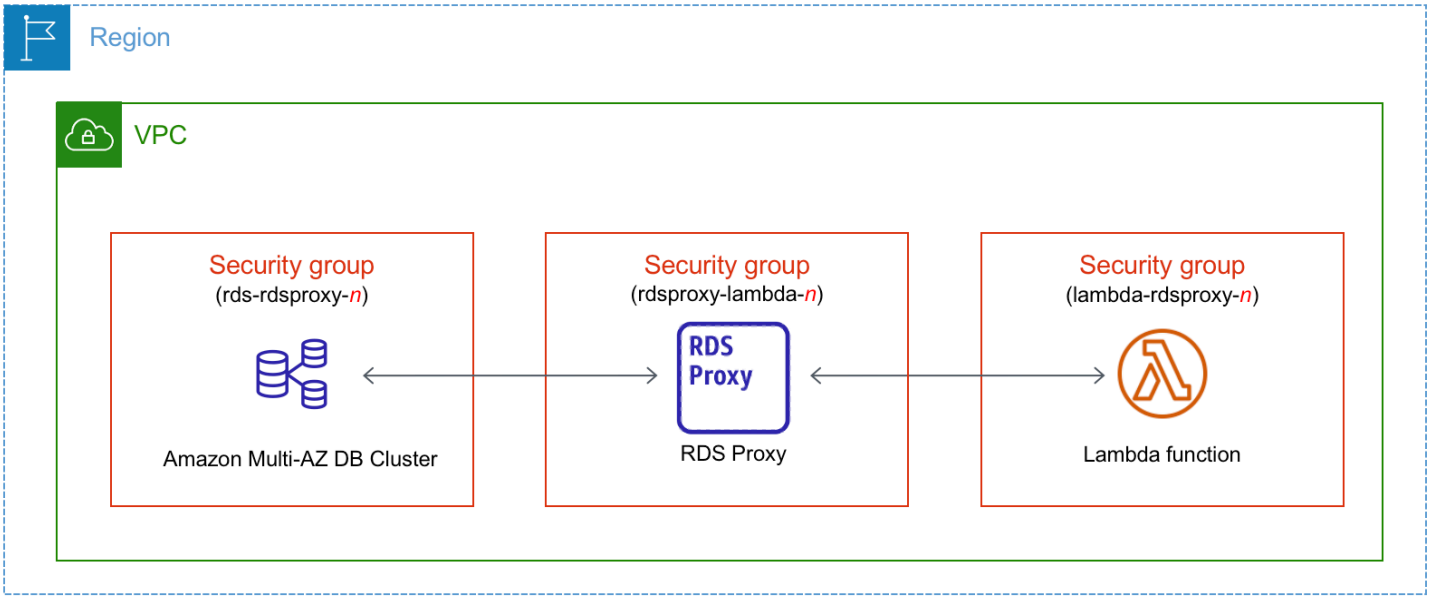
Vous pouvez utiliser la console RDS pour simplifier la configuration d'une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ. Vous pouvez utiliser la console RDS pour simplifier la configuration d'une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ. Souvent, votre cluster de bases de données se trouve dans un sous-réseau privé au sein d'un VPC. La fonction Lambda peut être utilisée par les applications pour accéder à votre cluster de bases de données multi-AZ privé.

L'image suivante montre une connexion directe entre votre cluster de bases de données multi-AZ et votre fonction Lambda.



Vous pouvez configurer la connexion entre votre fonction Lambda et votre base de données via un proxy RDS pour améliorer les performances et la résilience de votre base de données. Souvent, les fonctions Lambda établissent des connexions de base de données courtes et fréquentes qui bénéficient du regroupement de connexions offert par le proxy RDS. Vous pouvez profiter de toute authentification IAM dont vous disposez déjà pour les fonctions Lambda, plutôt que de gérer les informations d'identification de base de données dans votre code d'application Lambda. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .

Vous pouvez utiliser la console pour créer automatiquement un proxy pour votre connexion. Vous pouvez également sélectionner des proxys existants. La console met à jour le groupe de sécurité du proxy pour autoriser les connexions depuis votre base de données et la fonction Lambda. Vous pouvez saisir vos informations d'identification de base de données ou sélectionner le secret Secrets Manager dont vous avez besoin pour accéder à la base de données.



Rubriques

- Vue d'ensemble de la connectivité automatique avec une fonction Lambda
- Connexion automatique d'une fonction Lambda et d'un cluster de bases de données multi-AZ
- Affichage des ressources de calcul connectées

Vue d'ensemble de la connectivité automatique avec une fonction Lambda

Lorsque vous établissez automatiquement une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ, Amazon RDS configure le groupe de sécurité VPC pour votre fonction Lambda et pour votre cluster de bases de données.

Voici les conditions requises pour connecter une fonction Lambda à un cluster de bases de données multi-AZ :

- La fonction Lambda doit exister dans le même VPC que le cluster de bases de données multi-AZ.

  Si aucune fonction Lambda n'existe dans le même VPC, la console fournit un lien pour en créer une.
- L'utilisateur qui configure la connectivité doit avoir les autorisations nécessaires pour effectuer les opérations Amazon RDS, Amazon EC2, Lambda, Secrets Manager et IAM suivantes :
  - Amazon RDS
    - `rds:CreateDBProxies`

  - `rds:DescribeDBInstances`
  - `rds:DescribeDBProxies`
  - `rds:ModifyDBInstance`
  - `rds:ModifyDBProxy`
  - `rds:RegisterProxyTargets`
- Amazon EC2
  - `ec2:AuthorizeSecurityGroupEgress`
  - `ec2:AuthorizeSecurityGroupIngress`
  - `ec2:CreateSecurityGroup`
  - `ec2:DeleteSecurityGroup`
  - `ec2:DescribeSecurityGroups`
  - `ec2:RevokeSecurityGroupEgress`
  - `ec2:RevokeSecurityGroupIngress`
- Lambda
  - `lambda:CreateFunctions`
  - `lambda:ListFunctions`
  - `lambda:UpdateFunctionConfiguration`
- Secrets Manager
  - `sercetsmanager:CreateSecret`
  - `secretsmanager:DescribeSecret`
- IAM
  - `iam:AttachPolicy`
  - `iam:CreateRole`
  - `iam:CreatePolicy`
- AWS KMS
  - `kms:describeKey`

Lorsque vous établissez une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ, Amazon RDS configure le groupe de sécurité VPC pour votre fonction et pour votre cluster de bases de données multi-AZ. Si vous utilisez un proxy RDS, Amazon RDS configure également le groupe de sécurité VPC pour le proxy. Amazon RDS agit conformément à la

configuration actuelle des groupes de sécurité associés au cluster de bases de données multi-AZ, à la fonction Lambda et au proxy, comme décrit dans le tableau suivant.

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Amazon RDS n'entreprend aucune action car les groupes de sécurité de toutes les ressources suivent le modèle de dénomination correct et disposent des règles entrantes et sortantes appropriées.	Un ou plusieurs groupes de sécurité sont associés au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda- n` (où `rds-lambda- n` est un nombre) ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy comme source.	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds- n` ou `lambda-rdsproxy- n` (où *n* est un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité possède une seule règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy comme destination.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda- n` (où *n* est un nombre). Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité possède des règles entrantes et sortantes avec les groupes de sécurité VPC de la fonction Lambda et du cluster de bases de données multi-AZ.
		L'une des conditions suivantes s'applique :	RDS action: create new security groups

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. • Un ou plusieurs groupes de sécurité sont associés au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. Toutefois,	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. • Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ.	• Aucun groupe de sécurité n'est associé au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. • Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond à `rdsproxy-lambda-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ ou la fonction Lambda. Amazon RDS ne peut pas utiliser un groupe de sécurité dépourvu de règles entrantes	

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec la fonction Lambda. Amazon RDS ne peut pas utiliser comme source un groupe de sécurité dépourvu de toute règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié. Des exemples de modifications incluent l'ajout d'une règle ou la modification du port d'une règle existante.	Amazon RDS ne peut pas utiliser comme source un groupe de sécurité dépourvu de toute règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy comme source.	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ. Amazon RDS ne peut pas utiliser comme destination un groupe de sécurité dépourvu de toute règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ ou la fonction Lambda. Amazon RDS ne peut pas utiliser un groupe de sécurité dépourvu de règles entrantes et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	RDS action: create new security groups

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE` . Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité comprend une seule règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy comme source.	Il existe un groupe de sécurité Lambda valide pour la connexion, mais il n'est pas associé à la fonction Lambda. Le nom de ce groupe de sécurité correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Il n'a pas été modifié. Il possède une seule règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy comme destination.	Il existe un groupe de sécurité de proxy valide pour la connexion, mais il n'est pas associé au proxy. Le nom de ce groupe de sécurité correspond au modèle `rdsproxy-lambda-` *n*. Il n'a pas été modifié. Il possède des règles entrantes et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda.	RDS action: associate Lambda security group

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`. • Un ou plusieurs groupes de sécurité sont associés au cluster de bases de données multi-AZ avec un nom qui correspond au modèle `rds-lambda-` *n* ou si l'élément `TargetHealth` d'un proxy associé a pour valeur `AVAILABLE`	Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité possède une seule règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy comme destination.	Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. Un groupe de sécurité qui correspond au modèle n'a pas été modifié. Ce groupe de sécurité possède des règles entrantes et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda.	RDS action: create new security groups

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec la fonction Lambda ou le proxy. Amazon RDS ne peut pas utiliser comme source un groupe de sécurité dépourvu de toute règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.			

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
Un ou plusieurs groupes de sécurité sont associés au cluster de base de données multi-AZ avec un nom qui correspond au modèle `rds-rdsproxy-` *n* (où *n* est un nombre).	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. • Un ou plusieurs groupes de sécurité sont associés à la fonction Lambda avec un nom qui correspond au modèle `lambda-rds-` *n* ou `lambda-rdsproxy-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ.	L'une des conditions suivantes s'applique : • Aucun groupe de sécurité n'est associé au proxy avec un nom qui correspond au modèle `rdsproxy-lambda-` *n*. • Un ou plusieurs groupes de sécurité sont associés au proxy avec un nom qui correspond à `rdsproxy-lambda-` *n*. Toutefois, Amazon RDS ne peut utiliser aucun de ces groupes de sécurité pour la connexion avec le cluster de bases de données multi-AZ ou la fonction Lambda. Amazon RDS ne peut pas utiliser un groupe de sécurité dépourvu	RDS action: create new security groups

Configuration du groupe de sécurité RDS actuel	Configuration actuelle du groupe de sécurité Lambda	Configuration actuelle du groupe de sécurité du proxy	Action RDS
	Amazon RDS ne peut pas utiliser comme destination un groupe de sécurité dépourvu de toute règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou du proxy. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	de règles entrantes et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda. Amazon RDS ne peut pas non plus utiliser un groupe de sécurité qui a été modifié.	

Action RDS : créer de nouveaux groupes de sécurité

Amazon RDS entreprend les actions suivantes :

- Crée un nouveau groupe de sécurité qui correspond au modèle `rds-lambda-`*n*. Ce groupe de sécurité possède une règle entrante avec le groupe de sécurité VPC de la fonction Lambda ou du proxy comme source. Ce groupe de sécurité est associé au cluster de bases de données multi-AZ et permet à la fonction ou au proxy d'accéder au cluster de bases de données multi-AZ.
- Crée un nouveau groupe de sécurité qui correspond au modèle `lambda-rds-`*n*. Ce groupe de sécurité comprend une règle sortante avec le groupe de sécurité VPC du cluster de bases de données multi-AZ ou le proxy comme destination. Ce groupe de sécurité est associé à la fonction Lambda et permet à cette dernière d'envoyer du trafic vers le cluster de bases de données multi-AZ ou d'envoyer du trafic via un proxy.
- Crée un nouveau groupe de sécurité qui correspond au modèle `rdsproxy-lambda-`*n*. Ce groupe de sécurité possède des règles entrantes et sortantes avec le groupe de sécurité VPC du cluster de bases de données multi-AZ et de la fonction Lambda.

Action RDS : associer un groupe de sécurité Lambda

Amazon RDS associe le groupe de sécurité Lambda valide et existant à la fonction Lambda. Ce groupe de sécurité permet à la fonction d'envoyer du trafic vers le cluster de bases de données multi-AZ ou d'envoyer du trafic via un proxy.

Connexion automatique d'une fonction Lambda et d'un cluster de bases de données multi-AZ

Vous pouvez utiliser la console Amazon RDS pour connecter automatiquement une fonction Lambda à votre cluster de bases de données multi-AZ. Cela simplifie le processus de configuration d'une connexion entre ces ressources.

Vous pouvez également utiliser un proxy RDS pour inclure un proxy dans votre connexion. Les fonctions Lambda établissent des connexions de base de données courtes et fréquentes qui bénéficient du regroupement de connexions offert par le proxy RDS. Vous pouvez également utiliser toute authentification IAM que vous avez déjà configurée pour votre fonction Lambda, plutôt que de gérer les informations d'identification de base de données dans votre code d'application Lambda.

Vous pouvez connecter un cluster de bases de données multi-AZ existant aux fonctions Lambda nouvelles et existantes à l'aide de la page Configurer une connexion Lambda. Le processus de configuration configure automatiquement les groupes de sécurité requis pour vous.

Avant de configurer une connexion entre une fonction Lambda et un cluster de bases de données multi-AZ, assurez-vous que :

- Votre fonction Lambda et votre cluster de bases de données multi-AZ se trouvent dans le même VPC.
- Vous disposez des autorisations appropriées pour votre compte d'utilisateur. Pour plus d'informations sur les exigences, consultez Vue d'ensemble de la connectivité automatique avec une fonction Lambda.

Si vous modifiez les groupes de sécurité requis après avoir configuré la connectivité, ces modifications peuvent affecter la connexion entre la fonction Lambda et le cluster de bases de données multi-AZ.

> **Note**
>
> Vous pouvez configurer automatiquement une connexion entre un cluster de bases de données multi-AZ et une fonction Lambda uniquement dans la AWS Management Console.

Pour connecter une fonction Lambda, toutes les instances figurant dans le cluster de bases de données multi-AZ doivent être dans l'état Disponible.

Pour connecter automatiquement une fonction Lambda et un cluster de bases de données multi-AZ

<result>

Une fois que vous avez confirmé la configuration, Amazon RDS commence le processus de connexion de votre fonction Lambda, de votre proxy RDS (si vous avez utilisé un proxy) et de votre cluster de bases de données multi-AZ. La console affiche la boîte de dialogue Détails de connexion, qui répertorie les modifications de groupe de sécurité qui permettent les connexions entre vos ressources.
</result>

1. Connectez-vous à l'AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis le cluster de bases de données multi-AZ que vous souhaitez connecter à une fonction Lambda.
3. Pour Actions, choisissez Configurer la connexion Lambda.
4. Sur la page Configurer la connexion Lambda, sous Sélectionner une fonction Lambda, effectuez l'une des opérations suivantes :
   - Si vous avez déjà une fonction Lambda dans le même VPC que votre cluster de bases de données multi-AZ, choisissez Choisir une fonction existante, puis choisissez la fonction.
   - Si vous ne disposez pas d'une fonction Lambda dans le même VPC, choisissez Créer une nouvelle fonction, puis saisissez le Nom de la fonction. L'environnement d'exécution par défaut est défini sur Nodejs.18. Vous pouvez modifier les paramètres de votre nouvelle fonction Lambda dans la console Lambda après avoir terminé la configuration de la connexion.
5. (Facultatif) Sous Proxy RDS, sélectionnez Se connecter via un proxy RDS, puis effectuez l'une des opérations suivantes :
   - Si vous souhaitez utiliser un proxy existant, choisissez Choisir un proxy existant, puis choisissez le proxy.
   - Si vous n'avez pas de proxy et que vous souhaitez qu'Amazon RDS en crée un automatiquement pour vous, choisissez Créer un nouveau proxy. Ensuite, pour Informations d'identification de la base de données, effectuez l'une des opérations suivantes :

a. Choisissez Nom d'utilisateur et mot de passe de base de données, puis saisissez le Nom d'utilisateur et le Mot de passe de votre cluster de bases de données multi-AZ.

b. Choisissez Secret Secrets Manager. Ensuite, pour Sélectionner un secret, choisissez un secret AWS Secrets Manager. Si vous n'avez pas de secret Secrets Manager, choisissez Créer un nouveau secret Secrets Manager pour créer un nouveau secret. Après avoir créé le secret, pour Sélectionner un secret, choisissez le nouveau secret.

Après avoir créé le nouveau proxy, choisissez Choisir un proxy existant, puis choisissez le proxy. Notez qu'il peut s'écouler un certain temps avant que votre proxy soit disponible pour la connexion.

6. (Facultatif) Développez Récapitulatif de la connexion et vérifiez les mises à jour en surbrillance pour vos ressources.
7. Choisissez Set up (Configurer).

Affichage des ressources de calcul connectées

Vous pouvez utiliser la AWS Management Console pour visualiser les ressources de calcul qui sont connectées à votre cluster de bases de données multi-AZ. Les ressources affichées incluent les connexions de ressources de calcul qu'Amazon RDS a configurées automatiquement.

Les ressources de calcul répertoriées n'incluent pas celles qui sont connectées manuellement au cluster de bases de données multi-AZ. Par exemple, vous pouvez autoriser une ressource de calcul à accéder manuellement à votre cluster de bases de données multi-AZ en ajoutant une règle à votre groupe de sécurité VPC associé au cluster.

Pour que la console répertorie une fonction Lambda, les conditions suivantes doivent s'appliquer :

- Le nom du groupe de sécurité associé à la ressource de calcul correspond au modèle `lambda-rds-`*n* ou `lambda-rdsproxy-`*n* (où *n* est un nombre).
- Le groupe de sécurité associé à la ressource de calcul possède une règle sortante avec la plage de ports définie sur le port du cluster de bases de données multi-AZ ou un proxy associé. La destination de la règle sortante doit être définie sur un groupe de sécurité associé au cluster de bases de données multi-AZ ou un proxy associé.
- Le nom du groupe de sécurité attaché au proxy associé à votre base de données correspond au modèle `rds-rdsproxy-`*n* (où *n* est un nombre).

- Le groupe de sécurité associé à la fonction possède une règle sortante avec le port défini sur le port utilisé par le cluster de bases de données multi-AZ ou le proxy associé. La destination doit être définie sur un groupe de sécurité associé au cluster de bases de données multi-AZ ou au proxy associé.

Pour afficher les ressources de calcul automatiquement connectées à un cluster de bases de données multi-AZ

1. Connectez-vous à l'AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis choisissez le cluster de bases de données multi-AZ.
3. Dans l'onglet Connectivité et sécurité, examinez les ressources de calcul sous Ressources de calcul connectées.

Modification d'un cluster de base de données multi-AZ pour Amazon RDS

Un cluster de base de données multi-AZ compte une instance de base de données d'écriture et deux instances de base de données de lecture dans trois zones de disponibilité distinctes. Les clusters de base de données multi-AZ offrent une haute disponibilité, une capacité accrue pour les charges de travail en lecture et une moindre latence par rapport aux déploiements multi-AZ. Pour de plus amples informations sur les clusters de base de données multi-AZ, consultez Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.

Vous pouvez modifier un cluster de base de données multi-AZ pour en changer les paramètres. Vous pouvez également effectuer des opérations sur un cluster de base de données multi-AZ, notamment créer un instantané.

> **Important**
>
> Vous ne pouvez pas modifier les instances de base de données au sein d'un cluster de base de données multi-AZ. Toutes les modifications doivent être effectuées au niveau du cluster de base de données. La seule opération que vous pouvez effectuer sur une instance de base de données au sein d'un cluster de base de données multi-AZ est de la redémarrer.

Vous pouvez modifier un cluster de base de données multi-AZ à l'aide de l' AWS Management Console API, de AWS CLI, ou de l'API RDS.

console

Pour modifier un cluster de base de données multi-AZ

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données), puis le cluster de base de données multi-AZ que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier). La page Modify DB cluster (Modifier le cluster DB) s'affiche.
4. Modifiez les paramètres de votre choix. Pour plus d'informations sur chaque paramètre, consultez Paramètres de modification des clusters de base de données multi-AZ.
5. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez le résumé des modifications.

6. (Facultatif) Choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. La sélection de cette option peut entraîner des temps d'arrêt dans certains cas. Pour de plus amples informations, veuillez consulter Application immédiate des modifications.
7. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modify DB cluster (Modifier le cluster de base de données) pour enregistrer vos modifications.

   Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

AWS CLI

Pour modifier un cluster de base de données multi-AZ à l'aide de AWS CLI, appelez la modify-db-clustercommande. Spécifiez l'identifiant du cluster de base de données et les valeurs des options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter Paramètres de modification des clusters de base de données multi-AZ.

Example

Le code suivant modifie `my-multi-az-dbcluster` en définissant la période de rétention des sauvegardes sur 1 semaine (7 jours). Ce code active la protection contre la suppression en utilisant `--deletion-protection`. Pour désactiver la protection contre la suppression, utilisez `--no-deletion-protection`. Les modifications sont appliquées dans la prochaine fenêtre de maintenance à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`. Pour de plus amples informations, veuillez consulter Application immédiate des modifications.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-cluster \
    --db-cluster-identifier my-multi-az-dbcluster \
    --backup-retention-period 7 \
    --deletion-protection \
    --no-apply-immediately
```

Dans Windows:

```
aws rds modify-db-cluster ^
    --db-cluster-identifier my-multi-az-dbcluster ^
    --backup-retention-period 7 ^
    --deletion-protection ^
```

```
    --no-apply-immediately
```

API RDS

Pour modifier un cluster de base de données multi-AZ à l'aide de l'API Amazon RDS, appelez l'opération DBClusterModify. Spécifiez l'identifiant du cluster de base de données et les paramètres que vous souhaitez modifier. Pour plus d'informations sur chaque paramètre, consultez Paramètres de modification des clusters de base de données multi-AZ.

Application immédiate des modifications

Quand vous modifiez un cluster de base de données multi-AZ, vous pouvez appliquer immédiatement les modifications. Pour appliquer immédiatement les modifications, choisissez l'option Appliquer immédiatement dans l' AWS Management Console. Vous pouvez également utiliser l'`--apply-immediately`option lorsque vous appelez le AWS CLI ou définissez le `ApplyImmediately` paramètre sur `true` lorsque vous utilisez l'API Amazon RDS.

Si vous ne choisissez pas d'appliquer les modifications immédiatement, les modifications sont placées dans la file d'attente des modifications en attente. Au cours de la fenêtre de maintenance suivante, les modifications en attente sont appliquées. Si vous choisissez d'appliquer les modifications immédiatement, vos nouvelles modifications et les modifications placées dans la file d'attente des modifications en attente sont appliquées.

> **Important**
>
> Si des modifications en attente exigent que le cluster de base de données soit temporairement indisponible (temps d'arrêt), le choix de l'option Apply immediately (Appliquer immédiatement) peut entraîner une interruption inattendue.
> Si vous choisissez d'appliquer une modification immédiatement, les modifications en attente sont également appliquées immédiatement, au lieu d'attendre la fenêtre de maintenance suivante.
> Si vous ne souhaitez pas qu'une modification en attente soit appliquée lors de la fenêtre de maintenance suivante, vous pouvez modifier l'instance de base de données de façon à inverser la modification. Vous pouvez le faire en utilisant l'option AWS CLI et en spécifiant l'`--apply-immediately`option.

Les modifications de certains paramètres de base de données sont appliquées immédiatement, même si vous choisissez de reporter vos modifications. Pour savoir comment les différents

paramètres de base de données interagissent avec le paramètre Appliquer immédiatement, veuillez consulter Paramètres de modification des clusters de base de données multi-AZ.

Paramètres de modification des clusters de base de données multi-AZ

Pour obtenir des détails sur les paramètres que vous pouvez utiliser pour modifier un cluster de bases de données multi-AZ, consultez le tableau suivant. Pour plus d'informations sur les AWS CLI options, consultez modify-db-cluster. Pour plus d'informations sur les paramètres de l'API RDS, consultez Modifier DBCluster.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Stockage alloué	Capacité de stockage à allouer pour chaque instance de base de données du cluster de base de données (en gibioctets). Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.	Option de l'interface CLI :
		`--allocated-storage`
		Paramètre de l'API RDS :
		`Allocated Storage`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Mise à niveau automatiq ue de versions mineures	Activez la mise à niveau automatiq ue des versions mineures pour permettre au cluster de base	Option de l'interface CLI : `--auto-minor-version-upgrade`	La modification a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiatement.	Un temps d'arrêt a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

	de données de recevoir automatiquement les mises à niveau de la version mineure préférée du moteur de base de données lorsqu'elles sont disponibles. Amazon RDS effectue les mises à niveau automatiques des versions mineures dans la fenêtre de maintenance.	`--no-auto-minor-version-upgrade`
		
		Paramètre de l'API RDS :
		
		`AutoMinorVersionUpgrade`		

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Période de rétention des sauvegar es	Le nombre de jours pendant lesquels les sauvegardes automatiques de votre cluster de base de données sont conservées. Cette valeur doit être supérieure à zéro.
	
	Pour de plus amples informati ons, veuillez consulter Présentation des sauvegardes.
		Option de l'interface CLI :
		
		`--backup-retention-period`
		
		Paramètre de l'API RDS :
		
		`BackupRetentionPeriod`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Fenêtre de sauvegar e	Période durant laquelle Amazon RDS effectue automatiquement une sauvegarde de votre cluster de base de données. Si vous n'avez pas besoin que votre base de données soit sauvegardée à un moment précis, utilisez la valeur par défaut No preferenc e (Aucune préférence).
	
	Pour plus d'informations, consultez Présentation des sauvegardes.
		Option de l'interface CLI :
		
		`--preferred-backup-window`
		
		Paramètre de l'API RDS :
		
		`PreferredBackupWindow`
			La modification est appliquée de manière asynchron e, dès que possible.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Autorité de certifica tion	L'autorité de certification (CA) pour le certificat de serveur utilisé par le cluster de base de données.
	
	Pour de plus amples informati ons, veuillez consulter .
		Option de l'interface CLI :
		
		`--ca-cert ificate-i dentifier`
		
		Paramètre de l'API RDS :
		
		`CACertifi cateIdent ifier`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Un temps d'arrêt survient uniquemen t si le moteur de base de données ne prend pas en charge la rotation sans redémarrage. Vous pouvez utiliser la describe-db-engine -versions AWS CLI commande pour déterminer si le moteur de base de données prend en charge la rotation sans redémarrage.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Copier les balises aux instantan és	Cette option permet de copier toutes les identific ations de cluster de base de données dans un instantan é de base de données lorsque vous créez un instantané.
	
	Pour de plus amples informati ons, veuillez consulter Marquage d' Amazon RDS.
		Option de l'interface CLI :
		
		`-copy-tags-to-snapshot`
		
		`-no-copy-tags-to-snapshot`
		
		Paramètre de l'API RDS :
		
		`CopyTagsToSnapshot`
			La modification a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
Authentif ication de base de données	Pour les clusters de base de données multi-AZ, seule l'option Password authentication (Authentification par mot de passe) est prise en charge.	Aucun(e) car l'authentification par mot de passe est la valeur par défaut.	Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement. Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Identific ateur du cluster DB	Identifiant du cluster de bases de données. Cette valeur est stockée sous la forme d'une chaîne en minuscules.
	
	Lorsque vous modifiez l'identif iant du cluster de bases de données, le point de terminais on du cluster de bases de données change. Les identificateurs et les points de terminaison des instances de base de données du cluster de bases de données changent également. Le nom du nouveau cluster de bases de données doit être unique. La longueur
		Option de l'interface CLI :
		
		`--new-db-cluster-identifier`
		
		Paramètre de l'API RDS :
		
		`NewDBClusterIdentifier`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	maximale est de 63 caractères. Les noms des instances de base de données du cluster de bases de données sont modifiés pour correspon dre au nouveau nom du cluster de bases de données. Le nom d'une nouvelle instance de base de données ne peut pas être identique à celui d'une instance de base de données existante. Par exemple, si vous remplacez le nom du cluster de bases de données par `maz`, le nom d'une instance de base de données peut être remplacé par `maz-insta nce-1` . Dans			

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	ce cas, aucune instance de base de données existante ne peut être nommée `maz-insta nce-1` . Pour de plus amples informati ons, veuillez consulter Modification du nom d'un cluster de bases de données multi-AZ pour Amazon RDS.			

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Classe d'instanc e de cluster de base de données	Capacité de calcul et de mémoire de chaque instance de base de données contenue dans le cluster de base de données multi-AZ, par exemple `db.r6gd.x large` .
	
	Dans la mesure du possible, choisissez une classe d'instanc e de base de données suffisamment grande pour qu'un ensemble de travail de requête classique puisse tenir dans la mémoire. Lorsque les ensembles de travail sont en mémoire, le système peut éviter d'écrire sur le disque, ce
		Option de l'interface CLI :
		
		`--db-cluster-instance-class`
		
		Paramètre de l'API RDS :
		
		`DBCluster InstanceClass`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Un temps d'arrêt a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	qui améliore les performances. Pour de plus amples informati ons, veuillez consulter the section called “Disponibilité des classes d'instance pour les clusters de bases de données multi-AZ”.			

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Groupe de paramètr s de cluster de bases de données	Groupe de paramètres de cluster de bases de données que vous souhaitez associer au cluster de bases de données.
	
	Pour de plus amples informati ons, veuillez consulter Groupes de paramètres pour les clusters de bases de données multi-AZ.
		Option de l'interface CLI :
		
		`--db-cluster-parameter-grou p-name`
		
		Paramètre de l'API RDS :
		
		`DBCluster Parameter GroupName`
			La modification du groupe de paramètres a lieu immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modificat ion. Lorsque vous modifiez le groupe de paramètre s, les modificat ions apportées à certains paramètre s s'appliquent immédiatement aux instances de base de données du cluster de base de données multi-AZ, sans redémarra ge. Les modificat ions apportées aux autres paramètre s s'appliquent uniquement après le redémarrage des instances de base de données.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Version du moteur de base de données	Version du moteur de base de données que vous souhaitez utiliser.	Option de l'interface CLI :
		
		`--engine-version`
		
		Paramètre de l'API RDS :
		
		`EngineVersion`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Un temps d'arrêt a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Deletion protectio n (Protecti on contre la suppress on)	Sélectionnez Enable deletion protection (Activer la protection de la suppression) pour empêcher la suppression de votre cluster de bases de données.
	
	Pour de plus amples informati ons, veuillez consulter Suppressi on d'une instance DB.
		Option de l'interface CLI :
		
		`--deletion-protection`
		
		`--no-deletion-protection`
		
		Paramètre de l'API RDS :
		
		`DeletionP rotection`
			La modification a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Fenêtre de maintena ce	Fenêtre de 30 minutes durant laquelle les modifications en attente sont appliquées à votre cluster de base de données. Si la période n'a pas d'importa nce, choisisse z No preferenc e (Aucune préférence).
	
	Pour plus d'informations, consultez Fenêtre de maintenance Amazon RDS.
		Option de l'interface CLI :
		
		`--preferred-maintenance-window`
		
		Paramètre de l'API RDS :
		
		`PreferredMaintenanceWindow`
			La modification a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiatement.	Si une ou plusieurs actions en attente entraînent un temps d'arrêt et que la fenêtre de maintenance est modifiée pour inclure l'heure actuelle, les actions en attente sont appliquées immédiatement et un temps d'arrêt se produit.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Gérez les informati ons d'identif ication principal es dans AWS Secrets Manager	Sélection nez Gérer les informations d'identification principales dans AWS Secrets Manager pour gérer le mot de passe d'utilisateur principal dans un secret, dans Secrets Manager.
	
	Vous pouvez éventuellement choisir une clé KMS à utiliser pour protéger le secret. Choisisse z l'une des clés KMS de votre compte ou entrez la clé d'un autre compte.
	
	Si RDS gère déjà le mot de passe de l'utilisateur principal pour le cluster de bases de données, vous pouvez effectuer la
		Option de l'interface CLI :
		
		`--manage-master-user-password | --no-manage-master-user-password`
		
		`--master-user-secret-kms-key-id`
		
		`--rotate-master-user-password | --no-rotate-master-user-password`
		
		Paramètre de l'API RDS :
		
		`ManageMasterUserPassword`
		
		`MasterUserSecretKmsKeyId`
		
		`RotateMasterUserPassword`
			Si vous activez ou désactivez la gestion automatiq ue des mots de passe d'utilisa teur principal , la modificat ion se produit immédiatement. Cette modification ignore le paramètre d'application immédiate.
			
			Si vous effectuez la rotation du mot de passe de l'utilisa teur principal, vous devez spécifier que la modificat ion doit s'appliquer immédiatement.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	rotation du mot de passe de l'utilisa teur principal en choisissant Rotate secret immediate ly (Effectuer immédiatement une rotation du secret). Pour de plus amples informati ons, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.			

New master passworc	Mot de passe de votre compte utilisateur principal.	Option de l'interface CLI :
		
		`--master-user-password`
		
		Paramètre de l'API RDS :
		
		`MasterUse rPassword`
			La modification est appliquée de manière asynchron e, dès que possible. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

IOPS provision nés	Quantité d'IOPS provisionnés (opérations d'entrée/sortie par seconde) à allouer initialem ent pour le cluster de base de données.	Option de l'interface CLI :
		`--iops`
		Paramètre de l'API RDS :
		`Iops`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
Accès public	Publicly accessibl e (Accessible publiquement) dote le cluster de base de données d'une adresse IP publique, ce qui signifie qu'il est accessible en dehors du cloud privé virtuel (VPC). Pour être accessible publiquement, le cluster de base de données doit aussi se trouver dans un sous-réseau public du VPC. Non accessibl e publiquement pour rendre le cluster de base de données accessible uniquement à partir du VPC. Pour de plus amples informati ons, veuillez	Non disponible lors de la modification d'un cluster de base de données.	La modification a lieu immédiate ment. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	consulter Masquer un(e) instance de base de données dans un VPC depuis Internet. Pour pouvoir se connecter à un cluster de base de données en dehors de son VPC, il doit être accessible publiquement. De plus, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité du cluster de base de données, et d'autres condition s doivent être remplies. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de			

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt
	base de données Amazon RDS. Si votre cluster de base de données n'est pas accessible au public, vous pouvez utiliser une connexion AWS Site-to-Site VPN ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour de plus amples informations, veuillez consulter Confidentialité du trafic inter-réseau.			

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Type de stockage	Type de stockage pour votre cluster de base de données.
	
	Seul le stockage SSD à usage général (gp3), IOPS provision né (io1) et SSD IOPS provisionné (io2) est pris en charge.
	
	Pour de plus amples informati ons, veuillez consulter Types de stockage Amazon RDS.
		Option de l'interface CLI :
		
		`--storage-type`
		
		Paramètre de l'API RDS :
		
		`StorageType`
			Si vous choisisse z d'appliquer la modification immédiatement, elle se produit immédiatement.
			
			Si vous ne choisissez pas d'appliquer la modification immédiatement, elle se produit lors de la fenêtre de maintenance suivante.
				Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètr de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS	Lorsque la modificat ion a lieu	Remarques sur les temps d'arrêt

Groupe de sécurité VPC	Groupes de sécurité à associer au cluster de base de données.
	
	Pour de plus amples informati ons, veuillez consulter Présentation des groupes de sécurité VPC.
		Option de l'interface CLI :
		
		`--vpc-sec urity-group-ids`
		
		Paramètre de l'API RDS :
		
		`VpcSecuri tyGroupIds`
			La modification est appliquée de manière asynchron e, dès que possible. Ce paramètre ignore le paramètre Appliquer immédiatement.	Aucun temps d'arrêt n'a lieu pendant cette modification.

Paramètres non applicables pendant la modification de clusters de base de données multi-AZ

Les paramètres suivants de la AWS CLI commande modify-db-clusteret de l'opération Modify de l'API RDS DBCluster ne s'appliquent pas aux clusters de bases de données multi-AZ.

Vous ne pouvez pas non plus modifier ces paramètres pour les clusters de base de données multi-AZ dans la console.

AWS CLI réglage	Paramètre de l'API RDS
`--backtrack-window`	`BacktrackWindow`
`--cloudwatch-logs-export-co nfiguration`	`CloudwatchLogsExportConfigu ration`
`--copy-tags-to-snapshot | --no-copy-tags-to-snapshot`	`CopyTagsToSnapshot`

AWS CLI réglage	Paramètre de l'API RDS
`--db-instance-parameter-group-name`	`DBInstanceParameterGroupName`
`--domain`	`Domain`
`--domain-iam-role-name`	`DomainIAMRoleName`
`--enable-global-write-forwarding | --no-enable-global-write-forwarding`	`EnableGlobalWriteForwarding`
`--enable-http-endpoint | --no-enable-http-endpoint`	`EnableHttpEndpoint`
`--option-group-name`	`OptionGroupName`
`--port`	`Port`
`--scaling-configuration`	`ScalingConfiguration`
`--storage-type`	`StorageType`

Mise à niveau de la version du moteur d'un cluster de base de données multi-AZ pour Amazon RDS

Amazon RDS fournit des versions plus récentes de chaque moteur de base de données pris en charge afin que vous puissiez maintenir votre cluster de bases de données multi-AZ à jour. Cette rubrique explique le processus de mise à niveau d'un cluster de base de données multi-AZ vers des versions plus récentes.

La mise à niveau d'un cluster de base de données multi-AZ implique la sélection d'une nouvelle version de moteur compatible et la planification des temps d'arrêt potentiels. Le processus garantit une interruption minimale en utilisant les fonctionnalités de basculement de l'architecture multi-AZ. Au cours de la mise à niveau, l'instance principale est d'abord mise à jour, suivie d'un basculement vers l'instance de secours afin de maintenir la disponibilité. Les meilleures pratiques incluent l'exécution de mises à niveau pendant les périodes de faible trafic, les tests dans des environnements hors production et la vérification de la compatibilité des applications avec la nouvelle version.

Rubriques

- Mises à niveau de version mineure.
- Mises à niveau de version majeure.
- Mise à niveau d'un cluster de base de données multi-AZ
- Mise à niveau des répliques de lecture de clusters de bases de données multi-AZ

Mises à niveau de version mineure.

Une mise à niveau de version mineure contient uniquement des modifications rétrocompatibles avec les applications existantes. Lorsque vous lancez une mise à niveau d'une version mineure, Amazon RDS met d'abord à niveau les instances de base de données du lecteur une par une. Ensuite, l'une des instances de base de données du lecteur devient la nouvelle instance de base de données du rédacteur. Amazon RDS met ensuite à niveau l'ancienne instance d'écriture (qui est désormais une instance de lecteur).

Les temps d'arrêt pendant la mise à niveau sont limités au temps nécessaire à l'une des instances de base de données du lecteur pour devenir la nouvelle instance de base de données du rédacteur. Ce temps d'arrêt agit comme un basculement automatique. Pour de plus amples informations, veuillez consulter the section called “Défaillance d'un cluster de base de données multi-AZ”. Notez que le délai de réplication de votre cluster de base de données multi-AZ peut affecter le temps d'arrêt. Pour

de plus amples informations, veuillez consulter the section called “Retard de réplica et clusters de base de données multi-AZ”.

Pour les répliques de lecture de clusters de bases de données multi-AZ RDS pour PostgreSQL, Amazon RDS met à niveau les instances membres du cluster une par une. Les rôles du cluster de lecture et d'écriture ne changent pas pendant la mise à niveau. Par conséquent, votre cluster de base de données peut être indisponible pendant qu'Amazon RDS met à niveau l'instance du rédacteur de cluster.

> **Note**
>
> Le temps d'arrêt pour une mise à niveau d'une version mineure d'un cluster de base de données multi-AZ est généralement de 35 secondes. Lorsqu'il est utilisé avec le proxy RDS, vous pouvez réduire davantage les temps d'arrêt à une seconde ou moins. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS. Vous pouvez également utiliser un proxy de base de données open source tel que ProxySQL ou le pilote AWS Advanced JDBC Wrapper. PgBouncer

Mises à niveau de version majeure.

La mise à niveau d'une version majeure peut introduire des modifications non compatibles avec les applications existantes.

Lorsque vous lancez une mise à niveau de version majeure d'un cluster de bases de données multi-AZ RDS pour PostgreSQL, Amazon RDS met à niveau simultanément les instances de lecteur et d'écriture. Par conséquent, votre cluster de base de données risque de ne pas être disponible tant que la mise à niveau n'est pas terminée.

Lorsque vous lancez une mise à niveau de version majeure d'un cluster de base de données multi-AZ RDS pour MySQL, Amazon RDS met à niveau les instances membres du cluster une par une, de sorte que la réplication s'effectue d'une version inférieure du moteur vers une version supérieure. Il est important de vous assurer que votre charge de travail est compatible avec les versions du moteur source et cible lors d'une mise à niveau de version majeure, car les versions du moteur peuvent différer en termes de syntaxe et de fonctionnalités.

> **Note**
>
> Comme pour les mises à niveau de versions mineures, le temps d'arrêt pour une mise à niveau de version majeure de RDS pour MySQL est généralement de 35 secondes. Lorsqu'il est utilisé avec le proxy RDS, vous pouvez réduire davantage les temps d'arrêt à une seconde ou moins. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS.

Mise à niveau d'un cluster de base de données multi-AZ

Le processus de mise à niveau de la version du moteur d'un cluster de bases de données multi-AZ est identique au processus de mise à niveau de la version du moteur d'une instance de base de données. Pour obtenir des instructions, consultez the section called “Mise à niveau de la version du moteur”. La seule différence est que lorsque vous utilisez le AWS Command Line Interface (AWS CLI), vous utilisez la modify-db-clustercommande et spécifiez le `--db-cluster-identifier` paramètre (ainsi que le `--allow-major-version-upgrade` paramètre).

Pour plus d'informations sur les mises à niveau des versions majeures et mineures, consultez la documentation suivante relative à votre moteur de base de données :

- the section called “Mises à niveau du moteur de base de données PostgreSQL”
- the section called “Mises à niveau du moteur de base de données MySQL”

Mise à niveau des répliques de lecture de clusters de bases de données multi-AZ

Amazon RDS ne met pas automatiquement à niveau les réplicas en lecture des clusters de bases de données multi-AZ. Pour les mises à niveau de versions mineures, vous devez d'abord mettre à niveau manuellement toutes les répliques en lecture, puis mettre à niveau le cluster. Dans le cas contraire, la mise à niveau est bloquée. Quand vous effectuez une mise à niveau de version majeure d'un cluster, l'état de réplication de tous les réplicas en lecture devient résilié. Vous devez supprimer et recréer les réplicas en lecture une fois la mise à niveau terminée. Pour de plus amples informations, veuillez consulter the section called “Supervision de la réplication en lecture”.

Modification du nom d'un cluster de bases de données multi-AZ pour Amazon RDS

Vous pouvez renommer un cluster de base de données multi-AZ à l' AWS Management Console aide de la AWS CLI `modify-db-cluster` commande ou de l'opération Amazon RDS API`ModifyDBCluster`. Le renommage d'un cluster de bases de données multi-AZ peut avoir des effets importants. Vous trouverez ci-dessous une liste de considérations à prendre en compte avant de renommer un cluster de bases de données multi-AZ.

- Lorsque vous renommez un cluster de bases de données multi-AZ, les points de terminaison du cluster changent pour le cluster de bases de données multi-AZ. Ces points de terminaison changent car ils incluent le nom que vous avez attribué au cluster de bases de données multi-AZ. Vous pouvez rediriger le trafic d'un ancien point de terminaison vers un nouveau. Pour plus d'informations sur les points de terminaison du cluster de bases de données multi-AZ, consultez Connexion à un cluster de bases de données multi-AZ pour Amazon RDS.
- Lorsque vous renommez un cluster de base de données multi-AZ, l'ancien DNS nom qui était utilisé par le cluster de base de données multi-AZ est supprimé, bien qu'il puisse rester en cache pendant quelques minutes. Le nouveau DNS nom du cluster de base de données Multi-AZ renommé prend effet au bout de deux minutes environ. Le cluster de bases de données multi-AZ renommé n'est pas disponible tant que le nouveau nom ne devient pas effectif.
- Vous ne pouvez pas utiliser le nom d'un cluster de bases de données multi-AZ existant lorsque vous renommez un cluster.
- Les métriques et les événements associés au nom d'un cluster de bases de données multi-AZ sont conservés si vous réutilisez un nom de cluster de bases de données.
- Les identifications de cluster de bases de données multi-AZ restent avec le cluster de bases de données multi-AZ, quel que soit le renommage.
- Les instantanés de cluster de bases de données sont conservés pour un cluster de bases de données multi-AZ renommé.

> **Note**
>
> Un cluster de bases de données multi-AZ est un environnement de base de données isolé s'exécutant dans le cloud. Un cluster de bases de données multi-AZ peut héberger plusieurs bases de données. Pour plus d'informations sur le changement de nom d'une base de données, consultez la documentation de votre moteur de base de données.

Renommage pour remplacer un cluster de bases de données multi-AZ existant

Les scénarios les plus courants pour renommer un cluster de base de données multi-AZ incluent la restauration de données à partir d'un instantané de cluster de base de données ou l'exécution d'une point-in-time restauration ()PITR. En renommant le cluster de bases de données multi-AZ, vous pouvez remplacer le cluster de bases de données multi-AZ sans modifier le code d'application qui fait référence au cluster de bases de données multi-AZ. Dans chacun de ces cas, procédez comme suit :

1. Arrêtez tout le trafic en direction du cluster de bases de données multi-AZ. Vous pouvez rediriger le trafic pour l'empêcher d'accéder aux bases de données sur le cluster de bases de données multi-AZ ou choisir une autre manière d'empêcher le trafic d'accéder à vos bases de données sur le cluster de bases de données multi-AZ.
2. Renommez le cluster de bases de données multi-AZ existant.
3. Créez un nouveau cluster de bases de données multi-AZ en effectuant une restauration à partir d'un instantané de cluster de bases de données ou en effectuant une récupération ponctuelle dans le temps. Ensuite, attribuez au nouveau cluster de bases de données multi-AZ le nom du cluster de bases de données multi-AZ précédent.

Si vous supprimez l'ancien cluster de bases de données multi-AZ, vous êtes responsable de la suppression de tout instantané non voulu de cluster de bases de données multi-AZ de l'ancien cluster de bases de données multi-AZ.

Console

Pour renommer un cluster de bases de données multi-AZ

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le cluster de bases de données multi-AZ à renommer.
4. Sélectionnez Modifier.
5. Dans Settings (Paramètres), entrez un nouveau nom pour DB cluster identifier (Identifiant du cluster de bases de données).
6. Choisissez Continuer.

7. Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement. La sélection de cette option peut entraîner une interruption de service dans certains cas. Pour de plus amples informations, veuillez consulter Application immédiate des modifications.
8. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modifier le cluster pour enregistrer vos modifications.

   Sinon, choisissez Back (Précédent) pour éditer vos modifications ou Cancel (Annuler) pour les annuler.

AWS CLI

Pour renommer un cluster de base de données multi-AZ, utilisez la AWS CLI commande. modify-db-cluster Fournissez la valeur `--db-cluster-identifier` actuelle et le paramètre `--new-db-cluster-identifier` avec le nouveau nom du cluster de bases de données multi-AZ.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-cluster \
    --db-cluster-identifier DBClusterIdentifier \
    --new-db-cluster-identifier NewDBClusterIdentifier
```

Dans Windows:

```
aws rds modify-db-cluster ^
    --db-cluster-identifier DBClusterIdentifier ^
    --new-db-cluster-identifier NewDBClusterIdentifier
```

RDS API

Pour renommer un cluster de base de données multi-AZ, appelez RDS API l'opération Amazon ModifyDBCluster avec les paramètres suivants :

- `DBClusterIdentifier` : nom existant du cluster de bases de données.
- `NewDBClusterIdentifier` : nouveau nom du cluster de bases de données.

Redémarrage d'un cluster de base de données multi-AZ et d'instances de base de données de lecture pour Amazon RDS

Il peut parfois être nécessaire de redémarrer un cluster de base de données multi-AZ, généralement à des fins de maintenance. Par exemple, si vous effectuez certaines modifications ou si vous changez le groupe de paramètres du cluster de base de données associé à un cluster de base de données, vous redémarrez le cluster de base de données. Ce faisant, les modifications prennent effet.

Si un cluster de base de données n'utilise pas les dernières modifications apportées au groupe de paramètres de cluster de base de données associé, AWS Management Console affiche le groupe de paramètres de cluster de base de données avec l'état en attente de redémarrage. Le statut de groupe de paramètres pending-reboot n'entraîne pas de redémarrage automatique lors de la fenêtre de maintenance suivante. Pour appliquer les dernières modifications apportées aux paramètres de ce cluster de base de données, vous devez le redémarrer manuellement. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour les clusters de bases de données multi-AZ.

Le redémarrage d'un cluster de base de données entraîne celui du service du moteur de base de données. Le redémarrage d'un cluster de base de données entraîne une interruption momentanée, au cours de laquelle le statut du cluster de base de données est défini sur rebooting (redémarrage en cours).

Vous ne pouvez pas redémarrer votre cluster de base de données s'il n'est pas à l'état Available (Disponible). Votre base de données peut ne pas être disponible pour plusieurs raisons, par exemple une sauvegarde en cours ou une modification demandée précédemment par le client, ou encore une action de créneau de maintenance.

Le temps nécessaire au redémarrage de votre cluster de base de données dépend du processus de récupération en cas de panne, de l'activité de la base de données au moment du redémarrage et du comportement de votre cluster de base de données spécifique. Pour améliorer le délai de redémarrage, nous vous recommandons de réduire l'activité de base de donnée autant que possible pendant le processus de redémarrage. Cela a pour effet de réduire l'activité de restauration pour les transactions en transit.

> **Important**
>
> Les clusters de base de données multi-AZ ne prennent pas en charge le redémarrage avec basculement. Lorsque vous redémarrez l'instance de rédacteur d'un cluster de base

de données Multi-AZ, cela n'affecte pas les instances de base de données de lecteur dans ce cluster de base de données et aucun basculement ne se produit. Lorsque vous redémarrez une instance de base de données de lecture, aucun basculement ne se produit. Pour basculer sur un cluster de base de données multi-AZ, choisissez Failover dans la console, appelez la AWS CLI commande `failover-db-cluster`ou appelez l'APIopération. `FailoverDBCluster`

Console

Pour redémarrer un cluster de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données), puis le cluster de base de données multi-AZ que vous souhaitez redémarrer.
3. Pour Actions, choisissez Redémarrer.

   La page Reboot DB cluster (Redémarrer le cluster de base de données) s'affiche.
4. Choisissez Reboot (Redémarrer) pour redémarrer votre cluster de base de données.

   Ou choisissez Cancel (Annuler).

AWS CLI

Pour redémarrer un cluster de base de données multi-AZ à l'aide de AWS CLI, appelez la reboot-db-clustercommande.

```
aws rds reboot-db-cluster --db-cluster-identifier mymultiazdbcluster
```

RDS API

Pour redémarrer un cluster de base de données multi-AZ à l'aide d'Amazon RDSAPI, appelez l'ebootDBClusteropération R.

Défaillance d'un cluster de base de données multi-AZ pour Amazon RDS

En cas de panne planifiée ou imprévue de votre instance de base de données d'écriture dans un cluster de base de données multi-AZ, Amazon bascule RDS automatiquement vers une instance de base de données de lecteur située dans une autre zone de disponibilité. Cela garantit une haute disponibilité avec un minimum de perturbations. Des basculements peuvent se produire lors de pannes matérielles, de problèmes réseau ou de demandes manuelles. Cette rubrique décrit la détection automatique des défaillances, la séquence des événements lors du basculement et son impact sur les opérations de lecture et d'écriture. Il fournit également les meilleures pratiques pour surveiller et minimiser les temps de basculement.

La durée du basculement dépend de l'activité de base de données et d'autres conditions au moment où l'instance de base de données d'écriture est devenue indisponible. Les durées de basculement sont généralement inférieures à 35 secondes. Le basculement se termine lorsque les deux instances de base de données de lecture ont appliqué les transactions en suspens de l'instance d'écriture défaillante. Lorsque le basculement est terminé, la RDS console peut mettre plus de temps à refléter la nouvelle zone de disponibilité.

Rubriques

- Basculements automatiques
- Basculement manuel d'un cluster de base de données multi-AZ
- Déterminer si un cluster de base de données multi-AZ a basculé
- Configuration JVM TTL des recherches DNS de noms

Basculements automatiques

Amazon RDS gère automatiquement les basculements afin que vous puissiez reprendre les opérations de base de données le plus rapidement possible sans intervention administrative. Pour basculer, l'instance de base de données d'écriture bascule automatiquement sur une instance de base de données de lecture.

Basculement manuel d'un cluster de base de données multi-AZ

Si vous basculez manuellement sur un cluster de base de données multi-AZ, mettez d'RDSabord fin à l'instance de base de données principale. Ensuite, le système de surveillance interne détecte que l'instance de base de données principale est défectueuse et promeut une instance de base

de données de réplique lisible. Les durées de basculement sont généralement inférieures à 35 secondes.

Vous pouvez basculer manuellement sur un cluster de base de données multi-AZ à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Console

Pour faire basculer manuellement un cluster de base de données multi-AZ

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le cluster de base de données multi-AZ que vous voulez faire basculer.
4. Pour Actions, choisissez Failover (Basculement).

   La page Failover DB Cluster s'affiche.
5. Choisissez Failover (Basculement) pour confirmer le basculement manuel.

AWS CLI

Pour basculer manuellement sur un cluster de base de données multi-AZ, utilisez la AWS CLI commande failover-db-cluster.

Example

```
aws rds failover-db-cluster --db-cluster-identifier mymultiazdbcluster
```

RDS API

Pour basculer manuellement sur un cluster de bases de données multi-AZ, appelez Amazon RDS API F ailoverDBCluster et spécifiez le`DBClusterIdentifier`.

Déterminer si un cluster de base de données multi-AZ a basculé

Pour déterminer si votre cluster de base de données multi-AZ a basculé, voici ce que vous pouvez faire :

- Configurez des abonnements aux événements de base de données pour vous informer par e-mail ou SMS qu'un basculement a été initié. Pour plus d'informations sur les événements, consultez Utilisation des notifications d'RDSévénements Amazon.
- Consultez les événements de votre base de données à l'aide de la RDS console ou API des opérations Amazon.
- Consultez l'état actuel de votre cluster de base de données multi-AZ à l'aide de la RDS console Amazon, du AWS CLI, et du RDSAPI.

Pour plus d'informations sur la manière dont vous pouvez réagir en cas de basculement, réduire le temps de restauration et sur d'autres bonnes pratiques pour AmazonRDS, consultezBonnes pratiques relatives à Amazon RDS..

Configuration JVM TTL des recherches DNS de noms

Le mécanisme de basculement modifie automatiquement l'enregistrement Domain Name System (DNS) de l'instance de base de données pour qu'il pointe vers l'instance de base de données du lecteur. Par conséquent, vous devez rétablir toutes les connexions existantes à votre instance de base de données. Dans un environnement de machine virtuelle Java (JVM), en raison du fonctionnement du mécanisme de DNS mise en cache Java, il se peut que vous deviez reconfigurer JVM les paramètres.

Les recherches de DNS noms de JVM caches. Lorsque le JVM convertit un nom d'hôte en adresse IP, il met l'adresse IP en cache pendant une période spécifiée, connue sous le nom de time-to-live(TTL).

Étant donné que les AWS ressources utilisent des entrées de DNS nom qui changent parfois, nous vous recommandons de configurer votre nom JVM avec une TTL valeur ne dépassant pas 60 secondes. Cela garantit que lorsque l'adresse IP d'une ressource change, votre application peut recevoir et utiliser la nouvelle adresse IP de la ressource en demandant le. DNS

Sur certaines configurations Java, la JVM valeur par défaut TTL est définie de manière à ne jamais actualiser les DNS entrées avant JVM le redémarrage. Ainsi, si l'adresse IP d'une AWS ressource change alors que votre application est toujours en cours d'exécution, elle ne peut pas utiliser cette ressource tant que vous ne l'avez pas redémarrée manuellement JVM et que les informations IP mises en cache ne sont pas actualisées. Dans ce cas, il est essentiel de définir le JVM « s » TTL afin qu'il actualise régulièrement ses informations IP mises en cache.

> **Note**
>
> La valeur par défaut TTL peut varier en fonction de la version de votre ordinateur JVM et de l'installation ou non d'un gestionnaire de sécurité. Beaucoup JVMs fournissent une valeur par défaut TTL inférieure à 60 secondes. Si vous utilisez un tel gestionnaire JVM sans utiliser de gestionnaire de sécurité, vous pouvez ignorer le reste de cette rubrique. Pour de plus amples informations sur les responsables de la sécurité dans Oracle, veuillez consulter The Security Manager dans la documentation Oracle.

Pour modifier les « JVM s »TTL, définissez la valeur de la `networkaddress.cache.ttl`propriété. Utilisez l'une des méthodes suivantes selon vos besoins :

- Pour définir la valeur de propriété de manière globale pour toutes les applications utilisant leJVM, définissez `networkaddress.cache.ttl` dans le `$JAVA_HOME/jre/lib/security/java.security` fichier.

```
networkaddress.cache.ttl=60
```

- Pour définir la propriété localement pour votre application uniquement, définissez `networkaddress.cache.ttl` dans le code d'initialisation de votre application avant que les connexions réseau ne soient établies.

```
java.security.Security.setProperty("networkaddress.cache.ttl" , "60");
```

Configuration de la réplication SQL logique Postgre avec des clusters de bases de données multi-AZ pour Amazon RDS

En utilisant la réplication SQL logique Postgre avec votre cluster de base de données multi-AZ, vous pouvez répliquer et synchroniser des tables individuelles plutôt que l'instance de base de données complète. La réplication logique s'appuie sur un modèle publier et s'abonner pour répliquer les modifications depuis la source vers un ou plusieurs destinataires. Cela fonctionne en utilisant les enregistrements de modification du journal d'SQLécriture anticipée de Postgre (). WAL Pour de plus amples informations, veuillez consulter the section called “Exécution d'une réplication logique”.

Lorsque vous créez un nouvel emplacement de réplication logique sur l'instance de base de données d'enregistreur d'un cluster de bases de données multi-AZ, l'emplacement est copié de manière asynchrone sur chaque instance de base de données de lecteur dans le cluster. Les emplacements sur les instances de base de données de lecteur sont synchronisés en continu avec ceux figurant sur l'instance de base de données d'enregistreur.

La réplication logique est prise en charge pour les clusters de bases de données multi-AZ exécutés RDS pour Postgre SQL version 14.8-R2 et supérieure, et 15.3-R2 et version ultérieure.

> **Note**
>
> Outre la fonctionnalité native de réplication SQL logique Postgre, les clusters de base de données multi-AZ exécutés RDS pour Postgre prennent SQL également en charge l'extension. `pglogical`

Pour plus d'informations sur la réplication SQL logique Postgre, consultez la section Réplication logique dans la documentation PostgreSQL.

Rubriques

- Prérequis
- Configuration de la réplication logique
- Limitations et recommandations

Prérequis

Pour configurer la réplication SQL logique Postgre pour les clusters de bases de données multi-AZ, vous devez remplir les conditions préalables suivantes.

- Votre compte d'utilisateur doit être membre du groupe `rds_superuser` et disposer des privilèges `rds_superuser`. Pour de plus amples informations, veuillez consulter the section called “Comprendre les rôles et les autorisations PostgreSQL”.
- Votre cluster de bases de données multi-AZ doit être associé à un groupe de paramètres de cluster de bases de données personnalisé afin que vous puissiez configurer les valeurs de paramètres décrites dans la procédure suivante. Pour de plus amples informations, veuillez consulter the section called “Groupes de paramètres de cluster DB”.

Configuration de la réplication logique

Pour configurer la réplication logique pour un cluster de bases de données multi-AZ, vous devez activer des paramètres spécifiques dans le groupe de paramètres de cluster de bases de données associé, puis créer des emplacements de réplication logique.

> **Note**
>
> À partir de la SQL version 16 de Postgre, vous pouvez utiliser les instances de base de données de lecture du cluster de base de données multi-AZ pour la réplication logique.

Pour configurer la réplication logique pour et RDS pour un cluster de base de données Postgre SQL Multi-AZ

1. Ouvrez le groupe de paramètres de cluster de base de données personnalisé associé à votre cluster RDS de base de données Postgre SQL Multi-AZ.
2. Dans le champ de recherche Paramètres, localisez le paramètre statique `rds.logical_replication` et définissez sa valeur sur `1`. Ce changement de paramètre peut augmenter WAL la génération. Ne l'activez donc que lorsque vous utilisez des emplacements logiques.
3. Dans le cadre de cette modification, configurez les paramètres de cluster de bases de données suivants.

   - `max_wal_senders`

- `max_replication_slots`
- `max_connections`

En fonction de votre utilisation prévue, vous devrez peut-être également modifier les valeurs des paramètres suivants. Toutefois, dans de nombreux cas, les valeurs par défaut sont suffisantes.

- `max_logical_replication_workers`
- `max_sync_workers_per_subscription`

4. Redémarrez le cluster de bases de données multi-AZ pour que les valeurs des paramètres prennent effet. Pour obtenir des instructions, consultez the section called “Redémarrage d'un cluster de base de données multi-AZ”.
5. Créez un emplacement de réplication logique sur l'instance de base de données d'enregistreur du cluster de bases de données multi-AZ, comme cela est expliqué dans the section called “Utilisation des emplacements de réplication logique”. Cela nécessite que vous précisiez un plugin de décodage. Actuellement, RDS car Postgre SQL prend en charge les `pgoutput` plugins `test_decoding`wal2json,, et fournis avec SQL Postgre.

   L'emplacement est copié de manière asynchrone sur chaque instance de base de données de lecteur dans le cluster.
6. Vérifiez l'état de l'emplacement sur toutes les instances de base de données de lecteur du cluster de bases de données multi-AZ. Pour ce faire, inspectez la vue `pg_replication_slots` sur toutes les instances de base de données de lecteur et assurez-vous que l'état `confirmed_flush_lsn` progresse alors que l'application consomme activement des modifications logiques.

   Les commandes suivantes montrent comment inspecter l'état de réplication sur les instances de base de données de lecteur.

```
% psql -h test-postgres-instance-2.abcdefabcdef.us-west-2.rds.amazonaws.com

postgres=> select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;
  slot_name   | slot_type | confirmed_flush_lsn
--------------+-----------+---------------------
 logical_slot | logical   | 32/D0001700
(1 row)
```

```
postgres=> select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;
  slot_name   | slot_type | confirmed_flush_lsn
--------------+-----------+---------------------
 logical_slot | logical   | 32/D8003628
(1 row)

% psql -h test-postgres-instance-3.abcdefabcdef.us-west-2.rds.amazonaws.com

postgres=> select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;
  slot_name   | slot_type | confirmed_flush_lsn
--------------+-----------+---------------------
 logical_slot | logical   | 32/D0001700
(1 row)

postgres=> select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;
  slot_name   | slot_type | confirmed_flush_lsn
--------------+-----------+---------------------
 logical_slot | logical   | 32/D8003628
(1 row)
```

Une fois que vous avez terminé vos tâches de réplication, arrêtez le processus de réplication, supprimez les emplacements de réplication et désactivez la réplication logique. Pour désactiver la réplication logique, modifiez le groupe de paramètres de votre cluster de bases de données et réaffectez à `rds.logical_replication` la valeur `0`. Redémarrez le cluster pour que la modification des paramètres prenne effet.

Limitations et recommandations

Les limites et recommandations suivantes s'appliquent à l'utilisation de la réplication logique avec des clusters de base de données multi-AZ exécutant Postgre SQL version 16 :

- Vous ne pouvez utiliser que des instances de base de données Writer pour créer ou supprimer des emplacements de réplication logiques. Par exemple, la `CREATE SUBSCRIPTION` commande doit utiliser le point de terminaison du rédacteur du cluster dans la chaîne de connexion de l'hôte.
- Vous devez utiliser le point de terminaison du rédacteur de cluster lors de toute synchronisation ou resynchronisation de tables. Par exemple, vous pouvez utiliser les commandes suivantes pour resynchroniser une table récemment ajoutée :

```
Postgres=>ALTER SUBSCRIPTION subscription-name CONNECTION host=writer-endpoint
Postgres=>ALTER SUBSCRIPTION subscription-name REFRESH PUBLICATION
```

- Vous devez attendre que la synchronisation des tables soit terminée avant d'utiliser les instances de base de données du lecteur pour la réplication logique. Vous pouvez utiliser la table du pg_subscription_rel catalogue pour surveiller la synchronisation des tables. La synchronisation des tables est terminée lorsque la `srsubstate` colonne est définie sur ready (`r`).
- Nous recommandons d'utiliser des points de terminaison d'instance pour la connexion de réplication logique une fois la synchronisation initiale des tables terminée. La commande suivante réduit la charge sur l'instance de base de données du rédacteur en transférant la réplication vers l'une des instances de base de données du lecteur :

  ```
  Postgres=>ALTER SUBSCRIPTION subscription-name CONNECTION host=reader-instance-endpoint
  ```

  Vous ne pouvez pas utiliser le même emplacement sur plusieurs instances de base de données à la fois. Lorsque deux applications ou plus répliquent des modifications logiques provenant de différentes instances de base de données du cluster, certaines modifications peuvent être perdues en raison d'un basculement du cluster ou d'un problème réseau. Dans ces situations, vous pouvez utiliser les points de terminaison de l'instance pour la réplication logique dans la chaîne de connexion hôte. L'autre application utilisant la même configuration affichera le message d'erreur suivant :

  ```
  replication slot slot_name is already active for PID x providing immediate feedback.
  ```

- Lorsque vous utilisez l'`pglogical`extension, vous ne pouvez utiliser que le point de terminaison du rédacteur de cluster. L'extension présente des limites connues qui peuvent créer des emplacements de réplication logiques inutilisés lors de la synchronisation des tables. Les emplacements de réplication périmés réservent les fichiers journaux à écriture anticipée (WAL) et peuvent entraîner des problèmes d'espace disque.

Utilisation de répliques de lecture de clusters de bases de données multi-AZ pour Amazon RDS

Un réplica en lecture d'un cluster de base de données est un type spécial de cluster que vous créez à partir d'une instance de base de données source. Après la création d'un réplica en lecture, les mises à jour apportées à l'instance de base de données principale sont copiées de façon asynchrone sur le réplica en lecture du cluster de bases de données multi-AZ. Vous pouvez réduire la charge sur votre instance de base de données principale en acheminant les requêtes en lecture depuis vos applications vers le réplica en lecture. Les réplicas en lecture permettent une montée en puissance basée sur Elastic au-delà des contraintes de capacité d'une seule instance de base de données dans le cas de charges de travail de base de données à lecture intensive.

Vous pouvez également créer un ou plusieurs réplicas en lecture d'instance de base de données à partir d'un cluster de bases de données multi-AZ. Les réplicas en lecture d'instances de base de données vous permettent de dépasser la capacité de calcul ou d'E/S du cluster de bases de données multi-AZ source en dirigeant le trafic de lecture excédentaire vers les réplicas en lecture. Pour le moment, vous ne pouvez pas créer un réplica en lecture du cluster de bases de données multi-AZ à partir d'un cluster de bases de données multi-AZ existant.

Lorsque vous choisissez entre la migration vers un cluster de base de données multi-AZ à l'aide d'une réplique en lecture ou la création d'une réplique en lecture d'instance de base de données à partir d'un cluster de base de données multi-AZ, tenez compte de votre cas d'utilisation spécifique et de vos exigences de performance.

Migration vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture

> Cette approche est idéale lorsque vous devez améliorer la disponibilité et la durabilité de votre base de données tout en minimisant les temps d'arrêt. En utilisant une réplique en lecture pour passer à un cluster de base de données multi-AZ, vous pouvez garantir un fonctionnement continu et la cohérence des données. Cette méthode est particulièrement utile pour les environnements de production où le maintien de la disponibilité et la réduction de l'impact sur les charges de travail réelles sont essentiels.

Création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ

> Cette méthode convient lorsque vous souhaitez dimensionner les opérations de lecture ou décharger le trafic de lecture de votre instance de base de données principale. En créant une réplique en lecture à partir d'un cluster de base de données multi-AZ existant, vous pouvez

répartir les charges de travail lourdes en lecture et améliorer les performances sans affecter la stabilité de l'instance principale.

Le choix de la bonne approche dépend de votre priorité : garantir une disponibilité et une durabilité élevées ou améliorer les performances de lecture. Évaluez les caractéristiques de votre charge de travail et vos exigences opérationnelles pour prendre une décision éclairée.

Rubriques

- Migration vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture
- Création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ

Migration vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture

Pour migrer un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ vers un déploiement de cluster de bases de données multi-AZ avec un temps d'arrêt réduit, vous pouvez créer un réplica en lecture du cluster de bases de données multi-AZ. Pour la source, vous spécifiez l'instance de base de données dans le déploiement mono-AZ ou l'instance de base de données principale dans le déploiement d'instances de base de données multi-AZ. L'instance de base de données peut traiter les transactions d'écriture pendant la migration vers un cluster de bases de données multi-AZ.

Prenez en compte les points suivants avant de créer un réplica en lecture du cluster de base de données multi-AZ :

- L'instance de base de données source doit se trouver sur une version qui prend en charge les clusters de bases de données multi-AZ. Pour de plus amples informations, veuillez consulter Régions et moteurs de base de données pris en charge pour les clusters de bases de données multi-AZ dans Amazon RDS.
- Le réplica en lecture du cluster de bases de données multi-AZ doit se trouver sur la même version majeure que sa source et sur la même version mineure ou ultérieure.
- Vous devez activer les sauvegardes automatiques sur l'instance de base de données source en affectant à la période de rétention des sauvegardes une valeur différente de 0.
- Le stockage alloué à l'instance de base de données source doit être de 100 Go ou plus.
- Pour RDS for MySQL, les `enforce_gtid_consistency` paramètres `gtid-mode` et doivent être définis sur l'instance de `ON` base de données source. Vous devez utiliser un groupe de paramètres

personnalisé, pas le groupe de paramètres par défaut. Pour de plus amples informations, veuillez consulter the section called “Groupes de paramètres DB”.

- Une transaction de longue durée active peut ralentir le processus de création du réplica en lecture. Nous vous recommandons d'attendre que les transactions de longue durée se terminent pour créer un réplica en lecture.
- Si vous supprimez l'instance de base de données source pour un réplica en lecture du cluster de bases de données multi-AZ, le réplica en lecture est promu en cluster de bases de données multi-AZ autonome.

Création et promotion du réplica en lecture du cluster de bases de données multi-AZ

Vous pouvez créer et promouvoir une réplique de lecture d'un cluster de bases de données multi-AZ à l'aide du AWS Management Console AWS CLI, ou RDSAPI.

> **Note**
>
> Nous vous recommandons vivement de créer toutes les répliques de lecture dans le même cloud privé virtuel (VPC) basé sur Amazon VPC de l'instance de base de données source. Si vous créez une réplique en lecture dans une instance de base de données VPC différente de l'instance de base de données source, les plages de routage interdomaines sans classe (CIDR) peuvent se chevaucher entre la réplique et le système AmazonRDS. CIDRle chevauchement rend la réplique instable, ce qui peut avoir un impact négatif sur les applications qui s'y connectent. Si vous recevez une erreur lors de la création du réplica en lecture, choisissez un autre groupe de sous-réseaux de base de données de destination. Pour de plus amples informations, veuillez consulter Utilisation d'un(e) instance de base de données dans un VPC.

Console

Pour procéder à la migration d'un déploiement mono-AZ ou d'un déploiement d'instance de base de données multi-AZ vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture, effectuez les étapes suivantes à l'aide de la AWS Management Console.

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Créez le réplica en lecture du cluster de bases de données multi-AZ.

a. Dans le panneau de navigation, choisissez Databases (Bases de données).

b. Sélectionnez l'instance de base de données que vous voulez utiliser comme source pour votre réplica en lecture.

c. Sous Actions, choisissez Créer des réplicas en lecture.

d. Pour Availability and durability (Disponibilité et durabilité), choisissez Multi-AZ DB cluster (Cluster de bases de données multi-AZ).

e. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture.

f. Pour les sections restantes, spécifiez vos paramètres de cluster de base de données. Pour des informations sur un paramètre, consultez Paramètres de création de clusters de base de données multi-AZ.

g. Choisissez Créer un réplica en lecture.

3. Lorsque vous êtes prêt, promouvez le réplica en lecture pour en faire un cluster de bases de données multi-AZ autonome :

a. Arrêtez l'écriture de toute transaction sur l'instance de base de données source, puis attendez que toutes les mises à jour soient effectuées sur le réplica en lecture.

Les mises à jour de la base de données ont lieu sur le réplica en lecture après avoir eu lieu sur l'instance de base de données principale. Ce délai de réplication peut varier considérablement. Utilisez la métrique `ReplicaLag` pour déterminer à quel moment toutes les mises à jour ont été effectuées sur le réplica en lecture. Pour plus d'informations sur le retard de réplica, consultez Supervision de la réplication en lecture.

b. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

c. Dans la RDS console Amazon, sélectionnez Databases.

Le volet Bases de données s'affiche. Chaque réplica en lecture affiche Réplica dans la colonne Rôle.

d. Choisissez le réplica en lecture du cluster de bases de données multi-AZ que vous voulez promouvoir.

e. Pour Actions, choisissez Promote (Promouvoir).

f. Dans la page Promote read replica (Promouvoir le réplica en lecture), saisissez la période de rétention des sauvegardes et la fenêtre de sauvegarde pour le cluster de bases de données multi-AZ nouvellement promu.

g. Lorsque les paramètres sont tels que vous les souhaitez, sélectionnez Promote read replica (Promouvoir le réplica en lecture).

h. Attendez que l'état du cluster de bases de données multi-AZ promu soit `Available`.

i. Dirigez vos applications pour utiliser le cluster de bases de données multi-AZ promu.

(Facultatif) Supprimez le déploiement mono-AZ ou le déploiement d'instance de base de données multi-AZ s'il n'est plus nécessaire. Pour obtenir des instructions, consultez Suppression d'une instance DB.

AWS CLI

Pour procéder à la migration d'un déploiement mono-AZ ou d'un déploiement d'instance de base de données multi-AZ vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture, effectuez les étapes suivantes à l'aide de la AWS CLI.

1. Créez le réplica en lecture du cluster de bases de données multi-AZ.

   Pour créer une réplique en lecture à partir de l'instance de base de données source, utilisez la AWS CLI commande `create-db-cluster`. Pour `--replication-source-identifier`, spécifiez le nom de ressource Amazon (ARN) de l'instance de base de données source.

   Dans Linux, macOS, ou Unix:

```
aws rds create-db-cluster \
  --db-cluster-identifier mymultiazdbcluster \
  --replication-source-identifier arn:aws:rds:us-
east-2:123456789012:db:mydbinstance
  --engine postgres \
  --db-cluster-instance-class db.m5d.large \
  --storage-type io1 \
  --iops 1000 \
  --db-subnet-group-name defaultvpc \
  --backup-retention-period 1
```

   Dans Windows:

```
aws rds create-db-cluster ^
  --db-cluster-identifier mymultiazdbcluster ^
```

```
  --replication-source-identifier arn:aws:rds:us-
east-2:123456789012:db:mydbinstance
  --engine postgres ^
  --db-cluster-instance-class db.m5d.large ^
  --storage-type io1 ^
  --iops 1000 ^
  --db-subnet-group-name defaultvpc ^
  --backup-retention-period 1
```

2. Arrêtez l'écriture de toute transaction sur l'instance de base de données source, puis attendez que toutes les mises à jour soient effectuées sur le réplica en lecture.

   Les mises à jour de la base de données ont lieu sur le réplica en lecture après avoir eu lieu sur l'instance de base de données principale. Ce délai de réplication peut varier considérablement. Utilisez la métrique `Replica Lag` pour déterminer à quel moment toutes les mises à jour ont été effectuées sur le réplica en lecture. Pour plus d'informations sur le retard de réplica, consultez Supervision de la réplication en lecture.

3. Lorsque vous êtes prêt, promouvez le réplica en lecture pour en faire un cluster de bases de données multi-AZ autonome.

   Pour promouvoir un réplica en lecture du cluster de bases de données multi-AZ, utilisez la commande AWS CLI `promote-read-replica-db-cluster`. Pour `--db-cluster-identifier`, spécifiez l'identifiant du réplica en lecture du cluster de bases de données multi-AZ.

   ```
   aws rds promote-read-replica-db-cluster --db-cluster-identifier mymultiazdbcluster
   ```

4. Attendez que l'état du cluster de bases de données multi-AZ promu soit `Available`.
5. Dirigez vos applications pour utiliser le cluster de bases de données multi-AZ promu.

(Facultatif) Supprimez le déploiement mono-AZ ou le déploiement d'instance de base de données multi-AZ s'il n'est plus nécessaire. Pour obtenir des instructions, consultez Suppression d'une instance DB.

RDS API

Pour migrer un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ vers un cluster de base de données multi-AZ à l'aide d'une réplique en lecture, effectuez les étapes suivantes à l'aide du. RDS API

1. Créez le réplica en lecture du cluster de bases de données multi-AZ.

   Pour créer un réplica en lecture du cluster de bases de données multi-AZ, utilisez l'opération `CreateDBCluster` avec le paramètre `DBClusterIdentifier` requis. Pour`ReplicationSourceIdentifier`, spécifiez le nom de ressource Amazon (ARN) de l'instance de base de données source.

2. Arrêtez l'écriture de toute transaction sur l'instance de base de données source, puis attendez que toutes les mises à jour soient effectuées sur le réplica en lecture.

   Les mises à jour de la base de données ont lieu sur le réplica en lecture après avoir eu lieu sur l'instance de base de données principale. Ce délai de réplication peut varier considérablement. Utilisez la métrique `Replica Lag` pour déterminer à quel moment toutes les mises à jour ont été effectuées sur le réplica en lecture. Pour plus d'informations sur le retard de réplica, consultez Supervision de la réplication en lecture.

3. Lorsque vous êtes prêt, promouvez le réplica en lecture pour en faire un cluster de bases de données multi-AZ autonome.

   Pour promouvoir un réplica en lecture du cluster de bases de données multi-AZ, utilisez l'opération `PromoteReadReplicaDBCluster` avec le paramètre `DBClusterIdentifier` requis. Spécifiez l'identifiant du réplica en lecture du cluster de bases de données multi-AZ.

4. Attendez que l'état du cluster de bases de données multi-AZ promu soit `Available`.

5. Dirigez vos applications pour utiliser le cluster de bases de données multi-AZ promu.

(Facultatif) Supprimez le déploiement mono-AZ ou le déploiement d'instance de base de données multi-AZ s'il n'est plus nécessaire. Pour obtenir des instructions, consultez Suppression d'une instance DB.

Limites de création d'un réplica en lecture du cluster de bases de données multi-AZ

Les limites suivantes s'appliquent à la création d'un réplica en lecture du cluster de bases de données multi-AZ à partir d'un déploiement mono-AZ ou d'un déploiement d'instance de base de données multi-AZ.

- Vous ne pouvez pas créer une réplique de lecture d'un cluster de base de données multi-AZ dans un Compte AWS fichier différent de Compte AWS celui qui possède l'instance de base de données source.

- Vous ne pouvez pas créer une réplique de lecture d'un cluster de base de données multi-AZ dans une instance de base de données Région AWS différente de l'instance de base de données source.
- Vous ne pouvez pas restaurer un réplica en lecture du cluster de bases de données multi-AZ à un instant dans le passé.
- Le chiffrement du stockage doit avoir les mêmes paramètres sur l'instance de base de données source que sur le cluster de bases de données multi-AZ.
- Si l'instance de base de données source est chiffrée, la réplique lue du cluster de base de données multi-AZ doit être chiffrée à l'aide de la même KMS clé.
- Si l'instance de base de données source utilise un stockage à usage général SSD (gp3) et dispose de moins de 400 GiB de stockage alloué, vous ne pouvez pas modifier le IOPS provisionnement prévu pour la réplique de lecture du cluster de base de données multi-AZ.
- Pour effectuer une mise à niveau de version mineure sur l'instance de base de données source, vous devez d'abord effectuer la mise à niveau de version mineure sur le réplica en lecture du cluster de bases de données multi-AZ.
- Lorsque vous effectuez une mise à niveau de version mineure sur une réplique en lecture RDS d'un cluster de base de données Postgre SQL Multi-AZ, l'instance de base de données du lecteur ne passe pas à l'instance de base de données du rédacteur après la mise à niveau. Par conséquent, votre cluster de base de données peut être indisponible pendant qu'Amazon RDS met à niveau l'instance du rédacteur.
- Vous ne pouvez pas effectuer de mise à niveau de version majeure sur une réplique en lecture d'un cluster de bases de données multi-AZ.
- Vous pouvez effectuer une mise à niveau de version majeure sur l'instance de base de données source d'un réplica en lecture du cluster de bases de données multi-AZ, mais la réplication vers les réplicas en lecture s'arrête et ne peut pas être redémarrée.
- Le réplica en lecture du cluster de bases de données multi-AZ ne prend pas en charge les réplicas en lecture en cascade.
- RDSPour PostgreSQL, les répliques de lecture de clusters de bases de données multi-AZ ne peuvent pas basculer.

Création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ

Vous pouvez créer un réplica en lecture d'une instance de base de données à partir d'un cluster de bases de données multi-AZ afin de dimensionner au-delà de la capacité de calcul ou d'E/S du cluster pour les réplicas de base de données à lecture intensive. Vous pouvez diriger ce trafic en lecture excessif vers un ou plusieurs réplicas en lecture d'une instance de base de données. Vous pouvez également utiliser des réplicas en lecture pour migrer d'un cluster de bases de données multi-AZ vers une instance de base de données.

Pour créer un réplica en lecture, spécifiez un cluster de bases de données multi-AZ comme source de réplication. L'une des instances de lecteur du cluster de bases de données multi-AZ est toujours la source de la réplication, et non l'instance d'enregistreur. Cette condition garantit que le réplica est toujours synchronisé avec le cluster source, même en cas de basculement.

Rubriques

- Comparaison des instances de base de données en lecture et des réplicas en lecture d'instances de base de données
- Considérations
- Création d'un réplica en lecture d'une instance de base de données
- Transfert du réplica en lecture de l'instance de base de données
- Limites pour la création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ

Comparaison des instances de base de données en lecture et des réplicas en lecture d'instances de base de données

Un réplica en lecture d'une instance de base de données d'un cluster de bases de données multi-AZ est différente des instances de base de données de lecture du cluster de bases de données multi-AZ pour les raisons suivantes :

- Les réplicas en lecture font office de cibles de basculement automatique, contrairement aux réplicas en lecture d'instances de base de données.
- Les instances de base de données de lecteur doivent accuser réception d'une modification par l'instance de base de données d'enregistreur avant que la modification puisse être validée. Pour les réplicas en lecture d'instance de base de données, les mises à jour sont copiées de façon asynchrone sur les réplicas en lecture, sans accusé de réception nécessaire.

- Les instances de base de données de lecteur partagent toujours la même classe d'instance, le même type de stockage et la même version de moteur que l'instance de base de données d'enregistreur du cluster de bases de données multi-AZ. Les réplicas en lecture d'instances de base de données ne doivent toutefois pas nécessairement partager les mêmes configurations que le cluster source.
- Vous pouvez transformer un réplica en lecture d'une instance de base de données en une instance de base de données autonome. Vous ne pouvez pas transformer une instance de base de données de lecture d'un cluster de bases de données multi-AZ en instance autonome.
- Le point de terminaison de lecture achemine uniquement les demandes vers les instances de base de données de lecture du cluster de bases de données multi-AZ. Il n'achemine jamais les demandes vers un réplica en lecture d'une instance de base de données.

Pour plus d'informations sur les instances de base de données de lecteur et d'enregistreur, consultez the section called “Architecture de cluster de bases de données multi-AZ”.

Considérations

Prenez en compte les points suivants avant de créer un réplica en lecture d'une instance de base de données vers un cluster de bases de données multi-AZ :

- Lorsque vous créez le réplica en lecture d'une instance de base de données, il doit se trouver sur la même version majeure que son cluster source et sur la même version mineure ou ultérieure. Après l'avoir créé, vous pouvez éventuellement mettre à niveau le réplica en lecture vers une version mineure supérieure à celle du cluster source.
- Lorsque vous créez le réplica en lecture de l'instance de base de données, le stockage alloué doit être identique à celui du cluster de bases de données multi-AZ source. Vous pouvez modifier l'espace de stockage alloué après la création du réplica en lecture.
- Pour RDS for MySQL, le `gtid-mode` paramètre doit être défini sur le cluster `ON` de base de données multi-AZ source. Pour de plus amples informations, veuillez consulter the section called “Groupes de paramètres de cluster DB”.
- Une transaction de longue durée active peut ralentir le processus de création du réplica en lecture. Nous vous recommandons d'attendre que les transactions de longue durée se terminent pour créer un réplica en lecture.
- Si vous supprimez le cluster de bases de données multi-AZ source pour un réplica en lecture d'une instance de base de données, tous les réplicas en lecture sur lesquels il écrit sont promues en instance de base de données autonome.

Création d'un réplica en lecture d'une instance de base de données

Vous pouvez créer une réplique de lecture d'instance de base de données à partir d'un cluster de base de données multi-AZ à l'aide de AWS Management Console AWS CLI, ou RDSAPI.

> **Note**
>
> Nous vous recommandons vivement de créer toutes les répliques de lecture dans le même cloud privé virtuel (VPC) basé sur Amazon VPC du cluster de base de données multi-AZ source.
> Si vous créez une réplique en lecture dans un cluster de base de données multi-AZ VPC différent du cluster de base de données source, les plages de routage interdomaines sans classe (CIDR) peuvent se chevaucher entre la réplique et le système. RDS CIDRle chevauchement rend la réplique instable, ce qui peut avoir un impact négatif sur les applications qui s'y connectent. Si vous recevez une erreur lors de la création du réplica en lecture, choisissez un autre groupe de sous-réseaux de base de données de destination. Pour de plus amples informations, veuillez consulter the section called “Utilisation d'un(e) instance de base de données dans un VPC”.

Console

Pour créer un réplica en lecture d'une instance de base de données à partir d'un cluster de bases de données multi-AZ, effectuez les étapes suivantes à l'aide de la AWS Management Console.

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez le cluster de bases de données multi-AZ que vous voulez utiliser comme source pour votre réplica en lecture.
4. Sous Actions, choisissez Créer des réplicas en lecture.
5. Pour Source du réplica, assurez-vous que le cluster de bases de données multi-AZ correct est sélectionné.
6. Sous Identifiant de base de données, saisissez un nom pour le réplica en lecture.
7. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour des informations sur un paramètre, consultez the section called “Paramètres disponibles”.

> **Note**
>
> Le stockage alloué pour le réplica en lecture de l'instance de base de données doit être identique à celui du cluster de bases de données multi-AZ source.

8. Choisissez Créer un réplica en lecture.

AWS CLI

Pour créer une réplique de lecture d'instance de base de données à partir d'un cluster de base de données multi-AZ, utilisez la AWS CLI commande `create-db-instance-read-replica`. Pour `--source-db-cluster-identifier`, spécifiez l'identifiant du cluster de bases de données multi-AZ.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance-read-replica \
  --db-instance-identifier myreadreplica \
  --source-db-cluster-identifier mymultiazdbcluster
```

Dans Windows:

```
aws rds create-db-instance-read-replica ^
  --db-instance-identifier myreadreplica ^
  --source-db-cluster-identifier mymultiazdbcluster
```

RDS API

Pour créer un réplica en lecture d'une instance de base de données à partir d'un cluster de bases de données multi-AZ, utilisez l'opération `CreateDBInstanceReadReplica`.

Transfert du réplica en lecture de l'instance de base de données

Si vous n'avez plus besoin du réplica en lecture de l'instance de base de données, vous pouvez le transformer en une instance de base de données autonome. Lorsque vous effectuez la promotion d'un réplica en lecture, l'instance de bases de données est redémarrée avant de devenir disponible. Pour obtenir des instructions, consultez the section called “Promotion d'un réplica en lecture”.

Si vous utilisez le réplica en lecture pour procéder à la migration d'un déploiement de cluster de bases de données multi-AZ vers un déploiement d'instance de base de données mono-AZ ou

multi-AZ, assurez-vous d'arrêter toutes les transactions en cours d'écriture sur le cluster de bases de données source. Ensuite, attendez que toutes les mises à jour soient apportées au réplica en lecture. Les mises à jour de la base de données ont lieu sur les réplicas en lecture après avoir eu lieu sur l'une des instances de base de données de lecture du cluster de bases de données multi-AZ. Ce délai de réplication peut varier considérablement. Utilisez la métrique `ReplicaLag` pour déterminer à quel moment toutes les mises à jour ont été effectuées sur le réplica en lecture. Pour plus d'informations sur le retard de réplica, consultez the section called “Supervision de la réplication en lecture”.

Après avoir transféré le réplica en lecture, attendez que le statut de l'instance de base de données promue affiche `Available` avant de demander à vos applications d'utiliser l'instance de base de données promue. Vous pouvez éventuellement supprimer le déploiement du cluster de bases de données multi-AZ si vous n'en n'avez plus besoin. Pour obtenir des instructions, consultez the section called “Suppression d'un cluster de base de données multi-AZ”.

Limites pour la création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ

Les limites suivantes s'appliquent à la création d'un réplica en lecture d'une instance de base de données à partir d'un déploiement de cluster de bases de données multi-AZ.

- Vous ne pouvez pas créer une réplique lue d'instance de base de données dans un fichier différent de Compte AWS Compte AWS celui qui possède le cluster de base de données multi-AZ source.
- Vous ne pouvez pas créer une réplique de lecture d'instance de base de données dans un cluster de base de données multi-AZ Région AWS différent du cluster de base de données source.
- Vous ne pouvez pas restaurer un réplica en lecture d'une instance de base de données à un instant dans le passé.
- Le chiffrement du stockage doit avoir les mêmes paramètres sur le cluster de bases de données source multi-AZ et sur le réplica en lecture de l'instance de base de données.
- Si le cluster de base de données multi-AZ source est chiffré, la réplique lue de l'instance de base de données doit être chiffrée à l'aide de la même KMS clé.
- Pour effectuer une mise à niveau de version mineure sur le cluster de bases de données multi-AZ source, vous devez d'abord effectuer la mise à niveau de version mineure sur le réplica en lecture de l'instance de base de données.
- Le réplica en lecture de l'instance de base de données ne prend pas en charge les réplicas en lecture en cascade.

- RDSPour PostgreSQL, le cluster de base de données multi-AZ source doit exécuter Postgre SQL version 13.11, 14.8 ou 15.2.R2 ou supérieure afin de créer une réplique en lecture d'instance de base de données.
- Vous pouvez effectuer une mise à niveau de la version majeure sur le cluster de bases de données multi-AZ source d'un réplica en lecture de l'instance de base de données, mais la réplication vers le réplica en lecture s'arrête et ne peut pas être redémarrée.

Configuration de la réplication externe à partir de clusters de bases de données multi-AZ pour Amazon RDS

Vous pouvez configurer la réplication entre un cluster de bases de données multi-AZ et une base de données externe à AmazonRDS.

La réplication externe permet aux clusters de bases de données multi-AZ de répliquer des données entre une RDS instance de base de données et une base de données externe, sur site ou dans un autre environnement cloud. C'est avantageux pour la reprise après sinistre, la migration des données et le maintien de la cohérence entre les systèmes situés sur différents sites. La section couvre les conditions préalables à la configuration de la réplication, la manière de configurer le processus et les principales considérations telles que la latence de réplication, la bande passante et la compatibilité avec les différents moteurs de base de données.

RDSpour My SQL

Pour configurer la réplication externe RDS pour un cluster de base de données My SQL Multi-AZ, vous devez conserver les fichiers journaux binaires sur les instances de base de données du cluster suffisamment longtemps pour garantir que les modifications sont appliquées à la réplique avant qu'Amazon ne RDS supprime le fichier binlog. Pour ce faire, configurez la conservation des journaux binaires en appelant la procédure `mysql.rds_set_configuration` stockée et en spécifiant le `binlog retention hours` paramètre. Pour de plus amples informations, veuillez consulter the section called “nombre d'heures de conservation du journal binaire”.

La valeur par défaut pour `binlog retention hours` est`NULL`, ce qui signifie que les journaux binaires ne sont pas conservés (0 heure). Si vous souhaitez configurer une réplication externe pour un cluster de base de données multi-AZ, vous devez définir le paramètre sur une valeur autre que`NULL`.

Vous ne pouvez configurer la conservation des journaux binaires qu'à partir de l'instance de base de données Writer du cluster de base de données Multi-AZ, et le paramètre est propagé à toutes les instances de base de données de lecture de manière asynchrone.

En outre, nous vous recommandons vivement d'activer la réplication GTID basée sur votre réplique externe. Ensuite, si l'une des instances de base de données échoue, vous pouvez reprendre la réplication à partir d'une autre instance de base de données saine au sein du cluster. Pour plus d'informations, consultez la section Réplication avec des identifiants de transaction globaux dans la section Ma SQL documentation.

RDSpour Postgre SQL

Pour configurer la réplication externe RDS pour un cluster de base de données SQL multi-AZ Postgre, vous devez activer la réplication logique. Pour obtenir des instructions, consultez the section called “Réplication SQL logique Postgre avec des clusters de bases de données multi-AZ”.

Suppression d'un cluster de base de données multi-AZ pour Amazon RDS

Vous pouvez supprimer un cluster DB Multi-AZ à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Le temps nécessaire pour supprimer un cluster de base de données multi-AZ peut varier en fonction des facteurs suivants :

- La période de conservation des sauvegardes (c'est-à-dire le nombre de sauvegardes à supprimer).
- Combien de données sont supprimées.
- Si un instantané final est pris.

La protection contre la suppression doit être désactivée sur le cluster de base de données Multi-AZ avant de pouvoir le supprimer. Pour de plus amples informations, veuillez consulter the section called “Conditions préalables pour la suppression d'une instance de base de données”. Vous pouvez désactiver la protection contre la suppression en modifiant le cluster de base de données multi-AZ. Pour de plus amples informations, veuillez consulter the section called “Modification d'un cluster de base de données multi-AZ”.

Console

Pour supprimer un cluster de base de données multi-AZ

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données), puis le cluster de base de données multi-AZ que vous souhaitez supprimer.
3. Pour Actions, choisissez Supprimer.
4. Choisissez Create final snapshot? (Créer un instantané final ?)pour créer un instantané de base de données final pour le cluster de base de données multi-AZ.

   Si vous créez un instantané final, saisissez un nom dans Final snapshot name (Nom de l'instantané final).
5. Pour retenir les sauvegardes automatiques, choisissez Retain automated backups (Retenir les sauvegardes automatiques).
6. Saisissez **`delete me`** dans la zone.
7. Sélectionnez Delete (Supprimer).

AWS CLI

Pour supprimer un cluster de base de données multi-AZ à l'aide de AWS CLI, appelez la delete-db-clustercommande avec les options suivantes :

- `--db-cluster-identifier`
- `--final-db-snapshot-identifier` ou `--skip-final-snapshot`

Example Avec un instantané final

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-cluster \
    --db-cluster-identifier mymultiazdbcluster \
    --final-db-snapshot-identifier mymultiazdbclusterfinalsnapshot
```

Dans Windows:

```
aws rds delete-db-cluster ^
    --db-cluster-identifier mymultiazdbcluster ^
    --final-db-snapshot-identifier mymultiazdbclusterfinalsnapshot
```

Example Sans instantané final

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-cluster \
    --db-cluster-identifier mymultiazdbcluster \
    --skip-final-snapshot
```

Dans Windows:

```
aws rds delete-db-cluster ^
    --db-cluster-identifier mymultiazdbcluster ^
    --skip-final-snapshot
```

RDS API

Pour supprimer un cluster de base de données multi-AZ à l'aide d'Amazon RDSAPI, appelez l'eleteDBClusteropération D avec les paramètres suivants :

- `DBClusterIdentifier`
- `FinalDBSnapshotIdentifier` ou `SkipFinalSnapshot`

Limites des clusters de bases de données multi-AZ pour Amazon RDS

Un cluster de base de données multi-AZ compte une instance de base de données d'écriture et deux instances de base de données de lecture dans trois zones de disponibilité distinctes. Les clusters de base de données multi-AZ offrent une haute disponibilité, une capacité accrue pour les charges de travail en lecture et une moindre latence par rapport aux déploiements multi-AZ. Pour de plus amples informations sur les clusters de base de données multi-AZ, consultez Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.

Les limitations suivantes s'appliquent aux clusters de bases de données multi-AZ.

- Les clusters de base de données multi-AZ ne prennent pas en charge les fonctions suivantes :
  - IPv6 connexions (mode double pile)
  - Sauvegardes automatiques interrégionales
  - Authentification Kerberos
  - Modification du port. Il existe une solution de remplacement qui consiste à restaurer un cluster de base de données multi-AZ à un instant dans le passé et à spécifier un port différent.
  - Groupes d'options
  - Point-in-time-recovery (PITR) pour les clusters supprimés
  - Mise à l'échelle automatique du stockage en définissant le stockage maximal alloué. En guise d'alternative, vous pouvez mettre le stockage à l'échelle manuellement.
  - Arrêt et démarrage du cluster de base de données multi-AZ
  - Copie d'un instantané de cluster de base de données multi-AZ
  - Chiffrement d'un cluster de base de données Multi-AZ non chiffré
- Les clusters de base de données Multi-AZ RDS for MySQL ne prennent en charge que les procédures stockées système suivantes :
  - `mysql.rds_rotate_general_log`
  - `mysql.rds_rotate_slow_log`
  - `mysql.rds_show_configuration`
  - `mysql.rds_set_external_master_with_auto_position`
  - `mysql.rds_set_configuration`
- Les clusters de bases de données multi-AZ RDS pour PostgreSQL ne prennent pas en charge les extensions suivantes : et. `aws_s3` `pg_transport`

- Les clusters de base de données multi-AZ RDS for PostgreSQL ne prennent pas en charge l'utilisation d'un serveur DNS personnalisé pour l'accès réseau sortant.

Le support étendu RDS vous permet de continuer à exécuter une base de données sur une version majeure du moteur après la date de fin du support standard de RDS moyennant des frais supplémentaires.

Vous ne pouvez inscrire une base de données au support étendu RDS qu'en activant le support étendu RDS lorsque vous créez ou restaurez une instance de base de données pour la première fois. Vous ne pouvez pas mettre à jour votre statut d'inscription au RDS Extended Support sur les instances de base de données existantes, sauf si vous les restaurez.

Si vous avez activé le support étendu RDS lors de la création ou de la restauration d'une instance de base de données, Amazon RDS Amazon inscrira automatiquement l'instance de base de données au service RDS Extended Support après la date de fin du support standard de RDS . L'inscription automatique à RDS Extended Support ne modifie pas le moteur de base de données et n'a aucun impact sur le temps de disponibilité ou les performances de votre instance de base de données.

Le support étendu RDS fournit les mises à jour et le support technique suivants :

- Mises à jour de sécurité critiques et élevées CVEs pour votre instance de base de données ou votre cluster de base de données, y compris le moteur de base de données
- Corrections de bogues et correctifs pour les problèmes critiques
- Possibilité d'ouvrir des dossiers d'assistance et de recevoir de l'aide au dépannage dans le cadre du contrat de niveau de service Amazon RDS standard

Cette offre payante vous donne plus de temps pour passer à une version majeure du moteur compatible. Par exemple, la date de fin du support standard RDS pour RDS for MySQL 5.7 est le 29 février 2024. Cependant, vous n'êtes pas prêt à passer manuellement à la version 8.0 de RDS pour MySQL avant cette date. Dans ce cas, Amazon RDS inscrit automatiquement vos bases de données à RDS Extended Support le 29 février 2024, et vous pouvez continuer à exécuter RDS for MySQL version 5.7. À compter du 1er mars 2024, Amazon RDS vous facture automatiquement le support étendu RDS.

Le support étendu RDS est disponible jusqu'à 3 ans après la date de fin de support standard RDS pour une version majeure du moteur). Passé ce délai, si vous n'avez pas mis à niveau la version principale de votre moteur vers une version prise en charge, Amazon RDS () mettra automatiquement à niveau votre version principale du moteur. Nous vous recommandons de mettre à niveau vers une version majeure prise en charge du moteur dès que possible.

Pour plus d'informations sur les dates de fin du support standard RDS et les dates de fin du support étendu RDS, consultez le calendrier de publication Versions de MySQL majeures prises en charge sur Amazon RDS d'Amazon RDS for PostgreSQL.

Rubriques

- Présentation du support étendu d'Amazon RDS
- Frais d'Amazon RDS Extended Support
- Versions avec Amazon RDS Extended Support
- Amazon RDS et les responsabilités des clients avec Amazon RDS Extended Support
- Création d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support
- Afficher l'inscription de vos instances de base de données ou de vos clusters de bases de données multi-AZ, de vos clusters de base de dans Amazon RDS Extended Support
- Affichage des dates de support pour les versions du moteur dans Amazon RDS Extended Support
- Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Présentation du support étendu d'Amazon RDS

Après la date de fin du support standard de RDS , si vous n'avez pas désactivé RDS Extended Support lors de la création ou de la restauration de vos instances de base de données, Amazon RDS les inscrira automatiquement au RDS Extended Support. Amazon RDS met automatiquement à niveau votre instance de base de données vers la dernière version mineure publiée avant la date de fin du support standard de RDS , si vous n'utilisez pas déjà cette version. Amazon RDS ne mettra pas à niveau votre version mineure avant la fin de la date de support standard de RDS pour votre version principale du moteur.

Vous pouvez créer de nouvelles bases de données avec les principales versions du moteur qui ont atteint la date de fin de support standard de RDS . RDS inscrit automatiquement ces nouvelles bases de données au RDS Extended Support et vous facture cette offre.

Si vous effectuez une mise à niveau vers un moteur toujours couvert par le support standard de RDS avant la date de fin du support standard de RDS , Amazon RDS n'inscrira pas votre moteur au support étendu RDS.

Si la restauration échoue, Amazon RDS () inscrira automatiquement votre moteur à RDS Extended Support avec une version compatible avec le snapshot.

Vous pouvez mettre fin à l'inscription au RDS Extended Support à tout moment. Pour mettre fin à l'inscription, mettez à niveau chaque moteur inscrit vers une version plus récente qui bénéficie toujours du support standard de RDS . La fin de l'inscription au support étendu RDS prendra effet le jour où vous aurez effectué une mise à niveau vers une version plus récente du moteur toujours couverte par le support standard de RDS .

Pour plus d'informations sur les dates de fin du support standard RDS et les dates de fin du support étendu RDS, consultez le calendrier de publication Versions de MySQL majeures prises en charge sur Amazon RDS d'Amazon RDS for PostgreSQL.

Frais d'Amazon RDS Extended Support

Vous devrez payer des frais pour tous les moteurs inscrits au Support RDS étendu à compter du lendemain de la date de RDS fin de l'assistance standard. Pour connaître la date de RDS fin du support standard, consultez le Versions majeures calendrier de publication d'Amazon RDS pour Postgre SQL. RDSLes frais de support étendu s'appliquent aux instances de secours dans le cadre de déploiements multi-AZ.

Les frais supplémentaires pour le Support RDS étendu s'arrêtent automatiquement lorsque vous effectuez l'une des actions suivantes :

- Passez à une version du moteur couverte par le support standard.
- Supprimez la base de données qui exécute une version majeure après la date de fin du support standard d'RDS.

Les frais reprendront si la version de votre moteur cible bénéficie du Support RDS étendu à l'avenir.

Par exemple, RDSpour Postgre SQL 11 entre en support étendu le 1er mars 2024, mais les frais ne commencent pas avant le 1er avril 2024. Vous mettez à niveau votre base de données RDSpour Postgre SQL 11 vers RDSPostgre SQL Postgre 12 le 30 avril 2024. Vous ne serez facturé que pour 30 jours de Support étendu RDSpour Postgre SQL Aurora Postgre . Vous continuez RDSà exécuter Postgre SQL 12 sur cette instance de base de données après la date de RDS fin du support standard du 28 février 2025. Votre base de données sera à nouveau soumise à des frais de Support RDS étendu à compter du 1er mars 2025.

Pour plus d'informations, consultez les SQLtarifs Amazon RDS for My et Amazon RDS for Postgre, SQL les tarifs .

Éviter les frais liés à Amazon RDS Extended Support

Vous pouvez éviter d'être facturé pour le support RDS étendu en empêchant RDS de créer ou de restaurer une instance de base de données ou un cluster de base de données multi-AZ, un cluster après la date de fin du support standard d'RDS. Pour ce faire, utilisez le AWS CLI ou le RDSAPI.

Dans le AWS CLI, spécifiez `open-source-rds-extended-support-disabled` l'`--engine-lifecycle-support`option. Dans le RDSAPI, spécifiez `open-source-rds-extended-support-disabled` le `LifeCycleSupport` paramètre. Pour plus d'informations, consultez Création d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster ou Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster .

Versions avec Amazon RDS Extended Support

RDS Extended Support n'est disponible que pour les versions majeures. Il n'est pas disponible pour les versions mineures.

Le support étendu RDS est disponible pour RDS pour MySQL et pour RDS pour PostgreSQL. Pour plus d'informations, consultez le calendrier Versions majeures de publication d'Amazon RDS pour PostgreSQL dans les notes de mise à jour d'Amazon RDS pour PostgreSQL.

Vous pouvez également consulter les informations sur les dates de support pour les versions du moteur à l'aide de l'API AWS CLI ou de l'API RDS. Pour de plus amples informations, veuillez consulter Affichage des dates de support pour les versions du moteur dans Amazon RDS Extended Support.

Dénomination de la version d'Amazon RDS Extended Support

Amazon RDS publiera de nouvelles versions mineures avec des correctifs et des correctifs CVE pour les moteurs sur RDS Extended Support. Pour plus d'informations, consultez les mises à jour Versions de support étendu d'Amazon RDS pour RDS pour MySQL d'Amazon RDS Extended Support pour RDS pour PostgreSQL dans les notes de mise à jour d'Amazon RDS pour PostgreSQL.

Les noms de ces versions mineures seront au format major.minor-RDS.YYYYMMDD.patch.YYYYMMDD, par exemple, 5.7.44-RDS.20240208.R2.20240210 (pour RDS pour MySQL) ou 11.22-RDS.20240208.R2.20240210 (pour RDS pour PostgreSQL).

majeur

Pour MySQL, le numéro de version principal est à la fois le nombre entier et la première partie fractionnaire du numéro de version, par exemple 8.0. Une mise à niveau majeure augmente la partie majeure du numéro de version. Par exemple, une mise à niveau de 5.7.44 vers 8.0.33 est une mise à niveau de version majeure, où 5.7 et 8.0 sont les numéros de version principaux.

Pour PostgreSQL, le numéro de version principal est un entier, par exemple 11.

Minor-rds.yyyymmdd

Pour MySQL, le numéro de version secondaire est la troisième partie du numéro de version, par exemple le 44-RDS.20240208 in5.7.44-RDS.20240208.

Pour PostgeSQL, le numéro de version secondaire est la deuxième partie du numéro de version, par exemple, le 22-RDS.20240208 in. 11.22-RDS.20240208

La date est celle à laquelle Amazon RDS a créé la version mineure d'Amazon RDS.

patch

La version du correctif est celle qui suit la date à laquelle Amazon RDS a créé la version mineure d'Amazon RDS, par exemple, le R2 dans 5.7.44-RDS.20240208.R2 ou. 11.22-RDS.20240208.R2

Une version de correctif d'Amazon RDS inclut des corrections de bogues importantes ajoutées à une version mineure d'Amazon RDS après sa publication.

YYYYMMDD

La date correspond à laquelle Amazon RDS a créé la version du correctif, par exemple, le 20240210 dans ou. 5.7.44-RDS.20240208.R2.20240210 11.22-RDS.20240208.R2.20240210

Une version datée d'Amazon RDS est un correctif de sécurité qui inclut des correctifs de sécurité importants ajoutés à une version mineure après sa publication. Il n'inclut aucun correctif susceptible de modifier le comportement d'un moteur.

Amazon RDS et les responsabilités des clients avec Amazon RDS Extended Support

Le contenu suivant décrit les responsabilités d'Amazon RDS Amazon et vos responsabilités en matière de Support RDS étendu.

Rubriques

- Responsabilités RDS
- Vos responsabilités

Responsabilités RDS

Après la date de fin du support standard d'RDS, RDSAmazon fournira des correctifs, des corrections de bogues et des mises à niveau pour les moteurs inscrits au support RDS étendu. Cela se produira pendant une période maximale de 3 ans, ou jusqu'à ce que vous arrêtiez d'utiliser les moteurs, selon la première éventualité.

Les correctifs concerneront les niveaux de gravité critique et élevé, tels CVEs que définis par les cotes de CVSS gravité de la National Vulnerability Database (NVD). Pour plus d'informations, consultez Métriques de vulnérabilité (langue française non garantie).

Vos responsabilités

Vous êtes responsable de l'application des correctifs, des corrections de bogues et des mises à niveau fournis pour les instances de base de données ou les clusters de base de données multi-AZ, les inscrits au Support RDS étendu. Amazon RDS se réserve le droit de modifier, de remplacer ou de retirer ces correctifs, corrections de bogues et mises à niveau à tout moment. Si un correctif est nécessaire pour résoudre des problèmes de sécurité ou de stabilité critiques, RDSAmazon se réserve le droit de mettre à jour vos instances de base de données ou vos clusters de bases de données multi-AZ, vos clusters de base de avec le correctif, ou d'exiger que vous installiez le correctif.

Vous êtes également responsable de la mise à niveau de votre moteur vers une version plus récente avant la RDS fin de la date de Support étendu. La date de RDS fin du Support étendu est généralement 3 ans après la date de support RDS standard. Pour connaître la date de RDS fin du Support étendu pour la version majeure de votre moteur de base de données, consultez Versions majeures le calendrier de publication d'Amazon RDS pour Postgre SQL.

Si vous ne mettez pas à niveau votre moteur, après la RDS fin de la date de support étendu, Amazon RDS Amazon tentera de le mettre à niveau vers une version plus récente prise en charge dans le cadre du support standard d'RDS. Si la mise à niveau échoue, Amazon RDS Amazon se réserve le droit de supprimer l'instance de base de données ou le cluster de base de données multi-AZ, le qui exécute le moteur après la date de fin du support standard d'RDS. Toutefois, avant de le faire, Amazon RDS préservera vos données provenant de ce moteur.

Création d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ, , sélectionnez Activer le support RDS étendu dans la console ou utilisez l'option Support étendu dans le AWS CLI ou le paramètre dans le RDSAPI. Lorsque vous inscrivez une instance de base de données ou un cluster de base de données multi-AZ, un cluster de base de dans RDS Amazon Extended RDS Support, celui-ci est inscrit de façon permanente au support étendu pendant toute la durée de vie de l'instance de base de données ou du cluster de base de données multi-AZ, du cluster de base de données Aurora global.

Si vous utilisez la console, vous devez sélectionner Activer le support RDS étendu. Le paramètre n'est pas sélectionné par défaut.

Si vous utilisez le paramètre Support étendu AWS CLI ou si vous ne spécifiez pas le paramètre Support RDS étendu, Amazon RDS active par défaut le Support RDS étendu. RDS API Lorsque vous automatisez à l'aide de services AWS CloudFormationou d'autres services, ce comportement par défaut maintient la disponibilité de votre base de données après la date de fin du support standard d'RDS.

Vous pouvez empêcher l'inscription au Support RDS étendu en utilisant le AWS CLIou le RDSAPIpour créer une instance de base de données ou un cluster de base de données multi-AZ, un cluster .

Rubriques

- RDSComportement du Support étendu
- Considérations relatives RDS au Support étendu
- Créez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support

RDSComportement du Support étendu

Le tableau suivant récapitule ce qui se passe lorsqu'une version majeure du moteur atteint la fin du support standard d'RDS.

RDSStatut de support étendu*	Attitude
Activées	Amazon vous RDS facture le Support RDS étendu.
Désactivées	Amazon RDS met à niveau votre instance de base de données ou votre cluster de base de données multi-AZ, votre vers une version de moteur prise en charge. Cette mise à niveau a lieu à la date de fin du support standard d'RDS ou peu après.

* Dans la RDS console, le statut Support RDS étendu apparaît comme Oui ou Non. Dans le bloc AWS CLI opératoire RDSAPI, le statut Support RDS étendu apparaît sous la forme `open-source-rds-extended-support` ou`open-source-rds-extended-support-disabled`.

Considérations relatives RDS au Support étendu

Avant de créer une instance de base de données ou un cluster de base de données multi-AZ, un cluster , tenez compte des éléments suivants :

- Une fois la date de fin du support standard d'RDS passée, vous pouvez empêcher la création d'une nouvelle instance de base de données ou d'un nouveau cluster de base de données multi-AZ, d'un nouveau cluster et éviter les frais de support RDS étendu. Pour ce faire, utilisez le AWS CLI ou le RDSAPI. Dans le AWS CLI, spécifiez `open-source-rds-extended-support-disabled` l'`--engine-lifecycle-support`option. Dans le RDSAPI, spécifiez `open-source-rds-extended-support-disabled` le `LifeCycleSupport` paramètre. Si vous le spécifiez `open-source-rds-extended-support-disabled` et que la date de fin du support standard d'RDS est dépassée, la création d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster échouera toujours.
- RDSLe Support étendu est défini au niveau du cluster. Les membres d'un cluster auront toujours le même paramètre pour le Support RDS étendu dans la RDS console, `--engine-lifecycle-support` dans le AWS CLI et `EngineLifecycleSupport` dans le RDSAPI.

Pour plus d'informations, consultez Versions MySQL et publiez les calendriers d'Amazon RDS pour SQL Postgre.

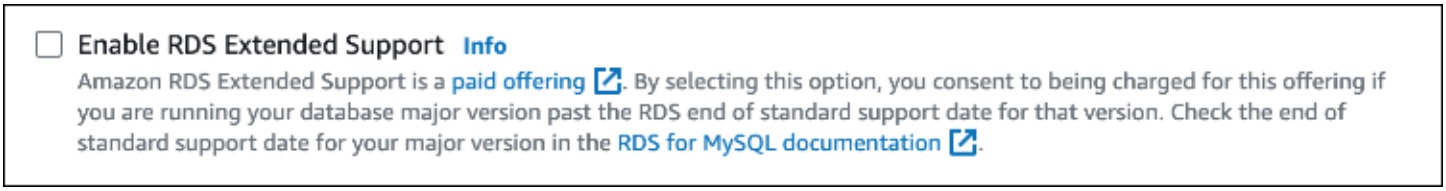
Créez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support

Vous pouvez créer une instance de base de données ou un cluster de base de données multi-AZ, avec une version de support RDS étendu en utilisant le AWS Management Console AWS CLI, le ou le RDSAPI.

Console

Lorsque vous créez , une instance de base de données ou un cluster de base de données multi-AZ, dans la section des options du moteur, sélectionnez Activer le support RDS étendu. Ce paramètre n'est pas sélectionné par défaut.

L'image suivante montre le paramètre Activer le support RDS étendu :



AWS CLI

Lorsque vous exécutez la AWS CLI commande or create-db-cluster(cluster de base de données multi-AZ), sélectionnez Support RDS étendu en spécifiant `open-source-rds-extended-support` l'`--engine-lifecycle-support`option. Par défaut, cette option est définie suropen-source-rds-extended-support.

Pour empêcher la création d'une nouvelle instance de base de données ou d'un cluster de base de données multi-AZ, après la date de fin de support standard d'RDS, spécifiez `open-source-rds-extended-support-disabled` l'`--engine-lifecycle-support`option. Ce faisant, vous éviterez les frais de Support RDS étendu associés.

RDS API

Lorsque vous utilisez l'RDSAPIopération Amazon C reateDBCluster (cluster de bases de données multi-AZ), sélectionnez RDS Extended Support en définissant le `EngineLifecycleSupport` paramètre sur. `open-source-rds-extended-support` Par défaut, ce paramètre est défini sur `open-source-rds-extended-support`.

Pour empêcher la création d'une nouvelle instance de base de données ou d'un cluster de base de données multi-AZ, après la date de fin de support standard d'RDS, spécifiez `open-source-`

`rds-extended-support-disabled` le `EngineLifecycleSupport` paramètre. Ce faisant, vous éviterez les frais de Support RDS étendu associés.

Pour plus d'informations, consultez les rubriques suivantes :

- Pour créer une instance de base de données, suivez les instructions relatives à votre moteur de base de données dans Création d'une instance de base de données Amazon RDS.
- Pour créer un cluster de base de données Multi-AZ, suivez les instructions pour votre moteur de base de données présentées dans Création d'un cluster de base de données multi-AZ pour Amazon RDS.

Afficher l'inscription de vos instances de base de données ou de vos clusters de bases de données multi-AZ, de vos clusters de base de dans Amazon RDS Extended Support

Vous pouvez consulter l'inscription de vos instances de base de données ou de vos clusters de bases de données multi-AZ, de vos clusters de bases de dans RDS Extended Support à l'aide de l' AWS Management Console API, de ou de l'API RDS. AWS CLI

> **Note**
>
> La colonne RDS Extended Support dans la console, l'`-engine-lifecycle-support`option - dans le AWS CLI, et le `EngineLifecycleSupport` paramètre dans l'API RDS indiquent uniquement l'inscription au RDS Extended Support. Les frais du support étendu RDS ne commencent que lorsque la version de votre moteur de base de données a atteint la fin du support standard RDS. Pour plus d'informations, consultez le calendrier Versions majeures de publication d'Amazon RDS pour PostgreSQL dans les notes de mise à jour d'Amazon RDS pour PostgreSQL.
> Par exemple, vous avez une base de données RDS for MySQL 5.7 inscrite au RDS Extended Support. Le 1er mars 2024, Amazon RDS a commencé à vous facturer le Support étendu RDS pour cette base de données. Le 31 juillet 2024, vous avez mis à niveau cette base de données vers RDS pour MySQL 8.0. Le statut de support étendu RDS pour cette base de données reste activé. Cependant, les frais de support étendu RDS pour cette base de données ont été interrompus car RDS for MySQL 8.0 n'avait pas encore atteint la fin du support standard. Amazon RDS ne vous facturera pas le support étendu RDS pour cette

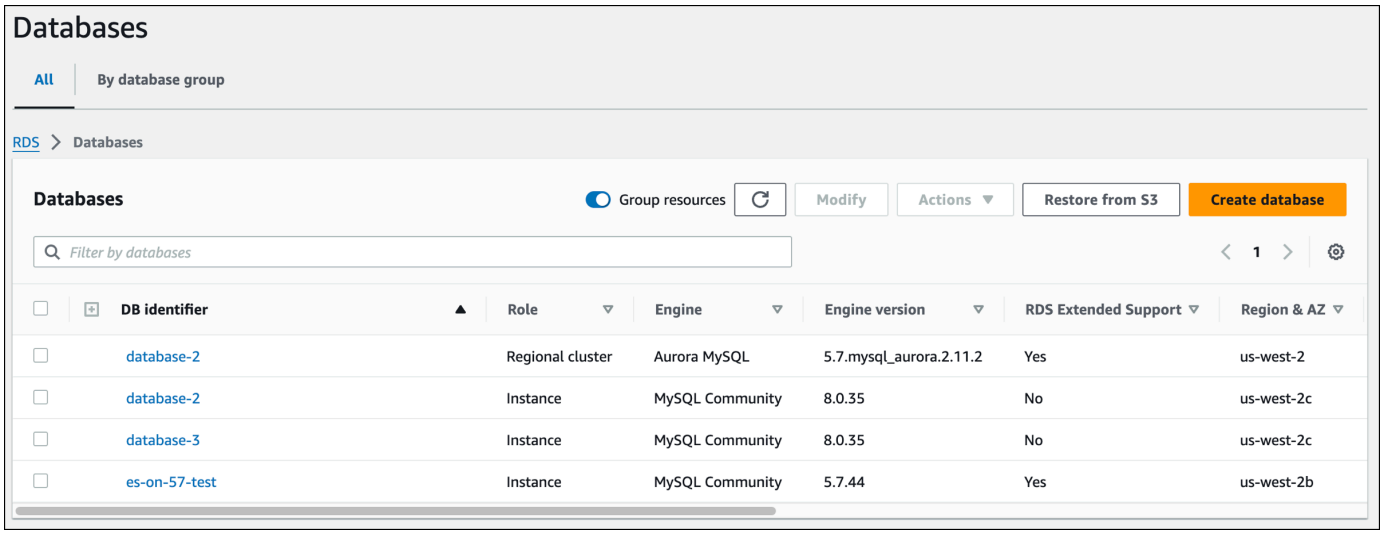
base de données avant le 1er août 2026, date à laquelle le support standard RDS prendra fin pour RDS pour MySQL 8.0.

Console

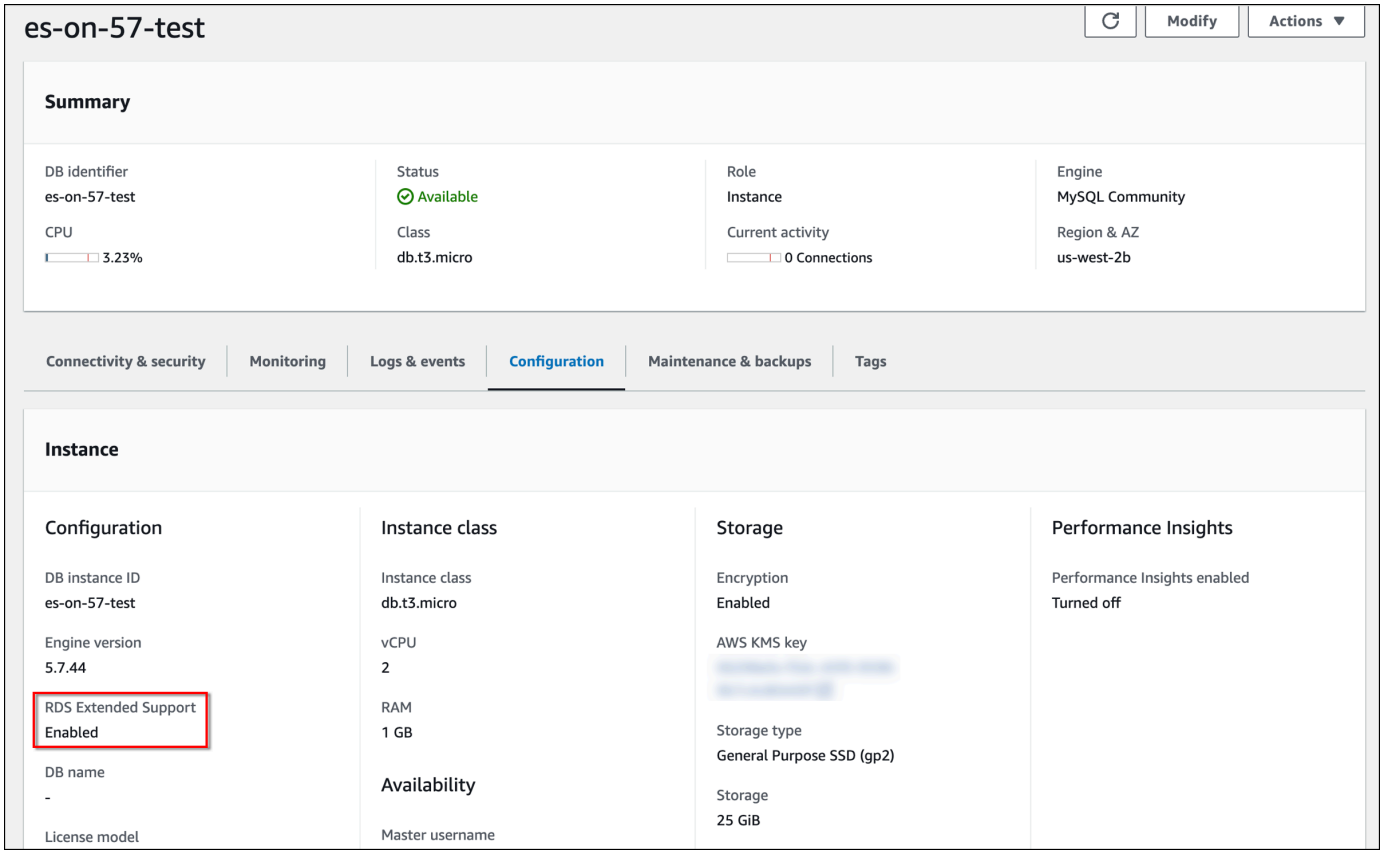
Pour consulter l'inscription de vos instances de base de données ou de vos clusters de bases de données multi-AZ, de vos clusters de base de dans RDS Extended Support

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données). La valeur sous RDS Extended Support indique si une instance de base de données ou un cluster de base de données multi-AZ, un cluster de base de est inscrit au RDS Extended Support. Si aucune valeur n'apparaît, cela signifie que le support étendu RDS n'est pas disponible pour votre base de données.

   > **Tip**
   >
   > Si la colonne Support étendu RDS n'apparaît pas, choisissez l'icône Préférences, puis activez le Support étendu RDS.



3. Vous pouvez également consulter l'inscription dans l'onglet Configuration pour chaque base de données. Choisissez une base de données sous identifiant de base de données. Dans l'onglet Configuration, regardez sous Support étendu pour voir si la base de données est inscrite ou non.



AWS CLI

Pour consulter l'inscription de vos bases de données à RDS Extended Support à l'aide de la AWS CLI commande (clusters de bases de données multi-AZ).

Si le Support étendu RDS est disponible pour une base de données, la réponse inclut le paramètre`EngineLifecycleSupport`. La valeur `open-source-rds-extended-support` indique qu'une instance de base de données ou un cluster de base de données multi-AZ, un cluster est inscrit au RDS Extended Support. La valeur `open-source-rds-extended-support-disabled` indique que l'inscription de l'instance de base de données ou du cluster de base de données multi-AZ, du cluster de base de dans RDS Extended Support a été désactivée.

Exemple

La commande suivante renvoie des informations pour toutes vos instances de base de données :

```
aws rds describe-db-instances
```

La réponse suivante indique qu'un moteur PostgreSQL exécuté sur l'`database-1`instance de base de données est inscrit au RDS Extended Support :

```
{
    "DBInstanceIdentifier": "database-1",
    "DBInstanceClass": "db.t3.large",
    "Engine": "postgres",
    ...
    "EngineLifecycleSupport": "open-source-rds-extended-support"
}
```

API RDS

https://docs.aws.amazon.com/AmazonRDS/latest/APIReference/API_DescribeDBClusters.html

Si le Support étendu RDS est disponible pour une base de données, la réponse inclut le paramètre`EngineLifecycleSupport`. La valeur `open-source-rds-extended-support` indique qu'une instance de base de données ou un cluster de base de données multi-AZ, un cluster est inscrit au RDS Extended Support. La valeur `open-source-rds-extended-support-disabled` indique que l'inscription de l'instance de base de données ou du cluster de base de données multi-AZ, du cluster de base de dans RDS Extended Support a été désactivée.

Affichage des dates de support pour les versions du moteur dans Amazon RDS Extended Support

Vous pouvez consulter les informations relatives aux dates de support des versions du moteur pour vos instances de base de données ou vos clusters de bases de données multi-AZ, vos clusters de base de dans Amazon RDS Extended Support à l'aide de l'API AWS CLI ou de l'API RDS. Ces informations peuvent vous aider à planifier les mises à niveau.

AWS CLI les commandes et les opérations de l'API RDS renvoient les dates de début et de fin pour le support standard RDS et le support étendu RDS. Ces dates figurent également dans les tableaux des principales versions du moteur. Pour plus d'informations, consultez le calendrier Versions de MySQL majeures prises en charge sur Amazon RDS de publication d'Amazon RDS pour PostgreSQL dans les notes de mise à jour d'Amazon RDS pour PostgreSQL.

AWS CLI

Pour consulter les dates de début et de fin du support standard de RDS et du support étendu de RDS pour les principales versions de votre moteur à l'aide de la AWS CLI commande describe-db-major-engine-versions.

Cette commande renvoie les paramètres pertinents suivants pour les moteurs open source (MariaDB, MySQL et PostgreSQL). Il ne renvoie pas ces paramètres pour les moteurs commerciaux (Db2, SQL Server et Oracle).

- `SupportedEngineLifecycles`— Ce paramètre est un tableau d'objets comprenant `LifecycleSupportNameLifecycleSupportStartDate`, et`LifecycleSupportEndDate`.
- `LifecycleSupportName`— Ce paramètre indique le type de support dont bénéficie la version du moteur : support standard RDS (`open-source-rds-standard-support`) ou support étendu RDS (`open-source-rds-extended-support`). Pour MariaDB, ce paramètre renvoie uniquement le support standard RDS (). `open-source-rds-standard-support`
- `LifecycleSupportStartDate`— Ce paramètre indique la date de début du support standard de RDS ou du support étendu de RDS pour la version principale du moteur, en fonction de la valeur de. `LifecycleSupportName`
- `LifecycleSupportEndDate`— Ce paramètre indique la date de fin du support standard de RDS ou du support étendu de RDS pour la version principale du moteur, en fonction de la valeur de. `LifecycleSupportName`

Exemple

L'exemple de réponse affiche les informations suivantes :

- Les dates de début et de fin des cycles de vie des moteurs pris en charge `open-source-rds-standard-support` et de `open-source-rds-extended-support` MySQL 5.7. RDS Extended Support est disponible pour MySQL 5.7.
- Dates de début et de fin du cycle de vie du moteur pris en charge `open-source-rds-standard-support` pour MariaDB 10.6. Support étendu RDS n'est pas disponible pour MariaDB 10.6.
- Aucune information sur les cycles de vie des moteurs pris en charge pour SQL Server Enterprise Edition 13, car SQL Server est un moteur commercial.

```
{
    "DBMajorEngineVersions": [
        {
            "Engine": "mysql",
            "MajorEngineVersion": "5.7",
            "SupportedEngineLifecycles": [
                {
                    "LifecycleSupportName": "open-source-rds-standard-support",
                    "LifecycleSupportStartDate": "2016-02-22T00:00:00+00:00",
                    "LifecycleSupportEndDate": "2024-02-29T23:59:59.999000+00:00"
                },
                {
                    "LifecycleSupportName": "open-source-rds-extended-support",
                    "LifecycleSupportStartDate": "2024-03-01T00:00:00+00:00",
                    "LifecycleSupportEndDate": "2027-02-28T23:59:59.999000+00:00"
                }
            ]
        },
        {
            "Engine": "mariadb",
            "MajorEngineVersion": "10.6",
            "SupportedEngineLifecycles": [
                {
                    "LifecycleSupportName": "open-source-rds-standard-support",
                    "LifecycleSupportStartDate": "2022-02-03T00:00:00+00:00",
                    "LifecycleSupportEndDate": "2026-07-31T23:59:59.999000+00:00"
                }
            ]
        },
        {
            "Engine": "sqlserver-ee",
            "MajorEngineVersion": "13.00"
        },
        ...
    ]
}
```


API RDS

Pour consulter les dates de début et de fin du support standard RDS et du support étendu RDS pour les principales versions de votre moteur à l'aide de l'API RDS, utilisez l'opération Describe. DBMajor EngineVersions

Si le support étendu RDS est disponible pour une version du moteur, la réponse inclut le paramètre `SupportedEngineLifeCycles` sous la forme d'un tableau contenant deux objets. L'un des objets inclut les dates de début et de fin du support standard de RDS . Le deuxième objet inclut les dates de début et de fin du Support étendu RDS.

Si le support étendu RDS n'est pas disponible pour une version de moteur open source (MariaDB, MySQL et PostgreSQL), la réponse inclut uniquement le paramètre sous forme de tableau contenant un seul objet`SupportedEngineLifeCycles`. Cet objet inclut les dates de début et de fin du support standard RDS.

Si la version du moteur est destinée à un moteur commercial (Db2, SQL Server et Oracle), la réponse n'inclut pas le paramètre`SupportedEngineLifeCycles`.

Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Lorsque vous restaurez une instance de base de données ou un cluster de base de données multi-AZ, , sélectionnez Activer le support RDS étendu dans la console ou utilisez l'option Support étendu dans le AWS CLI ou le paramètre dans le RDSAPI. Lorsque vous inscrivez une instance de base de données ou un cluster de base de données multi-AZ, un cluster de base de au RDS support étendu, celui-ci est inscrit de façon permanente au RDS support étendu pendant toute la durée de vie de l'instance de base de données ou du cluster de base de données multi-AZ, du cluster de base de données Aurora global.

La valeur par défaut du paramètre Support RDS étendu varie selon que vous utilisez la console AWS CLI, le ou le RDS API pour restaurer votre base de données. Si vous utilisez la console, que vous ne sélectionnez pas Activer le support RDS étendu et que la version principale du moteur que vous restaurez a dépassé la fin du support standard d'RDS, alors met RDS automatiquement à niveau votre instance de base de données vers une version du moteur plus récente. Si vous utilisez le AWS CLI ou le RDS API et que vous ne spécifiez pas le paramètre Support RDS étendu, Amazon active RDS par défaut le Support RDS étendu. Lorsque vous automatisez à l'aide de services AWS CloudFormationou d'autres services, ce comportement par défaut maintient la disponibilité de votre base de données après la date de fin du support standard d'RDS. Vous pouvez désactiver le Support RDS étendu en utilisant le AWS CLI ou le RDSAPI.

Rubriques

- RDSComportement du Support étendu
- Considérations relatives RDS au Support étendu
- Restaurez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support

RDSComportement du Support étendu

Le tableau suivant résume ce qui se passe lorsqu'une version majeure du moteur d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un que vous restaurez atteint la fin du support standard d'RDS.

RDSStatut de support étendu*	Attitude
Activées	Amazon vous RDS facture le Support RDS étendu.
Désactivé**	Une fois la restauration terminée, Amazon met RDS automatiquement à niveau votre instance de base de données ou votre cluster de base de données multi-AZ, votre vers une version de moteur plus récente (dans une future fenêtre de maintenance).

* Dans la RDS console, le statut Support RDS étendu apparaît comme Oui ou Non. Dans le bloc AWS CLI opératoire RDSAPI, le statut Support RDS étendu apparaît sous la forme `open-source-rds-extended-support` ou`open-source-rds-extended-support-disabled`.

** Cette option n'est disponible que lors de la restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ exécutant Postgre SQL 12 ou supérieur ou My SQL 8 et supérieur.

Considérations relatives RDS au Support étendu

Avant de restaurer une instance de base de données ou un cluster de base de données multi-AZ, un cluster , prenez en compte les éléments suivants :

- Une fois la date de fin du support standard d'RDS passée, si vous souhaitez restaurer une instance de base de données ou un cluster de base de données multi-AZ, un cluster depuis

Amazon S3, vous ne pouvez le faire qu'en utilisant le AWS CLI ou le RDSAPI. Utilisez l'`--engine-lifecycle-support`option de la AWS CLI commande restore-db-cluster-from-s3 ou le `EngineLifecycleSupport` paramètre de l'opération R estoreDBCluster fromS3. RDS API

- Si vous souhaitez empêcher RDS de restaurer vos bases de données vers des versions de Support RDS étendu, spécifiez `open-source-rds-extended-support-disabled` dans le AWS CLI ou le RDSAPI. Ce faisant, vous éviterez les frais de Support RDS étendu associés.

  Si vous spécifiez ce paramètre, Amazon RDS Amazon mettra automatiquement à niveau votre base de données restaurée vers une version majeure plus récente et prise en charge. Si la mise à niveau échoue aux vérifications préalables à la mise à niveau, RDSAmazon reviendra en toute sécurité à la version du moteur RDS Extended Support. Cette base de données restera en mode Support RDS étendu, et Amazon RDS Amazon vous facturera le support RDS étendu jusqu'à ce que vous mettiez manuellement votre base de données à niveau.

  Par exemple, si vous restaurez un instantané My SQL 5.7 sans utiliser RDS Extended Support, Amazon RDS essaiera de mettre automatiquement à niveau votre base de données vers My SQL 8.0. Si cette mise à niveau échoue en raison d'un problème que vous devez résoudre, Amazon RDS rétablira la base de données vers My SQL 5.7. Amazon RDS conservera la base de données sur RDS Extended Support jusqu'à ce que vous puissiez résoudre le problème. Par exemple, une mise à niveau peut échouer en raison d'un espace de stockage insuffisant. Une fois le problème résolu, vous devez lancer la mise à niveau. Après la première tentative de mise à niveau de votre base de données, Amazon RDS ne tentera plus de la mettre à niveau.
- RDSLe Support étendu est défini au niveau du cluster. Les membres d'un cluster auront toujours le même paramètre pour le Support RDS étendu dans la RDS console, `--engine-lifecycle-support` dans le AWS CLI et `EngineLifecycleSupport` dans le RDSAPI.

Pour plus d'informations, consultez Versions MySQL et publiez les calendriers d'Amazon RDS pour SQL Postgre.

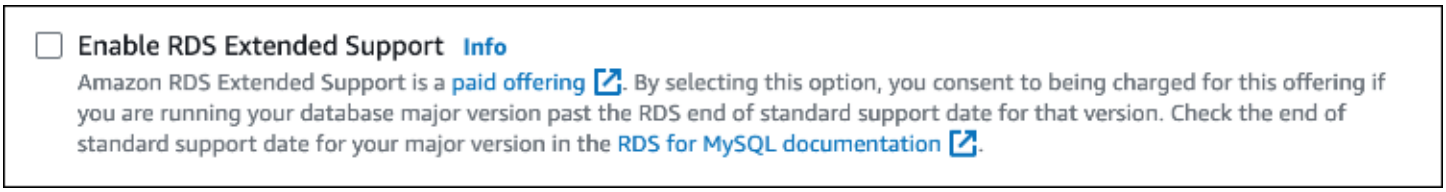
Restaurez une instance de base de données ou un cluster de base de données multi-AZ, un cluster avec RDS Extended Support

Vous pouvez restaurer une instance de base de données ou un cluster de base de données multi-AZ, avec une version de support RDS étendu en utilisant le AWS Management Console AWS CLI, le ou le RDSAPI.

Console

Lorsque vous restaurez , une instance de base de données ou un cluster de base de données multi-AZ, sélectionnez Activer le support RDS étendu dans la section des options du moteur. Si vous ne sélectionnez pas ce paramètre et que la version principale du moteur que vous restaurez a dépassé la fin du support standard d'RDS, RDSAmazon votre cluster de base de données multi-AZ vers une version compatible avec le support standard d'RDS.

L'image suivante montre le paramètre Activer le support RDS étendu :



AWS CLI

Lorsque vous exécutez la commande -db-snapshot ou restore-db-cluster-from-snapshot AWS CLI , sélectionnez RDS Support étendu en spécifiant l'option. `open-source-rds-extended-support --engine-lifecycle-support`

Si vous souhaitez éviter les frais associés au Support RDS étendu, définissez l'`--engine-lifecycle-support`option sur`open-source-rds-extended-support-disabled`. Par défaut, cette option est définie sur`open-source-rds-extended-support`.

Vous pouvez également spécifier cette valeur à l'aide des AWS CLI commandes suivantes :

- restore-db-cluster-from-s3
- restore-db-cluster-to-point-in-time
- restore-db-instance-from-s3
- restore-db-instance-to-point-in-time

RDS API

Lorsque vous utilisez l'RDSAPIopération R estoreDBInstance F romDBSnapshot ou R estoreDBCluster FromSnapshot Amazon, sélectionnez Support RDS étendu en définissant le `EngineLifecycleSupport` paramètre sur`open-source-rds-extended-support`.

Si vous souhaitez éviter les frais associés au Support RDS étendu, définissez le `EngineLifecycleSupport` paramètre sur`open-source-rds-extended-support-disabled`. Par défaut, ce paramètre est défini sur `open-source-rds-extended-support`.

Vous pouvez également spécifier cette valeur à l'aide RDS API des opérations suivantes :

- R à estoreDBCluster partir de S3
- RestoreDBClusterToPointInTime
- R à estoreDBInstance partir de S3
- RestoreDBInstanceToPointInTime

Pour plus d'informations sur la restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, suivez les instructions relatives à votre moteur de base de données dansRestauration vers une instance de base de données.

Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données

Un déploiement bleu/vert copie un environnement de base de données de production vers un environnement intermédiaire séparé et synchronisé. En utilisant Amazon RDS Blue/Green Deployments, vous pouvez apporter des modifications à la base de données dans l'environnement intermédiaire sans affecter l'environnement de production. Par exemple, vous pouvez mettre à niveau la version principale ou mineure du moteur de base de données ou modifier les paramètres de base de données dans l'environnement intermédiaire. Lorsque vous serez prêt, vous pourrez faire de l'environnement intermédiaire le nouvel environnement de base de données de production, avec des temps d'arrêt généralement inférieurs à une minute.

> **Note**
>
> Actuellement, les déploiements bleu/vert sont pris en charge pour RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL uniquement. Pour connaître la disponibilité d'Amazon Aurora, consultez la section Utilisation des déploiements Amazon Aurora Blue/Green pour les mises à jour des bases de données dans le guide de l'utilisateur Amazon Aurora.

Rubriques

- Présentation des déploiements Amazon RDS (Blue/Green)
-
-
-
-

Présentation des déploiements Amazon RDS (Blue/Green)

En utilisant Amazon RDS Blue/Green Deployments, vous pouvez apporter et tester des modifications de base de données avant de les implémenter dans un environnement de production. Un déploiement bleu/vert crée un environnement intermédiaire qui copie l'environnement de production. Dans un déploiement bleu/vert, l'environnement bleu est l'environnement de production actuel. L'environnement écologique est l'environnement de préparation et reste synchronisé avec l'environnement de production actuel.

Vous pouvez apporter des modifications aux instances de base de données RDS dans l'environnement vert sans affecter les charges de travail de production. Par exemple, vous pouvez mettre à niveau la version majeure ou mineure du moteur de base de données, mettre à niveau la configuration du système de fichiers sous-jacent, ou modifier les paramètres de la base de données dans l'environnement intermédiaire. Vous pouvez tester en profondeur les changements dans l'environnement vert. Lorsque vous êtes prêt, vous pouvez passer d'un environnement à un autre pour faire de l'environnement vert le nouvel environnement de production. La commutation prend généralement moins d'une minute, sans perte de données et sans qu'il soit nécessaire de modifier les applications.

Comme l'environnement vert est une copie de la topologie de l'environnement de production, il inclut les fonctionnalités utilisées par l'instance de base de données. Ces fonctionnalités comprennent les réplicas en lecture, la configuration du stockage, les instantanés de base de données, les sauvegardes automatiques, Performance Insights et la surveillance améliorée. Si l'instance de base de données bleue est un déploiement d'instance de base de données multi-AZ, alors l'instance de base de données verte est également un déploiement d'instance de base de données multi-AZ.

> **Note**
>
> Actuellement, blue/green les déploiements ne sont pris en charge que pour RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL. Pour connaître la disponibilité d'Amazon Aurora, consultez la présentation des déploiements Amazon Aurora Blue/Green dans le guide de l'utilisateur Amazon Aurora.
> Dans certaines conditions, RDS pour PostgreSQL utilise la réplication logique au lieu de la réplication physique pour synchroniser l'environnement vert avec l'environnement bleu. Pour de plus amples informations, veuillez consulter the section called “Méthodes de SQL réplication Postgre”.

Rubriques

- Disponibilité des régions et des versions
- Avantages de l'utilisation des déploiements Amazon RDS Blue/Green
- Flux de travail d'un blue/green déploiement
- Autoriser l'accès aux opérations de déploiement bleu/vert d'Amazon RDS
- Limites et considérations relatives aux déploiements Amazon RDS (blue/green)
- Bonnes pratiques pour les déploiements bleu/vert d'Amazon RDS
- Méthodes de SQL réplication Postgre pour les déploiements bleu/vert

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour de plus amples informations, veuillez consulter the section called “Déploiements bleu/vert”.

Avantages de l'utilisation des déploiements Amazon RDS Blue/Green

En utilisant les Blue/Green déploiements Amazon RDS, vous pouvez rester au courant des correctifs de sécurité, améliorer les performances des bases de données et adopter de nouvelles fonctionnalités de base de données avec des temps d'arrêt courts et prévisibles. Blue/green les déploiements réduisent les risques et les temps d'arrêt liés aux mises à jour de base de données, telles que les mises à niveau majeures ou mineures des versions du moteur.

Les déploiements bleu/vert offrent les avantages suivants :

- Créez facilement un environnement intermédiaire prêt pour la production.
- Répliquez automatiquement les modifications apportées aux bases de données de l'environnement de production à l'environnement intermédiaire.
- Testez les modifications apportées aux bases de données dans un environnement intermédiaire sûr sans affecter l'environnement de production.
- Restez à jour des correctifs de base de données et des mises à jour du système.
- Mettez en œuvre et testez les nouvelles fonctionnalités de base de données.
- Basculez votre environnement intermédiaire pour en faire le nouvel environnement de production sans modifier votre application.

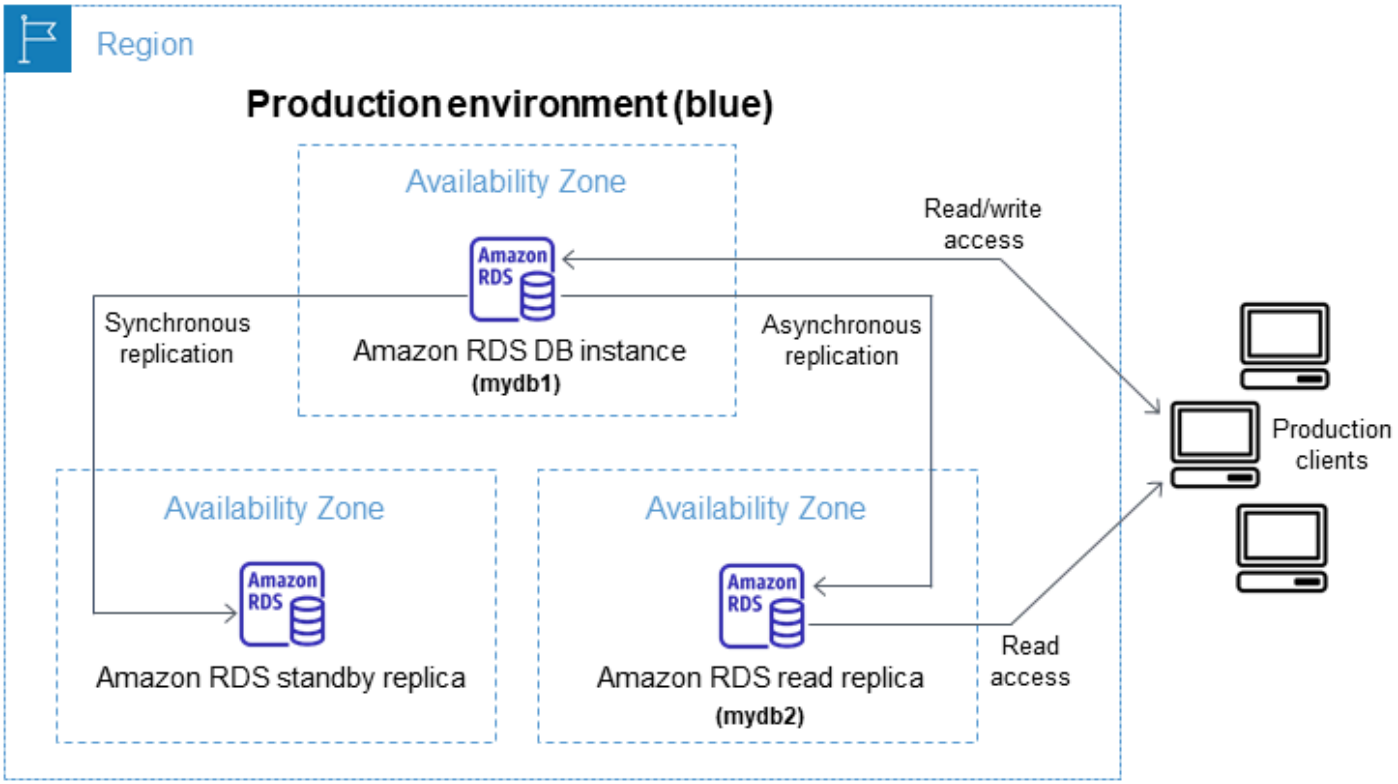
- Basculez en toute sécurité grâce aux barrières de protection de commutation intégrées.
- Éliminez les pertes de données pendant la commutation.
- Basculez rapidement, généralement en moins d'une minute en fonction de votre charge de travail.

Flux de travail d'un blue/green déploiement

Effectuez les étapes principales suivantes lorsque vous utilisez un blue/green déploiement pour des mises à jour de base de données.

1. Identifiez un environnement de production qui nécessite des mises à jour.

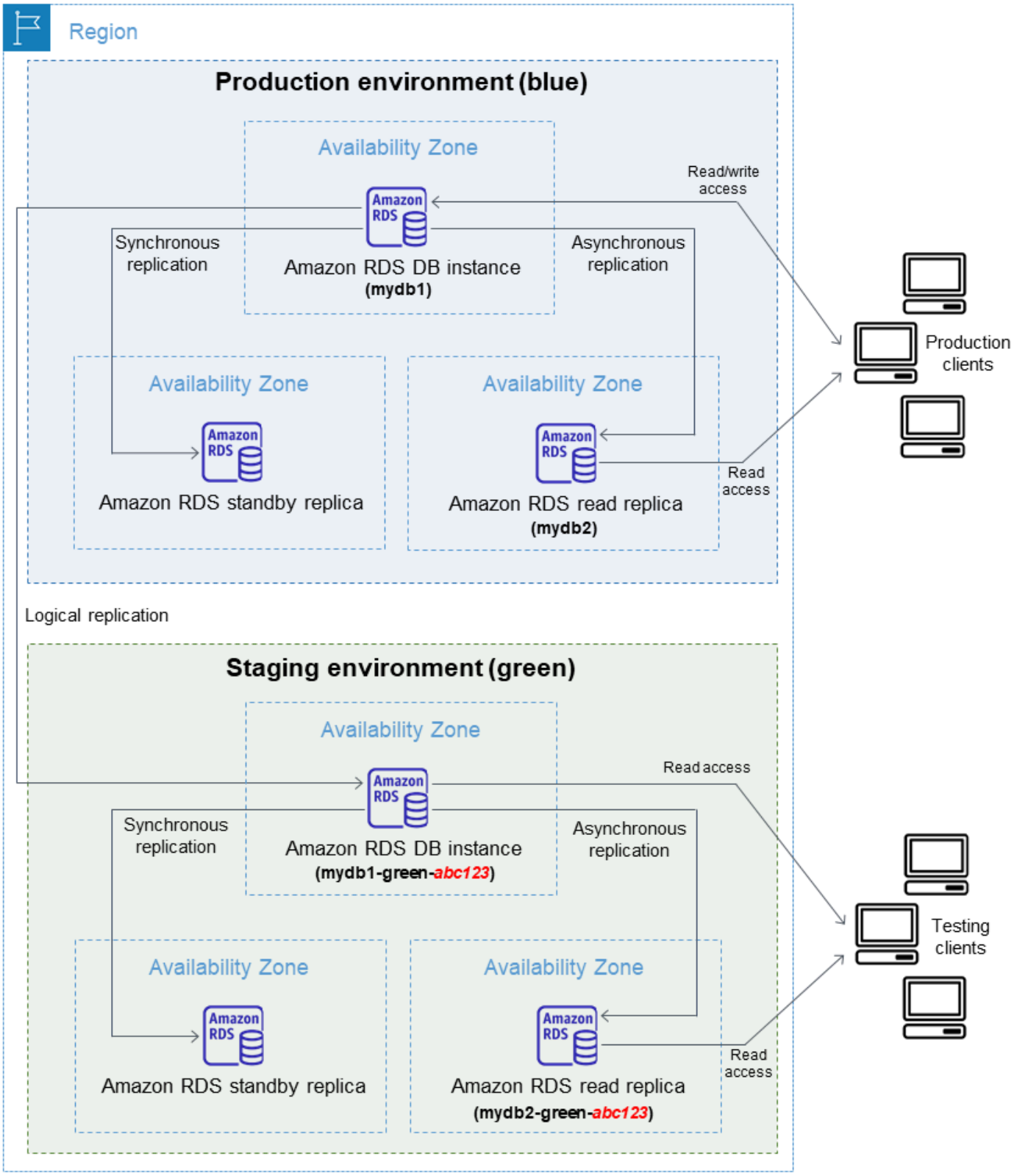
   Par exemple, l'environnement de production dans cette image comporte un déploiement d'instance de base de données multi-AZ (mydb1) et un réplica en lecture (mydb2).



2. Créez le blue/green déploiement. Pour obtenir des instructions, veuillez consulter .

   L'image suivante montre un exemple de blue/green déploiement de l'environnement de production à partir de l'étape 1. Lors de la création du blue/green déploiement, RDS copie la topologie et la configuration complètes de l'instance de base de données principale pour créer l'environnement écologique. Les noms des instances de base de données copiées sont complétés par `-green-`*`random-characters`*. L'environnement intermédiaire de l'image contient un déploiement

d'instance de base de données multi-AZ (mydb1-green-*abc123*) et une réplique de lecture (mydb2-green-). *abc123*



Lorsque vous créez le blue/green déploiement, vous pouvez mettre à niveau la version de votre moteur de base de données et spécifier un groupe de paramètres de base de données différent pour les instances de base de données dans l'environnement vert. RDS configure également la réplication de l'instance de base de données principale dans l'environnement bleu vers l'instance de base de données principale dans l'environnement vert.

Une fois le blue/green déploiement créé, l'instance de base de données dans l'environnement vert est en lecture seule par défaut.

3. Apportez des modifications supplémentaires à l'environnement intermédiaire, si nécessaire. Par exemple, vous pouvez modifier la classe d'instance de base de données utilisée par une ou plusieurs instances de base de données dans l'environnement vert.

   Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

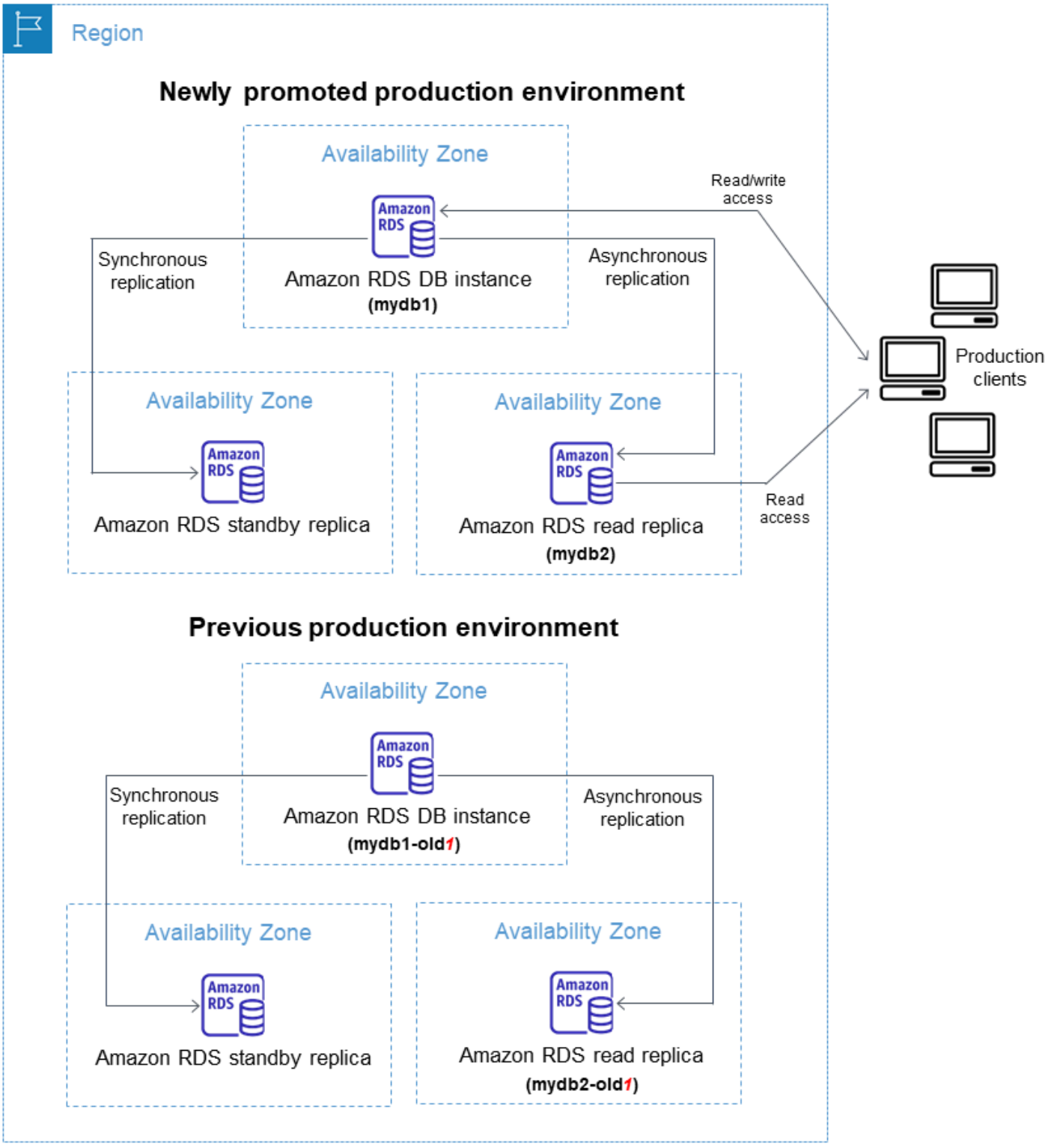
4. Testez votre environnement intermédiaire.

   Pendant les tests, nous vous recommandons de garder vos bases de données dans l'environnement vert en lecture seule. Activez les opérations d'écriture dans un environnement vert avec prudence, car elles peuvent entraîner des conflits de réplication. Elles peuvent également entraîner la présence de données involontaires dans les bases de données de production après la commutation. Pour activer les opérations d'écriture pour RDS pour MySQL, définissez le `read_only` paramètre sur `1` et attendez que le groupe de paramètres soit synchronisé. Comme `read_only` il s'agit d'un paramètre dynamique, aucun redémarrage n'est nécessaire. Une fois la synchronisation effectuée, passez `read_only` de `1` à. `0` Pour les déploiements RDS pour PostgreSQL qui utilisent la réplication logique, définissez `default_transaction_read_only` le paramètre `off` sur au niveau de la session. Pour ceux qui utilisent la réplication physique, vous ne pouvez pas activer les opérations d'écriture dans un environnement écologique.

5. Lorsque vous êtes prêt, passez à l'environnement de préparation au nouvel environnement de production. Pour obtenir des instructions, veuillez consulter .

   La commutation entraîne un temps d'arrêt. Le temps d'arrêt est généralement inférieur à une minute, mais il peut être plus long en fonction de votre charge de travail.

   L'image suivante présente les instances de base de données après la commutation.



Après la commutation, les instances de base de données qui se trouvaient dans l'environnement vert deviennent les nouvelles instances de base de données de production. Les noms et les points de terminaison de l'environnement de production actuel sont attribués à l'environnement de

production nouvellement basculé, ce qui ne nécessite aucune modification de votre application. En conséquence, votre trafic de production s'écoule désormais vers le nouvel environnement de production. Les instances de base de données dans l'environnement bleu précédent sont renommées en ajoutant `-old`*n* au nom actuel, où *n* est un numéro. Par exemple, supposons que le nom de l'instance de base de données dans l'environnement bleu est `mydb1`. Après le basculement, le nom de l'instance de base de données sera. `mydb1-old1`

Dans l'exemple de l'image, les changements suivants se produisent pendant la commutation :

- Le déploiement de l'instance de base de données multi-AZ de l'environnement vert nommé `mydb1-green-abc123` devient le déploiement de l'instance de base de données multi-AZ de production nommé `mydb1`.
- Le réplica en lecture nommé `mydb2-green-abc123` de l'environnement vert devient le réplica en lecture `mydb2` de l'environnement de production.
- Le déploiement de l'instance de base de données multi-AZ nommée `mydb1` de l'environnement bleu devient `mydb1-old1`.
- Le réplica en lecture nommé `mydb2` de l'environnement bleu devient `mydb2-old1`.

6. Si vous n'avez plus besoin d'un blue/green déploiement, vous pouvez le supprimer. Pour obtenir des instructions, veuillez consulter .

   Après la commutation, l'environnement de production précédent n'est pas supprimé afin que vous puissiez l'utiliser pour les tests de régression, si nécessaire.

Autoriser l'accès aux opérations de déploiement bleu/vert d'Amazon RDS

Les utilisateurs doivent disposer des autorisations requises pour effectuer les opérations liées aux déploiements bleu/vert. Vous pouvez créer des politiques IAM qui accordent aux utilisateurs et aux rôles l'autorisation d'effectuer des opérations d'API spécifiques sur les ressources spécifiées dont ils ont besoin. Vous pouvez ensuite attacher ces politiques aux jeux d'autorisations ou rôles IAM qui requièrent ces autorisations. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.

L'utilisateur qui crée un déploiement bleu/vert doit avoir les autorisations nécessaires pour effectuer les opérations RDS suivantes :

- `rds:CreateBlueGreenDeployment`
- `rds:AddTagsToResource`

- `rds:CreateDBInstanceReadReplica`

L'utilisateur qui bascule un déploiement bleu/vert doit avoir les autorisations nécessaires pour effectuer les opérations RDS suivantes :

- `rds:SwitchoverBlueGreenDeployment`
- `rds:ModifyDBInstance`
- `rds:PromoteReadReplica`

L'utilisateur qui supprime un déploiement bleu/vert doit avoir les autorisations nécessaires pour effectuer une ou plusieurs opérations RDS suivantes :

- `rds:DeleteBlueGreenDeployment`
- `rds:DeleteDBInstance`

Amazon RDS met en service et modifie les ressources dans l'environnement intermédiaire en votre nom. Ces ressources incluent des instances de base de données qui utilisent une convention de dénomination définie en interne. Par conséquent, les politiques IAM jointes ne peuvent pas contenir de modèles de noms de ressources partiels tels que`my-db-prefix-*`. Seuls les caractères génériques (*) sont pris en charge. En général, nous recommandons d'utiliser des balises de ressources et d'autres attributs pris en charge pour contrôler l'accès à ces ressources, plutôt que des caractères génériques. Pour plus d'informations, consultez Actions, ressources et clés de condition pour Amazon RDS.

Limites et considérations relatives aux déploiements Amazon RDS (blue/green)

Les déploiements bleu/vert dans Amazon RDS nécessitent une prise en compte attentive de facteurs tels que les emplacements de réplication, la gestion des ressources, le dimensionnement des instances et les impacts potentiels sur les performances de la base de données. Les sections suivantes fournissent des conseils pour vous aider à optimiser votre stratégie de déploiement afin de garantir des temps d'arrêt minimaux, des transitions fluides et une gestion efficace de votre environnement de base de données.

Rubriques

- Limitations relatives aux blue/green déploiements

- Considérations relatives aux blue/green déploiements

Limitations relatives aux blue/green déploiements

Les limitations suivantes s'appliquent aux blue/green déploiements.

Rubriques

- Limitations générales pour les déploiements bleu/vert
- Limitations d'RDS pour MySQL pour les déploiements blue/green
- Limitations blue/green de RDS pour PostgreSQL pour les déploiements avec réplication physique
- Limitations d' RDS pour PostgreSQL pour les déploiements avec réplication logique blue/green

Limitations générales pour les déploiements bleu/vert

Les limitations générales suivantes s'appliquent aux blue/green déploiements :

- Les déploiements bleu/vert ne prennent pas en charge la gestion des mots de passe des utilisateurs principaux avec. AWS Secrets Manager
- Si le volume de journal dédié (DLV) est activé sur la base de données bleue, il doit être activé sur toutes les instances de base de données, y compris les répliques de lecture.
- Lors de la commutation, les environnements bleu et vert ne peuvent pas avoir d'intégrations zéro ETL avec Amazon Redshift. Vous devez d'abord supprimer l'intégration et basculer, puis recréer l'intégration.
- Le planificateur d'événements (`event_scheduler`paramètre) doit être désactivé dans l'environnement vert lorsque vous créez un blue/green déploiement. Cela évite que des événements soient générés dans l'environnement vert et provoquent des incohérences.
- Vous ne pouvez pas transformer une instance de base de données non chiffrée en une instance de base de données chiffrée.
- Vous ne pouvez pas remplacer un d'instances de base de données bleu par une version de moteur supérieure à celle du d'instances de base de données vert correspondant.
- Les ressources de l'environnement bleu et de l'environnement vert doivent se trouver dans le même Compte AWS.
- Les déploiements bleu/vert ne sont pas pris en charge pour les fonctionnalités suivantes :
  - Proxy Amazon RDS
  - Réplicas en lecture en cascade

- Réplicas en lecture entre Régions
- AWS CloudFormation
- Déploiements de clusters de bases de données Multi-AZ

  Les déploiements bleu/vert sont pris en charge pour les déploiements d'instances de base de données multi-AZ. Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Limitations d'RDS pour MySQL pour les déploiements blue/green

Les limitations suivantes s'appliquent aux blue/green déploiements RDS pour MySQL :

- Le d'instances DB bleu ne peut pas être une réplique externe du journal binaire.
- Si la base de données source est associée à un groupe d'options personnalisé, vous ne pouvez pas spécifier de mise à niveau de version majeure lorsque vous créez le blue/green déploiement.

  Dans ce cas, vous pouvez créer un blue/green déploiement sans spécifier de mise à niveau de version majeure. Ensuite, vous pouvez mettre à niveau la base de données dans l'environnement vert. Pour de plus amples informations, veuillez consulter Mise à niveau de la version du moteur d'une instance de base de données.
- Les déploiements bleu/vert ne prennent pas en charge le pilote AWS JDBC pour MySQL. Pour plus d'informations, consultez la section Limitations connues sur GitHub.

Limitations blue/green de RDS pour PostgreSQL pour les déploiements avec réplication physique

Les limitations suivantes s'appliquent aux déploiements RDS pour PostgreSQL qui blue/green utilisent la réplication physique. Pour savoir dans quels cas blue/green les déploiements utilisent la réplication physique au lieu de la réplication logique, consultezthe section called “Méthodes de SQL réplication Postgre”.

- Une fois l'environnement écologique créé, vous ne pouvez pas effectuer de mise à niveau manuelle de la version majeure.
- Les déploiements bleu/vert qui utilisent la réplication physique ne prennent pas en charge les modifications de schéma dans l'environnement vert, car celui-ci est strictement en lecture seule.
- L'instance de base de données bleue ne peut pas être une source logique (éditeur) ou une réplique (abonné).

Limitations d' RDS pour PostgreSQL pour les déploiements avec réplication logique blue/green

Les limitations suivantes s'appliquent aux déploiements RDS pour PostgreSQL qui utilisent la réplication logique. Pour savoir dans quels cas blue/green les déploiements utilisent la réplication logique au lieu de la réplication physique, consultezthe section called “Méthodes de SQL réplication Postgre”.

- Les tables non enregistrées ne sont pas répliquées dans l'environnement vert .
- Le d'instances DB bleu ne peut pas être une source logique (éditeur) ou une réplique (abonné).
- Si l'instance de base de données bleue est configurée en tant que serveur externe d'une extension de l'encapsuleur de données externes (FDW), vous devez utiliser le nom du point de terminaison de l'instance au lieu des adresses IP. Ainsi, la configuration reste fonctionnelle après la commutation.
- Lors d'un blue/green déploiement, chaque base de données nécessite un emplacement de réplication logique. À mesure que le nombre de bases de données augmente, la surcharge en ressources augmente et peut potentiellement entraîner un retard de réplication, en particulier si l'instance de base de de bases de données n'est pas suffisamment dimensionnée. L'impact dépend de facteurs tels que la charge de travail de la base de données et le nombre de connexions. Pour atténuer ce problème, pensez à étendre votre classe d'instance de base de données ou à réduire le nombre de bases de données sur l'instance de source.
- Le processus d'application de la réplication logique dans un environnement écologique est monothread. Si l'environnement bleu génère un volume élevé de trafic d'écriture, l'environnement vert risque de ne pas être en mesure de suivre le rythme. Cela peut entraîner un retard ou un échec de réplication, en particulier pour les charges de travail qui produisent un débit d'écriture élevé en continu. Assurez-vous de tester minutieusement vos charges de travail. Pour les scénarios nécessitant des mises à niveau de version majeures et la gestion de charges de travail d'écriture de volumes élevés, envisagez d'autres approches telles que l'utilisation de AWS Database Migration Service (AWS DMS) .
- La création de nouvelles partitions sur des tables partitionnées n'est pas prise en charge lors des déploiements bleu/vert pour . La création de nouvelles partitions implique des opérations de langage de définition de données (DDL) telles que `CREATE TABLE` celles qui ne sont pas répliquées de l'environnement bleu vers l'environnement vert. Cependant, les tables partitionnées existantes et leurs données seront répliquées dans l'environnement vert.
- Les limitations suivantes s'appliquent aux extensions PostgreSQL :

- L'`pg_partman`extension doit être désactivée dans l'environnement bleu lorsque vous créez un blue/green déploiement. L'extension exécute des opérations DDL comme `CREATE TABLE`, qui interrompent la réplication logique de l'environnement bleu vers l'environnement vert.
- L'`pg_cron`extension doit rester désactivée sur toutes les bases de données vertes après la création du blue/green déploiement. L'extension dispose d'exécutants en arrière-plan qui s'exécutent en tant que superutilisateur et contournent le paramètre de lecture seule de l'environnement vert, ce qui peut provoquer des conflits de réplication.
- Les `pgactive` extensions `pglogical` et doivent être désactivées dans l'environnement bleu lorsque vous créez un blue/green déploiement. Après avoir remplacé l'environnement vert par le nouvel environnement de production, vous pouvez réactiver les extensions. En outre, la base de données bleue ne peut pas être un abonné logique d'une instance externe.
- Si vous utilisez l'`pgAudit`extension, elle doit rester dans les bibliothèques partagées (`shared_preload_libraries`) sur les groupes de paramètres de base de données personnalisés pour les instances de base de données bleues et vertes. Pour de plus amples informations, veuillez consulter the section called “Configuration de l' pgAudit extension”.

Limitations spécifiques à la réplication logique pour les déploiements blue/green

PostgreSQL impose certaines restrictions liées à la réplication logique, qui se traduisent par des limitations lors de la création de blue/green déploiements pour les clusters de bases de données .

Le tableau suivant décrit les limitations de réplication logique qui s'appliquent aux déploiements bleu/vert pour RDS for PostgreSQL. Pour plus d'informations, consultez Restrictions dans la documentation Réplication logique PostgreSQL.

Limitation	Explication
Les instructions DDL (Langage de définition de données), comme `CREATE TABLE` et `CREATE SCHEMA`, ne sont pas répliquées de l'environnement	Si Amazon RDS détecte une modification DDL dans l'environnement bleu, vos bases de données vertes entrent dans un état de réplication dégradée. Vous devez supprimer le blue/green déploiement et toutes les bases de données vertes, puis les recréer.

Limitation	Explication
bleu vers l'environnement vert.	
Les instructions du langage de contrôle des données (DCL), telles que `GRANT` et `REVOKE`, ne sont pas répliquées de l'environnement bleu vers l'environnement vert.	Si Amazon RDS PostgreSQL détecte une tentative d'exécution d'une instruction DCL dans l'environnement bleu, un message d'avertissement s'affiche. Aucune configuration ou API n'est disponible pour modifier ce comportement, car il s'agit d'une limitation du processus de blue/green déploiement.
Les opérations `NEXTVAL` sur les objets de séquence ne sont pas synchronisées entre l'environnement bleu et l'environnement vert.	Pendant la commutation, Amazon RDS incrémente les valeurs de séquence dans l'environnement vert pour les faire correspondre à celles dans l'environnement bleu. Si vous avez des milliers de séquences, cela peut retarder la commutation.

Limitation	Explication
Les objets de grande taille situés dans l'environnement bleu ne sont pas répliqués dans l'environnement vert. Cela inclut à la fois les grands objets existants et tous les grands objets nouvellement créés ou modifiés au cours du processus de blue/green déploiement.	Si Amazon RDS détecte dans l'environnement bleu la création ou la modification d'objets volumineux qui sont stockés dans la table système `pg_largeobject` , vos bases de données vertes entrent dans un état de réplication dégradée. Vous devez supprimer le blue/green déploiement et toutes les bases de données vertes, puis les recréer.
Les vues matérialisées ne sont pas automatiquement actualisées dans l'environnement vert.	L'actualisation des vues matérialisées dans l'environnement bleu n'actualise pas les vues dans l'environnement vert. Après le basculement, vous pouvez les actualiser manuellement à l'aide de la commande REFRESH MATERIALIZED VIEW ou planifier une actualisation.
Les opérations UPDATE et DELETE ne sont pas autorisées sur les tables dépourvues de clé primaire.	Avant de créer un blue/green déploiement, assurez-vous que toutes les tables disposent d'une clé primaire ou d'une utilisation`REPLICA IDENTITY FULL`. Toutefois, ne l'utilisez que `REPLICA IDENTITY FULL` s'il n'existe aucune clé primaire ou unique, car cela affecte les performances de réplication. Pour de plus amples informations, veuillez consulter la documentation sur PostgreSQL.

Considérations relatives aux blue/green déploiements

Amazon RDS suit les ressources lors des blue/green déploiements à l'aide de la `DbiResourceId` de chaque ressource. Cet identifiant de ressource est un identifiant Région AWS unique et immuable pour la ressource.

L'ID de ressource est distinct de l'ID d'instance de de base de données. Chacun d'entre eux est répertorié dans la configuration de base de données de la console RDS.

Le nom (ID d'instance) d'une ressource change lorsque vous changez de blue/green déploiement, mais chaque ressource conserve le même ID de ressource. Par exemple, l'identifiant d'une instance de base de données peut être `mydb` dans l'environnement bleu. Après la commutation, la même instance de base de données peut être renommée en `mydb-old1`. Cependant, l'ID de ressource de l'instance de base de données ne change pas pendant la commutation. Ainsi, lorsque vous remplacez les ressources vertes par les nouvelles ressources de production, leur ressource IDs ne correspond pas à la ressource bleue IDs qui était auparavant en production.

Après avoir transféré un blue/green déploiement, envisagez de mettre à jour la ressource en IDs fonction de celles des ressources de production récemment transférées pour les fonctionnalités et services intégrés que vous avez utilisés avec les ressources de production. Plus précisément, envisagez les mises à jour suivantes :

- Si vous effectuez un filtrage à l'aide de l'API et de la ressource RDS IDs, ajustez la ressource IDs utilisée pour le filtrage après le passage au numérique.
- Si vous l'utilisez CloudTrail pour auditer des ressources, ajustez les consommateurs de CloudTrail afin de suivre la nouvelle ressource IDs après le passage au numérique. Pour de plus amples informations, veuillez consulter Surveillance (Amazon RDS API appelle) AWS CloudTrail.
- Si vous utilisez l'API Performance Insights, ajustez la ressource IDs dans les appels à l'API après le passage au numérique. Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

  Vous pouvez surveiller une base de données avec le même nom après la commutation, mais elle ne contient pas les données d'avant la commutation.
- Si vous utilisez des ressources IDs dans les politiques IAM, assurez-vous d'ajouter la ressource IDs des ressources récemment transférées lorsque cela est nécessaire. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.

- Si des rôles IAM sont associés à votre instance de base de données de de base de données, veillez à les réassocier après le passage au numérique. Les rôles attachés ne sont pas automatiquement copiés dans l'environnement vert.
- Si vous vous authentifiez auprès de votre instance de base de données à l'aide de l'authentification de base de données IAM, veillez à ce que la politique IAM utilisée pour accéder à la base de données contienne à la fois les bases de données bleues et vertes répertoriées sous l'élément `Resource` de la politique. Cela est nécessaire pour se connecter à la base de données verte après la commutation. Pour de plus amples informations, veuillez consulter the section called “Création et utilisation d'une politique IAM pour l'accès à une base de données IAM”.
- Si vous avez l'habitude AWS Backup de gérer des sauvegardes automatisées des ressources dans un déploiement bleu/vert, ajustez la ressource IDs utilisée AWS Backup après le passage au numérique. Pour de plus amples informations, veuillez consulter Utilisation AWS Backup pour gérer les sauvegardes automatisées pour Amazon RDS.
- Si vous souhaitez restaurer un instantané de base de données manuel ou automatique pour une instance de base de données faisant partie d'un blue/green déploiement, assurez-vous de restaurer le cliché de base de données correct en examinant l'heure à laquelle le cliché a été pris. Pour de plus amples informations, veuillez consulter Restauration vers une instance de base de données.
- Si vous voulez décrire une sauvegarde automatisée de l'instance de base de données précédente de l'environnement bleu ou la restaurer à un moment donné, utilisez l'ID de ressource pour l'opération.

  Comme le nom de l'instance de base de données change pendant la commutation, vous ne pouvez pas utiliser son nom précédent pour les opérations `DescribeDBInstanceAutomatedBackups` ou `RestoreDBInstanceToPointInTime`.

  Pour de plus amples informations, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.
- Lorsque vous ajoutez un réplica en lecture à une instance de base de données dans l'environnement vert d'un déploiement bleu/vert, le nouveau réplica en lecture ne remplacera pas un réplica en lecture dans l'environnement bleu lors du basculement. Cependant, le nouveau réplica en lecture est conservé dans le nouvel environnement de production après la commutation.
- Après le changement, AWS Database Migration Service (AWS DMS) les tâches de réplication ne peuvent pas reprendre car le point de contrôle de l'environnement bleu n'est pas valide dans l'environnement vert. Vous devez recréer la tâche DMS avec un nouveau point de contrôle pour poursuivre la réplication.

- Lorsque vous supprimez une instance de base de données dans l'environnement écologique d'un blue/green déploiement, vous ne pouvez pas créer une nouvelle instance de base de données pour la remplacer dans le blue/green déploiement.

  Si vous créez une nouvelle instance de base de données avec le même nom et le même Amazon Resource Name (ARN) que l'instance de base de données supprimée, elle a une valeur `DbiResourceId` différente, de sorte qu'elle ne fait pas partie de l'environnement vert.

  Le comportement suivant survient si vous supprimez une instance de base de données dans l'environnement vert :

  - Si l'instance de base de données dans l'environnement bleu avec le même nom existe, elle ne sera pas basculée vers l'instance de base de données dans l'environnement vert. Cette instance de base de données ne sera pas renommée en ajoutant `-old`*n* au nom de l'instance de base de données.
  - Toute application qui pointe vers l'instance de base de données dans l'environnement bleu continue à utiliser la même instance de base de données après la commutation.

  Le même comportement s'applique aux instances de base de données et aux réplicas en lecture.

Bonnes pratiques pour les déploiements bleu/vert d'Amazon RDS

Les meilleures pratiques pour les déploiements bleu/vert sont les suivantes.

Rubriques

- Bonnes pratiques générales pour les déploiements bleu/vert
- RDS pour MySQL MySQL : bonnes pratiques pour les déploiements bleu/vert
- Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert

Bonnes pratiques générales pour les déploiements bleu/vert

Tenez compte des meilleures pratiques générales suivantes lorsque vous créez un déploiement bleu/vert.

- Testez minutieusement les instances de base de données dans l'environnement vert avant le basculement.
- Gardez vos bases de données dans l'environnement vert en lecture seule. Nous vous recommandons d'activer les opérations d'écriture sur l'environnement vert avec prudence, car elles

peuvent entraîner des conflits de réplication. Elles peuvent également entraîner la présence de données involontaires dans les bases de données de production après la commutation.

- Si vous utilisez un déploiement bleu/vert pour implémenter des modifications de schéma, apportez uniquement des modifications compatibles avec la réplication.

  Par exemple, vous pouvez ajouter de nouvelles colonnes à la fin d'un tableau sans perturber la réplication entre le déploiement bleu et le déploiement vert. Toutefois, les modifications de schéma, telles que le renommage de colonnes ou de tables, interrompent la réplication vers le déploiement vert.

  Pour plus d'informations sur les modifications compatibles avec la réplication, consultez Replication with Differing Table Definitions on Source and Replica dans la documentation MySQL, et Restrictions dans la documentation Réplication logique PostgreSQL.

  > **Note**
  >
  > Cette limitation ne s'applique pas aux déploiements bleu/vert de RDS pour PostgreSQL qui utilisent la réplication physique. Pour de plus amples informations, veuillez consulter the section called “Limitations blue/green de RDS pour PostgreSQL pour les déploiements avec réplication physique”.

- Après avoir créé le déploiement bleu/vert, gérez le chargement différé si nécessaire. Assurez-vous que le chargement des données est terminé avant de basculer. Pour de plus amples informations, veuillez consulter Chargement différé et initialisation du stockage pour les déploiements bleu/vert.
- Lorsque vous basculez vers un déploiement bleu/vert, suivez les bonnes pratiques de commutation. Pour de plus amples informations, veuillez consulter the section called “Bonnes pratiques de commutation”.

RDS pour MySQL MySQL : bonnes pratiques pour les déploiements bleu/vert

Tenez compte des meilleures pratiques suivantes lorsque vous créez un déploiement bleu/vert à partir d'un cluster de base de données Aurora MySQL d'une instance de base de données RDS .

- Évitez d'utiliser des moteurs de stockage non transactionnels, tels que MyISAM, qui ne sont pas optimisés pour la réplication.
- Optimisez les répliques de lecture et l'environnement écologique pour la réplication des journaux binaires. Si votre moteur de base de données le prend en charge, activez la réplication GTID,

parallel et crash-safe pour garantir la cohérence et la durabilité des données avant de créer votre déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation de GTID la réplication basée.

- Si l'environnement vert connaît un retard de réplication, tenez compte des points suivants :
  - Définissez temporairement le `innodb_flush_log_at_trx_commit` paramètre `2`dans le groupe de paramètres de base de données vert. Une fois que la réplication a rattrapé son retard, revenez à la valeur par défaut d'`1`avant le basculement. Si un arrêt ou un crash inattendu survient avec la valeur du paramètre temporaire, reconstruisez l'environnement écologique pour éviter toute corruption de données non détectée.
  - Pour réduire la latence d'écriture et améliorer le débit de réplication, remplacez temporairement les instances de base de données multi-AZ vertes par des instances de base de données mono-AZ. Réactivez Multi-AZ juste avant le passage au numérique.

Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert

Tenez compte des meilleures pratiques suivantes lorsque vous créez un déploiement bleu/vert à partir d'une instance de base de données RDS pour PostgreSQL.

Rubriques

- Bonnes pratiques générales de RDS pour PostgreSQL pour les déploiements bleu/vert
- Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert avec réplication physique
- Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert avec réplication logique

Bonnes pratiques générales de RDS pour PostgreSQL pour les déploiements bleu/vert

Tenez compte des meilleures pratiques générales suivantes lorsque vous créez un déploiement bleu/vert à partir d'une instance de base de données RDS pour PostgreSQL.

- Mettez à jour toutes vos extensions PostgreSQL vers la version la plus récente avant de créer un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter the section called “Mise à niveau des extensions Postgre SQL”.
- Les transactions de longue durée peuvent entraîner un retard de réplication important. Pour réduire le délai de réplication, pensez à effectuer les opérations suivantes :

- Réduisez les transactions de longue durée qui peuvent être retardées jusqu'à ce que l'environnement vert rattrape l'environnement bleu.
- Réduisez les opérations en vrac dans l'environnement bleu jusqu'à ce que l'environnement vert rattrape l'environnement bleu.
- Lancez une opération manuelle de congélation sous vide sur les tables occupées avant de créer le déploiement bleu/vert.
- Pour les versions 12 et supérieures de PostgreSQL, désactivez `index_cleanup` le paramètre sur les tables volumineuses ou occupées afin d'augmenter le taux de maintenance normale sur les bases de données bleues. Pour de plus amples informations, veuillez consulter the section called “Mise à vide d'une table le plus rapidement possible”.

  > **Note**
  >
  > Le fait de ne pas nettoyer régulièrement l'index pendant que vous passez l'aspirateur peut entraîner un gonflement de l'index, ce qui peut dégrader les performances de numérisation. Il est recommandé d'utiliser cette approche uniquement lors d'un déploiement bleu/vert. Une fois le déploiement terminé, nous vous recommandons de reprendre la maintenance et le nettoyage réguliers de l'index.

- La lenteur de la réplication peut entraîner des redémarrages fréquents des expéditeurs et des destinataires, ce qui retarde la synchronisation. Pour vous assurer qu'ils restent actifs, désactivez les délais d'expiration `0` en réglant le `wal_sender_timeout` paramètre sur l'environnement bleu et le `wal_receiver_timeout` paramètre sur `0` l'environnement vert.
- Pour éviter que les segments du journal d'écriture anticipée (WAL) ne soient supprimés de l'environnement bleu, définissez le `wal_keep_segments` paramètre sur 15625 pour PostgreSQL version 13 ou inférieure. Pour les versions 14 et supérieures, définissez le `wal_keep_size` paramètre sur 1 TiB, s'il y a suffisamment d'espace de stockage disponible.

Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert avec réplication physique

Avec la réplication physique, Amazon RDS crée une réplique en lecture de l'instance de base de données source. Pour les paramètres associés, la surveillance, le réglage et le dépannage, consultezthe section called “Utilisation de réplicas en lecture pour RDS for PostgreSQL”.

Pour savoir dans quels cas les déploiements bleu/vert utilisent la réplication physique au lieu de la réplication logique, consultez. the section called “Méthodes de SQL réplication Postgre”

Meilleures pratiques de RDS pour PostgreSQL pour les déploiements bleu/vert avec réplication logique

Tenez compte des meilleures pratiques suivantes lorsque vous créez un blue/green deployment that uses logical replication. For an explanation of when blue/green déploiement qui utilise la réplication logique plutôt que la réplication physique, voirthe section called “Méthodes de SQL réplication Postgre”.

- Si votre base de données dispose de suffisamment de mémoire libre, augmentez la valeur du paramètre `logical_decoding_work_mem` DB dans l'environnement bleu. Cela permet de réduire le nombre de décodages sur disque et d'utiliser de la mémoire à la place. Pour de plus amples informations, veuillez consulter la documentation sur PostgreSQL.
  - Vous pouvez surveiller le dépassement des transactions écrites sur le disque à l'aide de la `ReplicationSlotDiskUsage` CloudWatch métrique. Cette métrique fournit des informations sur l'utilisation des emplacements de réplication sur le disque, ce qui permet d'identifier les cas où les données de transaction dépassent la capacité de mémoire et sont stockées sur disque. Vous pouvez surveiller la mémoire disponible à l'aide de la `FreeableMemory` CloudWatch métrique. Pour de plus amples informations, veuillez consulter the section called “Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS”.
  - Dans les versions 14 et supérieures de RDS pour PostgreSQL, vous pouvez surveiller la taille des fichiers de dépassement logique à l'aide de la vue système. `pg_stat_replication_slots`
- Si vous utilisez l'`aws_s3`extension, donnez à l'instance de base de données verte l'accès à Amazon S3 via un rôle IAM une fois l'environnement vert créé. Cela permet aux commandes d'importation et d'exportation de continuer à fonctionner après la commutation. Pour obtenir des instructions, veuillez consulter the section called “Configuration de l'accès à un compartiment Amazon S3”.
- Vérifiez les performances de vos instructions UPDATE et DELETE et déterminez si la création d'un index sur la colonne utilisée dans la clause WHERE peut optimiser ces requêtes. Cela peut améliorer les performances lorsque les opérations sont rejouées dans un environnement vert.
- Si vous utilisez des déclencheurs, assurez-vous qu'ils n'interfèrent pas avec la création, la mise à jour et la suppression d'`pg_catalog.pg_publication`objets `pg_catalog.pg_replication_slots` dont le nom commence par « rds ». `pg_catalog.pg_subscription`

- Si vous spécifiez une version du moteur supérieure pour l'environnement écologique, exécutez l'`ANALYZE`opération sur toutes les bases de données pour actualiser le `pg_statistic` tableau. Les statistiques de l'optimiseur ne sont pas transférées lors d'une mise à niveau de version majeure. Vous devez donc régénérer toutes les statistiques pour éviter les problèmes de performances. Pour connaître les meilleures pratiques supplémentaires lors des mises à niveau majeures des versions, consultezthe section called “Comment effectuer une mise à niveau de version majeure”.
- Évitez de configurer les déclencheurs au fur `ENABLE REPLICA` `ENABLE ALWAYS` et à mesure que le déclencheur est utilisé sur la source pour manipuler des données. Dans le cas contraire, le système de réplication propage les modifications et exécute le déclencheur, ce qui entraîne une duplication.

Méthodes de SQL réplication Postgre pour les déploiements bleu/vert

Amazon RDS pour Postgre utilise SQL principalement la réplication physique pour blue/green deployments. However, if you request a major version upgrade when you create the blue/green le déploiement, et votre instance de base de données source exécute l'une des SQL versions de Postgre répertoriées dans le tableau ci-dessous. Amazon RDS utilise plutôt la réplication logique.

Le tableau suivant décrit les cas dans lesquels Amazon RDS utilise la réplication physique plutôt que la réplication logique pour les déploiements SQL bleu/vert de Postgre.

Version de l'SQLinstance de base de données Postgre source	Action de mise à niveau dans le cadre d'un déploiement bleu/vert	Méthode de réplication
• 16.1 et toutes les versions majeures et mineures supérieures • Version 15.4 et versions 15 ultérieures • Version 14.9 et versions 14 ultérieures • Versions 13.12 et supérieures 13	Mise à niveau de la version majeure (instance verte sur une version majeure du moteur supérieure à la version bleue)	Réplication logique

Version de l'SQLinstance de base de données Postgre source	Action de mise à niveau dans le cadre d'un déploiement bleu/vert	Méthode de réplication
• Versions 12.16 et supérieures 12 • Versions 11.21 et supérieures 11		
Toutes les versions prises en charge	Mise à niveau de version mineure ou absence de mise à niveau (instance verte sur la même version majeure du moteur que bleue)	Réplication physique

> **Note**
>
> Les mises à niveau des versions majeures ne sont pas prises en charge pour les déploiements bleu/vert avec source RDS pour SQL les versions 15.3 et inférieures de Postgre, 14.8 et inférieures, 13.11 et inférieures, 12.15 et inférieures, ou 11.20 et inférieures.

Pour plus d'informations sur les limites des déploiements bleu/vert qui utilisent la réplication physique et logique, consultez les sections suivantes :

- the section called “Limitations blue/green de RDS pour PostgreSQL pour les déploiements avec réplication physique”
- the section called “Limitations d' RDS pour PostgreSQL pour les déploiements avec réplication logique blue/green ”

Lorsque vous créez un déploiement bleu/vert, vous spécifiez l'instance de base de données source à copier dans le déploiement. L'instance de base de données que vous choisissez est l'instance de base de données de production, et elle devient l'instance de base de données principale dans

l'environnement bleu. Cette instance de base de données est copiée dans l'environnement vert, et RDS configure la réplication de l'instance de base de données de l'environnement bleu vers l'instance de base de données de l'environnement vert.

RDS copie la topologie et les fonctionnalités de l'environnement bleu dans une zone intermédiaire. Si l'instance de base de données bleue possède des répliques lues, elles sont copiées en tant que répliques de l'instance verte. Le stockage alloué à toutes les répliques vertes correspond à l'instance principale verte, tandis que les autres paramètres de stockage sont hérités des répliques bleues.

Si l'instance de base de données bleue est un déploiement d'instance de base de données multi-AZ, alors l'instance de base de données verte est créée comme un déploiement d'instance de base de données multi-AZ.

Rubriques

- Préparation d'un déploiement bleu/vert
- Spécification des modifications lors de la création d'un déploiement bleu/vert
- Chargement différé et initialisation du stockage pour les déploiements bleu/vert
- Création d'un déploiement bleu/vert
- Paramètres de création de déploiements bleu/vert

Préparation d'un déploiement bleu/vert

Vous devez suivre certaines étapes avant de créer un déploiement bleu/vert, en fonction du moteur sur lequel votre instance de base de données de est exécutée.

Rubriques

- Préparation d'une instance de base de données RDS pour MySQL ou RDS pour MariaDB pour un déploiement bleu/vert
- Préparation d'une instance de base de données RDS pour PostgreSQL pour un déploiement bleu/vert avec réplication physique
- Préparation d'une instance de base de données RDS pour PostgreSQL pour un déploiement bleu/vert avec réplication logique

Préparation d'une instance de base de données RDS pour MySQL ou RDS pour MariaDB pour un déploiement bleu/vert

Avant de créer un déploiement bleu/vert pour une instance de base de données RDS pour MySQL ou RDS pour MariaDB, vous devez activer les sauvegardes automatisées. Pour obtenir des instructions, consultez the section called “Activation des sauvegardes automatiques”.

Préparation d'une instance de base de données RDS pour PostgreSQL pour un déploiement bleu/vert avec réplication physique

Avant de créer un déploiement bleu/vert de RDS pour PostgreSQL utilisant la réplication physique, veillez à effectuer les opérations suivantes. Pour obtenir la liste des versions qui utilisent la réplication physique par opposition à la réplication logique, consultezthe section called “Méthodes de SQL réplication Postgre”.

- Activez les sauvegardes automatisées sur l'instance de base de données. Pour obtenir des instructions, consultez the section called “Activation des sauvegardes automatiques”.
- Vérifiez que l'instance de base de données n'est ni la source ni la cible de la réplication externe. Pour de plus amples informations, veuillez consulter the section called “Limitations générales”.

Préparation d'une instance de base de données RDS pour PostgreSQL pour un déploiement bleu/vert avec réplication logique

Avant de créer un déploiement bleu/vert de RDS pour PostgreSQL utilisant la réplication logique, veillez à effectuer les opérations suivantes. Pour obtenir la liste des versions qui utilisent la réplication logique par opposition à la réplication physique, consultezthe section called “Méthodes de SQL réplication Postgre”.

- Associez l'instance à un groupe de paramètres de base de données personnalisé avec la réplication logique (`rds.logical_replication`) activée. La réplication logique est requise pour la réplication de l'environnement bleu vers l'environnement vert. Pour obtenir des instructions, consultez the section called “Modification de paramètres dans un groupe de paramètres de bases de données”.

  Étant donné que les déploiements bleu/vert nécessitent au moins un assistant de fond par base de données, veillez à ajuster les paramètres de configuration suivants en fonction de votre charge de travail. Pour obtenir des instructions permettant de régler chaque paramètre, consultez la section Paramètres de configuration dans la documentation de PostgreSQL.

- `max_replication_slots`
- `max_wal_senders`
- `max_logical_replication_workers`
- `max_worker_processes`

Après avoir activé la réplication logique et défini toutes les options de configuration, veillez à redémarrer l'instance de base de données pour que vos modifications prennent effet. Les déploiements bleu/vert nécessitent que l'instance de base de données soit synchronisée avec le groupe de paramètres de base de données, sans quoi la création échoue. Pour de plus amples informations, veuillez consulter the section called “Redémarrage d'une instance DB”.

- Vérifiez que l'instance de base de données n'est ni la source ni la cible de la réplication externe. Pour de plus amples informations, veuillez consulter the section called “Limitations générales”.
- Assurez-vous que toutes les tables de l'instance de base de données possèdent une clé primaire. La réplication logique PostgreSQL n'autorise pas les opérations `UPDATE` ou `DELETE` sur les tables dépourvues de clé primaire.
- RDS pour PostgreSQL utilise la réplication logique native de PostgreSQL, en stockant les segments des journaux d'écriture anticipée (WAL) sur l'instance bleue jusqu'à ce qu'ils soient rejoués dans l'environnement vert. Avant de créer un déploiement bleu/vert, vérifiez que l'instance bleue dispose d'une capacité adéquate en vérifiant les mesures suivantes :
  - `FreeStorageSpace`
  - `TransactionLogsGeneration`
  - `TransactionLogsDiskUsage`
  - `OldestReplicationSlotLag`

  Pour estimer le stockage supplémentaire requis sur l'instance bleue, surveillez la `TransactionLogsGeneration` CloudWatch métrique pendant les périodes de charge de travail de pointe. Par exemple, si votre charge de travail génère 100 Go de données WAL en 24 heures, assurez-vous de disposer d'au moins 100 Go de stockage supplémentaire pour accueillir l'équivalent d'une journée de segments WAL. Pour de plus amples informations, veuillez consulter Surveillance des métriques dans une instance de base de données.

Spécification des modifications lors de la création d'un déploiement bleu/vert

Vous pouvez apporter les modifications suivantes à l'instance de base de données dans l'environnement vert lorsque vous créez le déploiement bleu/vert.

Vous pouvez apporter d'autres modifications à l'instance dans l'environnement vert après son déploiement. Par exemple, vous pouvez spécifier une version du moteur supérieure ou un autre groupe de paramètres.

Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Rubriques

- Spécifier une version de moteur supérieure
- Spécifier un groupe de paramètres de base de données différent
- Modifier les paramètres de stockage et de performance
- Activer l'option Écritures optimisées pour RDS
- Mettre à niveau la configuration du stockage

Spécifier une version de moteur supérieure

Vous pouvez spécifier une version supérieure du moteur si vous voulez tester une mise à niveau du moteur de base de données. Lors de la commutation, la base de données est mise à niveau vers la version majeure ou mineure du moteur de base de données que vous spécifiez.

Spécifier un groupe de paramètres de base de données différent

Vous pouvez tester la manière dont les changements de paramètres affectent les instances de base de données dans l'environnement vert ou spécifier un groupe de paramètres pour une nouvelle version majeure du moteur de base de données dans le cas d'une mise à niveau.

Si vous spécifiez un groupe de paramètres de base de données différent, le groupe de paramètres de base de données spécifié est associé à toutes les instances de base de données dans l'environnement vert. Si vous ne spécifiez pas de groupe de paramètres différent, chaque instance de base de données dans l'environnement vert est associée au groupe de paramètres de son instance de base de données bleue correspondante.

Modifier les paramètres de stockage et de performance

Ajustez les paramètres de stockage et de performance dans un environnement écologique afin d'optimiser l'allocation des ressources. Ces paramètres incluent le stockage alloué, les IOPS provisionnées, le type de stockage et le débit de stockage (pour le stockage gp3).

Vous pouvez modifier le type de stockage de l'instance de base de données verte en gp2, gp3, io1 ou io2. Pour le stockage gp3, vous pouvez également ajuster le débit de stockage afin d'améliorer les performances de transfert de données pour les charges de travail les plus exigeantes ou de réduire les coûts pour les applications moins intensives. Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.

Vous pouvez également choisir d'augmenter ou de diminuer le stockage alloué dans un environnement vert. Toutefois, une réduction du stockage ne se produit que si le stockage cible alloué est supérieur d'au moins 20 % à l'utilisation actuelle du stockage. Si vous diminuez le stockage alloué, Amazon RDS lance une mise à niveau de la configuration du stockage. Pour de plus amples informations, veuillez consulter the section called “Mettre à niveau la configuration du stockage”.

Si l'instance de base de données bleue utilise un stockage magnétique, vous devez remplacer l'instance de base de données verte par un type de stockage à usage général ou par IOPS provisionné afin d'augmenter ou de diminuer le stockage alloué.

Activer l'option Écritures optimisées pour RDS

Vous pouvez utiliser un déploiement bleu/vert pour passer à une classe d'instance de base de données qui prend en charge les écritures optimisées RDS. Vous ne pouvez activer l'option Écritures optimisées pour RDS que sur une base de données créée avec une classe d'instance de base de données prise en charge. Cette option crée une base de données verte qui utilise une classe d'instance de base de données prise en charge, ce qui vous permet d'activer l'option Écritures optimisées pour RDS sur l'instance de base de données verte.

Si vous effectuez une mise à niveau à partir d'une classe d'instance de base de données qui ne prend pas en charge l'option Écritures optimisées pour RDS vers une classe qui le fait, vous devez également mettre à niveau la configuration de stockage de l'instance de base de données verte. Pour de plus amples informations, veuillez consulter the section called “Mettre à niveau la configuration du stockage”.

Vous ne pouvez mettre à niveau que la classe d'instance de base de données de l'instance de base de données verte principale. Par défaut, les réplicas en lecture dans l'environnement vert héritent des

paramètres de l'instance de base de données de l'environnement bleu. Une fois l'environnement vert créé avec succès, vous devez modifier manuellement la classe d'instance de base de données des réplicas en lecture dans l'environnement vert.

Certaines mises à niveau de classe d'instance ne sont pas prises en charge selon la version du moteur et la classe d'instance de l'instance de base de données bleue. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter the section called “Classes d'instances de base de données”.

Mettre à niveau la configuration du stockage

Si votre base de données bleue n'utilise pas la dernière configuration de stockage, RDS peut migrer l'instance de base de données verte de l'ancienne configuration de stockage (système de fichiers 32 bits) vers la configuration préférée. Vous pouvez utiliser les déploiements bleu/vert RDS pour surmonter les limitations de mise à l'échelle en matière de stockage et de taille de fichier pour les anciens systèmes de fichiers 32 bits. En outre, ce paramètre modifie la configuration du stockage pour qu'elle soit compatible avec l'option Écritures optimisées pour RDS si la classe d'instance de base de données spécifiée prend en charge l'option Écritures optimisées.

> **Note**
>
> La mise à niveau de la configuration de stockage est une opération gourmande en E/S et entraîne des délais de création plus longs pour le solde blue/green deployments. The storage upgrade process is faster if the blue DB instance uses Provisioned IOPS SSD (io1 or io2 Block Express) storage, and if you provisoned the green environment with an instance size of 4xlarge or larger. Storage upgrades involving General Purpose SSD (gp2) storage can deplete your I/O créditeur, ce qui entraîne des délais de mise à niveau plus longs. Pour de plus amples informations, veuillez consulter the section called “Stockage d'instance de base de données”.
>
> Pendant la mise à niveau du stockage, l'instance de base de données verte est temporairement indisponible, tandis que l'instance de base de données bleue reste disponible. La réplication est interrompue pendant cette période. Surveillez le stockage sur l'instance bleue et envisagez de le dimensionner si le stockage atteint 90 %, car l'instance verte augmente automatiquement de 10 % après la mise à niveau.

Cette option n'est disponible que si votre base de données bleue ne possède pas la dernière configuration de stockage ou si vous modifiez la classe d'instance de base de données dans la même

demande. Vous ne pouvez mettre à niveau la configuration de stockage que lors de la création initiale d'un déploiement bleu/vert.

Chargement différé et initialisation du stockage pour les déploiements bleu/vert

Lorsque vous créez un déploiement bleu/vert, Amazon RDS crée l'instance de base de données principale dans l'environnement vert en restaurant à partir d'un instantané de base de données. Une fois créée, l'instance de base de données verte et ses répliques de lecture continuent de charger les données en arrière-plan par le biais d'un processus appelé chargement différé.

Le chargement différé ne charge les blocs de données que lorsque les applications les demandent. Si vous tentez d'accéder à des données qui n'ont pas encore été chargées, Amazon EBS les récupère immédiatement sur Amazon S3, tandis que les données restantes continuent de se charger en arrière-plan. Pour plus d'informations, consultez Instantanés Amazon EBS.

Pour accélérer les performances du volume complet, Amazon RDS propose une initialisation du stockage, qui lit tous les blocs du volume Green Environment. Amazon EBS télécharge des blocs de manière proactive depuis Amazon S3, offrant ainsi des performances de volume maximales dès la première utilisation. L'initialisation du stockage s'effectue entièrement en arrière-plan, ce qui garantit qu'il n'y a aucun impact sur la disponibilité de votre instance de base de données ou sur les activités en cours, telles que l'application de correctifs ou les mises à niveau.

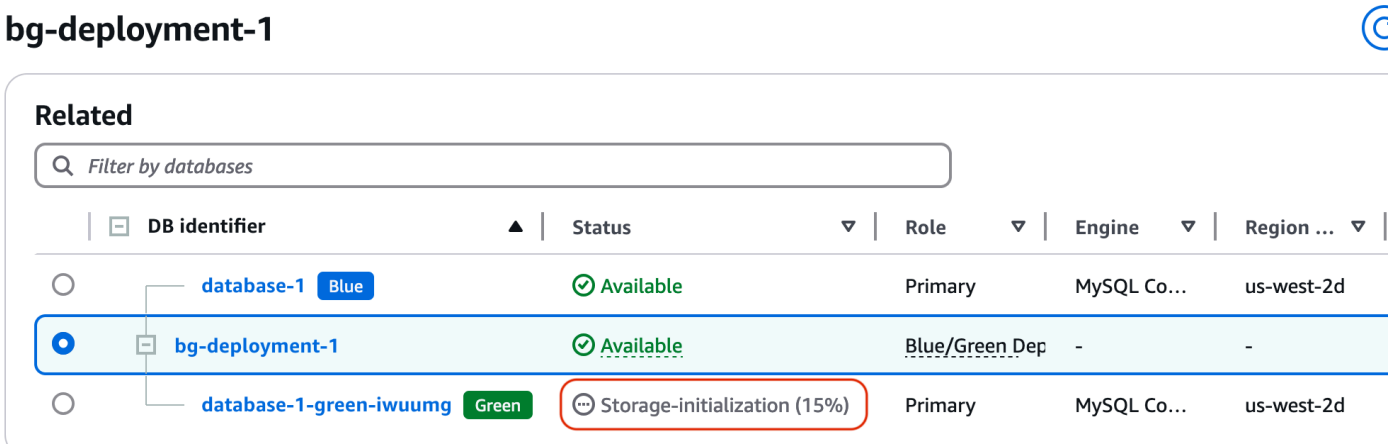
L'initialisation du stockage n'est disponible que pour les instances dans des déploiements bleu/vert avec des types de `gp2` volume, `gp3` `io1`, et. `io2` Il prend en charge toutes les classes d'instance à l'exception des familles t3 et t4. Si vous modifiez une instance de base de données verte dans un déploiement mono-AZ en un déploiement d'instance de base de données multi-AZ, l'initialisation du stockage inclut le nœud secondaire dans la configuration multi-AZ.

Pendant l'initialisation du stockage, l'instance reste entièrement disponible et utilisable pour les opérations de base de données, même si le stockage risque de ne pas atteindre ses performances optimales tant que l'initialisation n'est pas terminée. Lorsque l'initialisation du stockage est en cours, l'état général de l'instance passe à Initialisation du stockage, et l'indicateur de progression reflète le niveau d'initialisation minimal sur tous les volumes de l'instance de base de données.

Utilisez la console ou l' AWS CLI API Amazon RDS pour surveiller l'initialisation du stockage.

Console

Dans le AWS Management Console, vous pouvez voir la progression de l'initialisation du stockage avec l'état de l'instance de base de données.



AWS CLI

Avec le AWS CLI, vous pouvez surveiller l'initialisation du stockage à l'aide de la describe-db-instancescommande. Le `PercentProgress` champ de la réponse indique le pourcentage de données extraites d'Amazon S3.

```
aws rds describe-db-instances --db-instance-identifier my-db-instance

{
    "DBInstances": [
        {
            "DBInstanceIdentifier": "my-db-instance",
            "DBInstanceClass": "db.m5.2xlarge",
            "Engine": "postgres",
            "DBInstanceStatus": "storage-initialization",
            ...
            "PercentProgress": "34"
        }
    ]
}
```

Amazon RDS API

Avec l'API Amazon RDS, vous pouvez récupérer l'état de l'initialisation du stockage en appelant l'action DBInstancesDescribe.

L'indicateur de progression est mis à jour au fur et à mesure que la tâche d'initialisation en arrière-plan avance, ce qui vous permet de suivre l'état de préparation du stockage avant la fin de l'initialisation complète du stockage. L'initialisation du stockage permet d'optimiser les performances lorsque votre instance de base de données verte devient pleinement opérationnelle.

Création d'un déploiement bleu/vert

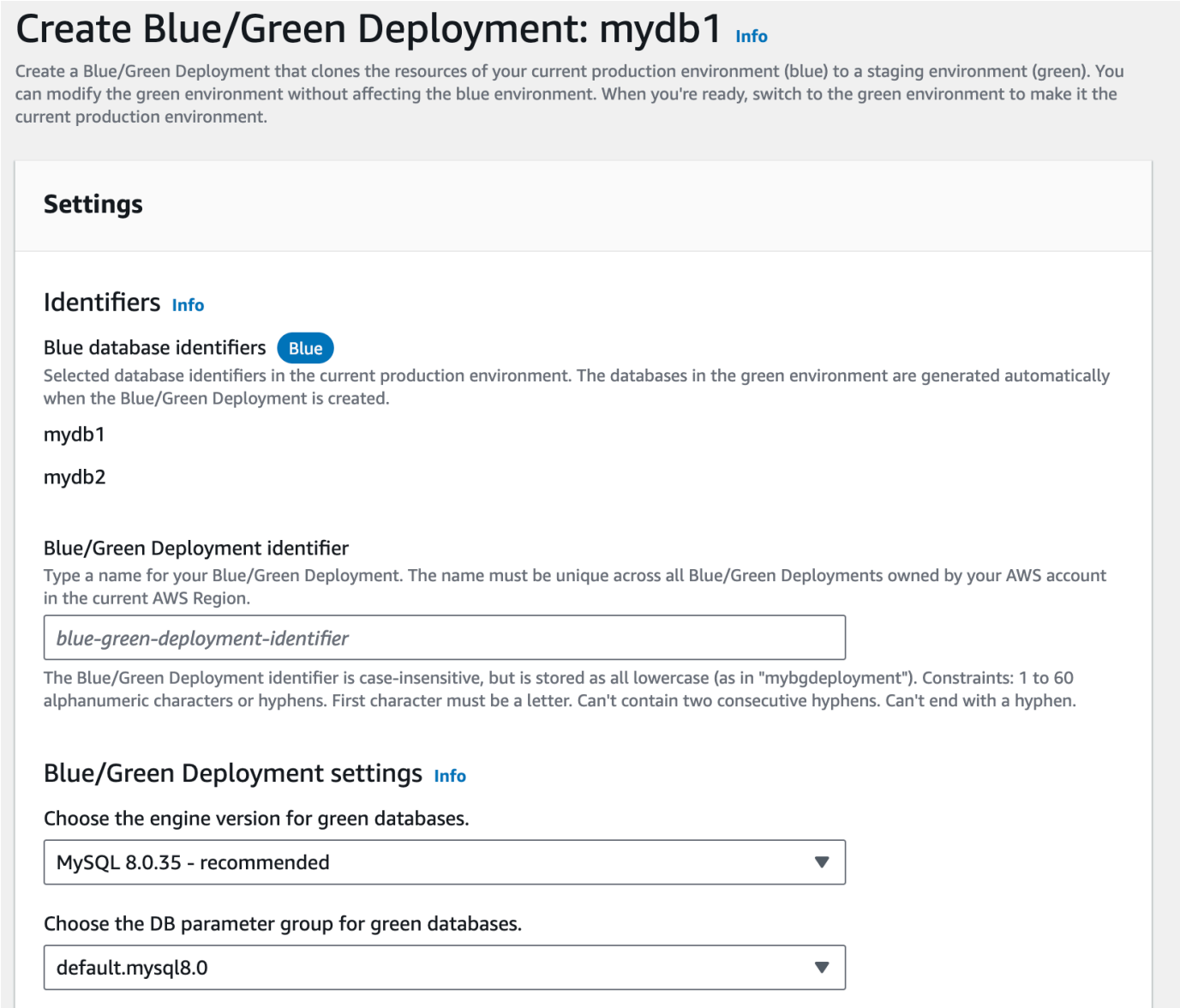
Vous pouvez créer un déploiement bleu/vert à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

console

Pour créer un déploiement bleu/vert

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis choisissez l'instance de base de données que vous voulez copier dans un environnement vert.
3. Choisissez Actions, puis Créer un déploiement bleu/vert.

   La page Créer un déploiement bleu/vert apparaît.



4. Passez en revue les identifiants de base de données bleus. Assurez-vous qu'elles correspondent aux instances de base de données que vous attendez dans l'environnement bleu. Si ce n'est pas le cas, choisissez Annuler.
5. Pour le nom du déploiement bleu/vert, entrez un nom pour votre déploiement bleu/vert.
6. Dans les sections restantes, spécifiez les paramètres de l'environnement vert. Pour obtenir des informations sur chaque paramètre, consultez the section called “Paramètres disponibles”.

   Vous pouvez apporter d'autres modifications aux bases de données dans l'environnement vert après son déploiement.
7. Sélectionnez Créer.

AWS CLI

Pour créer un déploiement bleu/vert à l'aide de AWS CLI, utilisez la create-blue-green-deploymentcommande. Pour plus d'informations sur toutes les options disponibles, consultezthe section called “Paramètres disponibles”.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-blue-green-deployment \
    --blue-green-deployment-name my-blue-green-deployment \
    --source arn:aws:rds:us-east-2:123456789012:db:mydb1 \
    --target-engine-version 8.0.31 \
    --target-db-parameter-group-name mydbparametergroup
```

Dans Windows:

```
aws rds create-blue-green-deployment ^
    --blue-green-deployment-name my-blue-green-deployment ^
    --source arn:aws:rds:us-east-2:123456789012:db:mydb1 ^
    --target-engine-version 8.0.31 ^
    --target-db-parameter-group-name mydbparametergroup
```

API RDS

Pour créer un déploiement bleu/vert à l'aide de l'API Amazon RDS, utilisez l'opération. `CreateBlueGreenDeployment` Pour plus d'informations sur chaque option, veuillez consulter the section called “Paramètres disponibles”.

Paramètres de création de déploiements bleu/vert

Le tableau suivant explique les paramètres que vous pouvez choisir lorsque vous créez un déploiement bleu/vert. Pour plus d'informations sur les AWS CLI options, consultez create-blue-green-deployment. Pour plus d'informations sur les paramètres de l'API RDS, consultez CreateBlueGreenDeployment.

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Stockage alloué	La quantité de stockage à allouer à votre instance de base de données verte (en gibioctets). Vous pouvez choisir d'augmenter ou de diminuer le stockage alloué.
	
	Si votre instance de base de données bleue utilise un stockage magnétique (`standard`), vous devez remplacer l'instance de base de données verte par un type de stockage à usage général ou IOPS provisionné afin de modifier le stockage alloué dans l'environnement vert.
	
	Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.
		Option de l'interface CLI :
		
		`--target-allocated-storage`
		
		Paramètre de l'API :
		
		`TargetAllocatedStorage`

Identifiant de déploiement bleu/vert	Un nom pour le déploiement bleu/vert.	Option de l'interface CLI :
		
		`--blue-green-deployment-name`
		
		Paramètre de l'API :
		
		`BlueGreenDeploymentName`

Identifiant de base de données bleu	Identifiant du d'instances que vous souhaitez copier dans l'environnement vert. Lorsque vous utilisez la CLI ou l'API, spécifiez le nom	Option de l'interface CLI : `--source` Paramètre de l'API :

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS
	de ressource Amazon (ARN) du d'instances.	`Source`

Groupe de paramètres de de bases de données pour les bases de données vertes	Un groupe de paramètres à associer aux bases de données dans l'environnement vert.	Option de l'interface CLI :
		
		`--target-db-parameter-group-name`
		
		`--target-db-cluster-parameter-group-name`
		
		Paramètre de l'API :
		
		`TargetDBParameterGroupName`
		
		`TargetDBClusterParameterGroupName`

Activer les écritures optimisées pour la base de données verte	Activez les écritures optimisées RDS sur l'instance de base de données principale verte. Pour de plus amples informations, veuillez consulter the section called “Activer l'option Écritures optimisées pour RDS”. Si vous passez d'une classe d'instance de base de données qui ne prend pas en charge l'option Écritures optimisées à une classe qui le fait, vous devez également mettre à niveau la configuration du stockage. Pour de plus amples informations, veuillez consulter the section called “Mettre à niveau la configuration du stockage”.	Pour la CLI et l'API, la spécification d'une classe d'instance de base de données cible qui prend en charge les écritures optimisées RDS l'active automatiquement sur l'instance de base de données principale verte.

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Version du moteur pour bases de données écologiques	Mettez à niveau le de bases de données dans l'environnement vert vers la version du moteur de base de données spécifiée.
	
	Si ce n'est pas spécifié, chaque base de données de base de données dans l'environnement vert est créée avec la même version de moteur que le de base de données correspondant dans l'environ nement bleu.
	
	Si vous choisissez une instance de RDS pour PostgreSQL qui utilise la réplication logique, examinez et reconnaissez les limites de la réplication logique. Pour de plus amples informations, veuillez consulter the section called “Limitations spécifiques à la réplication logique pour les déploiements blue/green ”.
		Option de l'interface CLI :
		
		`--target-engine-version`
		
		Paramètre de l'API RDS :
		
		`TargetEngineVersion`

Classe d'instance de base de données verte	La capacité de calcul et de mémoire de chaque instance de base de données dans un environnement vert, par exemple`db.m5d.xlarge` .
	
	Cette option n'est visible que lorsque vous activez les écritures optimisées RDS pour la base de données verte.
		Option de l'interface CLI :
		
		`--target-db-instance-class`
		
		Paramètre de l'API RDS :
		
		`TargetDBInstanceClass`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

IOPS provisionnés	Nombre d'opérations d'entrée/sortie provisionnées par seconde (IOPS) à allouer initialement à la base de données verte.
	
	Cette valeur s'applique uniquement à l'instance de base de données principale verte, et non aux répliques vertes.
		Option de l'interface CLI :
		
		`--target-iops`
		
		Paramètre de l'API RDS :
		
		`TargetIops`

Mise à niveau de configuration du stockage	Choisissez si vous souhaitez mettre à niveau la configuration de votre système de fichiers de stockage. Si vous activez ce paramètre, RDS fait migrer la base de données verte de l'ancien système de fichiers de stockage vers la configuration préférée.
	
	Cette option n'est disponible que si votre base de données bleue ne possède pas la dernière configuration de stockage ou si vous activez l'option Écritures optimisées pour RDS dans la même demande. Vous ne pouvez mettre à niveau la configuration de stockage que lors de la création initiale d'un déploiement bleu/vert.
	
	Pour de plus amples informations, veuillez consulter the section called “Mise à niveau du système de fichiers de stockage”.
		Option de l'interface CLI :
		
		`--upgrade-target-storage-config`
		
		Paramètre de l'API RDS :
		
		`UpgradeTargetStorageConfig`

Paramètre de la console	Description du paramètre	Option de l'interface CLI et paramètre de l'API RDS

Storage throughput (Débit de stockage)	Valeur du débit de stockage pour la base de données verte. Ce paramètre n'est visible que si vous choisissez le type de stockage SSD à usage général (gp3).
	
	Cette valeur s'applique uniquement à l'instance de base de données principale verte, et non aux répliques vertes.
	
	Pour de plus amples informations, veuillez consulter Stockage GP3 (recommandé).
		Option de l'interface CLI :
		
		`--target-storage-throughput`
		
		Paramètre de l'API RDS :
		
		`TargetStorageThroughput`

Type de stockage	Type de stockage de la base de données verte. Les types de stockage suivants sont pris en charge :
	
	• SSD à usage général (gp2)
	• SSD à usage général (gp3)
	• IOPS provisionnés (io1)
	• SSD IOPS provisionné (io2)
	
	Cette valeur s'applique uniquement à l'instance de base de données principale verte, et non aux répliques vertes.
	
	Pour de plus amples informations, veuillez consulter Types de stockage Amazon RDS.
		Option de l'interface CLI :
		
		`--target-storage-type`
		
		Paramètre de l'API RDS :
		
		`TargetStorageType`

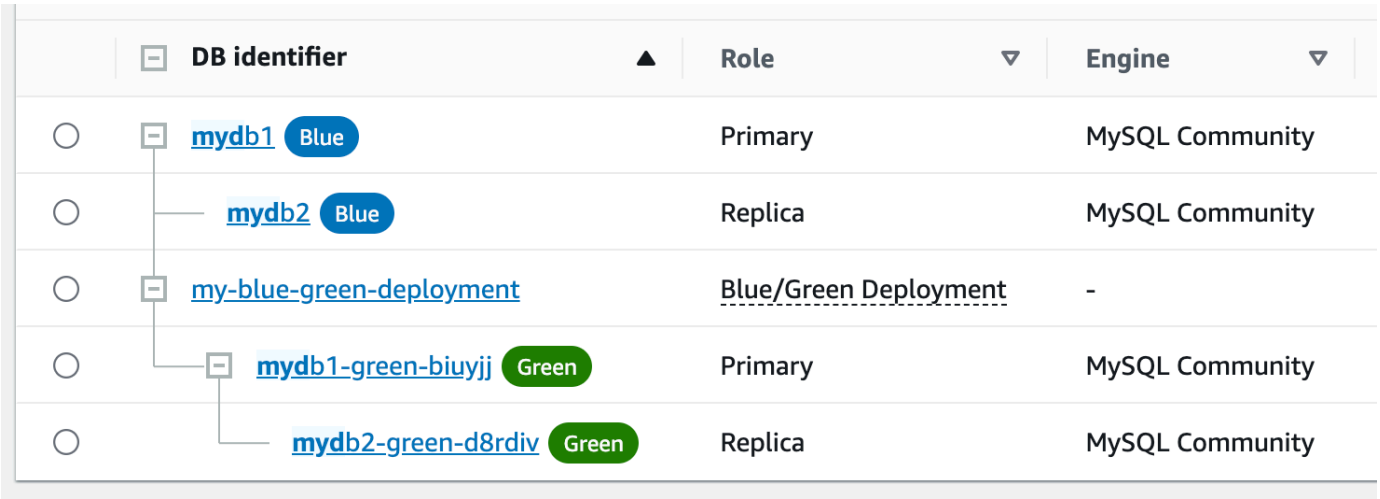
Vous pouvez consulter les détails d'un déploiement bleu/vert à l'aide de l'API AWS Management Console, de ou de l' AWS CLI API RDS.

Vous pouvez également consulter des événements et vous y abonner pour obtenir des informations sur un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Événements de déploiement bleu/vert.

Console

Pour afficher les détails d'un déploiement bleu/vert

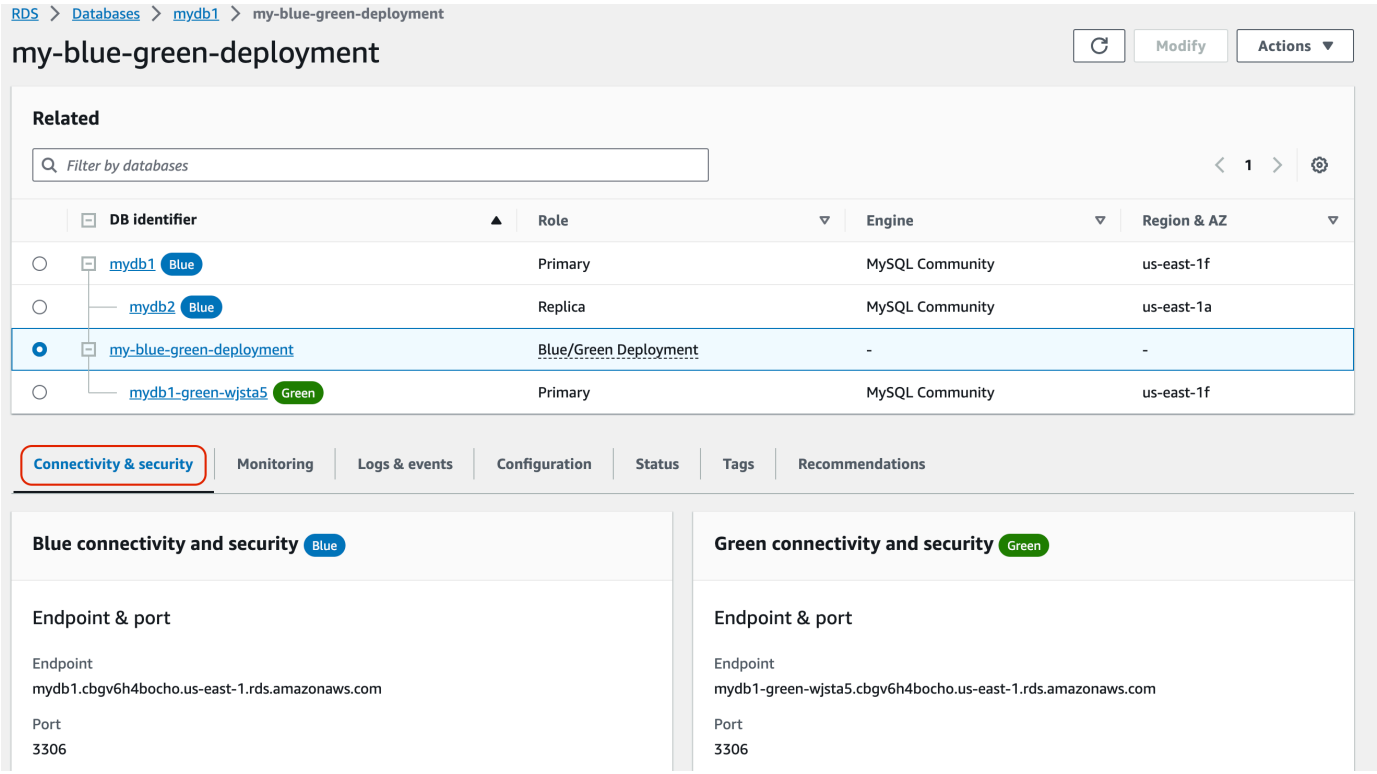
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis recherchez le déploiement bleu/vert dans la liste.



   La valeur Rôle pour le déploiement bleu/vert est Déploiement bleu/vert.
3. Choisissez le nom du déploiement bleu/vert que vous souhaitez visualiser pour afficher ses détails.

   Chaque onglet comporte une section pour le déploiement bleu et une section pour le déploiement vert. Par exemple, dans l'onglet Configuration, la version du moteur de base de données peut être différente dans l'environnement bleu et dans l'environnement vert si vous mettez à niveau la version du moteur de base de données dans l'environnement vert.

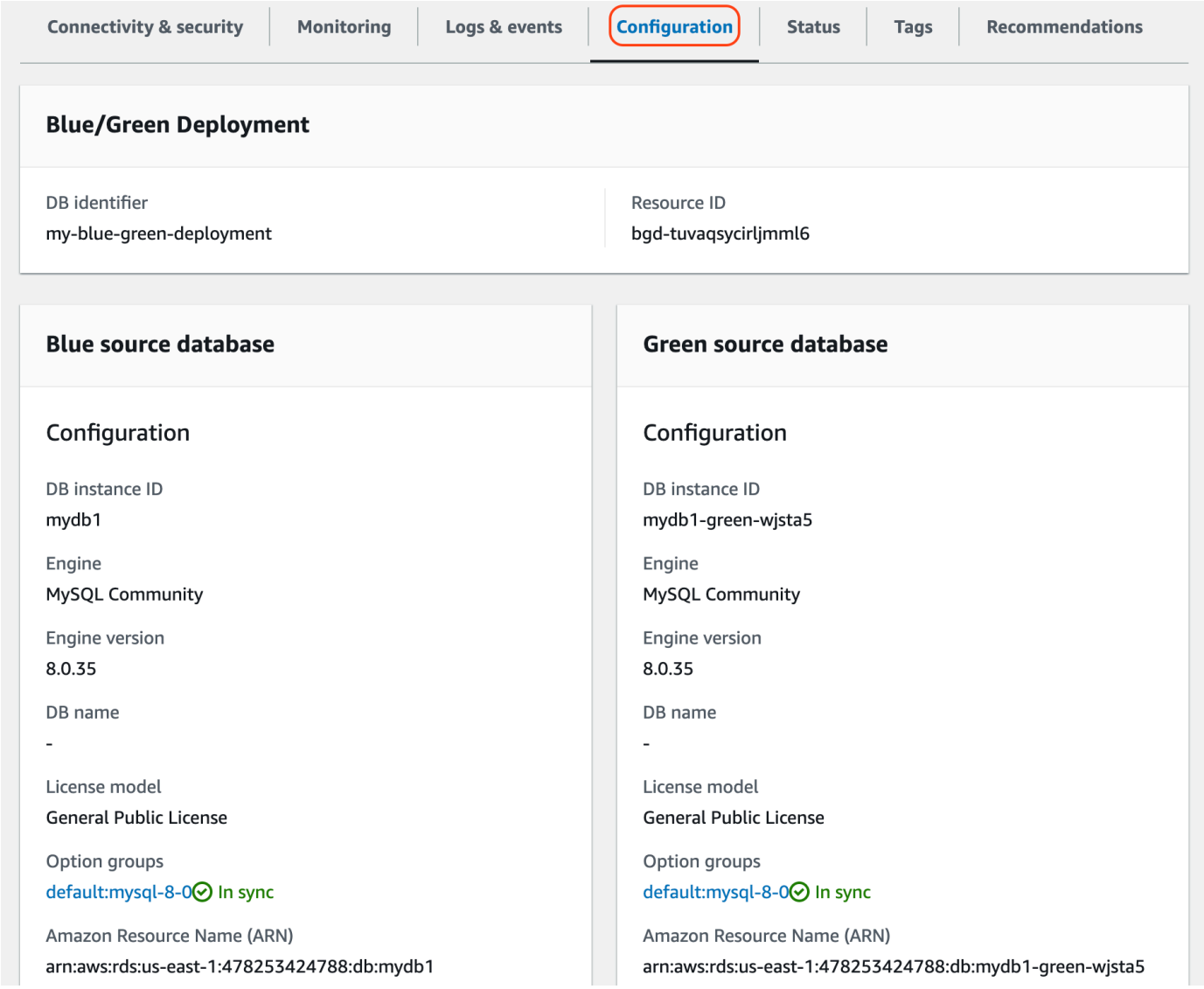
   L'image suivante montre un exemple de l'onglet Connectivité et sécurité :



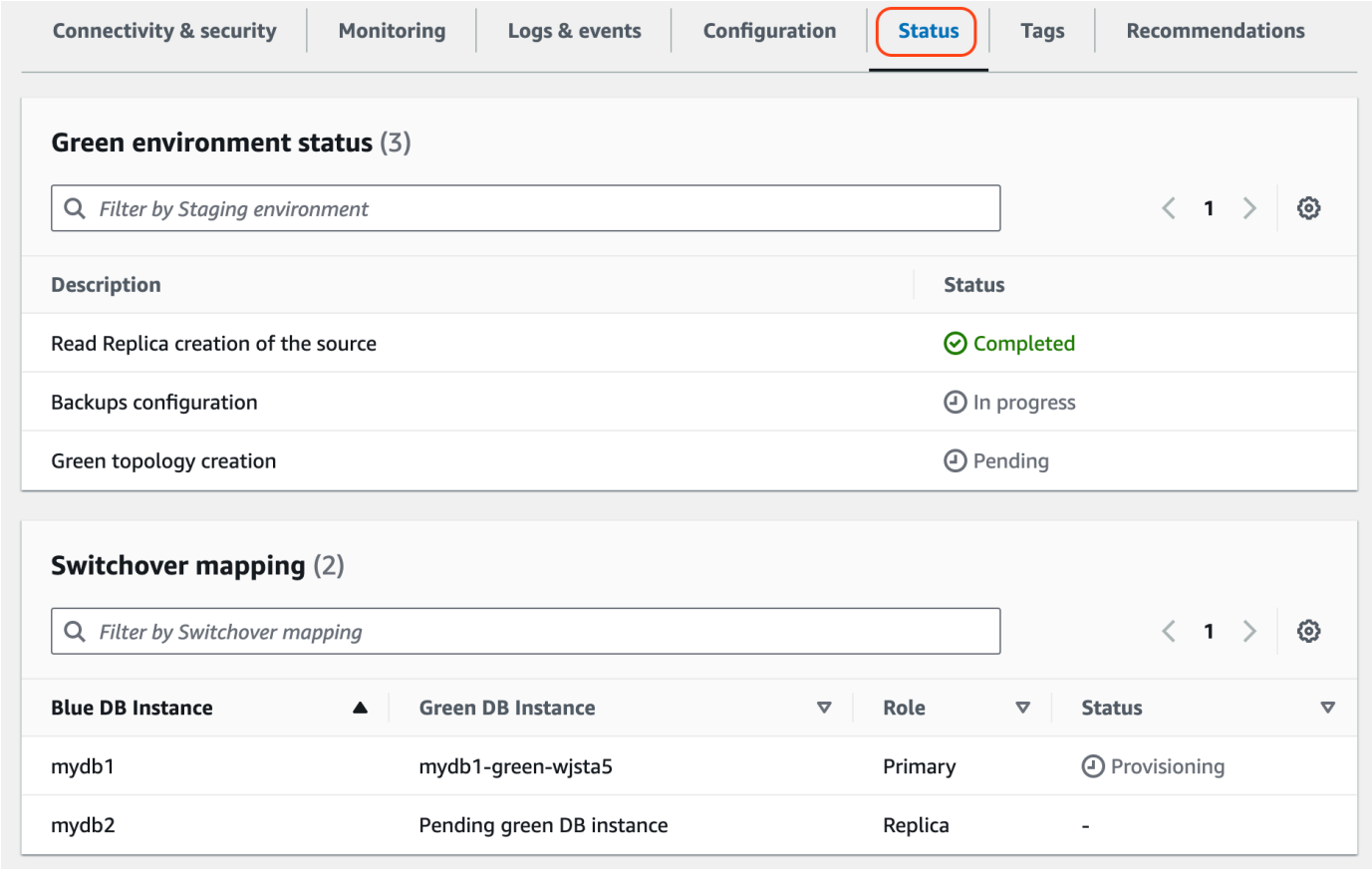
L'onglet Connectivité et sécurité inclut également une section intitulée Réplication, qui indique l'état actuel de la réplication et le décalage entre les environnements bleu et vert. Si l'état de réplication est défini sur `Replicating`, le déploiement bleu/vert se réplique correctement.

Dans le cas des déploiements bleu/vert d' RDS pour PostgreSQL qui utilisent la réplication logique, l'état de réplication peut devenir si vous apportez des modifications de DDL non prises en charge ou d'objets volumineux `Replication degraded` dans l'environnement bleu. Pour de plus amples informations, veuillez consulter the section called “Limitations spécifiques à la réplication logique pour les déploiements blue/green ”.

L'image suivante montre un exemple de l'onglet Configuration :



L'image suivante montre un exemple de l'onglet Status :



AWS CLI

Pour afficher les détails d'un déploiement bleu/vert à l'aide de AWS CLI, utilisez la describe-blue-green-deploymentscommande.

Example Affichage des détails d'un déploiement bleu/vert en filtrant sur son nom

Lorsque vous utilisez la describe-blue-green-deploymentscommande, vous pouvez filtrer sur`--blue-green-deployment-name`.

L'exemple suivant montre les détails d'un déploiement bleu/vert nommé *`my-blue-green-deployment`*.

```
aws rds describe-blue-green-deployments \
  --filters Name=blue-green-deployment-name,Values=my-blue-green-deployment
```

Example Affichage des détails d'un déploiement bleu/vert en spécifiant son identifiant

Lorsque vous utilisez la describe-blue-green-deploymentscommande, vous pouvez spécifier l'`--blue-green-deployment-identifier`option.

L'exemple suivant montre les détails d'un déploiement bleu/vert avec l'identifiant *`bgd-1234567890abcdef`*.

```
aws rds describe-blue-green-deployments \
  --blue-green-deployment-identifier bgd-1234567890abcdef
```

API RDS

Pour afficher les détails d'un déploiement bleu/vert à l'aide de l'API Amazon RDS, utilisez l'opération `DescribeBlueGreenDeployments` et spécifiez `BlueGreenDeploymentIdentifier`.

Lors du passage au numérique, l'environnement vert devient le nouvel environnement de production. Lorsque l'instance de base de données verte a lu des répliques, elles sont également transférées. Avant le basculement, le trafic de production est routé vers l'instance de base de données et les réplicas en lecture dans l'environnement bleu. Après le basculement, le trafic de production est routé vers l'instance de base de données et les réplicas en lecture dans l'environnement vert.

Le passage d'un déploiement bleu/vert n'est pas la même chose que la promotion du de base de données vert dans le déploiement bleu/vert. Si vous promouvez manuellement le de base de données vert en choisissant Promouvoir dans le menu Actions, la réplication entre les environnements bleu et vert est interrompue et le déploiement bleu/vert passe à l'état de configuration non valide.

Rubriques

- Délai de commutation
- Barrières de protection de commutation
- Actions de commutation
- Bonnes pratiques de commutation
- Vérification des CloudWatch métriques avant le passage au numérique
- Surveillance du délai de réplication avant le passage au numérique
- Basculement d'un déploiement bleu/vert

- Après la commutation

Délai de commutation

Vous pouvez spécifier un délai de commutation compris entre 30 secondes et 3 600 secondes (une heure). Si la commutation prend plus de temps que la durée spécifiée, toutes les modifications sont annulées et aucune modification n'est apportée à l'un ou l'autre des environnements. Le délai d'attente par défaut est de 300 secondes (cinq minutes).

Barrières de protection de commutation

Lorsque vous lancez une commutation, Amazon RDS effectue quelques vérifications de base pour tester la préparation des environnements bleu et vert à la commutation. Ces contrôles sont connus sous le nom de barrières de protection de commutation. Ces barrières de protection empêchent une commutation si les environnements ne sont pas prêts pour cela. Ils évitent donc un temps d'arrêt plus long que prévu et empêchent la perte de données entre les environnements bleu et vert qui pourrait survenir si la commutation était lancée.

Amazon RDS exécute les contrôles de barrière de protection suivants sur l'environnement vert :

- État de la réplication : vérifiez si l'état de réplication de l'instance de base de données principale verte est sain. L'instance de base de données principale verte est un réplica de l'instance de base de données principale bleue.
- Décalage de réplication : vérifiez si le retard de réplica de l'instance de base de données principale verte se situe dans les limites autorisées pour la commutation. Les limites autorisées sont basées sur le délai d'attente spécifié. Le retard de réplica indique dans quelle mesure l'instance de base de données principale verte est en retard sur son instance de base de données principale bleue. Pour de plus amples informations, veuillez consulter the section called “Surveillance du délai de réplication avant le passage au numérique”.
- Écritures actives : assurez-vous qu'aucune écriture n'est active sur l'instance de base de données principale verte.

Amazon RDS exécute les contrôles de barrière de protection suivants sur l'environnement bleu :

- Réplication externe : pour RDS pour PostgreSQL, assurez-vous que l'environnement bleu n'est pas une source logique autogérée (éditeur) ou une réplique (abonné). Si tel est le cas, nous vous recommandons de supprimer les emplacements de réplication autogérés et les abonnements dans

toutes les bases de données de l'environnement bleu, de procéder au basculement, puis de les recréer pour reprendre la réplication. Pour RDS pour MySQL et RDS pour MariaDB, vérifie si la base de données bleue n'est pas une réplique externe du journal binaire. Si tel est le cas, assurez-vous qu'il ne se réplique pas activement.

- Écritures actives de longue durée : assurez-vous qu'il n'y a pas d'écritures actives de longue durée sur l'instance de base de données principale bleue, car elles peuvent augmenter le retard de réplica.
- Instructions DDL de longue durée : assurez-vous qu'aucune instruction DDL de longue durée ne figure sur l'instance de base de données principale bleue, car elles peuvent augmenter le retard de réplica.
- Modifications PostgreSQL non prises en charge — Pour les déploiements bleu/vert RDS pour PostgreSQL utilisant la réplication logique, assurez-vous qu'aucune modification du DDL ni aucun ajout ou modification d'objets volumineux n'ont été effectués dans l'environnement bleu. Pour de plus amples informations, veuillez consulter the section called “Limitations spécifiques à la réplication logique pour les déploiements blue/green ”.

  Si Amazon RDS détecte des modifications PostgreSQL non prises en charge, il change l'état de réplication et vous indique que le basculement n'est pas disponible pour le déploiement `Replication degraded` et pour toutes les bases de données vertes. blue/green deployment. To proceed with switchover, we recommend that you delete and recreate the blue/green Pour ce faire, choisissez Actions, Supprimer avec les bases de données vertes.

Actions de commutation

Lorsque vous basculez un déploiement bleu/vert, RDS effectue les actions suivantes :

1. Exécute des contrôles de barrière de protection pour vérifier si les environnements bleu et vert sont prêts pour la commutation.
2. Arrête les nouvelles opérations d'écriture sur l'instance de base de données principale dans les deux environnements.
3. Supprime les connexions aux instances de base de données dans les deux environnements et ne permet pas de nouvelles connexions.
4. Attend que la réplication rattrape son retard dans l'environnement vert afin que celui-ci soit synchronisé avec l'environnement bleu.
5. Renomme les instances de base de données dans les deux environnements.

RDS renomme les instances de base de données dans l'environnement vert pour correspondre aux instances de base de données correspondantes dans l'environnement bleu. Par exemple, supposons que le nom d'une instance de base de données dans l'environnement bleu est `mydb`. Supposons également que le nom de l'instance de base de données correspondante dans l'environnement vert est `mydb-green-abc123`. Pendant la commutation, le nom de l'instance de base de données dans l'environnement vert devient `mydb`.

RDS renomme les instances de base de données dans l'environnement bleu en ajoutant `-old`*n* au nom actuel, où *n* est un nombre. Par exemple, supposons que le nom d'une instance de base de données dans l'environnement bleu est `mydb`. Après la commutation, le nom de l'instance de base de données pourrait être `mydb-old1`.

RDS renomme également les points de terminaison dans l'environnement vert pour qu'ils correspondent aux points de terminaison correspondants dans l'environnement bleu, de sorte que les changements d'application ne sont pas nécessaires.

6. Permet les connexions aux bases de données dans les deux environnements.
7. Autorise les opérations d'écriture sur le cluster de base de données dans le nouvel environnement de production.

   Après le basculement, le précédent de base de données principale de production autorise les opérations de lecture uniquement jusqu'à ce que vous définissiez le `read_only` paramètre (pour RDS pour MySQL) ou le `default_transaction_read_only` paramètre (pour RDS pour PostgreSQL) et que vous redémarriez l'instance de base de données. `0`

Vous pouvez surveiller l'état d'un passage au numérique à l'aide d'Amazon. EventBridge Pour de plus amples informations, veuillez consulter the section called “Événements de déploiement bleu/vert”.

Si vous avez configuré des balises dans l'environnement bleu, elles sont copiées dans le nouvel environnement de production lors du passage au numérique. Pour en savoir plus sur les identifications, consultez Marquage d' Amazon RDS.

Si la commutation commence et s'arrête avant la fin pour une raison quelconque, les modifications sont annulées et aucune modification n'est apportée à l'environnement.

Bonnes pratiques de commutation

Avant de basculer, nous vous recommandons vivement de respecter les bonnes pratiques en accomplissant les tâches suivantes :

- Testez minutieusement les ressources dans l'environnement vert. Assurez-vous qu'elles fonctionnent correctement et efficacement.
- Surveillez les CloudWatch statistiques Amazon pertinentes. Pour de plus amples informations, veuillez consulter the section called “Vérification des CloudWatch métriques avant le passage au numérique”.
- Déterminez le meilleur moment pour la commutation.

  Pendant la commutation, les écritures sont interrompues dans les bases de données des deux environnements. Identifiez un moment où le trafic est le plus faible dans votre environnement de production. Les transactions de longue durée, telles que les transactions actives DDLs, peuvent augmenter le temps de transition, ce qui se traduit par des temps d'arrêt plus longs pour vos charges de travail de production.

  S'il existe un grand nombre de connexions sur vos instances de base de données, votre de base de données, pensez à les réduire manuellement au minimum nécessaire pour votre application avant de passer au blue/green deployment. One way to achieve this is to create a script that monitors the status of the blue/green déploiement et de commencer à nettoyer les connexions lorsqu'il détecte que le statut est passé à`SWITCHOVER_IN_PROGRESS`.
- Assurez-vous que les instances de base de données dans les deux environnements sont dans l'état `Available`.
- Assurez-vous que l'instance de base de données principale dans l'environnement vert est dans un état sain et qu'il se réplique.
- Assurez-vous que les configurations de votre réseau et de votre client n'augmentent pas le cache DNS Time-To-Live (TTL) au-delà de cinq secondes, ce qui est la valeur par défaut pour les zones DNS RDS.
  Sinon, les applications continueront à envoyer du trafic d'écriture vers l'environnement bleu après le basculement.
- Assurez-vous que le chargement des données est terminé avant de basculer. Pour de plus amples informations, veuillez consulter the section called “Chargement différé et initialisation du stockage”.
- Pour les déploiements bleu/vert RDS pour PostgreSQL utilisant la réplication logique, procédez comme suit :
  - Passez en revue les limites de la réplication logique et prenez les mesures nécessaires avant le passage au numérique. Pour de plus amples informations, veuillez consulter the section called “Limitations spécifiques à la réplication logique pour les déploiements blue/green ”.

- Exécutez l'opération `ANALYZE` pour actualiser la table `pg_statistics`. Cela réduit le risque de problèmes de performances après le passage au numérique.

> **Note**
>
> Lors d'une commutation, vous ne pouvez modifier aucune instance de base de données incluse dans la commutation.

Vérification des CloudWatch métriques avant le passage au numérique

Avant de passer à un déploiement bleu/vert, nous vous recommandons de vérifier la valeur des métriques suivantes sur Amazon CloudWatch

- `DatabaseConnections` : utilisez cette métrique pour estimer le niveau d'activité du déploiement bleu/vert et assurez-vous que la valeur est à un niveau acceptable pour votre déploiement avant d'effectuer le basculement. Si l'analyse des performances est activée, `DBLoad` est une métrique plus précise.

Pour de plus amples informations, veuillez consulter the section called “CloudWatch métriques pour RDS”.

Surveillance du délai de réplication avant le passage au numérique

Avant de passer à un déploiement bleu/vert, assurez-vous que le délai de réplication est proche de zéro afin de réduire les temps d'arrêt.

RDS pour MySQL et RDS pour MariaDB

Pour les déploiements bleu/vert de MySQL et MariaDB, vérifiez CloudWatch la métrique dans l'environnement vert pour identifier le délai de réplication actuel. Pour de plus amples informations, veuillez consulter the section called “Diagnostic et résolution du retard entre réplicas en lecture”.

PostgreSQL

Pour les déploiements bleu/vert de PostgreSQL qui utilisent la réplication physique, the section called “Surveillance et réglage du processus de réplication” consultez les instructions permettant d'identifier le délai de réplication actuel.

Pour les déploiements bleu/vert de PostgreSQL qui utilisent la réplication logique, vérifiez `OldestReplicationSlotLag` CloudWatch la métrique dans l'environnement bleu pour identifier le délai de réplication actuel. Pour de plus amples informations, veuillez consulter the section called “Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS”.

En outre, vous pouvez exécuter la requête SQL suivante dans l'environnement bleu :

```
SELECT slot_name,
       confirmed_flush_lsn as flushed,
       pg_current_wal_lsn(),
       (pg_current_wal_lsn() - confirmed_flush_lsn) AS lsn_distance
FROM pg_catalog.pg_replication_slots
WHERE slot_type = 'logical';

slot_name        |    flushed    | pg_current_wal_lsn | lsn_distance
-----------------+---------------+--------------------+------------
logical_replica1 | 47D97/CF32980 | 47D97/CF3BAC8      | 37192
```

Le `confirmed_flush_lsn` représente le dernier numéro de séquence journal (LSN) envoyé à la réplique. Le `pg_current_wal_lsn` représente l'emplacement actuel de la base de données. Une `lsn_distance` valeur de 0 signifie que la réplique est rattrapée.

Pour savoir dans quels cas les déploiements bleu/vert utilisent la réplication physique par rapport à la réplication logique, voir. the section called “Méthodes de SQL réplication Postgre”

Basculement d'un déploiement bleu/vert

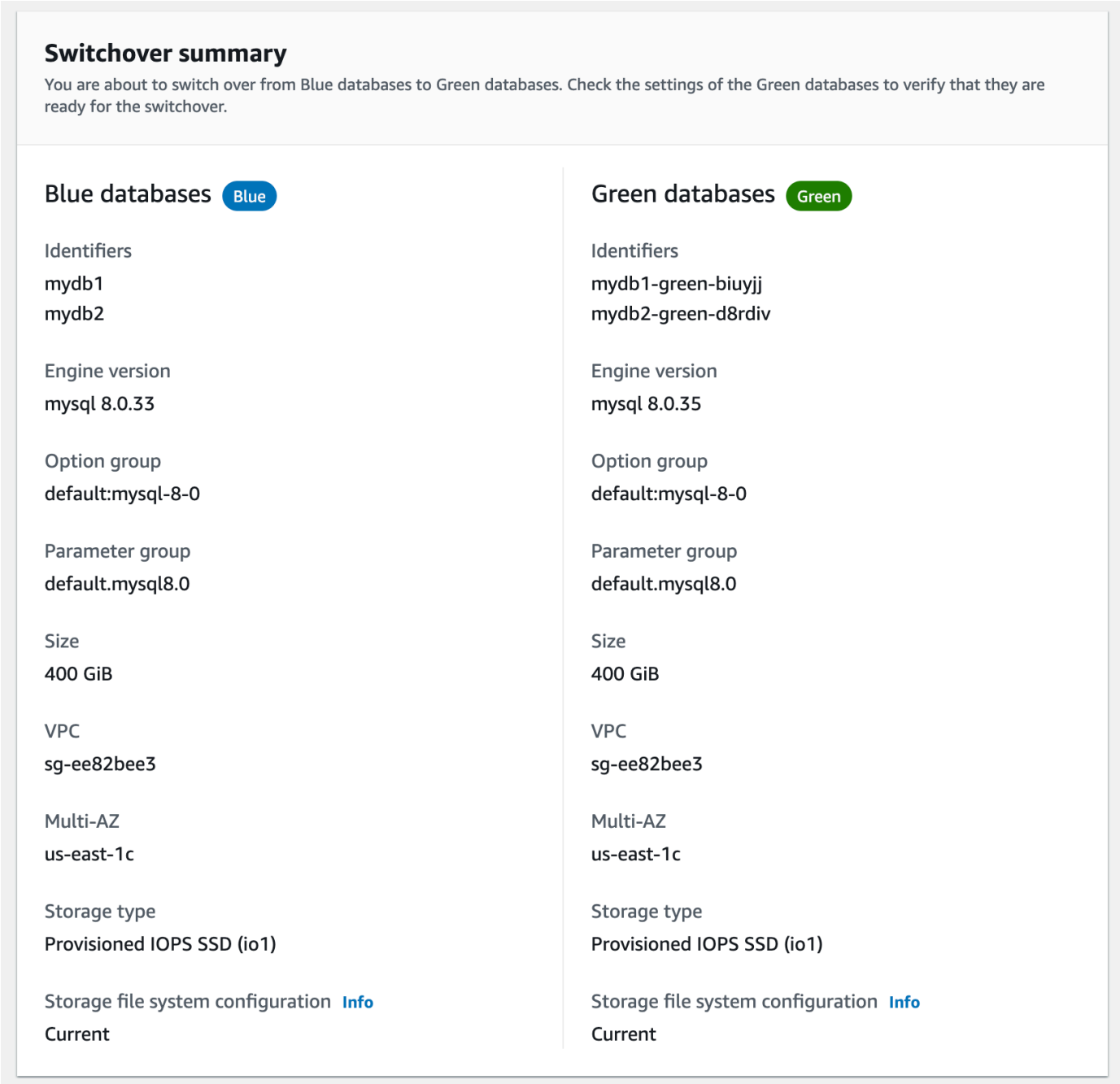
Vous pouvez passer d'un déploiement bleu/vert à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

Console

Pour effectuer un basculement de déploiement bleu/vert

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis sélectionnez le déploiement bleu/vert que vous souhaitez basculer.
3. Pour Actions, choisissez Basculer.

   La page Basculer apparaît.



4. Sur la page Basculer, consultez le résumé de la commutation. Assurez-vous que les ressources des deux environnements correspondent à ce que vous attendez. Si ce n'est pas le cas, choisissez Annuler.
5. Dans le champ Paramètre de délai d'attente, entrez le délai limite pour la commutation.
6. Si votre instance exécute RDS for PostgreSQL, passez en revue et confirmez les recommandations avant la commutation. Pour de plus amples informations, veuillez consulter the section called “Limitations spécifiques à la réplication logique pour les déploiements blue/green ”.

7. Choisissez Basculer.

AWS CLI

Pour passer d'un déploiement bleu/vert à l'aide de AWS CLI, utilisez la switchover-blue-green-deploymentcommande avec les options suivantes :

- `--blue-green-deployment-identifier`— Spécifiez l'ID de ressource du déploiement bleu/vert.
- `--switchover-timeout` : spécifiez la limite de temps pour la commutation, en secondes. La valeur par défaut est 300.

Example Basculement d'un déploiement bleu/vert

Dans Linux, macOS, ou Unix:

```
aws rds switchover-blue-green-deployment \
    --blue-green-deployment-identifier bgd-1234567890abcdef \
    --switchover-timeout 600
```

Dans Windows:

```
aws rds switchover-blue-green-deployment ^
    --blue-green-deployment-identifier bgd-1234567890abcdef ^
    --switchover-timeout 600
```

API RDS

Pour basculer un déploiement bleu/vert en utilisant l'API Amazon RDS, utilisez l'opération `SwitchoverBlueGreenDeployment` avec les paramètres suivants :

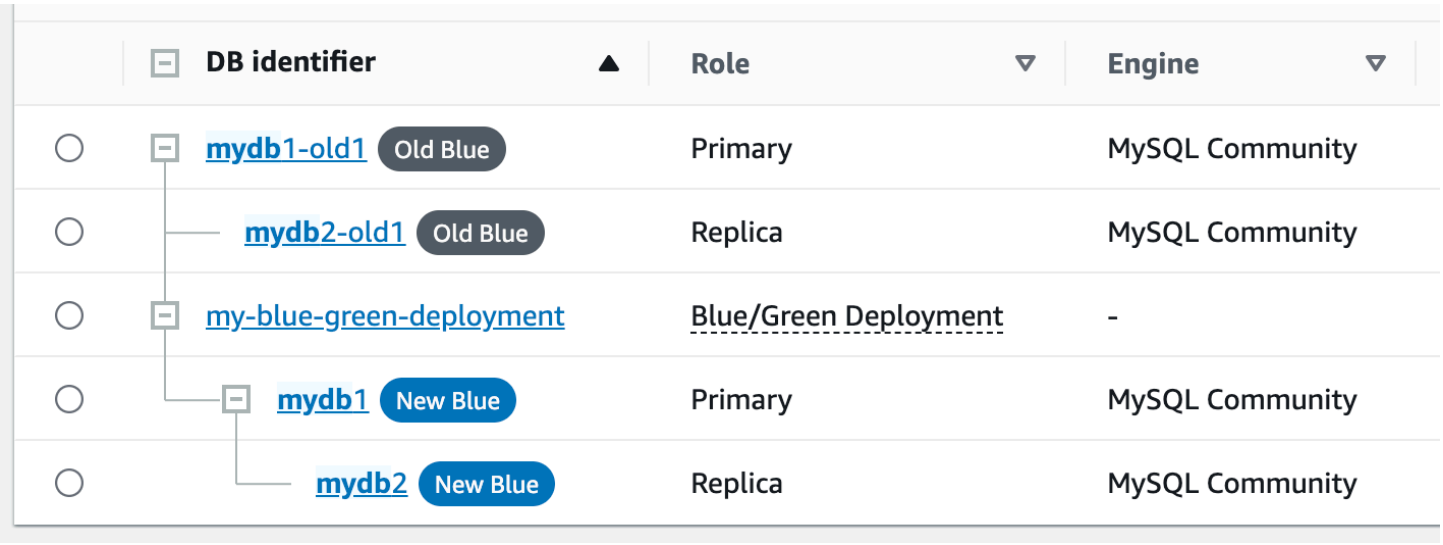
- `BlueGreenDeploymentIdentifier`— Spécifiez l'ID de ressource du déploiement bleu/vert.
- `SwitchoverTimeout` : spécifiez la limite de temps pour la commutation, en secondes. La valeur par défaut est 300.

Après la commutation

Après une commutation, les instances de base de données de l'environnement bleu précédent sont conservé(e)s. Les coûts standard s'appliquent à ces ressources. La réplication entre les environnements bleu et vert s'arrête.

RDS renomme les instances de base de données dans l'environnement bleu en ajoutant `-old`*n* au nom de la ressource actuelle, où *n* est un nombre. Les instances de base de données de l'ancien environnement bleu sont en lecture seule jusqu'à ce que vous définissiez le `read_only` paramètre (pour RDS pour MySQL) ou le `default_transaction_read_only` paramètre (pour RDS pour PostgreSQL) sur. `0` RDS nomme le de base de données dans l'environnement vert. `-new`*n*

Si vous supprimez la ressource de déploiement bleu/vert, RDS conserve les `-old`*n* ressources et. `-new`*n*



Mise à jour du nœud parent pour les consommateurs

RDS propose des répliques de lecture entièrement gérées. Cependant, il offre également la possibilité de configurer des répliques autogérées, également appelées répliques externes. Les répliques externes vous permettent d'utiliser des ressources tierces comme cibles de réplication.

Après avoir basculé vers un déploiement RDS pour MariaDB ou RDS pour MySQL MySQL de base de données contenait des répliques externes ou des consommateurs de journaux binaires avant le basculement, vous devez mettre à jour son nœud parent après le basculement afin de maintenir la continuité de la réplication.

Pour mettre à jour le nœud parent

1. Après le basculement, l'instance de base de données du qui se trouvait auparavant dans l'environnement vert émet un événement contenant le nom du fichier journal principal et la position du journal principal. Pour localiser l'événement, accédez à la console RDS et choisissez Events dans le volet de navigation de gauche.
2. Filtrez par événements dont la source est le nom de l'ancienne instance de base de données Green , avant le basculement.
3. Localisez l'événement qui contient les coordonnées binaires du journal. Le message de l'événement est similaire à :`Binary log coordinates in green environment after switchover: file mysql-bin-changelog.000003 and position 40134574`.
4. Assurez-vous que le consommateur ou la réplique a appliqué tous les journaux binaires de l'ancien environnement bleu. Utilisez ensuite les coordonnées binaires du journal fournies pour reprendre la réplication sur les consommateurs. Par exemple, si vous exécutez une réplique MySQL sur EC2, vous pouvez utiliser les commandes suivantes :

   MySQL 8.0.22 et versions inférieures majeures et mineures

   ```
   CHANGE MASTER TO MASTER_HOST='{new-writer-endpoint}', MASTER_LOG_FILE='mysql-bin-changelog.000003', MASTER_LOG_POS=40134574;
   ```

   MySQL 8.0.23 et versions supérieures majeures et mineures

   ```
   CHANGE REPLICATION SOURCE TO SOURCE_HOST='{new-writer-endpoint}',
    SOURCE_LOG_FILE='mysql-bin-changelog.000003', SOURCE_LOG_POS=40134574;
   ```

Si le client est une autre instance de base de données RDS pour MySQL ou RDS pour MariaDB, exécutez les procédures stockées suivantes dans l'ordre :

1. the section called “mysql.rds_stop_replication”
2. mysql.rds_reset_external_master (pour les versions 8.0 et antérieures) ou mysql_rds_reset_external_source (pour les versions 8.4 et supérieures)
3. mysql.rds_set_external_master (pour les versions 8.0 et antérieures) ou mysql_rds_set_external_source (pour les versions 8.4 et supérieures)
4. the section called “mysql.rds_start_replication”

Vous pouvez supprimer un déploiement bleu/vert avant ou après son basculement.

Lorsque vous supprimez un déploiement bleu/vert avant de le basculer, Amazon RDS supprime éventuellement les instances de base de données dans l'environnement vert :

- Si vous choisissez de supprimer les instances de bases de données dans l'environnement vert (`--delete-target`), veillez à ce que la protection contre la suppression ne soit pas activée pour elles.
- Si vous ne supprimez pas les instances de base de données dans l'environnement vert (`--no-delete-target`), elles sont conservées, mais ne font plus partie d'un déploiement bleu/vert. Pour RDS for MySQL , la réplication se poursuit entre les environnements. Pour RDS for PostgreSQL PostgreSQL, l'environnement écologique est promu en environnement autonome, de sorte que la réplication s'arrête.

L'option permettant de supprimer les bases de données vertes n'est pas disponible dans la console après le basculement. Lorsque vous supprimez des déploiements bleu/vert à l'aide de AWS CLI, vous ne pouvez pas spécifier l'`--delete-target`option si l'état du déploiement est. `SWITCHOVER_COMPLETED`

> **Important**
>
> La suppression d'un déploiement bleu/vert n'affecte pas l'environnement bleu.

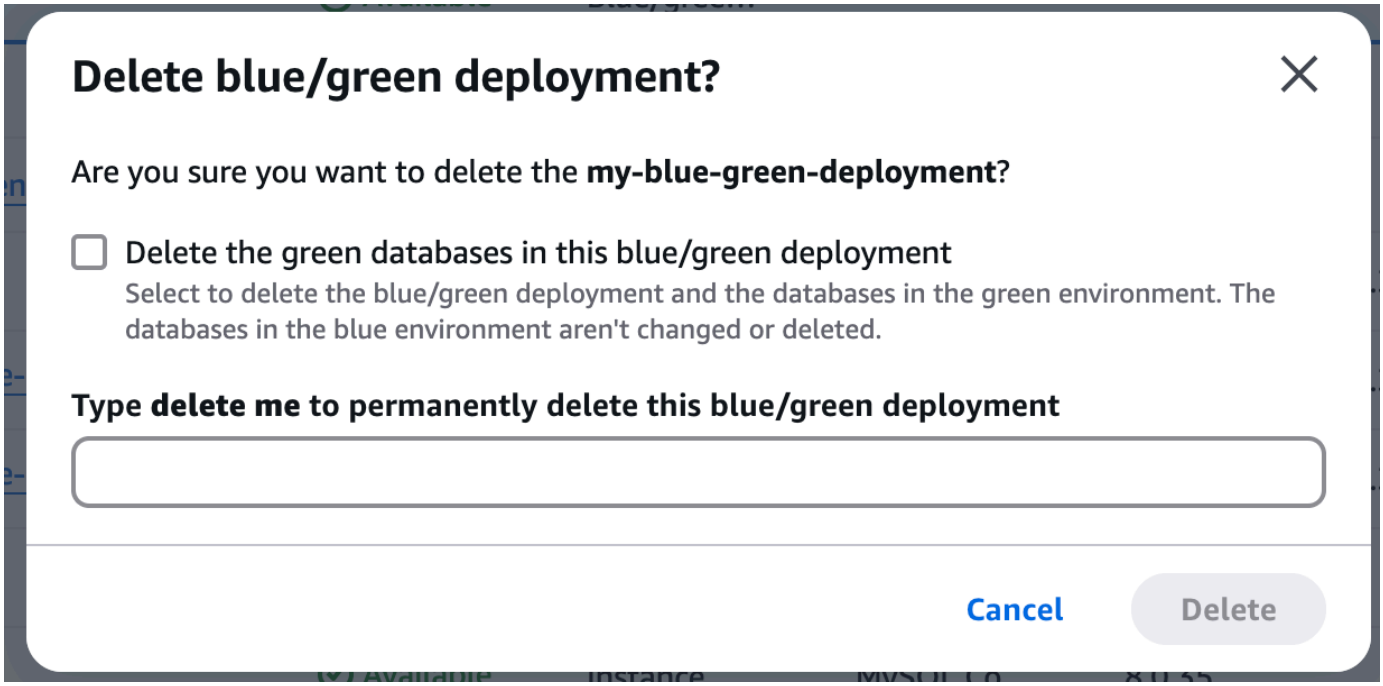
Vous pouvez supprimer un déploiement bleu/vert à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

console

Pour supprimer un déploiement bleu/vert

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données), puis choisissez le déploiement bleu/vert que vous souhaitez supprimer.
3. Pour Actions, choisissez Supprimer.

   Le déploiement bleu/vert de Delete ? une fenêtre apparaît.



Pour supprimer les bases de données vertes, sélectionnez Supprimer les bases de données vertes dans ce déploiement bleu/vert.

4. Saisissez **`delete me`** dans la zone.
5. Choisissez Supprimer.

AWS CLI

Pour supprimer un déploiement bleu/vert à l'aide de AWS CLI, utilisez la delete-blue-green-deploymentcommande avec les options suivantes :

- `--blue-green-deployment-identifier`— L'ID de ressource du déploiement bleu/vert à supprimer.
- `--delete-target` : spécifie que les instances de base de données dans l'environnement vert sont supprimées. Vous ne pouvez pas spécifier cette option si le statut du déploiement bleu/vert est `SWITCHOVER_COMPLETED`.
- `--no-delete-target` :spécifie que les instances de base de données dans l'environnement vert sont conservées.

Example Suppression d'un déploiement bleu/vert et des instances de base de données dans l'environnement vert

Dans Linux, macOS, ou Unix:

```
aws rds delete-blue-green-deployment \
    --blue-green-deployment-identifier bgd-1234567890abcdef \
    --delete-target
```

Dans Windows:

```
aws rds delete-blue-green-deployment ^
    --blue-green-deployment-identifier bgd-1234567890abcdef ^
    --delete-target
```

Example Suppression d'un déploiement bleu/vert mais conservation des instances de base de données dans l'environnement vert

Dans Linux, macOS, ou Unix:

```
aws rds delete-blue-green-deployment \
    --blue-green-deployment-identifier bgd-1234567890abcdef \
    --no-delete-target
```

Dans Windows:

```
aws rds delete-blue-green-deployment ^
    --blue-green-deployment-identifier bgd-1234567890abcdef ^
    --no-delete-target
```

API RDS

Pour supprimer un déploiement bleu/vert en utilisant l'API Amazon RDS, utilisez l'opération `DeleteBlueGreenDeployment` avec les paramètres suivants :

- `BlueGreenDeploymentIdentifier`— L'ID de ressource du déploiement bleu/vert à supprimer.
- `DeleteTarget` : spécifiez `TRUE` si vous souhaitez supprimer les instances de base de données dans l'environnement vert ou `FALSE` si vous souhaitez les conserver. Ne peut pas être `TRUE` si le statut du déploiement bleu/vert est `SWITCHOVER_COMPLETED`.

Sauvegarde, restauration et exportation de données

Cette section explique comment sauvegarder, restaurer et exporter des données à partir d'une instance de RDS base de données Amazon ou d'un cluster de base de données multi-AZ.

Pour plus d'informations sur la sauvegarde de votre instance Amazon RDS DB ou de votre cluster de base de données Multi-AZ, consultez les rubriques suivantes.

- Présentation des sauvegardes
- Gestion des sauvegardes automatisées
- Gestion des sauvegardes manuelles

Pour plus d'informations sur la restauration de votre RDS instance de base de données Amazon ou de votre cluster de base de données multi-AZ, consultezRestauration vers une instance de base de données.

Pour plus d'informations sur la copie, le partage ou l'exportation d'instantanés de base de données, consultez les rubriques suivantes.

- Copier un instantané de base de données pour Amazon RDS
- Partage d'un instantané de base de données pour Amazon RDS
- Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS

Pour plus d'informations sur l'affichage des sauvegardes automatisées gérées par AWS Backup, consultezUtilisation AWS Backup pour gérer les sauvegardes automatisées pour Amazon RDS.

Présentation des sauvegardes

Amazon RDS crée et enregistre des sauvegardes automatisées de votre instance de base de données ou de votre cluster de base de données multi-AZ pendant la fenêtre de sauvegarde de votre instance de base de données. RDScrée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de l'instance de base de données et pas uniquement les bases de données individuelles. RDSenregistre les sauvegardes automatisées de votre instance de base de données conformément à la période de conservation des sauvegardes que vous spécifiez. Si nécessaire, vous pouvez récupérer votre instance de base de données à n'importe quel moment pendant la période de conservation des sauvegardes.

Les sauvegardes automatiques suivent ces règles :

- Votre instance de base de données doit être dans l'état `available` pour que des sauvegardes automatiques soient exécutées. Les sauvegardes automatiques ne sont pas exécutées si votre instance de base de données est dans un état autre que `available`, par exemple `storage_full`.
- Les sauvegardes automatisées ne sont pas exécutées lorsqu'une copie d'un instantané de base de données est en cours d'exécution dans la même Région AWS pour la même base de données.

Vous pouvez également sauvegarder votre instance de base de données manuellement, en créant un instantané de bases de données. Pour plus d'informations sur la création manuelle d'un instantané de base de données, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Le premier instantané d'une instance de base de données contient les données de la base de données complète. Les instantanés suivants de la même base de données sont incrémentiels, ce qui signifie que seules les données qui ont changé depuis l'instantané le plus récent sont enregistrées.

Vous pouvez copier des instantanés de base de données automatiques et manuels, et partager les instantanés de base de données manuels. Pour plus d'informations sur la copie des instantanés de base de données, consultez Copier un instantané de base de données pour Amazon RDS. Pour plus d'informations sur le partage des instantanés de base de données, veuillez consulter Partage d'un instantané de base de données pour Amazon RDS.

Stockage de sauvegarde

Votre stockage de RDS sauvegarde Amazon pour chacune d'entre elles Région AWS est composé de sauvegardes automatisées et de snapshots de base de données manuels pour cette région. L'espace de stockage de sauvegarde total est égal à la somme des ressources de stockage de toutes les sauvegardes de cette région. Le déplacement d'un instantané de bases de données vers une autre région augmente le stockage de sauvegarde dans la région de destination. Les sauvegardes sont stockées dans Amazon S3.

Pour plus d'informations sur les coûts de stockage des sauvegardes, consultez RDSles tarifs Amazon.

Si vous choisissez de conserver les sauvegardes automatiques lorsque vous supprimez une instance de base de données, les sauvegardes automatiques sont enregistrées pour toute la période de conservation. Si vous ne choisissez pas Conserver les sauvegardes automatiques lors de la suppression d'une instance de base de données, toutes les sauvegardes automatiques sont supprimées en même temps que l'instance. Une fois supprimées, les sauvegardes automatiques ne peuvent pas être récupérées. Si vous choisissez de demander à Amazon de RDS créer un instantané de base de données final avant de supprimer votre instance de base de données, vous pouvez l'utiliser pour récupérer votre instance de base de données. En option, vous pouvez également utiliser un instantané manuel créé précédemment. Les instantanés manuels ne sont pas supprimés. Vous pouvez avoir jusqu'à 100 instantanés manuels par région.

Gestion des sauvegardes automatisées

Cette section explique comment gérer les sauvegardes automatisées pour les instances de base de données et les clusters de base de données multi-AZ.

Rubriques

- Fenêtre de sauvegarde
- Période de rétention des sauvegardes
- Activation des sauvegardes automatiques
- Conservation des sauvegardes automatiques
- Suppression des sauvegardes automatisées conservées
- Sauvegardes automatisées avec des moteurs My SQL Storage non pris en charge
- Sauvegardes automatiques avec moteurs de stockage MariaDB non pris en charge
- Réplication des sauvegardes automatisées vers une autre Région AWS

Fenêtre de sauvegarde

Les sauvegardes automatiques sont exécutées chaque jour pendant la fenêtre de sauvegarde préférée. Si la sauvegarde a besoin de plus de temps que la durée allouée par la fenêtre de sauvegarde, elle continue après la fin de la fenêtre jusqu'à ce qu'elle soit terminée. La fenêtre de sauvegarde ne peut pas chevaucher la fenêtre de maintenance hebdomadaire pour l'instance de base de données ou le cluster de base de données multi-AZ.

Pendant la fenêtre de sauvegarde automatique, les I/O de stockage peuvent être suspendues brièvement tandis que le processus de sauvegarde s'initialise (généralement en quelques secondes). Vous pouvez rencontrer des latences élevées pendant quelques minutes lors de sauvegardes de déploiements multi-AZ. Pour MariaDB, SQL My, Oracle et SQL Postgre, l'activité d'E/S n'est pas suspendue sur votre serveur principal pendant la sauvegarde pour les déploiements multi-AZ, car la sauvegarde est prise depuis le mode veille. Pour le SQL serveur, l'activité d'E/S est brièvement suspendue pendant la sauvegarde, tant pour les déploiements mono-AZ que pour les déploiements multi-AZ, car la sauvegarde est prise depuis le serveur principal. Pour Db2, l'activité d'E/S est également suspendue brièvement pendant la sauvegarde, même si la sauvegarde est prise depuis le mode veille.

Il peut arriver que des sauvegardes automatisées soient ignorées si l'instance ou le cluster de base de données a une charge de travail importante au moment où une sauvegarde est censée

démarrer. Si une sauvegarde est ignorée, vous pouvez toujours effectuer une point-in-time-recovery (PITR), et une sauvegarde est toujours tentée lors de la fenêtre de sauvegarde suivante. Pour plus d'informations surPITR, voirRestauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Si vous ne spécifiez pas de fenêtre de sauvegarde préférée lorsque vous créez l'instance de base de données ou le cluster de base de données multi-AZ, Amazon RDS attribue une fenêtre de sauvegarde par défaut de 30 minutes. Cette fenêtre est sélectionnée au hasard sur une période de 8 heures pour chacune Région AWS d'entre elles. Le tableau suivant répertorie les plages temporelles pour chacune des périodes Région AWS à partir desquelles les fenêtres de sauvegarde par défaut sont attribuées.

Nom de la région	Région	Bloc chronologique
US East (Virginie du Nord)	us-east-1	03 H 00 — 11 H 00 UTC
USA Est (Ohio)	us-east-2	03 H 00 — 11 H 00 UTC
USA Ouest (Californie du Nord)	us-west-1	6 H 00 — 14 H 00 UTC
USA Ouest (Oregon)	us-west-2	6 H 00 — 14 H 00 UTC
Afrique (Le Cap)	af-south-1	03 H 00 — 11 H 00 UTC
Asie-Pacifique (Hong Kong)	ap-east-1	6 H 00 — 14 H 00 UTC
Asie-Pacifique (Hyderabad)	ap-south-2	6 H 30 — 14 H 30 UTC
Asie-Pacifique (Jakarta)	ap-southeast-3	8 H 00 — 16 H 00 UTC
Asie-Pacifique (Malaisie)	ap-southeast-5	9 H 00 — 17 H 00 UTC

Nom de la région	Région	Bloc chronologique
Asie-Pacifique (Melbourne)	ap-southeast-4	11 H 00 — 19 H 00 UTC
Asie-Pacifique (Mumbai)	ap-south-1	16 H 30 — 00 H 30 UTC
Asie-Pacifique (Osaka)	ap-northeast-3	00:00 — 08:00 UTC
Asie-Pacifique (Séoul)	ap-northeast-2	13 H 00 — 21 H 00 UTC
Asie-Pacifique (Singapour)	ap-southeast-1	14 H 00 — 22 H 00 UTC
Asie-Pacifique (Sydney)	ap-southeast-2	12 H 00 — 20 H 00 UTC
Asie-Pacifique (Tokyo)	ap-northeast-1	13 H 00 — 21 H 00 UTC
Canada (Centre)	ca-central-1	03 H 00 — 11 H 00 UTC
Canada Ouest (Calgary)	ca-west-1	18 H 00 — 2 H 00 UTC
Chine (Beijing)	cn-north-1	6 H 00 — 14 H 00 UTC
Chine (Ningxia)	cn-northwest-1	6 H 00 — 14 H 00 UTC
Europe (Francfort)	eu-central-1	20 H 00 — 4 H 00 UTC
Europe (Irlande)	eu-west-1	22 H 00 — 6 H 00 UTC
Europe (Londres)	eu-west-2	22 H 00 — 6 H 00 UTC
Europe (Milan)	eu-south-1	02H00 — 10H00 UTC
Europe (Paris)	eu-west-3	07 H 29 — 14 H 29 UTC

Nom de la région	Région	Bloc chronologique
Europe (Espagne)	eu-south-2	02H00 — 10H00 UTC
Europe (Stockholm)	eu-north-1	23 H 00 — 7 H 00 UTC
Europe (Zurich)	eu-central-2	02H00 — 10H00 UTC
Israël (Tel Aviv)	il-central-1	03 H 00 — 11 H 00 UTC
Moyen-Orient (Bahreïn)	me-south-1	6 H 00 — 14 H 00 UTC
Moyen-Orient (UAE)	me-central-1	05 H 00 — 13 H 00 UTC
Amérique du Sud (São Paulo)	sa-east-1	23 H 00 — 7 H 00 UTC
AWS GovCloud (USA Est)	us-gov-east-1	17 H 00 — 1 H 00 UTC
AWS GovCloud (US-Ouest)	us-gov-west-1	6 H 00 — 14 H 00 UTC

Période de rétention des sauvegardes

Vous pouvez configurer une période de rétention des sauvegardes lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ. Si vous créez une instance de base de données à l'aide d'Amazon RDS API ou du AWS CLI et si vous ne définissez pas de période de conservation des sauvegardes, la période de rétention des sauvegardes par défaut est d'un jour. Si vous créez une instance de base de données à l'aide de la console, la période de conservation des sauvegardes par défaut est de sept jours.

Après avoir créé une instance ou un cluster de base de données, vous pouvez modifier la période de rétention des sauvegardes. Vous pouvez définir la période de rétention des sauvegardes d'une instance de base de données entre 0 et 35 jours. Le réglage de la période de rétention des sauvegardes sur 0 désactive les sauvegardes automatisées. Pour un cluster de base de données multi-AZ, vous pouvez définir la période de rétention des sauvegardes entre 1 et 35 jours. La limite des instantanés manuels (100 par région) ne s'applique pas aux sauvegardes automatiques.

> **Important**
>
> Une panne se produit si vous modifiez la période de rétention des sauvegardes d'une instance de base de données de 0 à une valeur différente de zéro ou d'une valeur non nulle à 0.

RDSn'inclut pas le temps passé dans l'`stopped`état dans lequel la période de conservation des sauvegardes est calculée. Les sauvegardes automatisées ne sont pas créées lorsqu'une instance ou un cluster de base de données est arrêté. Les sauvegardes peuvent être conservées au-delà de la période de conservation des sauvegardes si une instance de base de données a été arrêtée.

Activation des sauvegardes automatiques

Si les sauvegardes automatiques ne sont pas activées pour votre instance de base de données, vous pouvez les activer à tout moment. Pour activer les sauvegardes automatiques, vous devez définir la période de rétention des sauvegardes sur une valeur positive différente de zéro. Lorsque les sauvegardes automatiques sont activées, votre instance de base de données est mise hors ligne et une sauvegarde est immédiatement créée.

> **Note**
>
> Si vous gérez vos sauvegardes dans AWS Backup, vous ne pouvez pas activer les sauvegardes automatisées. Pour de plus amples informations, veuillez consulter Utilisation AWS Backup pour gérer les sauvegardes automatisées pour Amazon RDS.

console

Pour activer immédiatement les sauvegardes automatiques

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données ou le cluster de base de données multi-AZ que vous souhaitez modifier.
3. Sélectionnez Modifier.
4. Pour la période de conservation des sauvegardes, choisissez une valeur positive différente de zéro, par exemple trois jours.

5. Choisissez Continuer.
6. Choisissez Apply immediately (Appliquer immédiatement).
7. Choisissez Modifier l'instance de base de données ou Modifier le cluster pour enregistrer vos modifications et activer les sauvegardes automatisées.

AWS CLI

Pour activer les sauvegardes automatisées, utilisez la modify-db-clustercommande AWS CLI modify-db-instanceor.

Incluez les paramètres suivants :

- `--db-instance-identifier` (ou `--db-cluster-identifier` pour un cluster de base de données multi-AZ)
- `--backup-retention-period`
- `--apply-immediately` ou `--no-apply-immediately`

Dans l'exemple suivant, nous activons les sauvegardes automatiques en définissant la période de rétention des sauvegardes sur trois jours. Les modifications sont appliquées immédiatement.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my_db_instance  \
    --backup-retention-period 3 \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my_db_instance  ^
    --backup-retention-period 3 ^
    --apply-immediately
```

API RDS

Pour activer les sauvegardes automatisées, utilisez l'DBClusteropération Modify DBInstance ou Modify de l'API RDS avec les paramètres obligatoires suivants :

- `DBInstanceIdentifier` ou `DBClusterIdentifier`
- `BackupRetentionPeriod`

Affichage des sauvegardes automatiques

Pour afficher vos sauvegardes automatisées, choisissez Automated backups (Sauvegardes automatisées) dans le panneau de navigation. Pour afficher des instantanés individuels associés à une sauvegarde automatisée, choisissez Snapshots (Instantanés) dans le panneau de navigation. Vous pouvez également décrire des instantanés individuels associés à une sauvegarde automatique. À partir de là, vous pouvez restaurer une instance de base de données directement à partir d'un de ces instantanés.

Les noms des instantanés automatisés suivent le modèle`rds:<database-name>-yyyy-mm-dd-hh-mm`, en `yyyy-mm-dd-hh-mm` représentant la date et l'heure de création de l'instantané.

Pour décrire les sauvegardes automatisées de vos instances de base de données existantes à l'aide de AWS CLI, utilisez l'une des commandes suivantes :

```
aws rds describe-db-instance-automated-backups --db-instance-
identifier DBInstanceIdentifier
```

or

```
aws rds describe-db-instance-automated-backups --dbi-resource-id DbiResourceId
```

Pour décrire les sauvegardes automatiques conservées de vos instances de base de données existantes à l'aide de l'API RDS, appelez l'action `DescribeDBInstanceAutomatedBackups` avec un des paramètres suivants :

- `DBInstanceIdentifier`
- `DbiResourceId`

Conservation des sauvegardes automatiques

> **Note**
>
> Vous pouvez conserver uniquement les sauvegardes automatisées d'instances de base de données, et non de clusters de bases de données multi-AZ.

Vous pouvez choisir de conserver les sauvegardes automatisées lorsque vous supprimez une instance de base de données. Les sauvegardes automatisées peuvent être conservées pendant un nombre de jours égal à la période de conservation des sauvegardes définie pour l'instance de base de données au moment où vous la supprimez.

Les sauvegardes automatiques conservées contiennent des instantanés système et des journaux de transactions d'une instance de base de données. Elles incluent également les propriétés de votre instance de base de données (stockage alloué et classe de l'instance, par exemple) nécessaires pour la restaurer en tant qu'instance active.

Les sauvegardes automatisées conservées et les instantanés manuels sont facturés tant qu'ils ne sont pas supprimés. Pour de plus amples informations, veuillez consulter Coûts de conservation.

Vous pouvez conserver des sauvegardes automatisées pour les RDS instances exécutant les moteurs Db2, MariaDB, SQL My, SQL Postgre, Oracle et Microsoft Server. SQL

Vous pouvez restaurer ou supprimer les sauvegardes automatisées conservées à l'aide des RDS API boutons AWS Management Console, et AWS CLI.

Rubriques

- Période de conservation
- Affichage des sauvegardes retenues
- Restauration
- Coûts de conservation
- Limites

Période de conservation

Les instantanés système et les journaux de transactions contenus dans une sauvegarde automatique conservée expirent de la même façon que pour l'instance de base de données source. Dans la

mesure où aucun nouvel instantané ni journal n'est créé pour cette instance, les sauvegardes automatiques conservées finissent par toutes expirer. En effet, elles perdurent aussi longtemps que l'aurait fait leur dernier instantané système, sur la base des paramètres définis pour la période de rétention de l'instance source au moment où vous l'avez supprimée. Les sauvegardes automatiques conservées sont supprimées par le système après expiration de leur dernier instantané système.

Vous pouvez supprimer une sauvegarde automatique conservée de la même manière que vous supprimez une instance de base de données. Vous pouvez supprimer les sauvegardes automatisées conservées à l'aide de la console ou de l'RDSAPIopération`DeleteDBInstanceAutomatedBackup`.

Les instantanés finaux sont indépendants des sauvegardes automatiques conservées. Nous vous recommandons vivement de réaliser un instantané final même si vous conservez les sauvegardes automatisées car celles-ci finissent par expirer. L'instantané final n'expire jamais.

Affichage des sauvegardes retenues

Pour afficher vos sauvegardes automatisées conservées, choisissez Automated backups (Sauvegardes automatisées) dans le panneau de navigation, puis sélectionnez Retained (Conservées). Pour afficher des instantanés individuels associés à une sauvegarde automatisée conservée, choisissez Snapshots (Instantanés) dans le panneau de navigation. Vous pouvez également décrire les instantanés individuels associés à une sauvegarde automatique conservée. À partir de là, vous pouvez restaurer une instance de base de données directement à partir d'un de ces instantanés.

Pour décrire vos sauvegardes automatisées conservées à l'aide de AWS CLI, utilisez la commande suivante :

```
aws rds describe-db-instance-automated-backups --dbi-resource-id DbiResourceId
```

Pour décrire vos sauvegardes automatisées conservées à l'aide du RDSAPI, appelez l'`DescribeDBInstanceAutomatedBackups`action avec le `DbiResourceId` paramètre.

Restauration

Pour plus d'informations sur la restauration d'instances de base de données à partir de sauvegardes automatiques, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Coûts de conservation

Le coût d'une sauvegarde automatique conservée correspond au coût du stockage total des instantanés système associés à la sauvegarde. Il n'y a pas de frais supplémentaires pour les journaux de transactions ou les métadonnées de l'instance. Toutes les autres règles de tarification des sauvegardes s'appliquent aux instances restaurables.

Par exemple, imaginons que l'espace de stockage total alloué pour les instances en cours d'exécution soit de 100 Go. Supposons également que vous ayez 50 Go d'instantanés manuels plus 75 Go d'instantanés système associés à une sauvegarde automatique conservée. Dans ce cas, seuls les 25 Go supplémentaires de stockage de sauvegarde vous sont facturés, comme suit : (50 Go + 75 Go) – 100 Go = 25 Go.

Limites

Les limitations suivantes s'appliquent aux sauvegardes automatiques conservées :

- Le nombre maximum de sauvegardes automatisées conservées dans une AWS région est de 40. Ce nombre n'est pas inclus dans le quota d'instances de base de données. Vous pouvez avoir 40 instances de base de données en cours d'exécution et 40 sauvegardes automatiques conservées supplémentaires en même temps.
- Les sauvegardes automatiques conservées ne contiennent pas d'informations sur les paramètres ou les groupes d'options.
- Vous pouvez restaurer une instance supprimée à une date déterminée dans le passé, comprise dans la période de conservation au moment de la suppression.
- Vous ne pouvez pas modifier une sauvegarde automatique conservée car elle contient des sauvegardes système, des journaux de transactions et des propriétés d'instance de base de données qui existaient au moment de la suppression de l'instance source.

Suppression des sauvegardes automatisées conservées

Vous pouvez supprimer les sauvegardes automatiques conservées quand elles ne sont plus nécessaires.

Console

Pour supprimer une sauvegarde automatisée conservée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Sous l'onglet Conservé, choisissez la sauvegarde automatique conservée que vous souhaitez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Dans la page de confirmation, entrez **`delete me`** et choisissez Delete (Supprimer).

AWS CLI

Vous pouvez supprimer une sauvegarde automatique conservée à l'aide de la AWS CLI commande delete-db-instance-automated-backup avec l'option suivante :

- `--dbi-resource-id` – Identifiant de la ressource de l'instance de base de données source.

  Vous pouvez trouver l'identifiant de ressource pour l'instance de base de données source d'une sauvegarde automatique conservée en exécutant la AWS CLI commande describe-db-instance-automated-backups.

Example

L'exemple suivant supprime la sauvegarde automatisée conservée avec l'identifiant de ressource d'instance de base de données `db-123ABCEXAMPLE`.

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance-automated-backup \
    --dbi-resource-id db-123ABCEXAMPLE
```

Dans Windows:

```
aws rds delete-db-instance-automated-backup ^
    --dbi-resource-id db-123ABCEXAMPLE
```

RDS API

Vous pouvez supprimer une sauvegarde automatique conservée à l'aide de l'RDSAPIopération Amazon D eleteDBInstance AutomatedBackup avec le paramètre suivant :

- `DbiResourceId` – Identifiant de la ressource de l'instance de base de données source.

  Vous pouvez trouver l'identifiant de ressource pour l'instance de base de données source d'une sauvegarde automatique conservée à l'aide de l'RDSAPIopération Amazon escribeDBInstanceAutomatedBackupsD.

Désactivation des sauvegardes automatiques

Dans certains cas, vous pouvez avoir besoin de désactiver temporairement les sauvegardes automatiques, par exemple lorsque vous chargez un important volume de données.

> **Important**
>
> Nous vous déconseillons vivement de désactiver les sauvegardes automatisées, car cela point-in-time désactive la restauration. Le fait de désactiver les sauvegardes automatiques pour une instance de base de données ou un cluster de base de données multi-AZ supprime toutes les sauvegardes automatisées existantes pour la base de données. Si vous désactivez puis réactivez les sauvegardes automatiques, aucune restauration ne peut avoir lieu tant que ces dernières ne sont pas réactivées.

Console

Pour désactiver immédiatement les sauvegardes automatiques

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données ou le cluster de base de données multi-AZ que vous souhaitez modifier.
3. Sélectionnez Modifier.
4. Pour la Période de rétention des sauvegardes, choisissez 0 jours.
5. Choisissez Continuer.
6. Choisissez Apply immediately (Appliquer immédiatement).

7. Choisissez Modifier l'instance de base de données ou Modifier le cluster pour enregistrer vos modifications et désactiver les sauvegardes automatisées.

AWS CLI

Pour désactiver immédiatement les sauvegardes automatisées, utilisez la modify-db-clustercommande modify-db-instanceor et définissez la période de conservation des sauvegardes sur 0 avec`--apply-immediately`.

Example

L'exemple suivant désactive immédiatement les sauvegardes automatiques sur un cluster de base de données multi-AZ.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-cluster \
    --db-cluster-identifier mydbcluster \
    --backup-retention-period 0 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-cluster ^
    --db-cluster-identifier mydbcluster ^
    --backup-retention-period 0 ^
    --apply-immediately
```

Pour savoir quand la modification prend effet, appelez `describe-db-instances` pour l'instance de base de données (ou `describe-db-clusters` pour un cluster de base de données multi-AZ) jusqu'à ce que la valeur de la période de rétention des sauvegardes soit 0 et que l'état `mydbcluster` soit disponible.

```
aws rds describe-db-clusters --db-cluster-identifier mydcluster
```

RDS API

Pour désactiver immédiatement les sauvegardes automatiques, appelez l'odifyDBClusteropération M odifyDBInstance ou M avec les paramètres suivants :

- `DBInstanceIdentifier = mydbinstance` (ou `DBClusterIdentifier = mydbcluster`)

- `BackupRetentionPeriod = 0`

Example

```
https://rds.amazonaws.com/
    ?Action=ModifyDBInstance
    &DBInstanceIdentifier=mydbinstance
    &BackupRetentionPeriod=0
    &SignatureVersion=2
    &SignatureMethod=HmacSHA256
    &Timestamp=2009-10-14T17%3A48%3A21.746Z
    &AWSAccessKeyId=<&AWS; Access Key ID>
    &Signature=<Signature>
```

Sauvegardes automatisées avec des moteurs My SQL Storage non pris en charge

Pour le moteur My SQL DB, les sauvegardes automatisées ne sont prises en charge que pour le moteur de stockage InnoDB. L'utilisation de ces fonctionnalités avec d'autres moteurs SQL de stockage My, y compris MyISAM, peut entraîner un comportement peu fiable lors de la restauration à partir de sauvegardes. Plus précisément, étant donné que les moteurs de stockage tels que My ISAM ne permettent pas une restauration fiable en cas de panne, vos tables peuvent être corrompues en cas de panne. Pour cette raison, nous vous invitons à utiliser le moteur de stockage InnoDB.

- Pour convertir mes ISAM tables existantes en tables InnoDB, vous pouvez utiliser la `ALTER TABLE` commande, par exemple : `ALTER TABLE` *`table_name`* `ENGINE=innodb, ALGORITHM=COPY;`
- Si vous choisissez d'utiliser MyISAM, vous pouvez essayer de réparer manuellement les tables endommagées à la suite d'un crash à l'aide de la `REPAIR` commande. Pour plus d'informations, consultez la REPAIRTABLEdéclaration dans la section Ma SQL documentation. Cependant, comme indiqué dans la SQL section Ma documentation, il est fort probable que vous ne puissiez pas récupérer toutes vos données.
- Si vous souhaitez prendre un instantané de vos ISAM tables Mes tables avant de procéder à la restauration, procédez comme suit :
  1. Arrêtez toute activité sur Mes ISAM tables (c'est-à-dire fermez toutes les sessions).

     Vous pouvez fermer toutes les sessions en appelant la commande mysql.rds_kill pour chaque processus retourné à partir de la commande `SHOW FULL PROCESSLIST`.

2. Verrouillez et videz chacune de vos ISAM tables Mes tables. Par exemple, les commandes suivantes verrouillent et vident deux tables nommées `myisam_table1` et `myisam_table2` :

```
mysql> FLUSH TABLES myisam_table, myisam_table2 WITH READ LOCK;
```

3. Créez un instantané de votre instance de base de données ou cluster de base de données multi-AZ. Lorsque l'instantané est terminé, relâchez les verrous et reprenez l'activité sur Mes ISAM tables. Vous pouvez libérer les verrous sur vos tables à l'aide de la commande suivante :

```
mysql> UNLOCK TABLES;
```

Ces étapes obligent My ISAM à vider les données stockées en mémoire sur le disque, ce qui garantit un bon démarrage lors de la restauration à partir d'un instantané de base de données. Pour plus d'informations sur la création d'un instantané de base de données, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Sauvegardes automatiques avec moteurs de stockage MariaDB non pris en charge

Pour le moteur de base de données MariaDB, les sauvegardes automatiques sont uniquement prises en charge avec le moteur de stockage InnoDB. L'utilisation de ces fonctions avec d'autres moteurs de stockage MariaDB, dont Aria, peut entraîner un comportement non fiable lors de la restauration à partir de sauvegardes. Même si Aria est une alternative résistante aux pannes à MyISAM, vos tables peuvent toujours être corrompues en cas de plantage. Pour cette raison, nous vous invitons à utiliser le moteur de stockage InnoDB.

- Pour convertir les tables Aria existantes en tables InnoDB, vous pouvez utiliser la commande `ALTER TABLE`. Par exemple : `ALTER TABLE table_name ENGINE=innodb, ALGORITHM=COPY;`
- Si vous choisissez d'utiliser Aria, vous pouvez essayer de réparer manuellement les tables endommagées après un incident à l'aide de la commande `REPAIR TABLE`. Pour plus d'informations, consultez http://mariadb.com/kb/en/mariadb/repair-table/.
- Si vous souhaitez prendre un instantané de vos tables Aria avant la restauration, procédez comme suit :

  1. Arrêtez toutes les activités de vos tables Aria (autrement dit, fermez toutes les sessions).

2. Verrouillez et videz chacune de vos tables Aria.
3. Créez un instantané de votre instance de base de données ou cluster de base de données multi-AZ. Une fois l'instantané terminé, libérez les verrous et reprenez l'activité sur les tables Aria. Ces étapes obligent Aria à vider sur disque les données stockées en mémoire et à garantir ainsi un démarrage propre lors d'une restauration à partir d'un instantané de base de données.

Réplication des sauvegardes automatisées vers une autre Région AWS

Pour renforcer les capacités de reprise après sinistre, vous pouvez configurer votre instance de base de données Amazon RDS pour répliquer les instantanés et les journaux de transactions vers la destination Région AWS de votre choix. Lorsque la réplication des sauvegardes est configurée sur une instance de base de données, RDS lance une copie inter-région de tous les instantanés et journaux de transactions dès qu'ils sont prêts sur l'instance de base de données.

Des frais de copie des instantanés de bases de données s'appliquent au transfert des données. Une fois l'instantané de base de données copié, des frais standard s'appliquent au stockage dans la région de destination. Pour plus d'informations, consultez Tarification de RDS.

Pour un exemple d'utilisation de la réplication de sauvegarde, consultez la présentation technique AWS en ligne Managed Disaster Recovery with Amazon RDS for Oracle Cross-Region Automated Backups.

> **Note**
>
> La réplication automatique des sauvegardes n'est pas prise en charge pour les clusters de bases de données multi-AZ. Cependant, il est pris en charge pour les déploiements d'instances de base de données multi-AZ. Pour plus d'informations sur les limites des sauvegardes automatisées, consultezQuotas et contraintes pour Amazon RDS.

Pour plus d'informations sur la configuration et la gestion des sauvegardes automatisées pour Amazon RDS, consultez les rubriques suivantes.

Rubriques

- Activation des sauvegardes automatisées entre régions pour Amazon RDS
- Recherche d'informations sur les sauvegardes répliquées pour Amazon RDS
- Restauration à une heure spécifiée à partir d'une sauvegarde répliquée pour Amazon RDS
- Arrêt de la réplication automatique des sauvegardes pour Amazon RDS
- Suppression de sauvegardes répliquées pour Amazon RDS

Prise en charge du déploiement multi-AZ

La rétroréplication automatique entre régions est prise en charge pour les déploiements d'instances de base de données multi-AZ pour les moteurs suivants :

- RDS pour Db2
- RDS for MariaDB
- RDS for MySQL
- RDS for Oracle
- RDS for PostgreSQL
- RDS for SQL Server

La réplication de sauvegarde automatique entre régions n'est pas prise en charge pour les clusters de bases de données multi-AZ.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions avec les sauvegardes automatisées inter-régions, consultez Régions et moteurs de base de données pris en charge pour les sauvegardes automatisées entre régions dans Amazon RDS.

Région AWS Assistance à la source et à la destination

La réplication de sauvegarde est prise en charge entre les versions suivantes Régions AWS.

Région source	Régions de destination disponibles
Afrique (Le Cap)	Europe (Francfort), Europe (Irlande), Europe (Londres)
Asie-Pacifique (Hong Kong)	Asie-Pacifique (Singapour), Asie-Pacifique (Tokyo)
Asie-Pacifique (Hyderabad)	Asie-Pacifique (Mumbai)

Région source	Régions de destination disponibles
Asie-Pacifique (Melbourne)	Asie-Pacifique (Sydney)
Asie-Pacifique (Malaisie)	Asie-Pacifique (Singapour)
Asie-Pacifique (Mumbai)	Asie-Pacifique (Hyderabad), Asie-Pacifique (Singapour) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
Asia Pacific (Osaka)	Asia Pacific (Tokyo)
Asia Pacific (Seoul)	Asie-Pacifique (Singapour), Asie-Pacifique (Tokyo) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
Asie-Pacifique (Singapour)	Asie-Pacifique (Hong Kong), Asie-Pacifique (Malaisie), Asie-Pacifique (Mumbai), Asie-Pacifique (Séoul), Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
Asie-Pacifique (Sydney)	Asie-Pacifique (Melbourne), Asie-Pacifique (Singapour) USA Est (Virginie du Nord), USA Ouest (Californie du Nord), USA Ouest (Oregon)
Asie-Pacifique (Tokyo)	Asie-Pacifique (Hong Kong), Asie-Pacifique (Osaka), Asie-Pacifique (Séoul), Asie-Pacifique (Singapour) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
Canada (Centre)	Canada-Ouest (Calgary) Europe (Irlande) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Californie du Nord), USA Ouest (Oregon)
Canada-Ouest (Calgary)	Canada (Centre)

Région source	Régions de destination disponibles
Chine (Beijing)	Chine (Ningxia)
Chine (Ningxia)	Chine (Beijing)
Europe (Francfort)	Afrique (Le Cap) Europe (Irlande), Europe (Londres), Europe (Paris), Europe (Stockholm), Europe (Zurich) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
Europe (Irlande)	Afrique (Le Cap) Canada (Centre) Europe (Francfort), Europe (Londres), Europe (Paris), Europe (Stockholm), Europe (Zurich) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Californie du Nord), USA Ouest (Oregon)
Europe (Londres)	Afrique (Le Cap) Europe (Francfort), Europe (Irlande), Europe (Paris), Europe (Stockholm) US East (N. Virginia)
Europe (Paris)	Europe (Francfort), Europe (Irlande), Europe (Londres), Europe (Stockholm) US East (N. Virginia)
Europe (Stockholm)	Europe (Francfort), Europe (Irlande), Europe (Londres), Europe (Paris) USA Est (Virginie du Nord)
Europe (Zurich)	Europe (Francfort), Europe (Irlande)

Région source	Régions de destination disponibles
Amérique du Sud (São Paulo)	USA Est (Virginie du Nord), USA Est (Ohio)
AWS GovCloud (USA Est)	AWS GovCloud (US-Ouest)
AWS GovCloud (US-Ouest)	AWS GovCloud (USA Est)
USA Est (Virginie du Nord)	Asie-Pacifique (Mumbai), Asie-Pacifique (Séoul), Asie-Pacifique (Singapour), Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo) Canada (Central) Europe (Francfort), Europe (Irlande), Europe (Londres), Europe (Paris), Europe (Stockholm) South America (São Paulo) USA Est (Ohio), USA Ouest (Californie du Nord), USA Ouest (Oregon)
US East (Ohio)	Asie-Pacifique (Mumbai), Asie-Pacifique (Séoul), Asie-Pacifique (Singapour), Asie-Pacifique (Tokyo) Canada (Central) Europe (Francfort), Europe (Irlande) South America (São Paulo) USA Est (Virginie du Nord), USA Ouest (Californie du Nord), USA Ouest (Oregon)

Région source	Régions de destination disponibles
US West (N. California)	Asia Pacific (Sydney) Canada (Central) Europe (Ireland) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Oregon)
US West (Oregon)	Asie-Pacifique (Mumbai), Asie-Pacifique (Séoul), Asie-Pacifique (Singapour), Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo) Canada (Central) Europe (Francfort), Europe (Irlande) USA Est (Virginie du Nord), USA Est (Ohio), USA Ouest (Californie du Nord)

Vous pouvez également utiliser la `describe-source-regions` AWS CLI commande pour savoir lesquels Régions AWS peuvent se répliquer les uns sur les autres. Pour de plus amples informations, veuillez consulter Recherche d'informations sur les sauvegardes répliquées pour Amazon RDS.

Limites

Vous trouverez ci-dessous les limites relatives aux sauvegardes automatisées entre régions pour Amazon RDS.

- La réplication automatique des sauvegardes n'est pas prise en charge pour les clusters de bases de données multi-AZ.
- Par défaut, vous pouvez disposer d'un maximum de 20 sauvegardes automatisées interrégionales par région. Compte AWS

Activation des sauvegardes automatisées entre régions pour Amazon RDS

Vous pouvez activer la réplication de sauvegarde sur des instances de base de données nouvelles ou existantes à l'aide de la RDS console Amazon. Vous pouvez également utiliser la `start-db-instance-automated-backups-replication` AWS CLI commande ou

l'`StartDBInstanceAutomatedBackupsReplication`RDSAPIopération. Vous pouvez répliquer jusqu'à 20 sauvegardes vers chaque destination Région AWS pour chacune Compte AWS.

> **Note**
>
> Pour pouvoir répliquer des sauvegardes automatiques, vous devez les activer. Pour de plus amples informations, veuillez consulter Activation des sauvegardes automatiques.

Console

Vous pouvez activer la réplication des sauvegardes pour une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, procédez à l'activation au moment du lancement de celle-ci. Pour plus d'informations, consultez Paramètres des instances de base de données.
- Pour une instance de base de données existante, procédez comme suit.

Pour activer la réplication des sauvegardes sur une instance de base de données existante

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Sous l'onglet Région actuelle, choisissez l'instance de base de données pour laquelle vous souhaitez activer la réplication des sauvegardes.
4. Dans le champ Actions, choisissez Gérer la réplication entre les régions.
5. Sous Réplication des sauvegardes, choisissez Activer la réplication dans une autre Région AWS.
6. Choisissez la Région de destination.
7. Choisissez la Période de conservation des sauvegardes répliquées.
8. Si vous avez activé le chiffrement sur l'instance de base de données source, choisissez le AWS KMS keypour chiffrer les sauvegardes ou entrez une cléARN.
9. Choisissez Save (Enregistrer).

Dans la région source, les sauvegardes répliquées sont répertoriées sous l'onglet Région actuelle de la page Sauvegardes automatiques . Dans la région de destination, les sauvegardes répliquées sont répertoriées sous l'onglet Sauvegardes répliquées de la page Sauvegardes automatiques .

AWS CLI

Activez la réplication de sauvegarde à l'aide de la `start-db-instance-automated-backups-replication` AWS CLI commande.

L'CLIexemple suivant réplique des sauvegardes automatisées d'une instance de base de données située dans la région USA Ouest (Oregon) vers la région USA Est (Virginie du Nord). Il chiffre également les sauvegardes répliquées à l'aide d'un AWS KMS key dans la région de destination.

Pour activer la réplication des sauvegardes

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

```
aws rds start-db-instance-automated-backups-replication \
--region us-east-1 \
--source-db-instance-arn "arn:aws:rds:us-west-2:123456789012:db:mydatabase" \
--kms-key-id "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE" \
--backup-retention-period 7
```

  Dans Windows:

```
aws rds start-db-instance-automated-backups-replication ^
--region us-east-1 ^
--source-db-instance-arn "arn:aws:rds:us-west-2:123456789012:db:mydatabase" ^
--kms-key-id "arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE" ^
--backup-retention-period 7
```

  `--source-region`Cette option est requise lorsque vous cryptez des sauvegardes entre les régions AWS GovCloud (USA Est) et AWS GovCloud (USA Ouest). Pour `--source-region`, spécifiez la Région AWS de l'instance de base de données source.

  Si `--source-region` n'est pas spécifié, veillez à spécifier une valeur `--pre-signed-url`. Un présigné URL est un URL qui contient une demande signée Signature version 4 pour la `start-db-instance-automated-backups-replication` commande appelée dans la source Région AWS. Pour en savoir plus sur *`pre-signed-url`* cette option, consultez start-db-instance-automated-backups-replication dans le manuel de référence des commandes.AWS CLI

RDS API

Activez la réplication de sauvegarde en utilisant l'`StartDBInstanceAutomatedBackupsReplication`RDSAPIopération avec les paramètres suivants :

- `Region`(si vous n'appelez pas l'APIopération depuis la région de destination)
- `SourceDBInstanceArn`
- `BackupRetentionPeriod`
- `KmsKeyId` (facultatif)
- `PreSignedUrl` (requis si vous utilisez `KmsKeyId`)

> **Note**
>
> Si vous chiffrez les sauvegardes, vous devez également inclure un URL présigné. Pour plus d'informations sur les demandes présignéesURLs, consultez Authentification des demandes : utilisation des paramètres de requête (AWS Signature version 4) dans le manuel Amazon Simple Storage Service API Reference et le processus de signature Signature version 4 dans le document de référence AWS général.

Recherche d'informations sur les sauvegardes répliquées pour Amazon RDS

Vous pouvez utiliser les CLI commandes suivantes pour trouver des informations sur les sauvegardes répliquées :

- `describe-source-regions`
- `describe-db-instances`
- `describe-db-instance-automated-backups`

L'`describe-source-regions`exemple suivant répertorie la source Régions AWS à partir de laquelle les sauvegardes automatisées peuvent être répliquées vers la région de destination USA Ouest (Oregon).

Pour afficher des informations sur les régions sources

- Exécutez la commande suivante.

```
aws rds describe-source-regions --region us-west-2
```

La sortie montre que les sauvegardes peuvent être répliquées à partir de US East (N. Virginia), mais pas à partir de USA Est (Ohio) ou USA Ouest (Californie du Nord) dans USA Ouest (Oregon).

```
{
    "SourceRegions": [
        ...
        {
            "RegionName": "us-east-1",
            "Endpoint": "https://rds.us-east-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": true
        },
        {
            "RegionName": "us-east-2",
            "Endpoint": "https://rds.us-east-2.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        },
            "RegionName": "us-west-1",
            "Endpoint": "https://rds.us-west-1.amazonaws.com",
            "Status": "available",
            "SupportsDBInstanceAutomatedBackupsReplication": false
        }
    ]
}
```

L'exemple `describe-db-instances` suivant présente les sauvegardes automatiques d'une instance de base de données.

Pour afficher les sauvegardes répliquées d'une instance de base de données

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

  ```
  aws rds describe-db-instances \
  --db-instance-identifier mydatabase
  ```

Dans Windows:

```
aws rds describe-db-instances ^
--db-instance-identifier mydatabase
```

La sortie inclut les sauvegardes répliquées.

```
{
    "DBInstances": [
        {
            "StorageEncrypted": false,
            "Endpoint": {
                "HostedZoneId": "Z1PVIF0B656C1W",
                "Port": 1521,
            ...

            "BackupRetentionPeriod": 7,
            "DBInstanceAutomatedBackupsReplications":
 [{"DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-east-1:123456789012:auto-backup:ab-
L2IJCEXJP7XQ7HOJ4SIEXAMPLE"}]
        }
    ]
}
```

L'exemple `describe-db-instance-automated-backups` suivant présente les sauvegardes automatiques d'une instance de base de données.

Pour afficher les sauvegardes automatiques d'une instance de base de données

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

  ```
  aws rds describe-db-instance-automated-backups \
  --db-instance-identifier mydatabase
  ```

  Dans Windows:

  ```
  aws rds describe-db-instance-automated-backups ^
  ```

```
--db-instance-identifier mydatabase
```

La sortie affiche l'instance de base de données source et les sauvegardes automatiques de USA Ouest (Oregon), avec les sauvegardes répliquées dans US East (N. Virginia).

```
{
    "DBInstanceAutomatedBackups": [
        {
            "DBInstanceArn": "arn:aws:rds:us-west-2:868710585169:db:mydatabase",
            "DbiResourceId": "db-L2IJCEXJP7XQ7HOJ4SIEXAMPLE",
            "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-west-2:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE",
            "BackupRetentionPeriod": 7,
            "DBInstanceAutomatedBackupsReplications":
 [{"DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-east-1:123456789012:auto-backup:ab-
L2IJCEXJP7XQ7HOJ4SIEXAMPLE"}]
            "Region": "us-west-2",
            "DBInstanceIdentifier": "mydatabase",
            "RestoreWindow": {
                "EarliestTime": "2020-10-26T01:09:07Z",
                "LatestTime": "2020-10-31T19:09:53Z",
            }
            ...
        }
    ]
}
```

L'exemple `describe-db-instance-automated-backups` suivant utilise l'option `--db-instance-automated-backups-arn` pour afficher les sauvegardes répliquées dans la région de destination.

Pour afficher les sauvegardes répliquées

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

```
aws rds describe-db-instance-automated-backups \
--db-instance-automated-backups-arn "arn:aws:rds:us-east-1:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE"
```

Dans Windows:

```
aws rds describe-db-instance-automated-backups ^
--db-instance-automated-backups-arn "arn:aws:rds:us-east-1:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE"
```

La sortie présente l'instance de base de données source de la région USA Ouest (Oregon), avec les sauvegardes répliquées dans la région US East (N. Virginia).

```
{
    "DBInstanceAutomatedBackups": [
        {
            "DBInstanceArn": "arn:aws:rds:us-west-2:868710585169:db:mydatabase",
            "DbiResourceId": "db-L2IJCEXJP7XQ7HOJ4SIEXAMPLE",
            "DBInstanceAutomatedBackupsArn": "arn:aws:rds:us-east-1:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE",
            "Region": "us-west-2",
            "DBInstanceIdentifier": "mydatabase",
            "RestoreWindow": {
                "EarliestTime": "2020-10-26T01:09:07Z",
                "LatestTime": "2020-10-31T19:01:23Z"
            },
            "AllocatedStorage": 50,
            "BackupRetentionPeriod": 7,
            "Status": "replicating",
            "Port": 1521,
            ...
        }
    ]
}
```

Restauration à une heure spécifiée à partir d'une sauvegarde répliquée pour Amazon RDS

Vous pouvez restaurer une instance de base de données à un moment précis à partir d'une sauvegarde répliquée à l'aide de la RDS console Amazon. Vous pouvez également utiliser la `restore-db-instance-to-point-in-time` AWS CLI commande ou l'`RestoreDBInstanceToPointInTime`RDSAPIopération.

Pour des informations générales sur point-in-time recovery (PITR), voirRestauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

> **Note**
>
> Notez les restrictions du moteur de base de données suivantes lorsque des sauvegardes automatisées sont répliquées sur Régions AWS :
>
> - Sur SQL Server, RDS les groupes d'options ne sont pas copiés.
> - Sur RDS Oracle, les options suivantes ne sont pas copiées : `NATIVE_NETWORK_ENCRYPTION`OEM,`OEM_AGENT`, et`SSL`.
>
> Si vous avez associé un groupe d'options personnalisé à votre instance de base de données , vous pouvez recréer ce groupe d'options dans la région de destination. Restaurez ensuite l'instance de base de données dans la région de destination et associez le groupe d'options personnalisé à celle-ci. Pour de plus amples informations, veuillez consulter Utilisation de groupes d'options.

Console

Pour restaurer une instance de base de données à une heure spécifiée à partir d'une sauvegarde répliquée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez la région de destination (dans laquelle les sauvegardes seront répliquées) à partir du sélecteur de région.
3. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
4. Sous l'onglet Sauvegardes répliquées, choisissez l'instance de base de données à restaurer.
5. Sous Actions, sélectionnez Restaurer à un moment donné.
6. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

   Si vous choisissez Personnalisé, saisissez la date et l'heure de restauration souhaitée de l'instance.

> **Note**
>
> Les heures sont affichées dans votre fuseau horaire local, qui est indiqué par un décalage par rapport au temps universel coordonné (UTC). Par exemple, UTC -5 correspond à l'heure normale de l'Est/heure avancée du centre.

7. Pour Identifiant d'instance de base de données, entrez le nom de l'instance de base de données restaurée.
8. (Facultatif) Sélectionnez d'autres options selon vos besoins, comme l'activation de la mise à l'échelle automatique.
9. Choisissez Restaurer à un instant dans le passé.

AWS CLI

Utilisez la `restore-db-instance-to-point-in-time` AWS CLI commande pour créer une nouvelle instance de base de données.

Pour restaurer une instance de base de données à une heure spécifiée à partir d'une sauvegarde répliquée

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-automated-backups-arn "arn:aws:rds:us-
east-1:123456789012:auto-backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE" \
    --target-db-instance-identifier mytargetdbinstance \
    --restore-time 2020-10-14T23:45:00.000Z
```

  Dans Windows:

```
aws rds restore-db-instance-to-point-in-time ^
    --source-db-instance-automated-backups-arn "arn:aws:rds:us-
east-1:123456789012:auto-backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE" ^
    --target-db-instance-identifier mytargetdbinstance ^
    --restore-time 2020-10-14T23:45:00.000Z
```

RDS API

Pour restaurer une instance de base de données à une heure spécifiée, appelez l'RDSAPIopération `RestoreDBInstanceToPointInTime`Amazon avec les paramètres suivants :

- `SourceDBInstanceAutomatedBackupsArn`
- `TargetDBInstanceIdentifier`
- `RestoreTime`

Arrêt de la réplication automatique des sauvegardes pour Amazon RDS

Vous pouvez arrêter la réplication de sauvegarde pour les instances de base de données à l'aide de la RDS console Amazon. Vous pouvez également utiliser la `stop-db-instance-automated-backups-replication` AWS CLI commande ou l'`StopDBInstanceAutomatedBackupsReplication`RDSAPIopération.

Les sauvegardes répliquées sont conservées, conformément à la période de conservation des sauvegardes définie lors de leur création.

Console

Arrêtez la réplication des sauvegardes à partir de la page Sauvegardes automatiques de la région source.

Pour arrêter la réplication de sauvegarde vers un Région AWS

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez la région source dans le sélecteur de région.
3. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
4. Sous l'onglet Région actuelle, choisissez l'instance de base de données pour laquelle vous souhaitez arrêter la réplication des sauvegardes.
5. Dans le champ Actions, choisissez Gérer la réplication entre les régions.
6. Sous Backup replication (Réplication des sauvegardes), décochez la case Enable replication to another Région AWS (Activer la réplication dans une autre Région AWS).
7. Choisissez Save (Enregistrer).

Les sauvegardes répliquées sont répertoriées sous l'onglet Conservées de la page Sauvegardes automatiques de la région de destination.

AWS CLI

Arrêtez la réplication de sauvegarde à l'aide de la `stop-db-instance-automated-backups-replication` AWS CLI commande.

L'CLIexemple suivant empêche les sauvegardes automatiques d'une instance de base de données de se répliquer dans la région de l'ouest des États-Unis (Oregon).

Pour arrêter la réplication des sauvegardes

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

```
aws rds stop-db-instance-automated-backups-replication \
--region us-east-1 \
--source-db-instance-arn "arn:aws:rds:us-west-2:123456789012:db:mydatabase"
```

  Dans Windows:

```
aws rds stop-db-instance-automated-backups-replication ^
--region us-east-1 ^
--source-db-instance-arn "arn:aws:rds:us-west-2:123456789012:db:mydatabase"
```

RDS API

Arrêtez la réplication de sauvegarde en utilisant l'`StopDBInstanceAutomatedBackupsReplication`RDSAPIopération avec les paramètres suivants :

- `Region`
- `SourceDBInstanceArn`

Suppression de sauvegardes répliquées pour Amazon RDS

Vous pouvez supprimer les sauvegardes répliquées pour les instances de base de données à l'aide de la RDS console Amazon. Vous pouvez également utiliser la `delete-db-instance-automated-backups` AWS CLI commande ou l'`DeleteDBInstanceAutomatedBackup`RDSAPIopération.

Console

Vous pouvez supprimer des sauvegardes répliquées de la région de destination à partir de la page Sauvegardes automatiques.

Pour supprimer des sauvegardes répliquées

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez la région de destination dans le sélecteur de région.
3. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
4. Sous l'onglet Sauvegardes répliquées, choisissez l'instance de base de données pour laquelle vous souhaitez supprimer les sauvegardes répliquées.
5. Pour Actions, choisissez Supprimer.
6. Dans la page de confirmation, entrez **`delete me`** et choisissez Delete (Supprimer).

AWS CLI

Supprimez les sauvegardes répliquées à l'aide de la `delete-db-instance-automated-backup` AWS CLI commande.

Vous pouvez utiliser la `describe-db-instances`CLIcommande pour trouver les Amazon Resource Names (ARNs) des sauvegardes répliquées. Pour de plus amples informations, veuillez consulter Recherche d'informations sur les sauvegardes répliquées pour Amazon RDS.

Pour supprimer des sauvegardes répliquées

- Exécutez une des commandes suivantes :

  Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance-automated-backup \
```

```
--db-instance-automated-backups-arn "arn:aws:rds:us-east-1:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE"
```

Dans Windows:

```
aws rds delete-db-instance-automated-backup ^
--db-instance-automated-backups-arn "arn:aws:rds:us-east-1:123456789012:auto-
backup:ab-L2IJCEXJP7XQ7HOJ4SIEXAMPLE"
```

RDS API

Supprimez les sauvegardes répliquées à l'aide de l'DeleteDBInstanceAutomatedBackupRDSAPIopération associée au `DBInstanceAutomatedBackupsArn` paramètre.

Gestion des sauvegardes manuelles

Cette section explique comment gérer les sauvegardes manuelles pour les instances de base de données et les clusters de base de données.

Pour plus d'informations sur la création d'un instantané de base de données pour un Single-AZ Instance de base de données, voirCréation d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Pour plus d'informations sur la création d'un instantané de base de données pour un cluster de base de données multi-AZ, consultezCréation d'un instantané de cluster de base de données multi-AZ pour Amazon RDS.

Pour plus d'informations sur la suppression d'un instantané de base de données, consultezSupprimer un instantané de base de données pour Amazon RDS.

Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS

Amazon RDS crée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de l'instance de base de données et pas uniquement des bases de données individuelles. La création de cet instantané de base de données sur une instance de base de données mono-AZ entraîne une brève interruption I/O qui peut durer de quelques secondes à quelques minutes, en fonction de la taille et de la classe de votre instance de base de données. Pour MariaDB, SQL My, Oracle et SQL Postgre, l'activité d'E/S n'est pas suspendue sur votre serveur principal pendant la sauvegarde pour les déploiements multi-AZ, car la sauvegarde est prise depuis le mode veille. Pour le SQL serveur, l'activité d'E/S est brièvement suspendue pendant la sauvegarde pour les déploiements multi-AZ.

Lorsque vous créez un snapshot DB, vous devez identifier quelle instance de base de données vous allez sauvegarder, puis nommer votre snapshot DB afin de pouvoir effectuer une restauration à partir de ce dernier ultérieurement. Le temps nécessaire à la création d'un instantané varie en fonction de la taille de vos bases de données. Étant donné que l'instantané inclut l'intégralité du volume de stockage, la taille des fichiers, comme les fichiers temporaires, a également une incidence sur le temps nécessaire à la création de l'instantané.

> **Note**
>
> Votre instance de base de données doit être dans l'état `available` pour prendre un instantané de base de données.
> Pour les SQL instances de base de données Postgre, il est possible que les données des tables non enregistrées ne soient pas restaurées à partir de snapshots. Pour de plus amples informations, veuillez consulter Bonnes pratiques pour utiliser les moteurs de stockage PostgreSQL.

Contrairement aux sauvegardes automatisées, les instantanés manuels ne sont pas soumis à la période de rétention des sauvegardes. Les instantanés n'expirent pas.

Pour les sauvegardes à très long terme des données MariaDB, SQL My et SQL Postgre, nous recommandons d'exporter les données instantanées vers Amazon S3. Si la version majeure de votre moteur de base de données n'est plus prise en charge, vous ne pouvez pas restaurer cette version à partir d'un instantané. Pour de plus amples informations, veuillez consulter Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS.

Vous pouvez créer un instantané de base de données à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Console

Pour créer un instantané de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.

   La liste des instantanés manuels s'affiche.
3. Choisissez Prendre un instantané.

   La fenêtre Capture d'un instantané DB apparaît.
4. Choisissez l'instance de base de données pour laquelle vous souhaitez prendre un instantané.
5. Entrez le nom du snapshot.
6. Choisissez Prendre un instantané.

La liste des instantanés manuels apparaît, avec l'état du nouveau cliché de base de données indiqué sous `Creating` la forme. Une fois que l'état de l'instantané est `Available`, vous pouvez voir son heure de création.

AWS CLI

Lorsque vous créez un instantané de base de données à l'aide de AWS CLI, vous devez identifier l'instance de base de données que vous allez sauvegarder, puis donner un nom à votre instantané de base de données afin de pouvoir le restaurer ultérieurement. Pour ce faire, utilisez la AWS CLI `create-db-snapshot`commande avec les paramètres suivants :

- `--db-instance-identifier`
- `--db-snapshot-identifier`

Dans cet exemple, vous créez un instantané de base de données appelé *`mydbsnapshot`* pour une instance de base de données appelée *`mydbinstance`*.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-snapshot \
    --db-instance-identifier mydbinstance \
    --db-snapshot-identifier mydbsnapshot
```

Dans Windows:

```
aws rds create-db-snapshot ^
    --db-instance-identifier mydbinstance ^
    --db-snapshot-identifier mydbsnapshot
```

RDS API

Lorsque vous créez un instantané de base de données à l'aide d'Amazon RDSAPI, vous devez identifier l'instance de base de données que vous allez sauvegarder, puis donner un nom à votre instantané de base de données afin de pouvoir le restaurer ultérieurement. Vous pouvez le faire en utilisant la RDS API `CreateDBSnapshot`commande Amazon avec les paramètres suivants :

- `DBInstanceIdentifier`
- `DBSnapshotIdentifier`

Création d'un instantané de cluster de base de données multi-AZ pour Amazon RDS

Lorsque vous créez un instantané de cluster de base de données multi-AZ, veillez à identifier le cluster de base de donnes multi-AZ que vous allez sauvegarder, puis nommez votre instantané de cluster de base de données afin de pouvoir le restaurer par la suite. Vous pouvez également partager un instantané de cluster de base de données multi-AZ. Pour obtenir des instructions, consultez the section called “Partage d'un instantané de de base de données”.

Vous pouvez créer un instantané de cluster de base de données multi-AZ à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Pour les sauvegardes à très long terme, nous vous recommandons d'exporter les données d'instantané vers Amazon S3. Si la version majeure de votre moteur de base de données n'est plus prise en charge, vous ne pouvez pas restaurer cette version à partir d'un instantané. Pour de plus amples informations, veuillez consulter Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS.

Console

Pour créer un instantané de cluster de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Dans la liste, choisissez le cluster de base de données multi-AZ dont vous voulez prendre un instantané.
4. Sous Actions, choisissez Take snapshot (Prendre un instantané).

   La fenêtre Capture d'un instantané DB apparaît.
5. Dans Snapshot name (Nom de l'instantané), saisissez le nom de l'instantané.
6. Choisissez Prendre un instantané.

La page Snapshots (Instantanés) s'affiche avec le nouvel instantané de cluster de base de données multi-AZ dont l'état est `Creating`. Une fois que l'état de l'instantané est `Available`, vous pouvez voir son heure de création.

AWS CLI

Vous pouvez créer un instantané de cluster de base de données multi-AZ à l'aide de la AWS CLI create-db-cluster-snapshotcommande avec les options suivantes :

- `--db-cluster-identifier`
- `--db-cluster-snapshot-identifier`

Dans cet exemple, vous créez un instantané de cluster de base de données multi-AZ appelé *`instantané de mon cluster multiazdb`* pour un cluster de base de données appelé *`mon cluster multi-azdb`*.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-cluster-snapshot \
    --db-cluster-identifier mymultiazdbcluster \
    --db-cluster-snapshot-identifier mymultiazdbclustersnapshot
```

Dans Windows:

```
aws rds create-db-cluster-snapshot ^
    --db-cluster-identifier mymultiazdbcluster ^
    --db-cluster snapshot-identifier mymultiazdbclustersnapshot
```

RDS API

Vous pouvez créer un instantané de cluster de base de données multi-AZ à l'aide de l'opération Amazon RDS API C reateDBCluster Snapshot avec les paramètres suivants :

- `DBClusterIdentifier`
- `DBClusterSnapshotIdentifier`

Suppression d'un instantané de cluster de base de données multi-AZ

Vous pouvez supprimer les instantanés de base de données multi-AZ gérés par Amazon RDS lorsque vous n'en avez plus besoin. Pour obtenir des instructions, consultez the section called “Suppression d'un instantané de base de données”.

Supprimer un instantané de base de données pour Amazon RDS

Vous pouvez supprimer des instantanés de base de données gérés par Amazon RDS lorsque vous n'en avez plus besoin.

> **Note**
>
> Pour supprimer les sauvegardes gérées par AWS Backup, utilisez la AWS Backup console. Pour des informations sur AWS Backup, consultez le AWS Backup manuel du développeur .

Suppression d'un instantané de base de données

Vous pouvez supprimer un instantané de base de données manuel, partagé ou public à l'aide du AWS Management Console AWS CLI, ou du RDSAPI.

Pour supprimer un instantané partagé ou public, vous devez vous connecter au AWS compte propriétaire de l'instantané.

Si vous souhaitez supprimer des instantanés de base de données automatiques sans supprimer l'instance de base de données, définissez la période de conservation des sauvegardes de l'instance de base de données sur 0. Les instantanés automatiques sont supprimés lorsque la modification est appliquée. Vous pouvez appliquer la modification immédiatement si vous ne souhaitez pas attendre la prochaine période de maintenance. Une fois la modification terminée, vous pouvez réactiver les sauvegardes automatiques en définissant la période de conservation des sauvegardes sur un nombre supérieur à 0. Pour plus d'informations sur la modification d'une instance de base de données , consultez Modification d'une RDS instance de base de données Amazon.

Les sauvegardes automatisées conservées et les instantanés manuels sont facturés tant qu'ils ne sont pas supprimés. Pour de plus amples informations, veuillez consulter Coûts de conservation.

Si vous avez supprimé une instance de base de données, vous pouvez supprimer ses instantanés de base de données automatiques en supprimant les sauvegardes automatiques de l'instance de base de données. Pour plus d'informations sur les sauvegardes automatiques, consultez Présentation des sauvegardes.

Console

Pour supprimer un instantané de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.

   La liste des instantanés manuels s'affiche.
3. Choisissez l'instantané de base de données à supprimer.
4. Pour Actions, choisissez Delete snapshot (Supprimer la pile).
5. Dans la page de confirmation, sélectionnez Supprimer.

AWS CLI

Vous pouvez supprimer un instantané de base de données à l'aide de la AWS CLI commande delete-db-snapshot.

Les options suivantes sont utilisées pour supprimer un instantané de base de données.

- `--db-snapshot-identifier` – Identifiant de l'instantané de base de données.

Example

Le code suivant supprime l'instantané de base de données `mydbsnapshot`.

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-snapshot \
    --db-snapshot-identifier mydbsnapshot
```

Dans Windows:

```
aws rds delete-db-snapshot ^
    --db-snapshot-identifier mydbsnapshot
```

RDS API

Vous pouvez supprimer un instantané de base de données à l'aide de RDS API l'opération Amazon eleteDBSnapshotD.

Les paramètres suivants sont utilisés pour supprimer un instantané de base de données.

- `DBSnapshotIdentifier` – Identifiant de l'instantané de base de données.

Restauration vers une instance de base de données

Cette section explique comment effectuer une restauration sur une instance de base de données. Cette page explique comment effectuer une restauration sur une instance de base de données Amazon RDS à partir d'un instantané de base de données.

Amazon RDS crée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de cette dernière et pas seulement les bases de données. Vous pouvez créer une instance de base de données en effectuant une restauration à partir d'un instantané de base de données. Vous indiquez le nom de l'instantané de base de données à partir duquel opérer la restauration, puis un nom pour la nouvelle instance de base de données résultant de l'opération de restauration. Vous ne pouvez pas restaurer un instantané de base de données sur une instance de base de données existante ; une nouvelle instance de base de données est créée lors de la restauration.

Après restauration de l'instance de base de données, vous pouvez l'utiliser dès que son statut est `available`. L'instance de base de données continue de charger des données en arrière-plan. Cette opération s'appelle chargement différé.

Si vous accédez à des données qui n'ont pas encore été chargées, l'instance de base de données télécharge immédiatement les données demandées à partir d'Amazon S3, et continue à charger le reste des données en arrière-plan. Pour plus d'informations, consultez Instantanés Amazon EBS.

Pour atténuer les effets du chargement différé sur des tables auxquelles vous avez besoin de pouvoir accéder rapidement, vous pouvez effectuer des opérations impliquant des analyses de table entière, telles que `SELECT *`. Cela permet à Amazon RDS de télécharger toutes les données de table sauvegardées à partir de S3.

Vous pouvez restaurer une instance de base de données et utiliser un type de stockage différent que l'instantané de base de données source Dans ce cas, le processus de restauration est plus lent, à cause du travail supplémentaire nécessaire pour migrer les données vers le nouveau type de stockage. Si vous effectuez une restauration vers ou à partir d'un stockage magnétique, le processus de migration est plus lent. Ceci est dû au fait que le stockage magnétique ne dispose pas de la capacité IOPS du stockage IOPS provisionnés ou Usage général (SSD).

Vous pouvez l'utiliser AWS CloudFormation pour restaurer une instance de base de données à partir d'un instantané d'instance de base de données. Pour plus d'informations, consultez AWS : :RDS : : DBInstance dans le guide de l'AWS CloudFormation utilisateur.

> **Note**
>
> Vous ne pouvez pas restaurer une instance de base de données à partir d'un instantané de base de données qui est à la fois partagé et chiffré. Par contre, vous pouvez créer une copie de l'instantané de base de données et restaurer l'instance de base de données à partir de cette copie. Pour de plus amples informations, veuillez consulter Copier un instantané de base de données pour Amazon RDS.

Pour plus d'informations sur la restauration d'une instance de base de données avec une version RDS Extended Support, consultezRestauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support.

Restaurer à partir d'un instantané

Vous pouvez restaurer une instance de base de données à partir d'un instantané de base de données à l' AWS Management Console aide de l'API AWS CLI, de ou de l'API RDS.

> **Note**
>
> Vous ne pouvez pas réduire la quantité de stockage lorsque vous restaurez une instance de base de données. Lorsque vous augmentez la valeur du stockage alloué, vous devez le faire d'au moins 10 %. Si vous tentez d'augmenter la valeur de moins de 10 %, une erreur s'affiche. Vous ne pouvez pas augmenter le stockage alloué lors de la restauration des instances de base de données RDS for SQL Server.

console

Pour restaurer une instance de base de données à partir d'un instantané de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.

5. Sur la page Restore snapshot (Restaurer l'instantané), pour DB instance identifier (Identifiant d'instance de base de données), saisissez le nom de votre instance de base de données restaurée.
6. Spécifiez d'autres paramètres, tels que la taille de stockage allouée.

   Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.
7. Choisissez Restore DB Instance (Restaurer une instance de base de données).

AWS CLI

Pour restaurer une instance de base de données à partir d'un instantané de base de données, utilisez la AWS CLI commande restore-db-instance-from-db-snapshot.

Dans cet exemple, vous effectuez la restauration à partir d'un instantané de base de données précédemment créé, nommé `mydbsnapshot`. Vous effectuez la restauration à une nouvelle instance de base de données nommée `mynewdbinstance`. Cet exemple définit également la taille de stockage allouée.

Vous pouvez spécifier d'autres paramètres. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier mynewdbinstance \
    --db-snapshot-identifier mydbsnapshot \
    --allocated-storage 100
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier mynewdbinstance ^
    --db-snapshot-identifier mydbsnapshot ^
    --allocated-storage 100
```

La sortie générée lors de l'exécution de cette commande est semblable à ce qui suit :

```
DBINSTANCE  mynewdbinstance  db.t3.small  MySQL     50       sa              creating
 3  n  8.0.28  general-public-license
```

API RDS

Pour restaurer une instance de base de données à partir d'un instantané de base de données, appelez la fonction d'API Amazon RDS Restore DBInstance From DBSnapshot avec les paramètres suivants :

- `DBInstanceIdentifier`
- `DBSnapshotIdentifier`

Considérations

Pour les considérations relatives à la restauration sur une instance de base de données à partir d'un instantané de base de données, consultez les rubriques suivantes.

Rubriques

- Considérations relatives au groupe de paramètres
- Considérations relatives aux groupes de sécurité
- Considérations relatives au groupe d'options
- Considérations relatives à l'étiquetage des ressources
- Considérations relatives à DB2
- Considérations relatives à Microsoft SQL Server
- Considérations relatives à MySQL
- Considérations relatives à Oracle Database

Considérations relatives au groupe de paramètres

Nous vous recommandons de conserver le groupe de paramètres de base de données de tout instantané de bases de données que vous créez, de telle sorte que vous puissiez associer votre instance de base de données restaurée au groupe de paramètres approprié.

Le groupe de paramètres de base de données par défaut est associé à l'instance restaurée, sauf si vous en choisissez une autre. Aucun paramètre personnalisé n'est disponible dans le groupe de paramètres par défaut.

Vous pouvez spécifier le groupe de paramètres lorsque vous restaurez l'instance de base de données.

Pour plus d'informations sur les groupes de paramètres DB, consultez Groupes de paramètres pour (Amazon RDS).

Considérations relatives aux groupes de sécurité

Lorsque vous restaurez une instance de base de données, le cloud privé virtuel (VPC) par défaut, le groupe de sous-réseaux de base de données et le groupe de sécurité VPC sont associés à l'instance restaurée, sauf si vous en choisissez d'autres.

- Si vous utilisez la console Amazon RDS, vous pouvez spécifier un groupe de sécurité VPC personnalisé à associer à l'instance ou créer un nouveau groupe de sécurité VPC.
- Si vous utilisez le AWS CLI, vous pouvez spécifier un groupe de sécurité VPC personnalisé à associer à l'instance en incluant l'`--vpc-security-group-ids`option dans la `restore-db-instance-from-db-snapshot` commande.
- Si vous utilisez l'API Amazon RDS, vous pouvez inclure le paramètre `VpcSecurityGroupIds.VpcSecurityGroupId.N` dans l'action `RestoreDBInstanceFromDBSnapshot`.

Dès que la restauration est terminée et que votre nouvelle instance de base de données est disponible, vous pouvez également changer les paramètres de VPC en modifiant l'instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Considérations relatives au groupe d'options

Lorsque vous restaurez une instance de base de données, le groupe d'options de base de données par défaut est associé à l'instance de base de données restaurée dans la plupart des cas.

L'exception concerne l'instance de base de données source associée à un groupe d'options contenant une option persistante ou permanente. Par exemple, si l'instance de base de données source utilise le chiffrement des données Oracle Transparent Data Encryption (TDE), l'instance de base de données restaurée doit utiliser un groupe d'options ayant l'option TDE.

Si vous restaurez une instance de base de données dans un VPC différent, vous devez effectuer l'une des opérations suivantes pour affecter un groupe d'options de base de données :

- Affectez le groupe d'options par défaut de ce groupe VPC à l'instance.
- Affectez un autre groupe d'options qui est lié à ce VPC.
- Créez un nouveau groupe d'options et affectez-le à l'instance de base de données. Avec les options permanentes ou persistantes, telles qu'Oracle TDE, vous devez créer un nouveau groupe d'options incluant l'option persistante ou permanente.

Pour plus d'informations sur les groupes d'options de base de données, veuillez consulter Utilisation de groupes d'options.

Considérations relatives à l'étiquetage des ressources

Lorsque vous restaurez une instance de base de données depuis un instantané de base de données, RDS vérifie si vous spécifiez de nouvelles identifications. Si oui, les nouvelles identifications sont ajoutées à l'instance de base de données restaurée. S'il n'y a pas de nouvelles identifications, RDS ajoute les identifications de l'instance de base de données source au moment de la création de l'instantané dans l'instance de base de données restaurée.

Pour de plus amples informations, veuillez consulter Copier des balises dans des instantanés de base de données.

Considérations relatives à DB2

Avec le modèle BYOL, vos instances de base de données Amazon RDS pour DB2 doivent être associées à un groupe de paramètres personnalisé contenant votre IBM Site ID et votre IBM Customer ID. Dans le cas contraire, les tentatives de restauration d'une instance de base de données à partir d'un instantané échoueront. Vos instances de base de données Amazon RDS pour DB2 doivent également être associées à une licence AWS License Manager autogérée. Pour de plus amples informations, veuillez consulter Apportez votre propre licence (BYOL) pour Db2.

Avec le AWS Marketplace modèle de licence DB2, vous avez besoin d'un AWS Marketplace abonnement actif pour le IBM Db2 édition que vous souhaitez utiliser. Si vous n'en avez pas déjà un, abonnez-vous à Db2 AWS Marketplace pour cela IBM Db2 édition. Pour de plus amples informations, veuillez consulter Licence DB2 via AWS Marketplace.

Considérations relatives à Microsoft SQL Server

Lorsque vous restaurez un instantané de base de données RDS for Microsoft SQL Server sur une nouvelle instance, vous pouvez toujours effectuer une restauration sur la même édition que votre

instantané. Dans certains cas, vous pouvez également modifier l'édition de l'instance de base de données. Les limitations suivantes s'appliquent lors de la modification des éditions :

- L'instantané de base de données doit disposer de suffisamment de stockage alloué à la nouvelle édition.
- Seules les modifications d'édition suivantes sont prises en charge :
  - De Standard Edition vers Enterprise Edition
  - De Web Edition vers Standard Edition ou Enterprise Edition
  - D'Express Edition vers Web Edition, Standard Edition ou Enterprise Edition

Si vous voulez passer d'une édition à une nouvelle édition qui n'est pas prise en charge en restaurant un instantané, vous pouvez essayer d'utiliser la fonction de sauvegarde et de restauration native. SQL Server vérifie si votre base de données est compatible avec la nouvelle édition sur la base des fonctions SQL Server que vous avez activées sur la base de données. Pour de plus amples informations, veuillez consulter Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Considérations relatives à MySQL

Pour effectuer une restauration à partir d'un instantané de base de données RDS pour MySQL avec une version de moteur non prise en charge, vous devrez peut-être mettre à niveau votre instantané de base de données plusieurs fois. Pour plus d'informations sur les options de mise à niveau, consultezOptions de mise à niveau pour les instantanés de base de données avec des versions de moteur non prises en charge pour RDS for MySQL.

Pour plus d'informations sur la mise à niveau de la version du moteur d'un instantané de base de données RDS pour MySQL,Mise à niveau d'une version du moteur de snapshots de base de données MySQL.

Considérations relatives à Oracle Database

Lorsque vous restaurez une base de données Oracle à partir d'un instantané de base de données, tenez compte des points suivants :

- Avant de restaurer un instantané de base de données, vous pouvez le mettre à niveau vers une version ultérieure de base de données Oracle. Pour de plus amples informations, veuillez consulter Mise à niveau d'un instantané de base de données Oracle.

- Si vous restaurez un instantané d'une instance de CDB qui utilise la configuration à locataire unique, vous pouvez modifier le nom de la PDB. Vous ne pouvez pas modifier les noms de PDB lorsque votre instance de CDB utilise la configuration à locataires multiples. Pour de plus amples informations, veuillez consulter Sauvegarde et restauration d'une CDB.
- Vous ne pouvez pas modifier le nom de la base de données de conteneur (CDB), qui est toujours `RDSCDB`. Ce nom de CDB est le même pour toutes les instances de CDB.
- Vous ne pouvez pas interagir directement avec les bases de données locataire dans un instantané de base de données. Si vous restaurez un instantané d'une instance de CDB qui utilise la configuration à locataires multiples, vous restaurez toutes ses bases de données locataire. Vous pouvez utiliser describe-db-snapshot-tenant-databases pour inspecter les bases de données mutualisées dans un instantané de base de données avant de le restaurer.
- Si vous utilisez Oracle GoldenGate, conservez toujours le groupe de paramètres associé au `compatible` paramètre. Lorsque vous restaurez une instance de base de données depuis un instantané de bases de données, spécifiez le groupe de paramètres associé à une valeur `compatible` correspondante ou supérieure.
- Vous pouvez choisir de renommer votre base de données lorsque vous restaurez un instantané de base de données. Si la taille totale du journal de journalisation en ligne est supérieure à 20 Go, RDS peut rétablir la taille de votre journal de restauration en ligne à ses paramètres par défaut de 512 Mo (4 x 128 Mo). La taille réduite permet de terminer l'opération de restauration dans un délai raisonnable. Vous pouvez recréer les journaux de restauration en ligne ultérieurement et en modifier la taille.
- Vous pouvez gérer le mot de passe de votre utilisateur principal dans AWS Secrets Manager. Pour de plus amples informations, veuillez consulter Présentation de la gestion des mots de passe des utilisateurs principaux avec AWS Secrets Manager.

Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

Vous pouvez restaurer une instance de base de données à un moment précis, en créant une nouvelle instance de base de données sans modifier l'instance de base de données source.

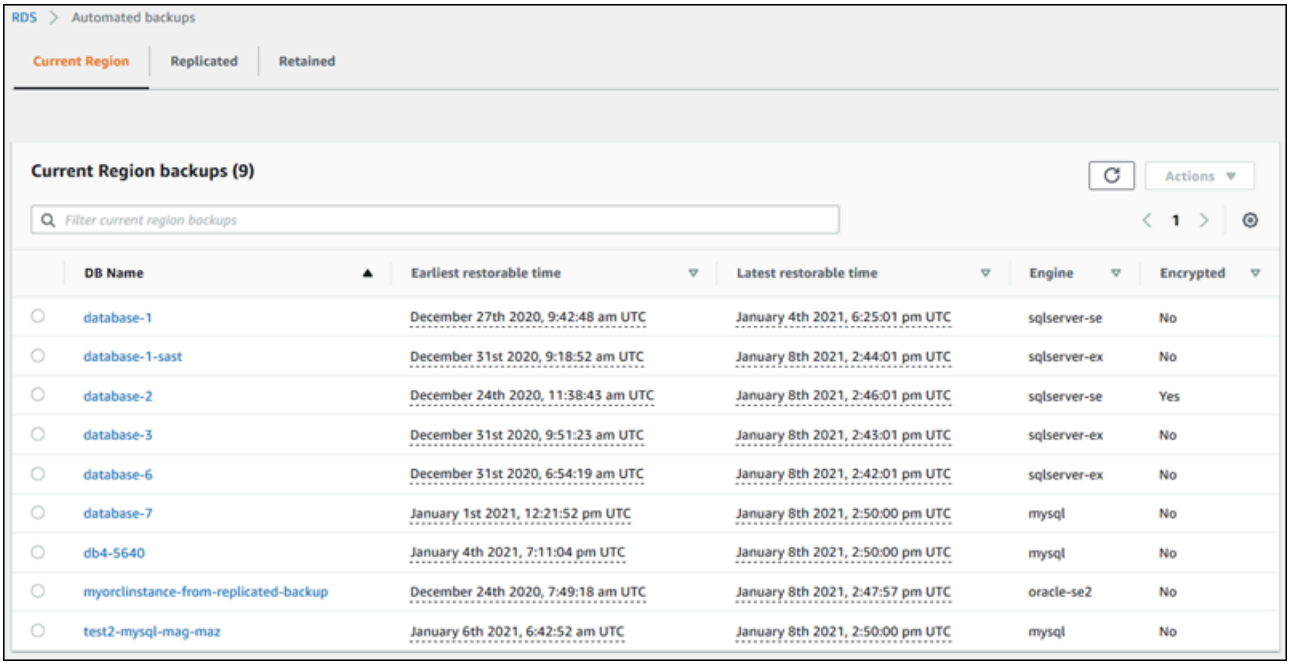
Lorsque vous restaurez une instance de base de données à un moment donné, vous pouvez choisir le groupe de sécurité Virtual Private Cloud (VPC) par défaut. Vous pouvez également appliquer un groupe de sécurité VPC personnalisé à votre instance de base de données.

Les instances de base de données restaurées sont automatiquement associées aux groupes de paramètres et d'options de base de données par défaut. Cependant, vous pouvez appliquer un groupe de paramètres ou d'options personnalisé en le définissant au moment de la restauration.

Si l'instance de base de données source possède des identifications de ressource, RDS ajoute les dernières identifications à l'instance de base de données restaurée.

RDS charge les journaux de transaction pour les instances de base de données Amazon S3 toutes les cinq minutes. Pour connaître l'heure de restauration la plus récente pour une instance de base de données, utilisez la AWS CLI describe-db-instancescommande et examinez la valeur renvoyée dans le `LatestRestorableTime` champ correspondant à l'instance de base de données. Pour afficher l'heure de restauration la plus récente pour chaque instance de base de données dans la console Amazon RDS, choisissez Automated backups (Sauvegardes automatisées).

Vous pouvez procéder à une restauration à n'importe quel moment dans le passé au cours de la période de rétention des sauvegardes. Pour afficher l'heure de restauration la plus ancienne pour chaque instance de base de données, choisissez Automated backups (Sauvegardes automatisées) dans la console Amazon RDS.



> **Note**
>
> Nous vous recommandons de restaurer la même taille d'instance de base de données—et les mêmes I/O par seconde ou similaires si vous utilisez le stockage IOPS provisionnés—comme instance DB source. Vous pouvez obtenir une erreur si, par exemple, vous choisissez une taille d'instance de base de données avec une valeur d'IOPS incompatible.

Pour plus d'informations sur la restauration d'une instance de base de données avec une version RDS Extended Support, consultezRestauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support.

Certains moteurs de base de données Amazon RDS doivent tenir compte de considérations particulières lors de la restauration à partir d'un certain point dans le temps :

- Si vous utilisez l'authentification par mot de passe avec une instance de base de données Amazon RDS pour DB2, les actions de gestion des utilisateurs, y compris`rdsadmin.add_user`, ne seront pas enregistrées dans les journaux. Ces actions nécessitent une sauvegarde instantanée complète.

  Avec le modèle BYOL, vos instances de base de données RDS pour DB2 doivent être associées à un groupe de paramètres personnalisé qui contient votre IBM Site ID et votre IBM Customer ID. Dans le cas contraire, les tentatives de restauration d'une instance de base de données à un

moment précis échoueront. Vos instances de base de données Amazon RDS pour DB2 doivent également être associées à une licence AWS License Manager autogérée. Pour de plus amples informations, veuillez consulter Apportez votre propre licence (BYOL) pour Db2.

Avec le AWS Marketplace modèle de licence DB2, vous avez besoin d'un AWS Marketplace abonnement actif pour le IBM Db2 édition que vous souhaitez utiliser. Si vous n'en avez pas déjà un, abonnez-vous à Db2 AWS Marketplace pour cela IBM Db2 édition. Pour de plus amples informations, veuillez consulter Licence DB2 via AWS Marketplace.

- Lorsque vous restaurez une instance de base de données RDS pour Oracle à un moment donné, vous pouvez spécifier un moteur de base de données, un modèle de licence et un DBName (SID) différents pour l'instance de base de données restaurée. Vous pouvez également spécifier que RDS doit stocker les mots de passe des utilisateurs principaux dans AWS Secrets Manager. Pour de plus amples informations, veuillez consulter Présentation de la gestion des mots de passe des utilisateurs principaux avec AWS Secrets Manager.
- Lorsque vous restaurez une instance de base de données Microsoft SQL Server à un instant dans le passé, chaque base de données au sein de cette instance est restaurée à un point dans le temps situé au sein d'1 seconde de chaque autre base de données de l'instance. Les transactions qui couvrent plusieurs bases de données au sein de l'instance peuvent ne pas être restaurées de manière cohérente.
- Pour une instance de base de données SQL Server, les modes `OFFLINE`, `EMERGENCY` et `SINGLE_USER` ne sont pas pris en charge. La configuration d'une base de données avec l'un de ces modes entraîne le blocage de la date/heure de restauration la plus récente pour la totalité de l'instance.
- Certaines actions, telles que la modification du modèle de restauration d'une base de données SQL Server, peuvent interrompre la séquence des journaux utilisés pour la point-in-time restauration. Dans certains cas, Amazon RDS peut détecter ce problème et la date/heure de restauration la plus récente est bloquée. Dans d'autres cas, par exemple lorsqu'une base de données SQL Server utilise le modèle de récupération `BULK_LOGGED`, la rupture de la séquence de journalisation n'est pas détectée. Il peut s'avérer impossible de restaurer une instance de base de données SQL Server à un instant dans le passé s'il existe une rupture dans la séquence de journalisation. Pour ces raisons, Amazon RDS ne prend pas en charge le modèle de récupération des bases de données SQL Server.

Vous pouvez également l'utiliser AWS Backup pour gérer les sauvegardes des instances de base de données Amazon RDS. Si votre instance de base de données est associée à un plan de sauvegarde dans AWS Backup, ce plan de sauvegarde est utilisé pour la point-in-time restauration.

Les sauvegardes créées avec des noms AWS Backup se terminant par`awsbackup:AWS-Backup-job-number`. Pour plus d'informations à ce sujet AWS Backup, consultez le guide du AWS Backup développeur.

> **Note**
>
> Les informations de cette rubrique s'appliquent à Amazon RDS. Pour de plus amples informations sur la restauration d'un cluster de base de données Amazon Aurora, veuillez consulter Restauration d'un cluster de base de données à un instant spécifié.

Vous pouvez restaurer une instance de base de données à un moment donné à l'aide de l' AWS Management Console API AWS CLI, de ou de l'API RDS.

> **Note**
>
> Vous ne pouvez pas réduire la quantité de stockage lorsque vous restaurez une instance de base de données. Lorsque vous augmentez la valeur du stockage alloué, vous devez le faire d'au moins 10 %. Si vous tentez d'augmenter la valeur de moins de 10 %, une erreur s'affiche. Vous ne pouvez pas augmenter le stockage alloué lors de la restauration des instances de base de données RDS for SQL Server.

console

Pour restaurer une instance de base de données à un moment donné

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).

   Les sauvegardes automatisées sont affichées dans l'onglet Current Region (Région actuelle).
3. Choisissez l'instance de base de données que vous souhaitez restaurer.
4. Sous Actions, sélectionnez Restaurer à un moment donné.

   La fenêtre Restaurer à un instant dans le passé s'affiche.
5. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

Si vous choisissez Custom (Personnalisé), saisissez la date et l'heure auxquelles vous souhaitez restaurer l'instance.

> **Note**
>
> Les heures sont exprimées dans votre fuseau horaire local, qui est indiqué par son décalage par rapport à l'heure UTC. Par exemple, UTC-5 est l'heure normale de l'Est/ heure avancée du Centre.

6. Pour Identifiant d'instance de base de données, entrez le nom de l'instance de base de données restaurée. Le nom doit être unique.
7. Choisissez d'autres options selon vos besoins, telles que la classe d'instance de base de données et le stockage, ou le fait que vous voulez utiliser la mise à l'échelle automatique du stockage.

   Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.
8. Choisissez Restaurer à un instant dans le passé.

AWS CLI

Pour restaurer une instance de base de données à une heure spécifiée, utilisez la AWS CLI commande restore-db-instance-to- point-in-time pour créer une nouvelle instance de base de données. Cet exemple définit également la taille de stockage allouée et active la mise à l'échelle automatique du stockage.

L'étiquetage des ressources est pris en charge pour cette opération. Lorsque vous utilisez l'option `--tags`, les identifications d'instance de base de données source sont ignorées et celles qui sont fournies sont utilisées. Sinon, les dernières identifications de l'instance source sont utilisées.

Vous pouvez spécifier d'autres paramètres. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-to-point-in-time \
```

```
    --source-db-instance-identifier mysourcedbinstance \
    --target-db-instance-identifier mytargetdbinstance \
    --restore-time 2017-10-14T23:45:00.000Z \
    --allocated-storage 100 \
    --max-allocated-storage 1000
```

Dans Windows:

```
aws rds restore-db-instance-to-point-in-time ^
    --source-db-instance-identifier mysourcedbinstance ^
    --target-db-instance-identifier mytargetdbinstance ^
    --restore-time 2017-10-14T23:45:00.000Z ^
    --allocated-storage 100 ^
    --max-allocated-storage 1000
```

API RDS

Pour restaurer une instance de base de données à une date spécifiée, appelez l'opération d'API Amazon RDS `RestoreDBInstanceToPointInTime` avec les paramètres suivants :

- `SourceDBInstanceIdentifier`
- `TargetDBInstanceIdentifier`
- `RestoreTime`

Restauration d'un cluster de base de données multi-AZ à une date définie

Vous pouvez restaurer un cluster de base de données Multi-AZ à un moment précis dans le temps, en créant un nouveau cluster de base de données Multi-AZ.

RDS charge de façon continue les journaux de transactions pour les clusters de base de données multi-AZ sur Amazon S3. Vous pouvez procéder à une restauration à n'importe quel moment dans le passé au cours de la période de rétention des sauvegardes. Pour connaître l'heure de restauration la plus proche pour un cluster de base de données multi-AZ, utilisez la AWS CLI describe-db-clusterscommande. Consultez la valeur renvoyée dans le champ `EarliestRestorableTime` pour le cluster de base de données. Pour connaître la dernière heure de restauration d'un cluster DB Multi-AZ, regardez la valeur renvoyée dans le champ `LatestRestorableTime` correspondant au cluster DB.

Lorsque vous restaurez un cluster de base de données multi-AZ à un moment donné, vous pouvez choisir le groupe de sécurité VPC par défaut pour votre cluster de base de données multi-AZ, ou vous pouvez appliquer un groupe de sécurité VPC personnalisé à votre cluster de base de données multi-AZ.

Les clusters de base de données multi-AZ restaurés sont automatiquement associés au groupe de paramètres de cluster de base de données par défaut. Toutefois, vous pouvez appliquer un groupe de paramètres de cluster de base de données personnalisé en le spécifiant lors d'une restauration.

Si le cluster de base de données source possède des balises de ressources, RDS ajoute les dernières balises au cluster de base de données restauré.

> **Note**
>
> Nous vous recommandons de procéder à la restauration à une taille de cluster de base de données Multi-AZ identique ou similaire à celle du cluster de base de données source. Nous vous recommandons également de procéder à la restauration avec une valeur d'IOPS identique ou similaire si vous utilisez un stockage à IOPS provisionnés. Vous pouvez obtenir une erreur si, par exemple, vous choisissez une taille de cluster de base de données avec une valeur d'IOPS incompatible.
> Si le cluster de base de données multi-AZ source utilise un stockage SSD à usage général (gp3) et dispose de moins de 400 GiB de stockage alloué, vous ne pouvez pas modifier les IOPS provisionnées pour le cluster de base de données restauré.

Pour plus d'informations sur la restauration d'un cluster de base de données multi-AZ avec une version RDS Extended Support, consultez. Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Vous pouvez restaurer un cluster de base de données multi-AZ à un moment donné à l'aide de l'API AWS Management Console AWS CLI, de ou de l'API RDS.

Console

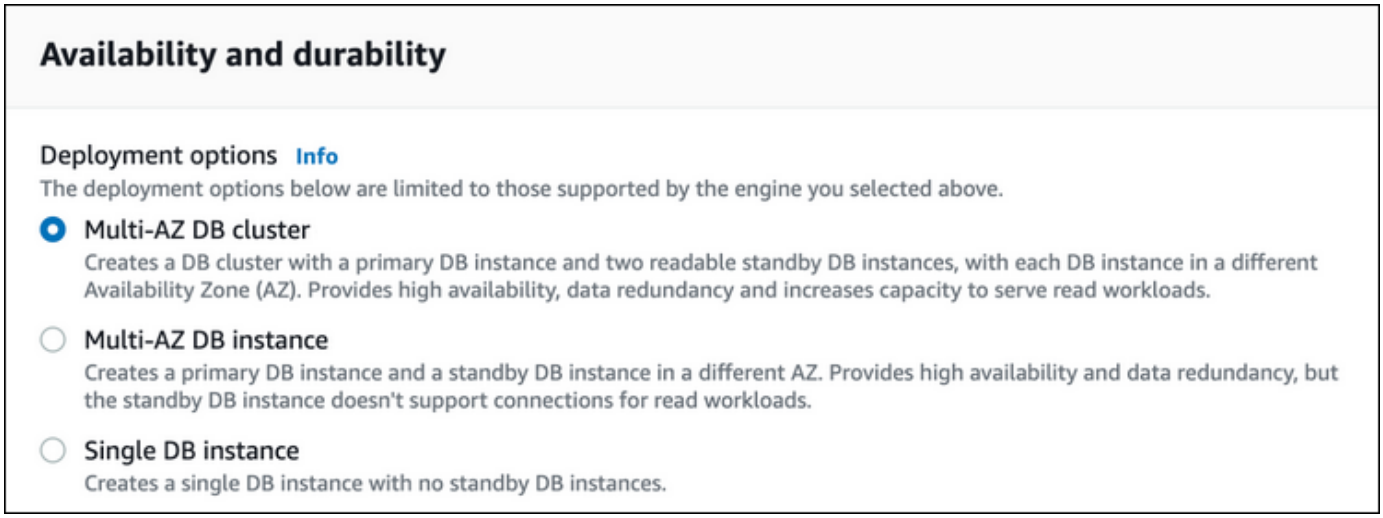
Pour restaurer un cluster de base de données multi-AZ à un instant dans le passé

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le cluster de base de données multi-AZ à restaurer.
4. Sous Actions, sélectionnez Restaurer à un moment donné.

   La fenêtre Restaurer à un instant dans le passé s'affiche.
5. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

   Si vous choisissez Custom (Personnalisé), saisissez la date et l'heure auxquelles vous souhaitez restaurer le cluster de base de données multi-AZ.

   > **Note**
   >
   > Les heures sont exprimées dans votre fuseau horaire local, qui est indiqué par son décalage par rapport à l'heure UTC. Par exemple, UTC-5 est l'heure normale de l'Est/ heure avancée du Centre.
6. Dans DB cluster identifier (Identifiant du cluster de base de données), saisissez le nom du cluster de base de données multi-AZ que vous avez restauré.
7. Dans Availability and durability (Disponibilité et durabilité), choisissez Multi-AZ DB cluster (Cluster de base de données Multi-AZ).



8. Pour DB instance class (Classe d'instance de base de données), choisissez une classe d'instance de base de données.

   Actuellement, les clusters DB Multi-AZ ne prennent en charge que les classes d'instance de base de données db.m6gd et db.r6gd. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .
9. Pour les sections restantes, spécifiez vos paramètres de cluster de base de données. Pour plus d'informations sur chaque paramètre, consultez Paramètres de création de clusters de base de données multi-AZ.
10. Choisissez Restaurer à un instant dans le passé.

AWS CLI

Pour restaurer un cluster de base de données multi-AZ à une heure spécifiée, utilisez la AWS CLI commande restore-db-cluster-to- point-in-time pour créer un nouveau cluster de base de données multi-AZ.

Actuellement, les clusters DB Multi-AZ ne prennent en charge que les classes d'instance de base de données db.m6gd et db.r6gd. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .

Example

Pour LinuxmacOS, ou Unix :

```
aws rds restore-db-cluster-to-point-in-time \
    --source-db-cluster-identifier mysourcemultiazdbcluster \
```

```
    --db-cluster-identifier mytargetmultiazdbcluster \
    --restore-to-time 2021-08-14T23:45:00.000Z \
    --db-cluster-instance-class db.r6gd.xlarge
```

Dans Windows :

```
aws rds restore-db-cluster-to-point-in-time ^
    --source-db-cluster-identifier mysourcemultiazdbcluster ^
    --db-cluster-identifier mytargetmultiazdbcluster ^
    --restore-to-time 2021-08-14T23:45:00.000Z ^
    --db-cluster-instance-class db.r6gd.xlarge
```

API RDS

Pour restaurer un cluster de base de données à une heure spécifiée, appelez l'ClusterToPointInTimeopération RestoreDB de l'API Amazon RDS avec les paramètres suivants :

- `SourceDBClusterIdentifier`
- `DBClusterIdentifier`
- `RestoreToTime`

Restauration d'un instantané dans un cluster de base de données multi-AZ

Vous pouvez restaurer un instantané sur un cluster de base de données multi-AZ à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Vous pouvez restaurer chacun des types d'instantanés suivants dans un cluster de base de données multi-AZ :

- Un instantané de déploiement mono-AZ
- Un instantané du déploiement d'un cluster de base de données multi-AZ avec une seule instance de base de données
- Un instantané de cluster de base de données multi-AZ

Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

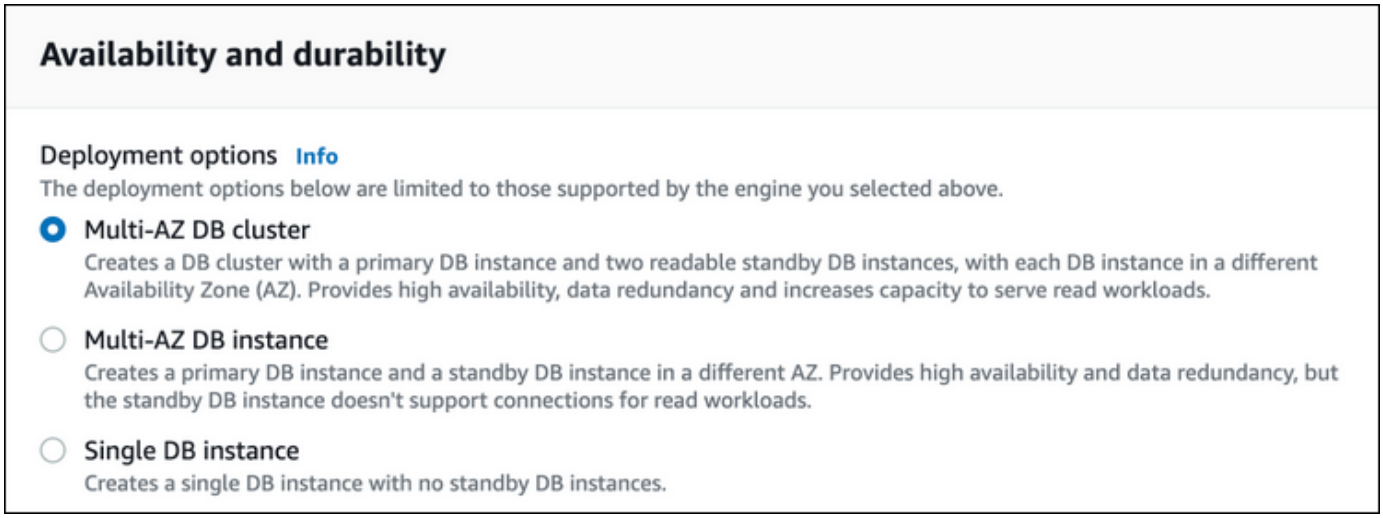
> **Tip**
>
> Vous pouvez migrer un déploiement mono-AZ ou un déploiement de cluster de bases de données multi-AZ vers un déploiement de cluster de bases de données multi-AZ en restaurant un instantané.

Pour plus d'informations sur la restauration d'un cluster de base de données multi-AZ avec une version RDS Extended Support, consultez. Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Console

Pour restaurer un instantané dans un cluster de base de données multi-AZ

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots (Instantanés).
3. Choisissez l'snapshot à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.
5. Sur la page Restore snapshot (Restaurer l'instantané), dans Availability and durability (Disponibilité et durabilité), choisissez Multi-AZ DB cluster (Cluster de base de données Multi-AZ).



6. Dans DB cluster identifier (Identifiant du cluster de base de données), saisissez le nom du cluster de base de données multi-AZ que vous avez restauré.
7. Pour les sections restantes, spécifiez vos paramètres de cluster de base de données. Pour plus d'informations sur chaque paramètre, consultez Paramètres de création de clusters de base de données multi-AZ.
8. Choisissez Restore DB Instance (Restaurer une instance de base de données).

AWS CLI

Pour restaurer un instantané sur un cluster de base de données multi-AZ, utilisez la AWS CLI commande restore-db-cluster-from-snapshot.

Dans l'exemple suivant, vous effectuez la restauration à partir d'un instantané créé précédemment sous le nom `mysnapshot`. Vous effectuez la restauration dans un cluster de base de données multi-AZ nommé `mynewmultiazdbcluster`. Vous spécifiez également la classe d'instance de base de données utilisée par les instances de base de données du cluster de base de données multi-AZ. Spécifiez `mysql` ou `postgres` pour le moteur de base de données.

Pour l'option `--snapshot-identifier`, vous pouvez utiliser le nom ou l'Amazon Resource Name (ARN) pour spécifier un instantané de cluster de bases de données. Cependant, vous pouvez utiliser uniquement l'ARN pour spécifier un instantané de base de données.

Pour l'option `--db-cluster-instance-class`, spécifiez la classe d'instance de base de données du nouveau cluster de bases de données multi-AZ. Les clusters de bases de données multi-AZ ne prennent en charge que des classes d'instance de base de données spécifiques, telles que les

classes d'instance de base de données db.m6gd et db.r6gd. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .

Vous pouvez également spécifier d'autres options.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds restore-db-cluster-from-snapshot \
    --db-cluster-identifier mynewmultiazdbcluster \
    --snapshot-identifier mysnapshot \
    --engine mysql|postgres \
    --db-cluster-instance-class db.r6gd.xlarge
```

Dans Windows :

```
aws rds restore-db-cluster-from-snapshot ^
    --db-cluster-identifier mynewmultiazdbcluster ^
    --snapshot-identifier mysnapshot ^
    --engine mysql|postgres ^
    --db-cluster-instance-class db.r6gd.xlarge
```

Une fois le cluster de bases de données restauré, vous pouvez ajouter le cluster de bases de données multi-AZ au groupe de sécurité associé au cluster de bases de données ou à l'instance de base de données qui a servi à créer l'instantané, le cas échéant. Cette action offre les même fonctions que le cluster de bases de données précédent ou l'instance de base de données précédente.

API RDS

Pour restaurer un instantané sur un cluster de base de données multi-AZ, appelez l'opération d'API RDS RestoreDB ClusterFromSnapshot avec les paramètres suivants :

- `DBClusterIdentifier`
- `SnapshotIdentifier`
- `Engine`

Vous pouvez également spécifier d'autres paramètres facultatifs.

Une fois le cluster de bases de données restauré, vous pouvez ajouter le cluster de bases de données multi-AZ au groupe de sécurité associé au cluster de bases de données ou à l'instance de base de données qui a servi à créer l'instantané, le cas échéant. Cette action offre les même fonctions que le cluster de bases de données précédent ou l'instance de base de données précédente.

Restauration d'un instantané de cluster de bases de données multi-AZ dans une instance de base de données

Multi-AZ DB cluster snapshot (Instantané de cluster de bases de données multi-AZ) crée un instantané du volume de stockage de votre cluster de bases de données en sauvegardant l'intégralité de ce dernier, et pas seulement les bases de données. Vous pouvez restaurer un instantané de cluster de bases de données multi-AZ dans un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ. Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

> **Note**
>
> Vous pouvez également restaurer un instantané de cluster de bases de données multi-AZ dans un nouveau cluster de bases de données multi-AZ. Pour obtenir des instructions, veuillez consulter Restauration d'un instantané dans un cluster de base de données multi-AZ.

Pour plus d'informations sur la restauration d'un cluster de base de données multi-AZ avec une version RDS Extended Support, consultez. Restauration d'une instance de base de données ou d'un cluster de base de données multi-AZ, d'un cluster avec Amazon RDS Extended Support

Utilisez l'API AWS Management Console, la ou l' AWS CLI API RDS pour restaurer un instantané de cluster de base de données multi-AZ dans le cadre d'un déploiement mono-AZ ou d'un déploiement d'instance de base de données multi-AZ.

Console

Pour restaurer un instantané de cluster de bases de données multi-AZ dans un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots (Instantanés).
3. Choisissez l'instantané de cluster de bases de données multi-AZ à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.
5. Sur la page Restore snapshot (Restaurer un instantané), dans Availability and durability (Disponibilité et durabilité), sélectionnez l'une des options suivantes :

- Single DB instance (Instance de base de données unique) : restaure l'instantané sur une instance de base de données sans instance de base de données de secours.
- Multi-AZ DB instance (Instance de base de données Multi-AZ) : restaure l'instantané dans un déploiement d'instance de base de données multi-AZ avec une instance de base de données primaire et une instance de base de données de secours.

6. Pour DB instance identifier (Identifiant de l'instance de base de données), saisissez le nom de l'instance de base de données restaurée.
7. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.
8. Choisissez Restore DB Instance (Restaurer une instance de base de données).

AWS CLI

Pour restaurer un instantané de cluster de base de données multi-AZ sur un déploiement d'instance de base de données, utilisez la AWS CLI commande restore-db-instance-from-db-snapshot.

Dans l'exemple suivant, vous effectuez la restauration à partir d'un instantané de cluster de bases de données multi-AZ créé précédemment sous le nom `myclustersnapshot`. Vous effectuez la restauration vers un déploiement d'instance de base de données multi-AZ avec une instance de base de données primaire nommée `mynewdbinstance`. Pour l'option `--db-cluster-snapshot-identifier`, spécifiez le nom de l'instantané du cluster de bases de données multi-AZ.

Pour l'option `--db-instance-class`, spécifiez la classe d'instance de base de données pour le déploiement de la nouvelle instance de base de données. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Classes d'instances de base de données .

Vous pouvez également spécifier d'autres options.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier mynewdbinstance \
    --db-cluster-snapshot-identifier myclustersnapshot \
    --engine mysql \
    --multi-az \
```

```
    --db-instance-class db.r6g.xlarge
```

Dans Windows :

```
aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier mynewdbinstance ^
    --db-cluster-snapshot-identifier myclustersnapshot ^
    --engine mysql ^
    --multi-az ^
    --db-instance-class db.r6g.xlarge
```

Une fois l'instance de base de données restaurée, vous pouvez l'ajouter au groupe de sécurité associé au cluster de bases de données multi-AZ qui a servi à créer l'instantané, le cas échéant. Cette action offre les même fonctions que pour le cluster de bases de données multi-AZ précédent.

API RDS

Pour restaurer un instantané de cluster de base de données multi-AZ sur un déploiement d'instance de base de données, appelez l'opération d'API RDS RestoreDB InstanceFrom DBSnapshot avec les paramètres suivants :

- `DBInstanceIdentifier`
- `DBClusterSnapshotIdentifier`
- `Engine`

Vous pouvez également spécifier d'autres paramètres facultatifs.

Une fois l'instance de base de données restaurée, vous pouvez l'ajouter au groupe de sécurité associé au cluster de bases de données multi-AZ qui a servi à créer l'instantané, le cas échéant. Cette action offre les même fonctions que pour le cluster de bases de données multi-AZ précédent.

Tutoriel : Restaurer une RDS instance de base de données Amazon à partir d'un instantané de base de données

Lorsque vous travaillez avec Amazon, RDS vous pouvez avoir une instance de base de données avec laquelle vous travaillez de temps en temps, mais dont vous n'avez pas besoin à plein temps. Supposons, par exemple, que vous organisiez une enquête trimestrielle auprès des clients qui utilise une EC2 instance Amazon pour héberger un site Web d'enquête auprès des clients. Vous disposez également d'une instance de base de données qui est utilisée pour stocker les résultats de l'enquête. Une façon d'économiser dans un tel scénario est de prendre un instantané de la base de données de l'instance de la base de données après la génération de l'enquête. Vous supprimez ensuite l'instance de base de données et vous la restaurez lorsque vous avez besoin de réaliser à nouveau l'enquête.

Lorsque vous restaurez une instance de base de données, vous fournissez le nom de l'instantané de la base de données à restaurer. Vous fournissez ensuite un nom pour la nouvelle instance de base de données qui est créée à partir de l'opération de restauration.

Pour plus d'informations sur la restauration d'instances de base de données à partir d'instantanés, veuillez consulter Restauration vers une instance de base de données.

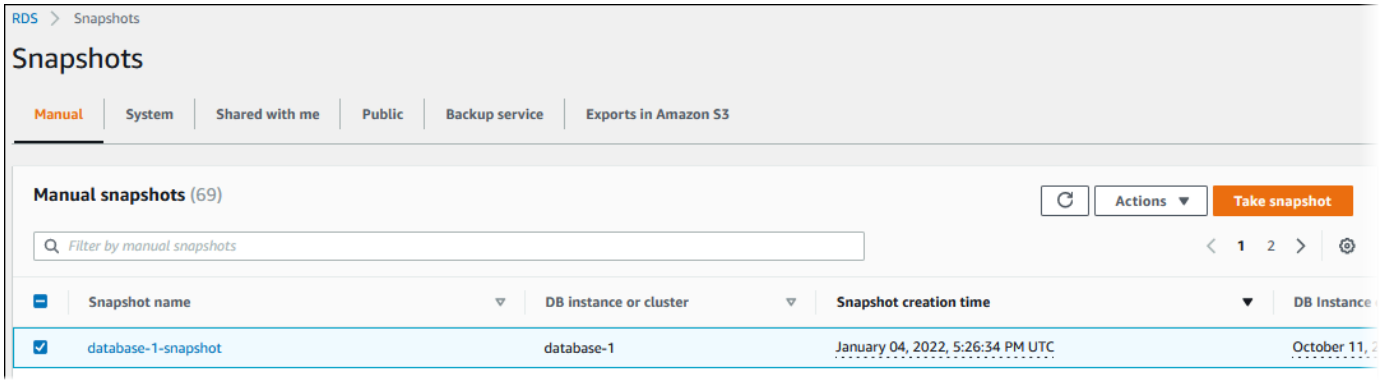
Pour plus d'informations sur AWS KMS gestion des clés pour AmazonRDS, voirAWS KMS key gestion.

Restauration d'une instance de base de données à partir d'un instantané de base de données

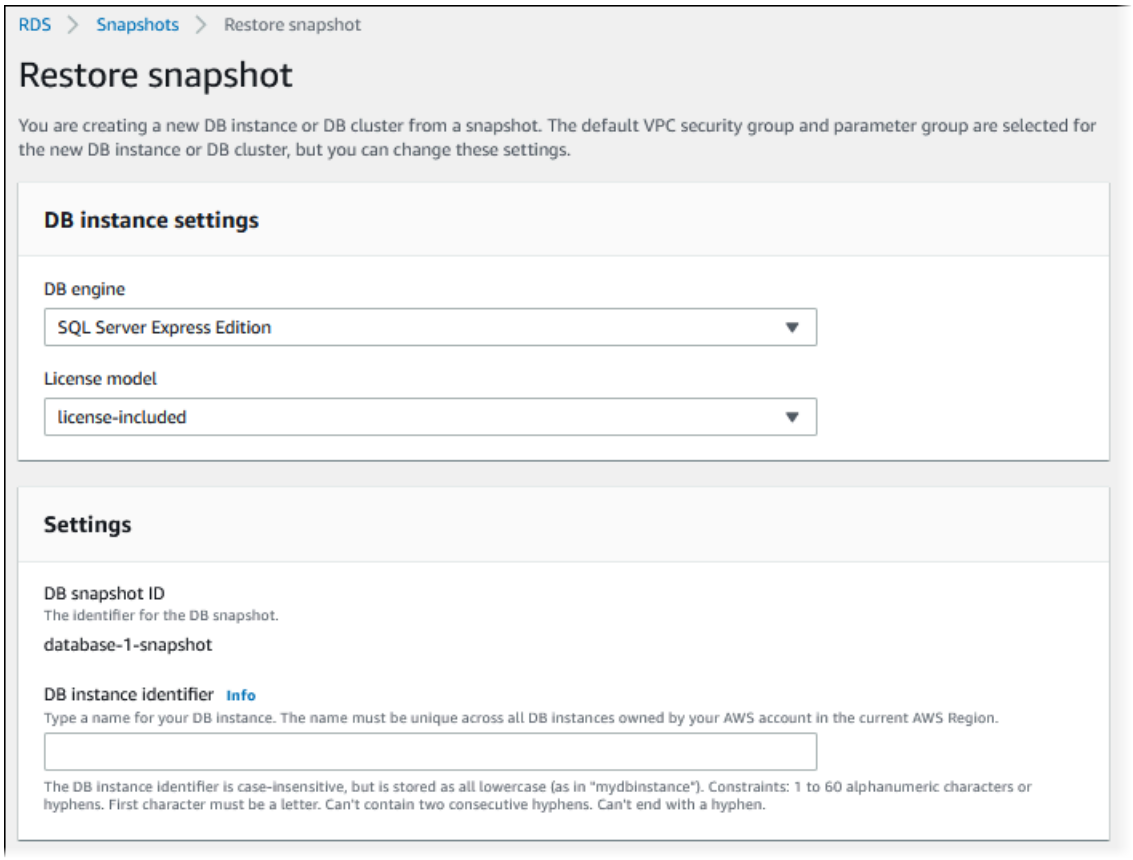
Utilisez la procédure suivante pour effectuer une restauration à partir d'un instantané dans AWS Management Console.

Pour restaurer une instance de base de données à partir d'un instantané de base de données

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.



La page Restaurer l'instantané s'affiche.



5. Sous Paramètres de l'instance de base de données, utilisez les paramètres par défaut pour le moteur de base de données et le modèle de licence (pour Oracle ou Microsoft SQL Server).
6. Sous Settings (Paramètres), pour DB instance identifier (Identifiant d'instance de base de données), saisissez le nom unique que vous voulez utiliser pour l'instance de base de données restaurée, par exemple **`mynewdbinstance`**.

Si vous restaurez à partir d'une instance de base de données que vous avez supprimée après avoir effectué l'instantané de base de données, vous pouvez utiliser le nom de cette instance de base de données.

7. Sous Disponibilité et durabilité, choisissez si vous créez une instance de secours dans une autre zone de disponibilité.

   Pour ce didacticiel, ne créez pas d'instance de secours.

8. Sous Connectivity (Connectivité), utilisez les paramètres par défaut pour les éléments suivants :

   - Cloud privé virtuel (VPC)
   - Groupe de sous-réseaux de base de données
   - Accès public
   - VPCgroupe de sécurité (pare-feu)

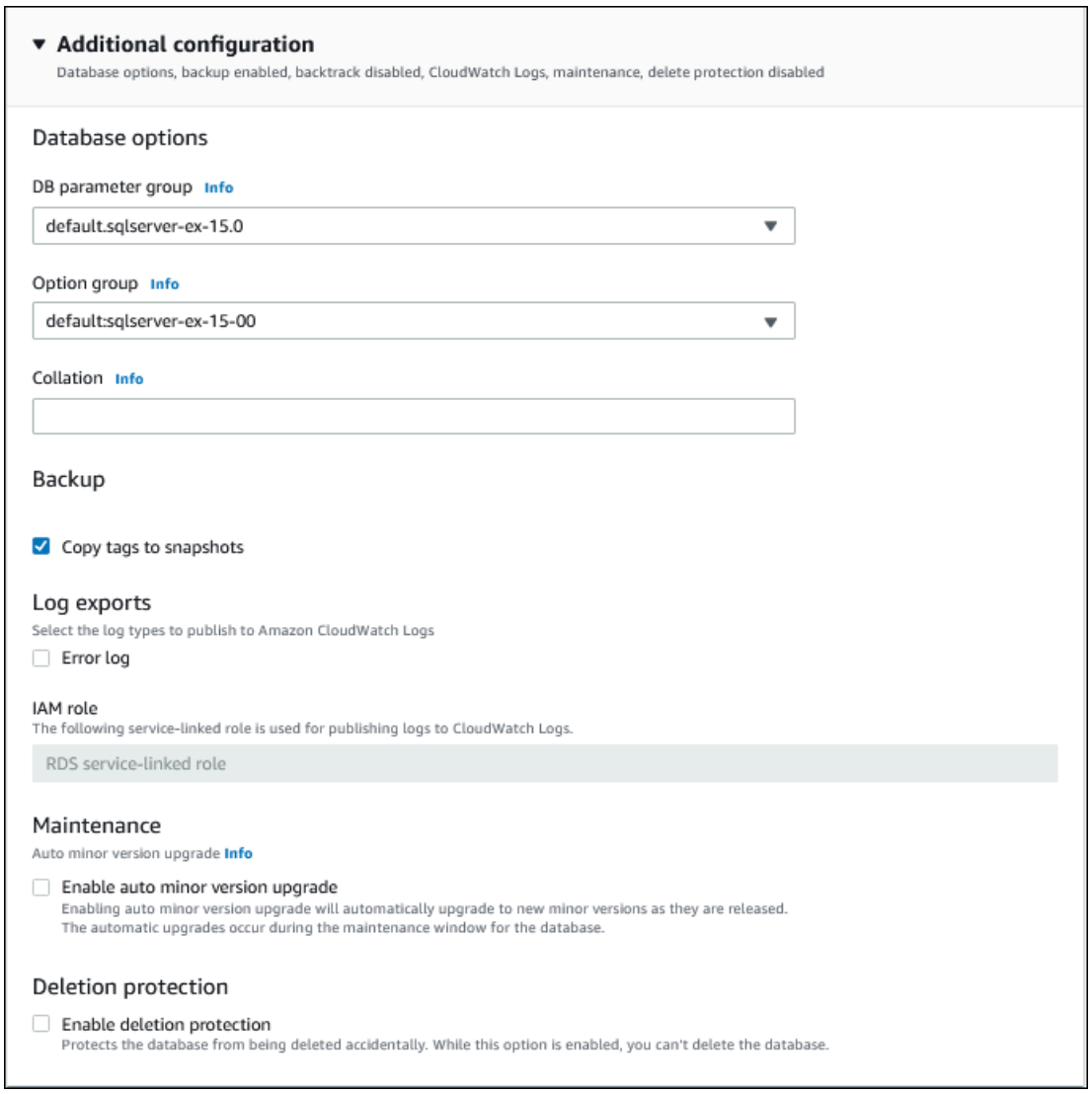
9. Choisissez la classe d'instance de base de données.

   Pour ce didacticiel, choisissez Burstable classes (includes t classes) (Classes à capacité extensible (inclut les classes t)), puis db.t3.small.

10. Pour Encryption (Chiffrement), utilisez les paramètres par défaut.

    Si l'instance de base de données source de l'instantané a été chiffrée, l'instance de base de données restaurée est également chiffrée. Vous ne pouvez pas la rendre non chiffrée.

11. Développez Additional configuration (Configuration supplémentaire) en bas de la page.



12. Effectuez les opérations suivantes sous Database options (Options de base de données) :

    a. Choisissez le DB parameter group (Groupe de paramètres de base de données).

       Pour ce didacticiel, utilisez le groupe de paramètres par défaut.

    b. Choisissez le Option group (Groupe d'options).

       Pour ce didacticiel, utilisez le groupe d'options par défaut.

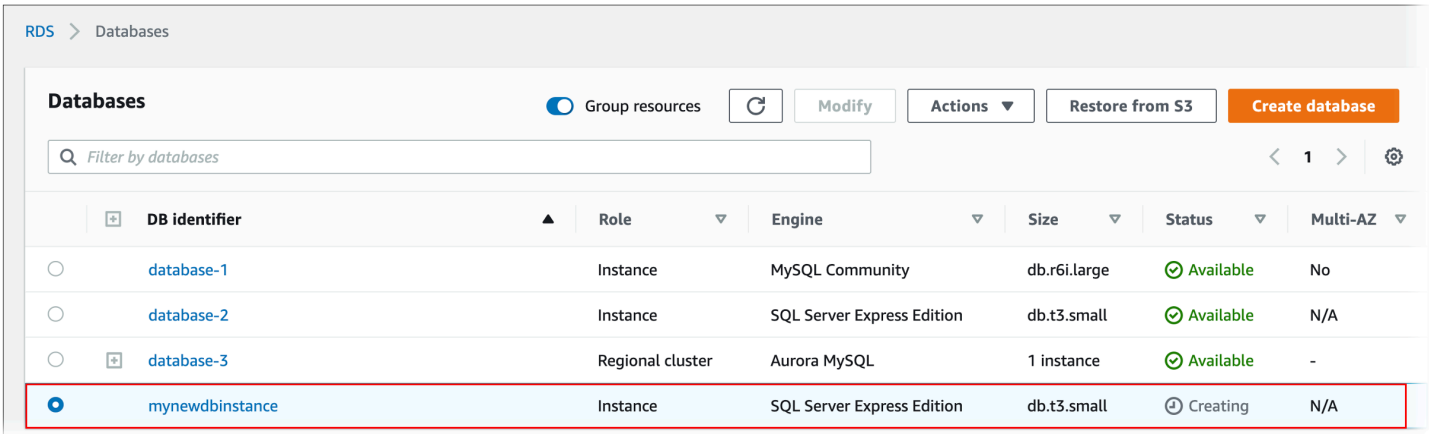
       > **Important**
       >
       > Dans certains cas, vous pouvez effectuer une restauration à partir de l'instantané de base de données d'une instance de base de données qui utilise une option

persistante ou permanente. Si tel est le cas, assurez-vous de choisir un groupe d'options qui utilise la même option.

c. Pour Deletion protection (Protection contre la suppression), cochez la case Enable deletion protection (Activer la protection contre la suppression).

13. Choisissez Restore DB Instance (Restaurer une instance de base de données).

La page Databases (Bases de données) affiche l'instance de base de données restaurée, avec le statut `Creating`.



Copier un instantané de base de données pour Amazon RDS

Avec Amazon RDS, vous pouvez copier des sauvegardes automatisées ou des instantanés de base de données manuels. Après avoir copié un instantané, la copie est un instantané manuel. Vous pouvez effectuer plusieurs copies d'une sauvegarde automatisée ou d'un instantané manuel, mais chaque copie doit avoir un identifiant unique.

Vous pouvez copier un instantané à l'intérieur de celui-ci Région AWS, vous pouvez copier un instantané par-dessus Régions AWS et vous pouvez copier des instantanés partagés.

Copie d'un instantané de base de données

Pour chacun Compte AWS, vous pouvez copier jusqu'à 20 instantanés de base de données à la fois de l'un Région AWS à l'autre. Si vous copiez un instantané de base de données vers un autre Région AWS, vous créez un instantané de base de données manuel qui y est conservé Région AWS. La copie d'un instantané de base de données depuis la source Région AWS entraîne des frais de transfert de données Amazon RDS.

Pour de plus amples informations sur la tarification du transfert des données, veuillez consulter la Tarification Amazon RDS.

Une fois que la copie instantanée de base de données a été créée dans le nouveau Région AWS, elle se comporte de la même manière que tous les autres instantanés de base de données qu'elle contient. Région AWS

Vous pouvez copier un instantané de base de données à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

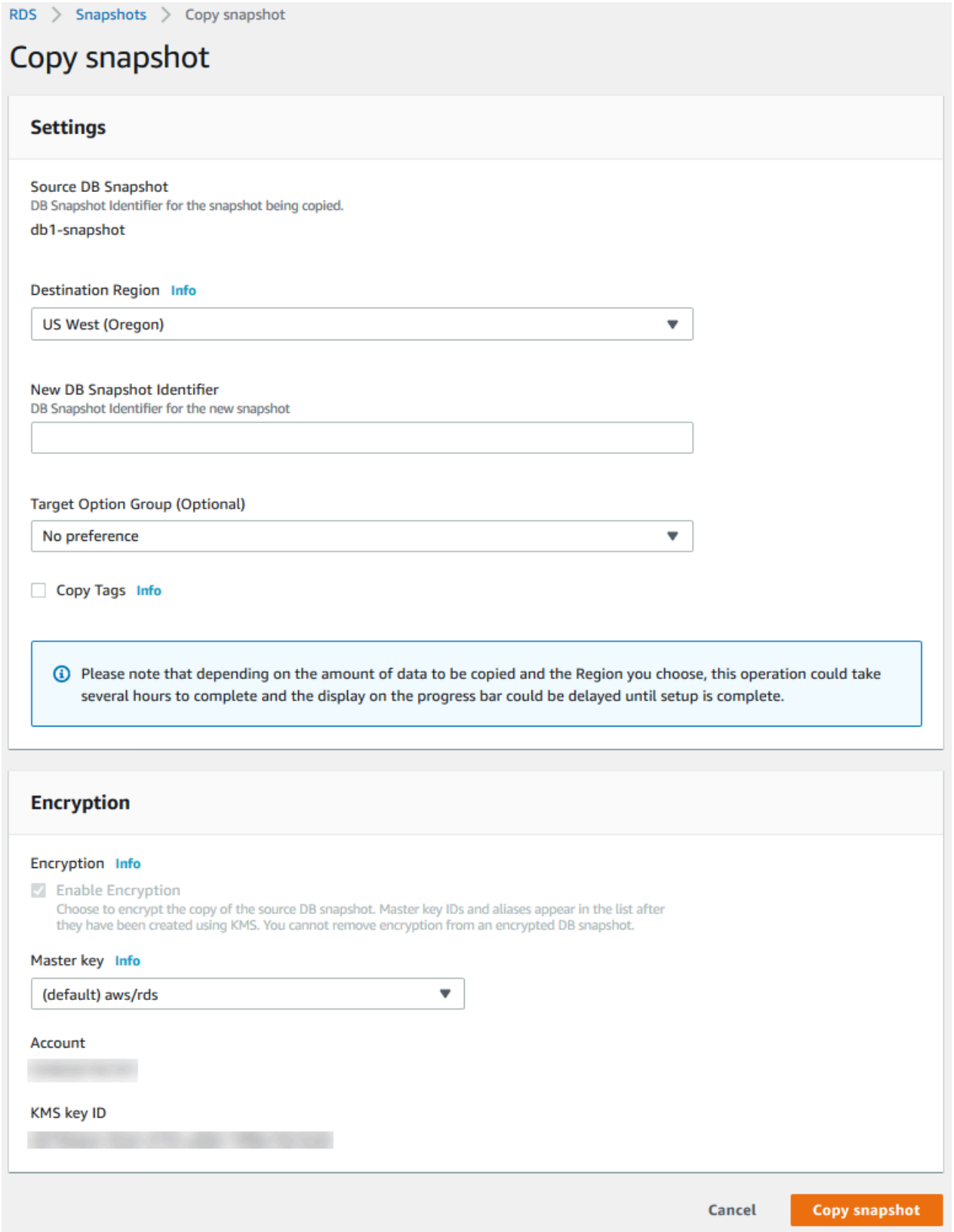
console

La procédure suivante copie un instantané de base de données chiffré ou non chiffré, dans la même région Région AWS ou dans plusieurs régions, à l'aide du AWS Management Console.

Pour copier un instantané de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Sélectionnez l'instantané de base de données que vous voulez copier.
4. Sous Actions, choisissez Copier un instantané.

La page Copier un instantané apparaît.



5. Pour Target option group (optional) (Groupe d'options cible (facultatif)), choisissez un nouveau groupe d'options si vous le souhaitez.

   Spécifiez cette option si vous copiez un instantané de l'un Région AWS vers l'autre et si votre instance de base de données utilise un groupe d'options autre que celui par défaut.

   Si votre instance de base de données source utilise le chiffrement TDE (Transparent Data Encryption) pour Oracle ou Microsoft SQL Server, vous devez spécifier cette option lors d'une copie entre régions. Pour de plus amples informations, veuillez consulter Considérations relatives aux groupes d'options.
6. (Facultatif) Pour copier le cliché de base de données vers un autre Région AWS, choisissez le nouveau pour Région de destination Région AWS.

   > **Note**
   >
   > La destination Région AWS doit disposer de la même version de moteur de base de données disponible que la source Région AWS.

7. Pour New DB snapshot identifier (Nouvel identificateur d'instantané de base de données), saisissez le nom de la copie de l'instantané de base de données.

   Vous pouvez effectuer plusieurs copies d'une sauvegarde automatisée ou d'un instantané manuel, mais chaque copie doit avoir un identifiant unique.
8. (Facultatif) Pour copier les balises et les valeurs de l'instantané vers la copie de cet instantané, choisissez Copier les balises.
9. (Facultatif) Pour Chiffrement, procédez comme suit :

   a. Si l'instantané de bases de données n'est pas chiffré mais que vous souhaitez chiffrer la copie, choisissez Enable encryption (Activer le chiffrement).

      > **Note**
      >
      > Si l'instantané de bases de données est chiffré, vous devez chiffrer la copie, de sorte que la case à cocher est déjà activée.

   b. Pour AWS KMS key, spécifiez l'identifiant de clé KMS à utiliser pour chiffrer la copie de l'instantané de base de données.
10. Choisissez Copy snapshot (Copier un instantané).

AWS CLI

Vous pouvez copier un instantané de base de données à l'aide de la AWS CLI commande copy-db-snapshot. Si vous copiez le cliché dans un nouveau Région AWS, exécutez la commande dans le nouveau Région AWS.

Les options suivantes sont utilisées pour copier un instantané de base de données. Toutes les options ne sont pas requises pour tous les scénarios. Utilisez les descriptions et les exemples qui suivent pour déterminer quelles options utiliser.

- `--source-db-snapshot-identifier` – L'identifiant de l'instantané de base de données source.
  - Si l'instantané source est Région AWS identique à la copie, spécifiez un identifiant d'instantané de base de données valide. Par exemple, `rds:mysql-instance1-snapshot-20130805`.
  - Si l'instantané source est identique à la copie et Région AWS qu'il a été partagé avec vous Compte AWS, spécifiez un ARN d'instantané de base de données valide. Par exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805`.
  - Si l'instantané source se trouve dans un emplacement différent Région AWS de celui de la copie, spécifiez un ARN d'instantané de base de données valide. Par exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805`.
  - Si vous effectuez la copie à partir d'un instantané de base de données manuel partagé, ce paramètre doit être l'Amazon Resource Name (ARN) de l'instantané de base de données partagé.
  - Si vous copiez un instantané chiffré, ce paramètre doit être au format ARN de la source Région AWS et doit correspondre `SourceDBSnapshotIdentifier` à celui du `PreSignedUrl` paramètre.
- `--target-db-snapshot-identifier` – L'identifiant de la nouvelle copie de l'instantané de base de données chiffré.
- `--copy-option-group` : copiez le groupe d'options à partir d'un instantané qui a été partagé avec votre Compte AWS.
- `--copy-tags` – Inclut l'option de copie des balises pour copier les balises et les valeurs de l'instantané vers la copie de cet instantané.
- `--option-group-name` – Groupe d'options à associer à la copie de l'instantané.

Spécifiez cette option si vous copiez un instantané de l'un Région AWS vers l'autre et si votre instance de base de données utilise un groupe d'options autre que celui par défaut.

Si votre instance de base de données source utilise le chiffrement TDE (Transparent Data Encryption) pour Oracle ou Microsoft SQL Server, vous devez spécifier cette option lors d'une copie entre régions. Pour de plus amples informations, veuillez consulter Considérations relatives aux groupes d'options.

- `--kms-key-id` – Identificateur de clé KMS pour un instantané de base de données chiffré. L'identificateur de clé KMS est l'ARN (Amazon Resource Name), l'identificateur de clé ou l'alias de clé pour la clé KMS.
  - Si vous copiez un instantané de base de données chiffré depuis votre Compte AWS, vous pouvez spécifier une valeur pour ce paramètre afin de chiffrer la copie avec une nouvelle clé KMS. Si vous ne spécifiez pas de valeur pour ce paramètre, la copie de l'instantané de base de données est chiffrée avec la même clé KMS que l'instantané de base de données source.
  - Si vous copiez un instantané de base de données chiffré partagé depuis un autre Compte AWS, vous devez spécifier une valeur pour ce paramètre.
  - Si vous spécifiez ce paramètre lorsque vous copiez un instantané non chiffré, la copie est chiffrée.
  - Si vous copiez un instantané chiffré vers un autre Région AWS, vous devez spécifier une clé KMS pour la destination Région AWS. Les clés KMS sont spécifiques à l' Région AWS endroit dans lequel elles ont été créées, et vous ne pouvez pas utiliser les clés de chiffrement les unes Région AWS des autres Région AWS.

Example source non chiffrée, même région de destination

Le code suivant crée une copie d'un instantané, sous le nouveau nom`mydbsnapshotcopy`, Région AWS identique à l'instantané source. Lorsque la copie est réalisée, les identifications et le groupe d'options de base de données sur l'instantané d'origine sont copiés dans la copie de l'instantané.

Pour LinuxmacOS, ou Unix :

```
aws rds copy-db-snapshot \
    --source-db-snapshot-identifier arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805 \
    --target-db-snapshot-identifier mydbsnapshotcopy \
    --copy-option-group \
```

```
    --copy-tags
```

Dans Windows :

```
aws rds copy-db-snapshot ^
    --source-db-snapshot-identifier arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805 ^
    --target-db-snapshot-identifier mydbsnapshotcopy ^
    --copy-option-group ^
    --copy-tags
```

Example source non chiffrée, autre région de destination

Le code suivant crée une copie d'un instantané, portant le nouveau nom`mydbsnapshotcopy`, Région AWS dans lequel la commande est exécutée.

Pour LinuxmacOS, ou Unix :

```
aws rds copy-db-snapshot \
    --source-db-snapshot-identifier arn:aws:rds:us-east-1:123456789012:snapshot:mysql-instance1-snapshot-20130805 \
    --target-db-snapshot-identifier mydbsnapshotcopy
```

Dans Windows :

```
aws rds copy-db-snapshot ^
    --source-db-snapshot-identifier arn:aws:rds:us-east-1:123456789012:snapshot:mysql-instance1-snapshot-20130805 ^
    --target-db-snapshot-identifier mydbsnapshotcopy
```

Example source chiffrée, autre région de destination

L'exemple de code suivant copie un instantané de bases de données chiffré à partir de la région USA Ouest (Oregon) vers la région US East (N. Virginia). Exécutez la commande dans la région de destination (us-east-1).

Pour LinuxmacOS, ou Unix :

```
aws rds copy-db-snapshot \
    --source-db-snapshot-identifier arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20161115 \
```

```
    --target-db-snapshot-identifier mydbsnapshotcopy \
    --kms-key-id my-us-east-1-key \
    --option-group-name custom-option-group-name
```

Dans Windows :

```
aws rds copy-db-snapshot ^
    --source-db-snapshot-identifier arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20161115 ^
    --target-db-snapshot-identifier mydbsnapshotcopy ^
    --kms-key-id my-us-east-1-key ^
    --option-group-name custom-option-group-name
```

Le `--source-region` paramètre est obligatoire lorsque vous copiez un instantané chiffré entre les régions AWS GovCloud (USA Est) et AWS GovCloud (USA Ouest). Pour `--source-region`, spécifiez la Région AWS de l'instance de base de données source.

Si `--source-region` n'est pas spécifié, spécifiez une valeur `--pre-signed-url`. Une URL présignée est une URL qui contient une demande signée via Signature Version 4 pour la commande `copy-db-snapshot` qui est appelée dans la Région AWS source. Pour en savoir plus sur `pre-signed-url` cette option, consultez copy-db-snapshotla référence des AWS CLI commandes.

API RDS

Vous pouvez copier un instantané de base de données à l'aide de l'opération Copy DBSnapshot de l'API Amazon RDS. Si vous copiez le cliché dans un nouveau Région AWS, effectuez l'action dans le nouveau Région AWS.

Les paramètres suivants sont utilisés pour copier un instantané de base de données. Tous les paramètres ne sont pas requis pour tous les scénarios. Utilisez les descriptions et les exemples qui suivent pour déterminer quels paramètres utiliser.

- `SourceDBSnapshotIdentifier` – L'identifiant de l'instantané de base de données source.
  - Si l'instantané source est Région AWS identique à la copie, spécifiez un identifiant d'instantané de base de données valide. Par exemple, `rds:mysql-instance1-snapshot-20130805`.
  - Si l'instantané source est identique à la copie et Région AWS qu'il a été partagé avec vous Compte AWS, spécifiez un ARN d'instantané de base de données valide. Par exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805`.

  - Si l'instantané source se trouve dans un emplacement différent Région AWS de celui de la copie, spécifiez un ARN d'instantané de base de données valide. Par exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20130805`.
  - Si vous effectuez la copie à partir d'un instantané de base de données manuel partagé, ce paramètre doit être l'Amazon Resource Name (ARN) de l'instantané de base de données partagé.
  - Si vous copiez un instantané chiffré, ce paramètre doit être au format ARN de la source Région AWS et doit correspondre `SourceDBSnapshotIdentifier` à celui du `PreSignedUrl` paramètre.
- `TargetDBSnapshotIdentifier` – L'identifiant de la nouvelle copie de l'instantané de base de données chiffré.
- `CopyOptionGroup` : affectez à ce paramètre la valeur `true` pour copier le groupe d'options à partir d'un instantané partagé vers la copie de cet instantané. L'argument par défaut est `false`.
- `CopyTags` – Affectez à ce paramètre la valeur `true` pour copier les balises et les valeurs de l'instantané vers la copie de cet instantané. La valeur par défaut est `false`.
- `OptionGroupName` – Groupe d'options à associer à la copie de l'instantané.

  Spécifiez ce paramètre si vous copiez un instantané de l'un Région AWS vers l'autre et si votre instance de base de données utilise un groupe d'options autre que celui par défaut.

  Si votre instance de base de données source utilise le chiffrement TDE (Transparent Data Encryption) pour Oracle ou Microsoft SQL Server, vous devez spécifier ce paramètre lors d'une copie entre régions. Pour de plus amples informations, veuillez consulter Considérations relatives aux groupes d'options.
- `KmsKeyId` – Identificateur de clé KMS pour un instantané de base de données chiffré. L'identificateur de clé KMS est l'ARN (Amazon Resource Name), l'identificateur de clé ou l'alias de clé pour la clé KMS.
  - Si vous copiez un instantané de base de données chiffré depuis votre Compte AWS, vous pouvez spécifier une valeur pour ce paramètre afin de chiffrer la copie avec une nouvelle clé KMS. Si vous ne spécifiez pas de valeur pour ce paramètre, la copie de l'instantané de base de données est chiffrée avec la même clé KMS que l'instantané de base de données source.
  - Si vous copiez un instantané de base de données chiffré partagé depuis un autre Compte AWS, vous devez spécifier une valeur pour ce paramètre.
  - Si vous spécifiez ce paramètre lorsque vous copiez un instantané non chiffré, la copie est chiffrée.

- Si vous copiez un instantané chiffré vers un autre Région AWS, vous devez spécifier une clé KMS pour la destination Région AWS. Les clés KMS sont spécifiques à l' Région AWS endroit dans lequel elles ont été créées, et vous ne pouvez pas utiliser les clés de chiffrement les unes Région AWS des autres Région AWS.

- `PreSignedUrl`— L'URL contenant une demande signée Signature Version 4 pour l'opération d'`CopyDBSnapshot`API dans la source Région AWS qui contient l'instantané de base de données source à copier.

  Spécifiez ce paramètre lorsque vous copiez un instantané de base de données chiffré depuis un autre à Région AWS l'aide de l'API Amazon RDS. Vous pouvez spécifier l'option de région source à la place de ce paramètre lorsque vous copiez un instantané de base de données chiffré à partir d'une autre Région AWS en utilisant AWS CLI.

  L'URL pré-signée doit être une demande valide pour l'opération d'API `CopyDBSnapshot` qui peut être exécutée dans la Région AWS source qui contient l'instantané de base de données chiffré à copier. La demande d'URL pré-signée doit contenir les valeurs de paramètres suivantes :

  - `DestinationRegion`— L'endroit Région AWS où le snapshot de base de données chiffré sera copié. C' Région AWS est la même que celle dans laquelle est appelée l'`CopyDBSnapshot`opération contenant cette URL présignée.

    Par exemple, supposons que vous copiez un instantané de bases de données chiffré à partir de la région us-west-2 vers la région us-east-1. Vous appelez ensuite l'opération `CopyDBSnapshot` dans la région us-east-1 et fournissez une URL pré-signée qui contient un appel à l'opération `CopyDBSnapshot` dans la région us-west-2. Pour cet exemple, il convient d'affecter la région us-east-1 au paramètre `DestinationRegion` figurant dans l'URL présignée.

  - `KmsKeyId` – Identificateur de clé KMS de la clé à utiliser pour chiffrer la copie de l'instantané de base de données dans la Région AWS de destination. Il s'agit du même identifiant pour l'`CopyDBSnapshot`opération appelée dans la destination Région AWS et pour l'opération contenue dans l'URL présignée.

  - `SourceDBSnapshotIdentifier` – Identifiant d'instantané de bases de données pour l'instantané chiffré à copier. Cet identifiant doit être au format Amazon Resource Name (ARN) pour la Région AWS source. Par exemple, si vous copiez un instantané de base de données chiffré depuis la région us-west-2, `SourceDBSnapshotIdentifier` votre image ressemble à l'exemple suivant :. `arn:aws:rds:us-west-2:123456789012:snapshot:mysql-instance1-snapshot-20161115`

Pour plus d'informations sur les demandes signées par Signature Version 4, consultez les informations suivantes :

- Authentification des demandes : utilisation des paramètres de requête (AWS signature version 4) dans le manuel Amazon Simple Storage Service API Reference
- Processus de signature de la version 4 dans le Références générales AWS

Example source non chiffrée, même région de destination

Le code suivant crée une copie d'un instantané, sous le nouveau nom`mydbsnapshotcopy`, Région AWS identique à l'instantané source. Lorsque la copie est réalisée, toutes les balises de l'instantané d'origine sont copiées dans la copie de l'instantané.

```
https://rds.us-west-1.amazonaws.com/
 ?Action=CopyDBSnapshot
 &CopyTags=true
 &SignatureMethod=HmacSHA256
 &SignatureVersion=4
 &SourceDBSnapshotIdentifier=mysql-instance1-snapshot-20130805
 &TargetDBSnapshotIdentifier=mydbsnapshotcopy
 &Version=2013-09-09
 &X-Amz-Algorithm=AWS4-HMAC-SHA256
 &X-Amz-Credential=AKIADQKE4SARGYLE/20140429/us-west-1/rds/aws4_request
 &X-Amz-Date=20140429T175351Z
 &X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
 &X-Amz-Signature=9164337efa99caf850e874a1cb7ef62f3cea29d0b448b9e0e7c53b288ddffed2
```

Example source non chiffrée, autre région de destination

Le code suivant crée une copie d'instantané, avec le nouveau nom `mydbsnapshotcopy`, dans la région USA Ouest (Californie du Nord).

```
https://rds.us-west-1.amazonaws.com/
 ?Action=CopyDBSnapshot
 &SignatureMethod=HmacSHA256
 &SignatureVersion=4
 &SourceDBSnapshotIdentifier=arn%3Aaws%3Ards%3Aus-east-1%3A123456789012%3Asnapshot
%3Amysql-instance1-snapshot-20130805
 &TargetDBSnapshotIdentifier=mydbsnapshotcopy
 &Version=2013-09-09
```

```
 &X-Amz-Algorithm=AWS4-HMAC-SHA256
 &X-Amz-Credential=AKIADQKE4SARGYLE/20140429/us-west-1/rds/aws4_request
 &X-Amz-Date=20140429T175351Z
 &X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
 &X-Amz-Signature=9164337efa99caf850e874a1cb7ef62f3cea29d0b448b9e0e7c53b288ddffed2
```

Example source chiffrée, autre région de destination

Le code suivant crée une copie d'instantané, avec le nouveau nom `mydbsnapshotcopy`, dans la région US East (N. Virginia).

```
https://rds.us-east-1.amazonaws.com/
    ?Action=CopyDBSnapshot
    &KmsKeyId=my-us-east-1-key
    &OptionGroupName=custom-option-group-name
    &PreSignedUrl=https%253A%252F%252Frds.us-west-2.amazonaws.com%252F
         %253FAction%253DCopyDBSnapshot
         %2526DestinationRegion%253Dus-east-1
         %2526KmsKeyId%253Dmy-us-east-1-key
         %2526SourceDBSnapshotIdentifier%253Darn%25253Aaws%25253Ards%25253Aus-
west-2%25253A123456789012%25253Asnapshot%25253Amysql-instance1-snapshot-20161115
         %2526SignatureMethod%253DHmacSHA256
         %2526SignatureVersion%253D4
         %2526Version%253D2014-10-31
         %2526X-Amz-Algorithm%253DAWS4-HMAC-SHA256
         %2526X-Amz-Credential%253DAKIADQKE4SARGYLE%252F20161117%252Fus-west-2%252Frds
%252Faws4_request
         %2526X-Amz-Date%253D20161117T215409Z
         %2526X-Amz-Expires%253D3600
         %2526X-Amz-SignedHeaders%253Dcontent-type%253Bhost%253Buser-agent%253Bx-amz-
content-sha256%253Bx-amz-date
         %2526X-Amz-Signature
%253D255a0f17b4e717d3b67fad163c3ec26573b882c03a65523522cf890a67fca613
    &SignatureMethod=HmacSHA256
    &SignatureVersion=4
    &SourceDBSnapshotIdentifier=arn%3Aaws%3Ards%3Aus-west-2%3A123456789012%3Asnapshot
%3Amysql-instance1-snapshot-20161115
    &TargetDBSnapshotIdentifier=mydbsnapshotcopy
    &Version=2014-10-31
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Credential=AKIADQKE4SARGYLE/20161117/us-east-1/rds/aws4_request
    &X-Amz-Date=20161117T221704Z
    &X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-date
```

```
&X-Amz-Signature=da4f2da66739d2e722c85fcfd225dc27bba7e2b8dbea8d8612434378e52adccf
```

Limites

Vous trouverez ci-dessous certaines limites qui s'appliquent lorsque vous copiez des instantanés :

- Vous ne pouvez pas copier un instantané vers ou depuis la région Chine (Beijing) ou la région Chine (Ningxia).
- Vous pouvez copier un instantané entre AWS GovCloud (US-East) et AWS GovCloud (US-West). Toutefois, vous ne pouvez pas copier un instantané entre ces régions GovCloud (États-Unis) et les régions qui ne sont pas des régions GovCloud (États-Unis).
- Si vous supprimez un instantané source avant que l'instantané cible ne soit disponible, la copie d'instantané peut échouer. Vérifiez que l'instantané cible a le statut `AVAILABLE` avant de supprimer un instantané source.
- Vous pouvez avoir jusqu'à 20 demandes de copie d'instantanés en cours vers une même région de destination par compte.
- Lorsque vous demandez plusieurs copies d'instantanés pour la même instance de base de données source, elles sont mises en file d'attente en interne. Les copies demandées ultérieurement ne démarreront pas tant que les copies de l'instantané précédent ne seront pas terminées. Pour plus d'informations, voir Pourquoi la création de mon instantané EC2 AMI ou EBS est-elle lente ? dans le AWS Knowledge Center.
- En fonction du type Régions AWS concerné et de la quantité de données à copier, la réalisation d'une copie instantanée entre régions peut prendre des heures. Dans certains cas, il peut y avoir un grand nombre de demandes de copie d'instantanés d'une région à une autre à partir d'une région source donnée. Dans de tels cas, Amazon RDS peut mettre en file d'attente les nouvelles demandes de copie entre régions provenant de cette région source en attendant que certaines copies en cours se terminent. Aucune information d'avancement n'est affichée sur les demandes de copie quand elles sont en file d'attente. Les informations d'avancement sont affichées lorsque la copie commence.
- Si une copie est toujours en attente lorsque vous démarrez une autre copie, la deuxième copie ne démarre qu'une fois la première copie terminée.
- Vous ne pouvez pas copier un instantané d'un cluster de base de données multi-AZ.
- Vous pouvez uniquement copier des instantanés d'instances de base de données qui utilisent des volumes io2 vers des emplacements Régions AWS où les volumes io2 Block Express sont disponibles. Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.

Considérations

Pour connaître les points à prendre en compte lors de la copie d'un instantané de base de données, consultez les rubriques suivantes.

Rubriques

- Conservation des instantanés
- Considérations relatives à la copie d'instantanés partagés
- Considérations relatives à la copie d'instantanés de chiffrement
- Considérations relatives à la copie incrémentielle d'instantanés
- Considérations relatives à la copie d'instantanés entre régions
- Considérations relatives aux groupes d'options
- Considérations relatives au groupe de paramètres

Conservation des instantanés

Amazon RDS supprime les sauvegardes automatisées dans plusieurs situations :

- A la fin de leur période de conservation.
- Lorsque vous désactivez les sauvegardes automatisées pour une instance de base de données.
- Lorsque vous supprimez une instance de base de données.

Si vous souhaitez conserver une sauvegarde automatisée à plus long terme, copiez-la pour créer un instantané de base de données manuel qui sera conservé jusqu'à ce que vous le supprimiez. Des coûts de stockage Amazon RDS peuvent s'appliquer aux instantanés manuels si ces derniers dépassent votre espace de stockage par défaut.

Pour de plus amples informations sur les coûts de stockage des sauvegardes, veuillez consulter Tarification Amazon RDS.

Considérations relatives à la copie d'instantanés partagés

Vous pouvez copier des instantanés que d'autres Comptes AWS personnes vous ont partagés. Dans certains cas, vous pouvez copier un instantané chiffré qui a été partagé depuis un autre Compte AWS. Dans ces cas, vous devez avoir accès à celui AWS KMS key qui a été utilisé pour chiffrer l'instantané.

> **Note**
>
> Les frais de stockage Amazon RDS s'appliquent aux instantanés partagés que vous copiez. Amazon RDS peut associer l'ARN de l'instance de base de données source à l'instantané que vous avez copié.

Vous pouvez copier un instantané de base de données partagé Régions AWS si celui-ci n'est pas chiffré. Cependant, si l'instantané de bases de données partagé est chiffré, vous ne pouvez le copier que dans la même région.

> **Note**
>
> La copie d'instantanés incrémentiels partagés dans un même fichier Région AWS est prise en charge lorsqu'ils ne sont pas chiffrés ou chiffrés à l'aide de la même clé KMS que l'instantané complet initial. Si vous utilisez une clé KMS différente pour chiffrer les instantanés suivants lors de leur copie, ces instantanés partagés sont des instantanés complets. Pour de plus amples informations, veuillez consulter Considérations relatives à la copie incrémentielle d'instantanés.

Considérations relatives à la copie d'instantanés de chiffrement

Vous pouvez copier un instantané qui a été chiffré à l'aide d'une clé KMS. Si vous copiez un instantané chiffré, la copie de l'instantané doit également être chiffrée. Si vous copiez un instantané chiffré dans celui-ci Région AWS, vous pouvez chiffrer la copie avec la même clé KMS que l'instantané d'origine. Ou vous pouvez spécifier une clé KMS différente.

Si vous copiez un instantané chiffré entre régions, vous devez spécifier une clé KMS valide dans la Région AWS de destination. Il peut s'agir d'une clé KMS spécifique à une Région ou d'une clé multi-Régions. Pour plus d'informations sur les clés KMS multi-Régions, veuillez consulter la section Utilisation de clés multi-Régions dans AWS KMS.

L'instantané source reste chiffré pendant tout le processus de copie. Pour plus d'informations, consultez Limitations des instances de base de données chiffrées Amazon RDS.

Vous pouvez également chiffrer une copie d'un instantané non chiffré. De cette façon, vous pouvez ajouter rapidement un chiffrement à une instance de base de données non chiffrée au préalable.

Pour ce faire, créez un instantané de votre instance de base de données lorsque vous êtes prêt à le chiffrer. Vous créez ensuite une copie de cet instantané et spécifiez une clé KMS pour chiffrer cette copie d'instantané. Vous pouvez ensuite restaurer une instance de base de données chiffrée à partir de l'instantané chiffré.

> **Important**
>
> Lorsque vous utilisez des déclarations de refus explicites pour toutes les ressources (*) dans les politiques AWS KMS clés des services gérés tels qu'Amazon RDS, vous devez spécifier une condition pour autoriser le compte propriétaire de la ressource. L'opération de copie peut échouer sans cette condition, même si la règle de refus inclut des exceptions pour votre utilisateur IAM.

Pour plus d'informations sur la gestion des AWS KMS clés pour Amazon RDS, consultezAWS KMS key gestion.

Considérations relatives à la copie incrémentielle d'instantanés

Un instantané incrémentiel contient uniquement les données qui ont été modifiées après l'instantané le plus récent de la même instance de base de données. La copie d'instantanés incrémentiels est plus rapide est entraîne des coûts de stockage plus faibles que la copie d'instantanés complets.

Le caractère incrémentiel d'une copie instantanée dépend de la dernière copie d'instantané terminée et de l'instantané source. Si la dernière copie d’instantané a été supprimée, la copie suivante est une copie complète, non une copie incrémentielle. Une copie instantanée sera du même type que la capture d'écran source. Si l'instantané source est un instantané incrémentiel, la copie du cliché sera un instantané incrémentiel.

Lorsque vous copiez un instantané Comptes AWS, il s'agit d'une copie incrémentielle uniquement si toutes les conditions suivantes sont remplies :

- La copie instantanée la plus récente provient de la même instance de base de données source et existe toujours dans le compte de destination.
- Toutes les copies de l'instantané dans le compte de destination sont soit non chiffrées, soit chiffrées avec la même clé KMS.
- Si l'instance de base de données source est une instance multi-AZ, elle n'a pas basculé vers une autre AZ depuis que le dernier instantané a été créé à partir de celle-ci.

Les exemples suivants illustrent la différence entre les instantanés complets et incrémentiels. Ils s'appliquent tant aux instantanés partagés qu'aux instantanés non partagés.

Instantané	Clé de chiffrement	Complet ou incrémentiel
S1	K1	Complet
S2	K1	Incrémentiel de S1
S3	K1	Incrémentiel de S2
S4	K1	Incrémentiel de S3
Copie de S1 (S1C)	K2	Complet
Copie de S2 (S2C)	K3	Complet
Copie de S3 (S3C)	K3	Incrémentiel de S2C
Copie de S4 (S4C)	K3	Incrémentiel de S3C
Copie 2 de S4 (S4C2)	K4	Complet

> **Note**
>
> Dans ces exemples, les instantanés S2, S3 et S4 ne sont incrémentiels que si l'instantané précédent existe toujours.
> Il en va de même des copies. Les copies d'instantané S3C et S4C ne sont incrémentielles que si la copie précédente existe toujours.

Pour plus d'informations sur la copie d'instantanés incrémentiels Régions AWS, consultez. Copies complètes et incrémentielles

Considérations relatives à la copie d'instantanés entre régions

Vous pouvez copier les instantanés de base de données entre Régions AWS. Toutefois, il existe certaines contraintes et considérations en ce qui concerne la copie d'instantanés entre régions.

Demander une copie d'instantané de bases de données entre régions

Pour communiquer avec la région source afin de demander une copie d'instantané de bases de données entre régions, le demandeur (rôle IAM ou utilisateur IAM) doit avoir accès à l'instantané de bases de données source et à la région source.

Certaines conditions de la politique IAM du demandeur peuvent occasionner l'échec de la demande. Les exemples suivants supposent que vous copiez l'instantané de bases de données à partir de USA Est (Ohio) vers US East (N. Virginia). Ces exemples illustrent les conditions de la politique IAM du demandeur qui occasionnent l'échec de la demande :

- La stratégie du demandeur est assortie d'une condition pour la `aws:RequestedRegion`.

```
...
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "aws:RequestedRegion": "us-east-1"
    }
}
```

  La demande échoue car la politique interdit l'accès à la région source. Pour qu'une demande aboutisse, spécifiez les régions source et destination.

```
...
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": "*",
"Condition": {
    "StringEquals": {
        "aws:RequestedRegion": [
            "us-east-1",
            "us-east-2"
        ]
    }
}
```

- La politique du demandeur n'autorise pas l'accès à l'instantané de bases de données source.

```
...
```

```
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": "arn:aws:rds:us-east-1:123456789012:snapshot:target-snapshot"
...
```

Pour une demande réussie, spécifiez les instantanés source et cible.

```
...
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": [
    "arn:aws:rds:us-east-1:123456789012:snapshot:target-snapshot",
    "arn:aws:rds:us-east-2:123456789012:snapshot:source-snapshot"
]
...
```

- La stratégie du demandeur refuse `aws:ViaAWSService`.

```
...
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": "*",
"Condition": {
    "Bool": {"aws:ViaAWSService": "false"}
}
```

La communication avec la région source est effectuée par RDS pour le compte du demandeur. Pour que la demande soit acceptée, ne refusez pas les appels passés par AWS les services.

- La stratégie du demandeur est assortie d'une condition pour `aws:SourceVpc` ou `aws:SourceVpce`.

Ces demandes peuvent échouer car l'appel effectué par RDS à la région distante ne provient pas du point de terminaison VPC ou du VPC spécifié.

Si vous devez utiliser l'une des conditions précédentes, qui sont susceptibles d'occasionner l'échec d'une requête, vous pouvez inclure une deuxième instruction avec `aws:CalledVia` dans votre politique pour que la demande soit couronnée de succès. Par exemple, vous pouvez utiliser `aws:CalledVia` avec `aws:SourceVpce` comme indiqué ici :

```
...
```

```
"Effect": "Allow",
"Action": "rds:CopyDBSnapshot",
"Resource": "*",
"Condition": {
    "Condition" : {
        "ForAnyValue:StringEquals" : {
          "aws:SourceVpce": "vpce-1a2b3c4d"
        }
     }
},
{
    "Effect": "Allow",
    "Action": [
        "rds:CopyDBSnapshot"
    ],
    "Resource": "*",
    "Condition": {
        "ForAnyValue:StringEquals": {
            "aws:CalledVia": [
                "rds.amazonaws.com"
            ]
        }
    }
}
```

Pour de plus amples informations, veuillez consulter Policies and permissions in IAM (Stratégies et autorisations dans IAM) dans le IAM Guide de l'utilisateur.

Autorisation de la copie d'instantané

Après qu'une demande de copie d'instantané de bases de données entre régions renvoie `success`, RDS démarre la copie en arrière-plan. Une autorisation permettant à RDS d'accéder à l'instantané source est créée. Cette autorisation associe l'instantané de bases de données source à l'instantané de bases de données cible et permet à RDS de ne copier que vers l'instantané cible spécifié.

L'autorisation est vérifiée par RDS à l'aide de l'autorisation `rds:CrossRegionCommunication` dans le rôle IAM lié au service. Si la copie est autorisée, RDS communique avec la région source et réalise la copie.

RDS n'a pas accès aux instantanés de bases de données qui n'étaient pas autorisés précédemment par une demande `CopyDBSnapshot`. L'autorisation est révoquée lorsque la copie est terminée.

RDS utilise le rôle lié au service afin de vérifier l'autorisation dans la région source. Si vous supprimez le rôle lié au service pendant le processus de copie, la copie échoue.

Pour plus d'informations, consultez Utilisation des rôles liés à un service dans le IAM Guide de l'utilisateur.

Utilisation des AWS Security Token Service informations d'identification

Les jetons de session provenant du point de terminaison global AWS Security Token Service (AWS STS) ne sont valides Régions AWS que s'ils sont activés par défaut (régions commerciales). Si vous utilisez les informations d'identification issues de l'opération d'`assumeRole`API dans AWS STS, utilisez le point de terminaison régional si la région source est une région optionnelle. Sinon, la demande échoue. Cela se produit parce que vos informations d'identification doivent être valides dans les deux régions, ce qui n'est vrai pour les régions optionnelles que lorsque le point de AWS STS terminaison régional est utilisé.

Pour utiliser le point de terminaison global, assurez-vous qu'il est activé dans les opérations pour les deux régions. Définissez le point de terminaison global sur `Valid in all Régions AWS` dans les paramètres du AWS STS compte.

La même règle s'applique aux informations d'identification dans le paramètre URL pré-signé.

Pour plus d'informations, consultez la section Gestion du AWS STS dans et Région AWS dans le guide de l'utilisateur IAM.

Latence et demandes de copie multiple

En fonction du type Régions AWS concerné et de la quantité de données à copier, la réalisation d'une copie instantanée entre régions peut prendre des heures.

Dans certains cas, il peut y avoir un grand nombre de demandes de copie d'instantanés d'une région à une autre à partir d'une Région AWS source donnée. Dans de tels cas, Amazon RDS peut placer les nouvelles demandes de copie interrégionales provenant de cette source Région AWS dans une file d'attente jusqu'à ce que certaines copies en cours soient terminées. Aucune information d'avancement n'est affichée sur les demandes de copie quand elles sont en file d'attente. Les informations d'avancement sont affichées lorsque la copie commence.

Copies complètes et incrémentielles

Lorsque vous copiez un instantané vers un instantané Région AWS différent de l'instantané source, la première copie est un instantané complet, même si vous copiez un instantané incrémentiel. Une copie complète d'un instantané conserve toutes les données et métadonnées requises pour

restaurer l'instance de base de données. Après la première copie instantanée, vous pouvez copier des instantanés incrémentiels de la même instance de base de données vers la même région de destination au sein de la même instance. Compte AWS Pour de plus amples informations sur les instantanés incrémentiels, veuillez consulter Considérations relatives à la copie incrémentielle d'instantanés.

La copie incrémentielle d'instantanés Régions AWS est prise en charge à la fois pour les instantanés chiffrés et non chiffrés.

Lorsque vous copiez un instantané Régions AWS, il s'agit d'une copie incrémentielle si les conditions suivantes sont remplies :

- L'instantané a été préalablement copié dans la région de destination.
- La dernière copie d'instantané existe toujours dans la région de destination.
- Toutes les copies de l'instantané dans la région de destination sont soit non chiffrées, soit chiffrées avec la même clé KMS.

Considérations relatives aux groupes d'options

Les groupes d'options de base de données sont spécifiques à ceux dans Région AWS lesquels ils sont créés, et vous ne pouvez pas utiliser un groupe d'options de l'un Région AWS à l'autre Région AWS.

Pour les bases de données Oracle, vous pouvez utiliser l'API AWS CLI ou RDS pour copier le groupe d'options de base de données personnalisé à partir d'un instantané partagé avec votre Compte AWS. Vous pouvez copier des groupes d'options uniquement au sein d'une même Région AWS. Le groupe d'options n'est pas copié s'il a déjà été copié dans le compte de destination et qu'aucune modification n'y a été apportée depuis sa copie. Si le groupe d'options source a été copié auparavant, mais a changé depuis, RDS copie la nouvelle version dans le compte de destination. Les groupes d'options par défaut ne sont pas copiés.

Lorsque vous copiez un instantané d'une région à une autre, vous pouvez spécifier un nouveau groupe d'options pour l'instantané. Nous vous recommandons de préparer le nouveau groupe d'options avant de copier l'instantané. Dans la destination Région AWS, créez un groupe d'options avec les mêmes paramètres que l'instance de base de données d'origine. S'il en existe déjà un dans le nouveau Région AWS, vous pouvez l'utiliser.

Dans certains cas, vous pouvez copier un instantané et ne pas spécifier de nouveau groupe d'options pour l'instantané. Dans ces cas, lorsque vous restaurez l'instantané, l'instance de base de données

obtient le groupe d'options par défaut. Pour fournir à la nouvelle instance de base de données les mêmes options que la version d'origine, procédez comme suit :

1. Dans la destination Région AWS, créez un groupe d'options avec les mêmes paramètres que l'instance de base de données d'origine. S'il en existe déjà un dans le nouveau Région AWS, vous pouvez l'utiliser.
2. Après avoir restauré le snapshot dans la destination Région AWS, modifiez la nouvelle instance de base de données et ajoutez le groupe d'options nouveau ou existant de l'étape précédente.

Considérations relatives au groupe de paramètres

Lorsque vous copiez un instantané d'une région à une autre, la copie n'inclut pas le groupe de paramètres utilisé par l'instance de base de données d'origine. Lorsque vous restaurez un instantané pour créer une nouvelle instance de base de données, cette instance de base de données obtient le groupe de paramètres par défaut pour celui dans Région AWS lequel elle a été créée. Pour fournir à la nouvelle instance de base de données les mêmes paramètres que la version d'origine, procédez comme suit :

1. Dans la destination Région AWS, créez un groupe de paramètres de base de données avec les mêmes paramètres que l'instance de base de données d'origine. S'il en existe déjà un dans le nouveau Région AWS, vous pouvez l'utiliser.
2. Après avoir restauré le cliché dans la destination Région AWS, modifiez la nouvelle instance de base de données et ajoutez le groupe de paramètres nouveau ou existant de l'étape précédente.

Partage d'un instantané de base de données pour Amazon RDS

À l'aide d'AmazonRDS, vous pouvez partager un instantané de base de données manuel de différentes manières :

- Le partage manuel d'un instantané de base de données, qu'il soit chiffré ou non, permet Comptes AWS aux personnes autorisées de copier l'instantané.
- Le partage d'un instantané de base de données manuel non chiffré permet aux Comptes AWS personnes autorisées de restaurer directement une instance de base de données à partir de l'instantané au lieu d'en prendre une copie et de la restaurer à partir de celui-ci. Toutefois, vous ne pouvez pas restaurer une instance de base de données à partir d'un instantané de base de données qui est à la fois partagé et chiffré. Par contre, vous pouvez créer une copie de l'instantané de base de données et restaurer l'instance de base de données à partir de cette copie.

> **Note**
>
> Pour partager un instantané de base de données automatisé, créez un instantané de base de données manuel en copiant l'instantané automatisé, puis partagez cette copie. Ce processus s'applique également aux ressources générées par AWS Backup.

Pour plus d'informations sur la copie d'un instantané, consultez Copier un instantané de base de données pour Amazon RDS. Pour plus d'informations sur la restauration d'une instance de base de données à partir d'un instantané de base de données, consultez Restauration vers une instance de base de données.

Vous pouvez partager un instantané manuel avec un maximum de 20 autres personnes Comptes AWS.

Les restrictions suivantes s'appliquent lorsque vous partagez des instantanés manuels avec d'autres Comptes AWS personnes :

- Lorsque vous restaurez une instance de base de données à partir d'un instantané partagé à l'aide de AWS Command Line Interface (AWS CLI) ou Amazon RDSAPI, vous devez spécifier le nom de ressource Amazon (ARN) du cliché partagé comme identifiant de capture d'écran.

- Vous ne pouvez pas partager un instantané de bases de données qui utilise un groupe d'options comportant des options permanentes ou persistantes, sauf pour les instances de base de données Oracle qui possèdent l'option `Timezone` et/ou `OLS`.

  Une option permanente ne peut pas être supprimée d'un groupe d'options. Les groupes d'options avec des options persistantes ne peuvent pas être supprimés d'une instance de base de données une fois que le groupe d'options a été assigné à l'instance de base de données.

  Le tableau suivant répertorie les options permanentes et persistantes ainsi que leurs moteurs de base de données associés.

Nom d'option	Persistante	Permanent	Moteur de base de données
TDE	Oui	Non	Édition Microsoft SQL Server Enterprise
TDE	Oui	Oui	Oracle Enterprise Edition
Fuseau horaire	Oui	Oui	Oracle Enterprise Edition Oracle Standard Edition Oracle Standard Edition One Oracle Standard Edition 2

  Pour les instances de base de données Oracle, vous pouvez copier les instantanés de bases de données partagés qui possèdent l'option `Timezone` et/ou `OLS`. Pour ce faire, spécifiez un groupe d'options cibles qui inclut ces options lorsque vous copiez l'instantané de bases de données. L'OLSoption est permanente et persistante uniquement pour les instances de base de données Oracle exécutant Oracle version 12.2 ou supérieure. Pour de plus amples informations sur ces options, veuillez consulter Fuseau horaire Oracle et Oracle Label Security.

- Vous ne pouvez pas partager un instantané d'un cluster de base de données multi-AZ.

Consultez les rubriques suivantes pour plus d'informations sur le partage d'instantanés publics, le partage d'instantanés chiffrés et l'arrêt du partage d'instantanés.

Rubriques

- Partage d'instantanés publics pour Amazon RDS
- Partage d'instantanés chiffrés pour Amazon RDS
- Arrêter le partage d'instantanés pour Amazon RDS

Partage d'un instantané

Vous pouvez partager un instantané de base de données à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Console

À l'aide de la RDS console Amazon, vous pouvez partager un instantané de base de données manuel avec un maximum de 20 personnes Comptes AWS. Vous pouvez également utiliser la console pour arrêter le partage d'un instantané manuel avec un ou plusieurs comptes.

Pour partager un instantané de base de données manuel à l'aide de la RDS console Amazon

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Sélectionnez l'instantané manuel que vous voulez partager.
4. Pour Actions, choisissez Share snapshot (Partager l'instantané).
5. Choisissez l'une des options suivantes pour DB snapshot visibility (Visibilité d'instantané de base de données).

    - Si la source n'est pas chiffrée, choisissez Public pour autoriser tous les AWS comptes à restaurer une instance de base de données à partir de votre instantané de base de données manuel, ou choisissez Privé pour autoriser uniquement Comptes AWS ce que vous spécifiez pour restaurer une instance de base de données à partir de votre instantané de base de données manuel.

    > **Warning**
    >
    > Si vous définissez la visibilité des instantanés de base de données sur Public, tous Comptes AWS peuvent restaurer une instance de base de données à partir de votre

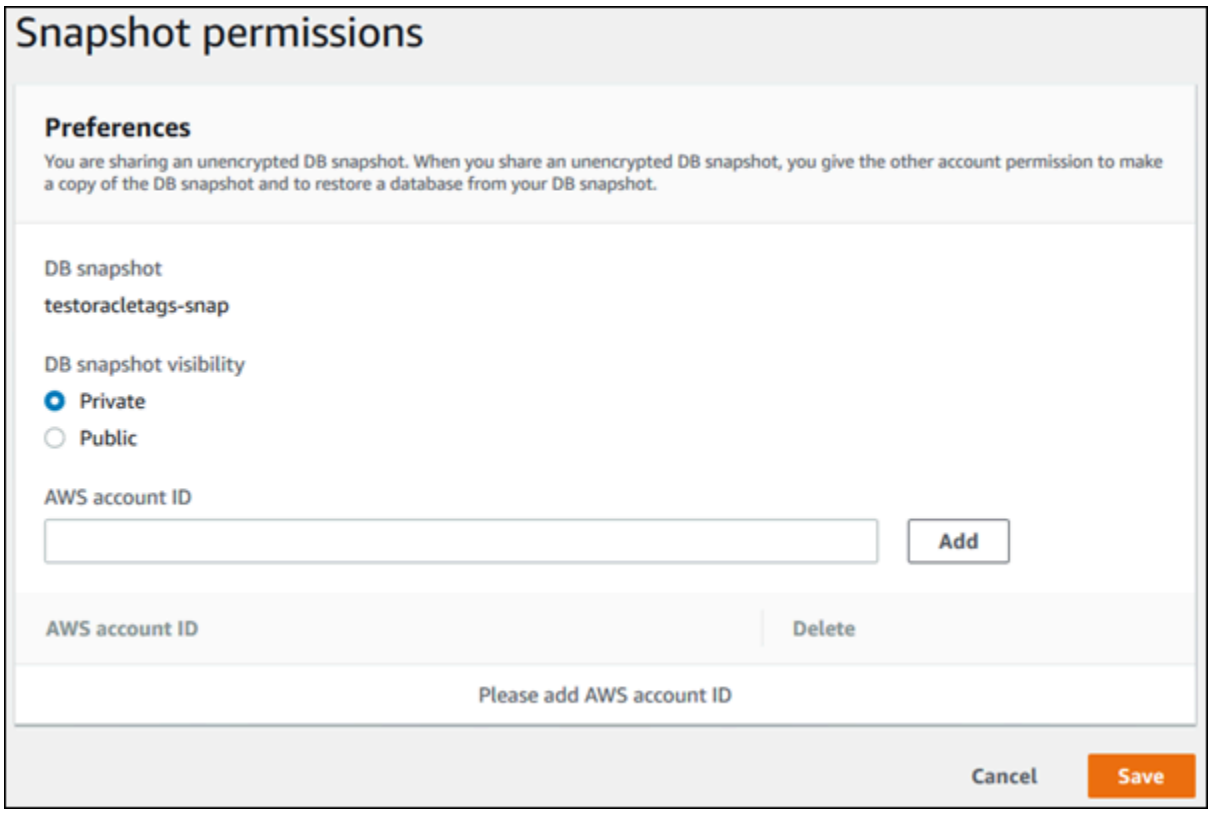
instantané de base de données manuel et avoir accès à vos données. Ne partagez aucun instantané de base de données manuel contenant des informations privées en le marquant comme Public.
Pour de plus amples informations, veuillez consulter Partage d'instantanés publics pour Amazon RDS.

- Si la source est chiffrée, la Visibilité d'instantané de base de données est définie sur Privé, car les instantanés chiffrés ne peuvent pas être partagés s'ils sont marqués comme étant publics.

> **Note**
>
> Les instantanés chiffrés avec la valeur par défaut ne AWS KMS key peuvent pas être partagés. Pour plus d'informations sur la manière de contourner ce problème, consultezPartage d'instantanés chiffrés pour Amazon RDS.

6. Pour ID de AWS compte, entrez l' Compte AWS identifiant du compte que vous souhaitez autoriser à restaurer une instance de base de données à partir de votre instantané manuel, puis choisissez Ajouter. Répétez l'opération pour inclure des Compte AWS identifiants supplémentaires, jusqu'à 20 Comptes AWS.

   Si vous commettez une erreur lors de l'ajout d'un Compte AWS identifiant à la liste des comptes autorisés, vous pouvez le supprimer de la liste en choisissant Supprimer à droite de l' Compte AWS identifiant incorrect.



7. Après avoir ajouté des identifiants pour tous les éléments Comptes AWS que vous souhaitez autoriser à restaurer l'instantané manuel, choisissez Enregistrer pour enregistrer vos modifications.

AWS CLI

Pour partager un instantané de base de données, utilisez la commande `aws rds modify-db-snapshot-attribute`. Utilisez le `--values-to-add` paramètre pour ajouter une liste des IDs personnes Comptes AWS autorisées à restaurer l'instantané manuel.

Example de partager un instantané avec un seul compte

L'exemple suivant permet `123456789012` à Compte AWS l'identifiant de restaurer le snapshot de base de données nommé`db7-snapshot`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot-attribute \
--db-snapshot-identifier db7-snapshot \
--attribute-name restore \
--values-to-add 123456789012
```

Dans Windows:

```
aws rds modify-db-snapshot-attribute ^
--db-snapshot-identifier db7-snapshot ^
--attribute-name restore ^
--values-to-add 123456789012
```

Example de partager un instantané avec plusieurs comptes

L'exemple suivant active deux Compte AWS identifiants `111122223333` et `444455556666` permet de restaurer le snapshot de base de données nommé`manual-snapshot1`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot-attribute \
--db-snapshot-identifier manual-snapshot1 \
--attribute-name restore \
--values-to-add {"111122223333","444455556666"}
```

Dans Windows:

```
aws rds modify-db-snapshot-attribute ^
--db-snapshot-identifier manual-snapshot1 ^
--attribute-name restore ^
--values-to-add "[\"111122223333\",\"444455556666\"]"
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Pour répertorier les Comptes AWS personnes autorisées à restaurer un instantané, utilisez la `describe-db-snapshot-attributes` AWS CLI commande.

RDS API

Vous pouvez également partager un instantané de base de données manuel avec d'autres personnes en Comptes AWS utilisant Amazon RDSAPI. Pour ce faire, appelez l'opération

ModifyDBSnapshotAttribute. Spécifiez `restore` pour `AttributeName` et utilisez le `ValuesToAdd` paramètre pour ajouter une liste des IDs pour Comptes AWS qui sont autorisés à restaurer le cliché manuel.

Pour rendre un instantané manuel public et restaurable par tous Comptes AWS, utilisez la valeur`all`. Veillez toutefois à ne pas ajouter de `all` valeur aux instantanés manuels contenant des informations privées que vous ne souhaitez pas rendre accessibles à tous Comptes AWS. De même, ne spécifiez pas la valeur `all` pour les instantanés chiffrés, car il est impossible de rendre tous ces instantanés publics.

Pour répertorier toutes les Comptes AWS personnes autorisées à restaurer un instantané, utilisez l'DescribeDBSnapshotAttributesAPIopération.

Partage d'instantanés publics pour Amazon RDS

Vous pouvez partager un instantané manuel non chiffré en tant que public, ce qui le rend accessible à tous Comptes AWS. Lors du partage d'un instantané marqué comme public, assurez-vous de n'inclure aucune information privée dans l'instantané public.

Lorsqu'un instantané est partagé publiquement, il donne à tous les Comptes AWS droits nécessaires à la fois pour le copier et pour créer des instances de base de données à partir de celui-ci.

Le stockage de sauvegarde des snapshots publics appartenant à d'autres comptes n'est pas facturé. Seuls les instantanés que vous possédez vous sont facturés.

Si vous copiez un instantané public, vous êtes propriétaire de la copie. Le stockage de sauvegarde de votre copie d'instantané vous est facturé. Si vous créez une instance de base de données à partir d'un instantané public, cette instance de base de données vous est facturée. Pour obtenir RDS des informations sur les tarifs Amazon, consultez la page RDS produit Amazon.

Vous ne pouvez supprimer que les instantanés publics que vous possédez. Pour supprimer un instantané partagé ou public, assurez-vous de vous connecter au Compte AWS propriétaire de l'instantané.

Afficher des instantanés publics appartenant à d'autres Comptes AWS

Vous pouvez consulter les instantanés publics détenus par d'autres comptes dans une AWS région donnée dans l'onglet Public de la page Snapshots de la console AmazonRDS. Vos instantanés (ceux appartenant à votre compte) n'apparaissent pas dans cet onglet.

Pour afficher des instantanés publics

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'onglet Public.

   Les instantanés publics s'affichent. Vous pouvez voir quel compte possède un instantané public dans la colonne Owner (Propriétaire).

   > **Note**
   >
   > Pour voir cette colonne, vous devrez peut-être modifier les préférences de la page en sélectionnant l'icône en forme d'engrenage en haut à droite de la liste Public snapshots (Instantanés publics).

Affichage de vos propres Instantanés publics

Vous pouvez utiliser la AWS CLI commande suivante (Unix uniquement) pour afficher les instantanés publics que vous possédez Compte AWS dans une région donnée AWS .

```
aws rds describe-db-snapshots --snapshot-type public --include-public |
 grep account_number
```

La sortie renvoyée est semblable à l'exemple suivant si vous avez des instantanés publics.

```
"DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mysnapshot1",
"DBSnapshotArn": "arn:aws:rds:us-east-1:123456789012:snapshot:mysnapshot2",
```

> **Note**
>
> Vous pouvez voir des entrées en double pour `DBSnapshotIdentifier` ou `SourceDBSnapshotIdentifier`.

Partage d'instantanés publics à partir de versions obsolètes du moteur de base de données

La restauration ou la copie d'instantanés publics à partir de versions obsolètes du moteur de base de données n'est pas prise en charge.

Les moteurs de SQL base de données RDS pour Oracle et RDS pour Postgre prennent directement en charge la mise à niveau des versions du moteur de capture d'écran de base de données. Vous pouvez mettre à jour vos instantanés, puis les partager à nouveau publiquement. Pour plus d'informations, consultez les ressources suivantes :

- Mise à niveau d'un instantané de base de données Oracle
- Mise à niveau d'une version du moteur de snapshots SQL de base de données Postgre

Pour les autres moteurs de base de données, effectuez les étapes suivantes pour rendre votre instantané public non pris en charge existant disponible pour restauration ou copie :

1. Marquez l'instantané comme privé.
2. Restaurez l'instantané.
3. Mettez à niveau l'instance de base de données restaurée vers une version du moteur prise en charge.
4. Créez un nouvel instantané.
5. Partagez à nouveau l'instantané publiquement.

Partage d'instantanés chiffrés pour Amazon RDS

Vous pouvez partager des instantanés de bases de données qui ont été chiffrés « au repos » en utilisant l'algorithme de chiffrement AES-256, comme décrit dans Chiffrement des ressources Amazon RDS.

Les restrictions suivantes s'appliquent au partage d'instantanés chiffrés :

- Vous ne pouvez pas partager des instantanés chiffrés marqués comme publics.
- Vous ne pouvez pas partager des instantanés Oracle ou Microsoft SQL Server qui sont chiffrés à l'aide de Transparent Data Encryption (TDE).
- Vous ne pouvez pas partager un instantané chiffré à l'aide de la clé KMS par défaut de celle Compte AWS qui a partagé l'instantané.

Pour plus d'informations sur la gestion des AWS KMS clés pour Amazon RDS, consultezAWS KMS key gestion.

Pour contourner le problème de clé KMS par défaut, effectuez les tâches suivantes :

1. Créez une clé gérée par le client et donnez-lui accès.
2. Copiez et partagez l'instantané depuis le compte source.
3. Copiez l'instantané partagé dans le compte cible.

Créez une clé gérée par le client et donnez-lui accès

Vous devez d'abord créer une clé KMS personnalisée Région AWS identique à l'instantané de base de données chiffré. Lors de la création de la clé gérée par le client, vous permettez à un autre utilisateur d'y accéder Compte AWS.

> **Note**
>
> Vous pouvez également utiliser une clé KMS provenant d'un autre AWS compte lorsque la politique des clés autorise l'accès aux comptes source et cible.

Pour créer une clé gérée par le client et y donner accès

1. Connectez-vous au AWS Management Console depuis la source Compte AWS.
2. Ouvrez la AWS KMS console à l'adresse https://console.aws.amazon.com/kms.
3. Pour modifier le Région AWS, utilisez le sélecteur de région dans le coin supérieur droit de la page.
4. Dans le volet de navigation, sélectionnez Clés gérées par le client.
5. Choisissez Create key.
6. Sur la page Configurer la clé :
    a. Pour Type de clé, sélectionnez Symétrique.
    b. Pour Utilisation des clés, sélectionnez Chiffrer et déchiffrer.
    c. (Facultatif) Développez Options avancées.

d. Pour Origine du matériau clé, sélectionnez KMS.
e. Pour Régionalité, sélectionnez la clé à région unique.
f. Choisissez Next (Suivant).

7. Sur la page Ajouter des étiquettes :

a. Pour Alias. Entrez un nom d'affichage pour votre clé KMS, par exemple**`share-snapshot`**.
b. (Facultatif) Entrez une description pour votre clé KMS.
c. (Facultatif) Ajoutez des balises à votre clé KMS.
d. Choisissez Next (Suivant).

8. Sur la page Définir des autorisations d'administration de clé, choisissez Suivant.
9. Sur la page Définir les autorisations d'utilisation des clés :

a. Pour Autre Comptes AWS, choisissez Ajouter un autre Compte AWS.
b. Entrez l'identifiant Compte AWS auquel vous souhaitez donner accès.

   Vous pouvez donner accès à plusieurs Comptes AWS.
c. Choisissez Next (Suivant).

10. Vérifiez votre clé KMS, puis choisissez Terminer.

Copiez et partagez l'instantané depuis le compte source

Ensuite, vous copiez l'instantané de base de données source vers un nouvel instantané à l'aide de la clé gérée par le client. Ensuite, vous le partagez avec la cible Compte AWS.

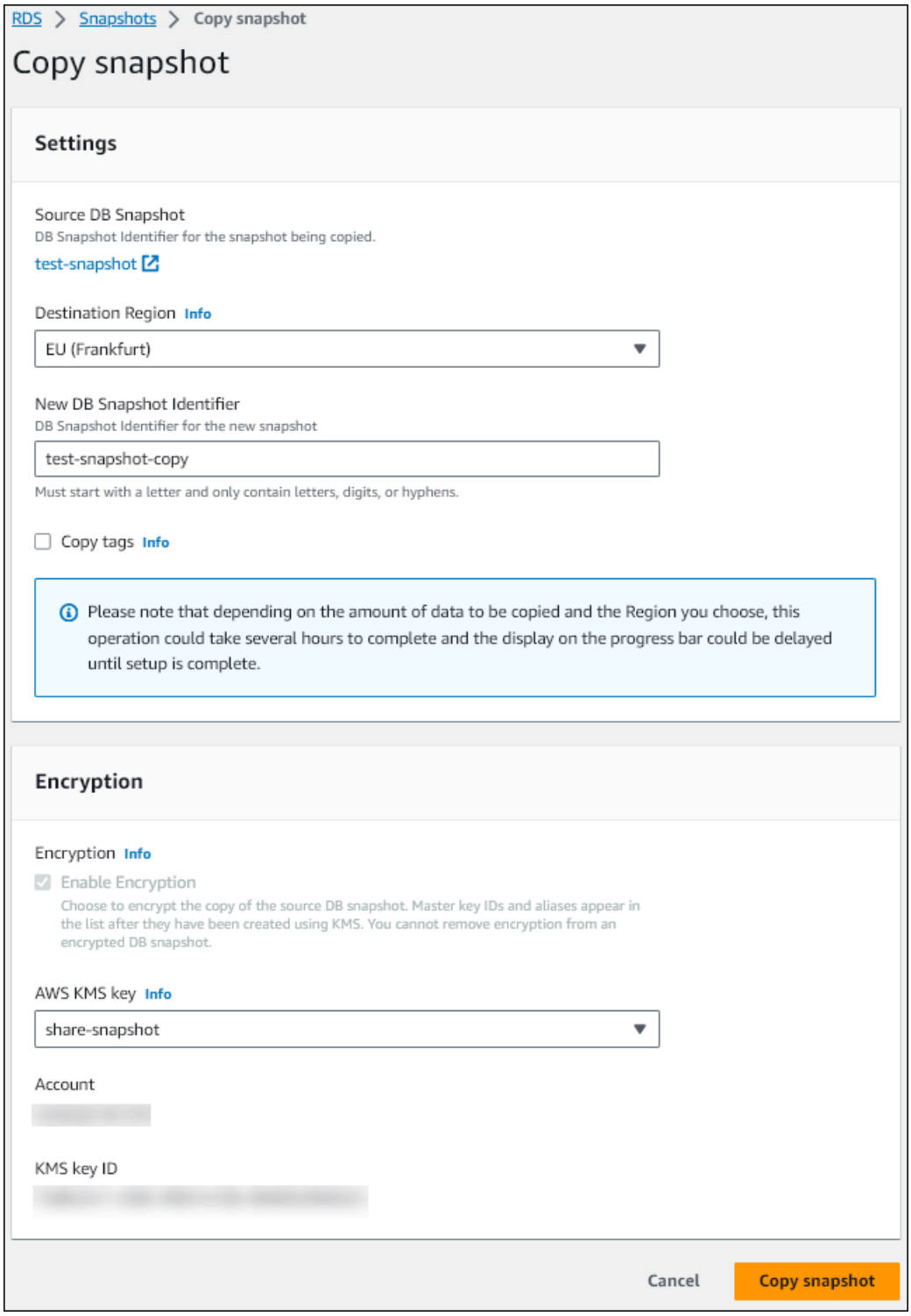
Pour copier et partager l'instantané

1. Connectez-vous au AWS Management Console depuis la source Compte AWS.
2. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/
3. Dans le panneau de navigation, choisissez Snapshots.
4. Sélectionnez le snapshot de base de données que vous souhaitez copier.
5. Sous Actions, choisissez Copier un instantané.
6. Sur la page Copier un instantané :

a. Pour Région de destination, choisissez l' Région AWS endroit où vous avez créé la clé gérée par le client lors de la procédure précédente.

b. Entrez le nom de la copie instantanée de base de données dans New DB Snapshot Identifier.

c. Pour AWS KMS key, choisissez la clé gérée par le client que vous avez créée.



d. Choisissez Copy snapshot (Copier un instantané).

7. Lorsque la copie instantanée est disponible, sélectionnez-la.
8. Pour Actions, choisissez Share snapshot (Partager l'instantané).
9. Sur la page des autorisations relatives aux instantanés :

    a. Entrez l'Compte AWS identifiant avec lequel vous partagez la copie instantanée, puis choisissez Ajouter.

    b. Choisissez Enregistrer.

    L'instantané est partagé.

Copiez l'instantané partagé dans le compte cible

Vous pouvez maintenant copier le cliché partagé dans la cible Compte AWS.

Pour copier le cliché partagé

1. Connectez-vous au AWS Management Console depuis la cible Compte AWS.
2. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/
3. Dans le panneau de navigation, choisissez Snapshots.
4. Choisissez l'onglet Partagé avec moi.
5. Sélectionnez l'instantané partagé.
6. Sous Actions, choisissez Copier un instantané.
7. Choisissez vos paramètres pour copier l'instantané comme dans la procédure précédente, mais utilisez un AWS KMS key paramètre appartenant au compte cible.

    Choisissez Copy snapshot (Copier un instantané).

Arrêter le partage d'instantanés pour Amazon RDS

Pour arrêter de partager un instantané de base de données, vous supprimez l'autorisation de la cible Compte AWS.

Console

Pour arrêter de partager un instantané de base de données manuel avec un Compte AWS

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Sélectionnez l'instantané manuel que vous voulez cesser de partager.
4. Choisissez Actions, puis Share snapshot (Partager l'instantané).
5. Pour supprimer l'autorisation pour un Compte AWS, choisissez Supprimer comme identifiant de AWS compte pour ce compte dans la liste des comptes autorisés.
6. Choisissez Save pour enregistrer les changements.

CLI

Pour supprimer un Compte AWS identifiant de la liste, utilisez le `--values-to-remove` paramètre.

Example de l'arrêt du partage d'instantanés

L'exemple suivant empêche l' Compte AWS ID 444455556666 de restaurer le snapshot.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot-attribute \
--db-snapshot-identifier manual-snapshot1 \
--attribute-name restore \
--values-to-remove 444455556666
```

Dans Windows:

```
aws rds modify-db-snapshot-attribute ^
--db-snapshot-identifier manual-snapshot1 ^
--attribute-name restore ^
--values-to-remove 444455556666
```

RDS API

Pour supprimer l'autorisation de partage pour un Compte AWS, utilisez l'`ModifyDBSnapshotAttribute`opération avec `AttributeName` set to `restore` et le

`ValuesToRemove` paramètre. Pour marquer un instantané manuel comme privé, supprimez la valeur `all` de la liste des valeurs pour l'attribut `restore`.

Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS

Vous pouvez exporter des données d'instantanés de bases de données vers un compartiment Amazon S3. Le processus d'exportation s'exécute en arrière-plan et n'affecte pas les performances de votre base de données active.

Lorsque vous exportez un instantané de base de données, Amazon RDS extrait les données de l'instantané et les stocke dans un compartiment Amazon S3. Les données sont stockées dans un format Apache Parquet qui est compressé et cohérent.

Vous pouvez exporter tous les types de instantanés de base de données, y compris les instantanés manuels, les instantanés du système automatisés et les instantanés créés par le service. AWS Backup Par défaut, toutes les données de l'instantané sont exportées. Toutefois, vous pouvez choisir d'exporter des ensembles spécifiques de bases de données, de schémas ou de tables.

Une fois les données exportées, vous pouvez les analyser directement via des outils tels que Amazon Athena ou Amazon Redshift Spectrum. Pour plus d'informations sur l'utilisation d'Athena pour lire les données de Parquet, consultez Parquet SerDe dans le guide de l'utilisateur d'Amazon Athena. Pour plus d'informations sur l'utilisation de Redshift Spectrum pour lire des données Parquet, consultez COPY depuis les formats de données en colonnes dans le Guide du développeur de base de données Amazon Redshift.

> **Warning**
>
> Vous ne pouvez pas restaurer les données de capture exportées de S3 vers une nouvelle instance de base de données ou importer des données de capture d'écran de S3 dans une instance de base de données existante. Toutefois, vous pouvez traiter les données à l'aide d'Amazon Athena ou de Redshift Spectrum à des fins d'analyse. En outre, vous pouvez transformer AWS Glue les données, puis les importer dans Amazon RDS à l'aide d'outils tels que AWS DMS des scripts personnalisés.

Pour plus d'informations sur l'exportation de snapshots de base de données vers Amazon S3, consultez les rubriques suivantes.

Rubriques

- Surveillance des exportations de snapshots pour Amazon RDS

- Annulation d'une tâche d'exportation d'instantanés pour Amazon RDS
- Messages d'échec pour les tâches d'exportation Amazon S3 pour Amazon RDS
- Résolution RDS des erreurs liées aux SQL autorisations Postgre
- Conventions de dénomination des fichiers pour les exportations vers Amazon S3 pour Amazon RDS
- Conversion de données lors de l'exportation vers un compartiment Amazon S3 pour Amazon RDS

Présentation de l'exportation des données d'instantané

Vous utilisez le processus suivant pour exporter des données d'instantané de base de données vers un compartiment Amazon S3. Pour plus de détails, consultez les sections suivantes.

1. Identifiez l'instantané à exporter.

   Utilisez un instantané automatique ou manuel existant, ou créez un instantané manuel d'une instance de base de données ou d'un cluster de base de données multi-AZ.
2. Configurez l'accès au compartiment Amazon S3.

   Un compartiment est un conteneur d'objets ou de fichiers Amazon S3. Pour fournir les informations permettant d'accéder à un compartiment, procédez comme suit :

   a. Identifiez le compartiment S3 vers lequel l'instantané doit être exporté. Le compartiment S3 doit se trouver dans la même AWS région que le snapshot. Pour de plus amples informations, veuillez consulter Identification du compartiment Amazon S3 pour l'exportation.

   b. Créez un rôle AWS Identity and Access Management (IAM) qui accorde à la tâche d'exportation de snapshots l'accès au compartiment S3. Pour de plus amples informations, veuillez consulter Fournir l'accès à un compartiment Amazon S3 à l'aide d'un rôle IAM.
3. Créez un chiffrement symétrique AWS KMS key pour le chiffrement côté serveur. La clé KMS est utilisée par la tâche d'exportation de snapshots pour configurer le chiffrement AWS KMS côté serveur lors de l'écriture des données d'exportation dans S3.

   La politique de clés KMS doit inclure à la fois les autorisations `kms:CreateGrant` et `kms:DescribeKey`. Pour plus d'informations sur l'utilisation des clés KMS dans Amazon RDS, consultez AWS KMS key gestion.

   Si votre politique de clé KMS contient une déclaration de refus, assurez-vous d'exclure explicitement le principal du AWS service`export.rds.amazonaws.com`.

Vous pouvez utiliser une clé KMS dans votre AWS compte, ou vous pouvez utiliser une clé KMS entre comptes. Pour de plus amples informations, veuillez consulter Utilisation d'un compte croisé AWS KMS key pour chiffrer les exportations Amazon S3.

4. Exportez l'instantané vers Amazon S3 à l'aide de la console ou de la commande de CLI `start-export-task`. Pour de plus amples informations, veuillez consulter Exportation d'un instantané de base de données vers un compartiment Amazon S3.
5. Pour accéder aux données exportées dans le compartiment Amazon S3, consultez Chargement, téléchargement et gestion d'objets dans le Guide de l'utilisateur Amazon Simple Storage Service.

Configuration de l'accès à un compartiment Amazon S3

Pour exporter des données d'instantané de base de données vers un fichier Amazon S3, vous accordez d'abord à l'instantané l'autorisation d'accéder au compartiment Amazon S3. Vous créez ensuite un rôle IAM pour autoriser le service Amazon RDS à écrire dans le compartiment Amazon S3.

Rubriques

- Identification du compartiment Amazon S3 pour l'exportation
- Fournir l'accès à un compartiment Amazon S3 à l'aide d'un rôle IAM
- Utilisation d'un compartiment Amazon S3 entre comptes
- Utilisation d'un compte croisé AWS KMS key pour chiffrer les exportations Amazon S3

Identification du compartiment Amazon S3 pour l'exportation

Identifiez le compartiment Amazon S3 vers lequel exporter l'instantané de base de données. Utilisez un compartiment S3 existant ou créez un nouveau compartiment S3.

> **Note**
>
> Le compartiment S3 vers lequel effectuer l'exportation doit se trouver dans la même AWS région que le snapshot.

Pour plus d'informations sur l'utilisation des Amazon S3 compartiments, veuillez consulter les points suivants dans le Guide de l'utilisateur Amazon Simple Storage Service :

- Comment afficher les propriétés d'un compartiment S3 ?
- Comment activer le chiffrement par défaut pour un compartiment Amazon S3 ?
- Comment créer un compartiment S3 ?

Fournir l'accès à un compartiment Amazon S3 à l'aide d'un rôle IAM

Avant d'exporter les données d'instantané de base de données vers Amazon S3, vous devez accorder aux tâches d'exportation d'instantané une autorisation d'accès en écriture au compartiment Amazon S3.

Pour accorder cette autorisation, créez une politique IAM qui donne accès au compartiment, puis créez un rôle IAM et attachez la politique au rôle. Vous affectez ultérieurement le rôle IAM à votre tâche d'exportation d'instantané.

Pour plus d'informations sur les autres outils de gestion d'accès Amazon S3, consultez la section Contrôle d'accès dans Amazon S3 dans le guide de l'utilisateur d'Amazon S3.

> **Important**
>
> Si vous prévoyez d'utiliser le AWS Management Console pour exporter votre instantané, vous pouvez choisir de créer la politique IAM et le rôle automatiquement lorsque vous exportez le cliché. Pour obtenir des instructions, veuillez consulter Exportation d'un instantané de base de données vers un compartiment Amazon S3.

Pour accorder aux tâches d'instantané de base de données l'accès à Amazon S3

1. Créez une politique IAM. Cette politique fournit les autorisations d'accès au compartiment et aux objets qui permettent à votre tâche d'exportation d'instantané d'accéder à Amazon S3.

   Dans la politique, incluez les actions obligatoires suivantes pour permettre le transfert de fichiers depuis Amazon RDS vers un compartiment S3.

   - `s3:PutObject*`
   - `s3:GetObject*`
   - `s3:ListBucket`
   - `s3:DeleteObject*`
   - `s3:GetBucketLocation`

Dans la politique, incluez les ressources suivantes pour identifier le compartiment S3 et les objets qu'il contient. La liste de ressources suivante indique le format Amazon Resource Name (ARN) pour l'accès à Amazon S3.

- `arn:aws:s3:::amzn-s3-demo-bucket`
- `arn:aws:s3:::amzn-s3-demo-bucket/*`

Pour plus d'informations sur la création d'une politique IAM pour Amazon RDS, veuillez consulter Création et utilisation d'une politique IAM pour l'accès à une base de données IAM. Consultez également Didacticiel : création et attachement de votre première politique gérée par le client dans le Guide de l'utilisateur IAM.

La AWS CLI commande suivante crée une politique IAM nommée `ExportPolicy` avec ces options. Elle accorde un accès à un compartiment nommé *amzn-s3-demo-bucket*.

> **Note**
>
> Après avoir créé la politique, notez son ARN. Vous en aurez besoin par la suite pour attacher la politique à un rôle IAM.

```
aws iam create-policy  --policy-name ExportPolicy --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ExportPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
```

```
        }
    ]
}'
```

2. Créez un rôle IAM, afin qu'Amazon RDS puisse endosser ce rôle IAM en votre nom pour accéder à vos compartiments Amazon S3. Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

   L'exemple suivant montre comment utiliser la AWS CLI commande pour créer un rôle nommé`rds-s3-export-role`.

```
aws iam create-role  --role-name rds-s3-export-role  --assume-role-policy-document
 '{
    "Version": "2012-10-17",
    "Statement": [
      {
        "Effect": "Allow",
        "Principal": {
           "Service": "export.rds.amazonaws.com"
         },
        "Action": "sts:AssumeRole"
      }
    ]
  }'
```

3. Attachez la politique IAM que vous avez créée au rôle IAM que vous venez de créer.

   La AWS CLI commande suivante associe la politique créée précédemment au rôle nommé`rds-s3-export-role`. Remplacez *`your-policy-arn`* par l'ARN de stratégie que vous avez noté lors d'une étape précédente.

```
aws iam attach-role-policy  --policy-arn your-policy-arn  --role-name rds-s3-
export-role
```

Utilisation d'un compartiment Amazon S3 entre comptes

Vous pouvez utiliser des compartiments Amazon S3 sur plusieurs AWS comptes. Pour utiliser un compartiment entre comptes, ajoutez une politique de compartiment afin d'autoriser l'accès au rôle IAM que vous utilisez pour les exportations S3. Pour plus d'informations, consultez Exemple 2 :

propriétaire d'un compartiment accordant à ses utilisateurs des autorisations entre comptes sur un compartiment.

Attachez une politique de compartiment à votre compartiment, comme illustré dans l'exemple suivant.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/Admin"
            },
            "Action": [
                "s3:PutObject*",
                "s3:ListBucket",
                "s3:GetObject*",
                "s3:DeleteObject*",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-destination-bucket",
                "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
            ]
        }
    ]
}
```

Utilisation d'un compte croisé AWS KMS key pour chiffrer les exportations Amazon S3

Vous pouvez utiliser un compte croisé AWS KMS key pour chiffrer les exportations Amazon S3. Tout d'abord, vous ajoutez une politique de clé au compte local, puis vous ajoutez des politiques IAM au compte externe. Pour plus d'informations, consultez la section Autorisation des utilisateurs d'autres comptes à utiliser une clé KMS.

Pour utiliser une clé KMS entre comptes

1. Ajoutez une politique de clé au compte local.

L'exemple suivant accorde `ExampleRole` et `ExampleUser` dans les autorisations 444455556666 du compte externe dans le compte local 123456789012.

```
{
    "Sid": "Allow an external account to use this KMS key",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::444455556666:role/ExampleRole",
            "arn:aws:iam::444455556666:user/ExampleUser"
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:RetireGrant"
    ],
    "Resource": "*"
}
```

2. Ajoutez des politiques IAM au compte externe.

   L'exemple de stratégie IAM suivant autorise le principal à utiliser la clé KMS dans le compte 123456789012 pour les opérations cryptographiques. Pour accorder cette autorisation aux `ExampleRole` et `ExampleUser` du compte 444455556666, attachez-leur la politique dans ce compte.

```
{
    "Sid": "Allow use of KMS key in account 123456789012",
    "Effect": "Allow",
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:CreateGrant",
        "kms:DescribeKey",
        "kms:RetireGrant"
```

```
    ],
    "Resource": "arn:aws:kms:us-
west-2:123456789012:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

Exportation d'un instantané de base de données vers un compartiment Amazon S3

Vous pouvez avoir jusqu'à cinq tâches simultanées d'exportation de snapshots de base de données en cours par Compte AWS.

> **Note**
>
> L'exportation d'instantanés RDS peut prendre un certain temps en fonction du type et de la taille de votre base de données. La tâche d'exportation commence par restaurer et mettre à l'échelle l'ensemble de la base de données avant d'extraire les données vers Amazon S3. La progression de la tâche au cours de cette phase s'affiche sous l'intitulé Starting. Lorsque la tâche passe à l'exportation de données vers S3, la progression affiche l'intitulé En cours. La durée nécessaire à l'exportation dépend des données stockées dans la base de données. Par exemple, l'exportation des tables comportant des colonnes numériques d'index ou de clé primaire bien distribuées est la plus rapide. L'opération prend plus de temps pour les tables qui ne contiennent pas de colonne adaptée au partitionnement et les tables avec un seul index sur une colonne basée sur une chaîne. Ce délai d'exportation est plus long car l'exportation utilise un processus à thread unique plus lent.

Vous pouvez exporter un instantané de base de données vers Amazon S3 à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Pour exporter un instantané de base de données vers un compartiment Amazon S3 multi-comptes, utilisez l'API AWS CLI ou l'API RDS.

Si vous utilisez une fonction Lambda pour exporter un instantané, ajoutez l'action `kms:DescribeKey` à la stratégie de fonction Lambda. Pour plus d'informations, consultez Autorisations AWS Lambda.

console

L'option de console Exporter vers Amazon S3 s'affiche uniquement pour les instantanés pouvant être exportés vers Amazon S3. Un instantané peut ne pas être disponible pour l'exportation pour les raisons suivantes :

- Le moteur de base de données n'est pas pris en charge pour l'exportation S3.
- La version du moteur de base de données n'est pas prise en charge pour l'exportation S3.
- L'exportation S3 n'est pas prise en charge dans la AWS région où l'instantané a été créé.

Pour exporter un instantané de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Dans les onglets, choisissez le type d'instantané que vous souhaitez exporter.
4. Dans la liste des instantanés, choisissez celui que vous souhaitez exporter.
5. Pour actions, choisissez Export to Amazon S3 (Exporter vers Amazon S3).

   La fenêtre Export to Amazon S3 (Exporter vers Amazon S3) apparaît.
6. Dans Export Identifier (Identifiant d'exportation), entrez un nom pour identifier la tâche d'exportation. Cette valeur est également utilisée pour le nom du fichier créé dans le compartiment S3.
7. Choisissez les données à exporter :

   - Choisissez All (Tout) pour exporter toutes les données de l'instantané.
   - Choisissez Partial (Partiel) pour exporter des parties spécifiques de l'instantané. Pour identifier les parties de l'instantané à exporter, entrez un(e) ou plusieurs bases de données, schémas ou tables pour Identifiers (Identifiants), séparés par des espaces.

     Utilisez le format suivant :

     ```
     database[.schema][.table] database2[.schema2][.table2] ... databasen[.schemaN]
     [.tablen]
     ```

     Exemples :

```
mydatabase mydatabase2.myschema1 mydatabase2.myschema2.mytable1
 mydatabase2.myschema2.mytable2
```

8. Pour S3 bucket (Compartiment S3), choisissez le compartiment vers lequel exporter.

   Pour affecter les données exportées à un chemin d'accès de dossier dans le compartiment S3, entrez le chemin d'accès facultatif pour S3 prefix (Préfixe S3).

9. Pour Rôle IAM, choisissez un rôle qui vous accorde un accès en écriture au compartiment S3 choisi, ou créez un nouveau rôle.

   - Si vous avez créé un rôle en suivant les étapes décrites dans Fournir l'accès à un compartiment Amazon S3 à l'aide d'un rôle IAM, choisissez ce rôle.
   - Si vous n'avez pas créé un rôle qui vous accorde un accès en écriture au compartiment S3 que vous avez choisi, choisissez Create a new role (Créer un nouveau rôle) pour créer le rôle automatiquement. Ensuite, saisissez un nom pour le rôle dans Nom du rôle IAM.

10. Pour AWS KMS key, entrez l'ARN de la clé à utiliser pour chiffrer les données exportées.
11. Choisissez Export to Amazon S3 (Exporter vers Amazon S3).

AWS CLI

Pour exporter un instantané de base de données vers Amazon S3 à l'aide de AWS CLI, utilisez la start-export-taskcommande avec les options requises suivantes :

- `--export-task-identifier`
- `--source-arn`
- `--s3-bucket-name`
- `--iam-role-arn`
- `--kms-key-id`

Dans les exemples suivants, la tâche d'exportation d'instantanés est nommée*`my-snapshot-export`*. Elle exporte un instantané vers un compartiment S3 nommé*`amzn-s3-demo-bucket`*.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds start-export-task \
    --export-task-identifier my-snapshot-export \
    --source-arn arn:aws:rds:AWS_Region:123456789012:snapshot:snapshot-name \
    --s3-bucket-name amzn-s3-demo-bucket \
    --iam-role-arn iam-role \
    --kms-key-id my-key
```

Dans Windows :

```
aws rds start-export-task ^
    --export-task-identifier my-snapshot-export ^
    --source-arn arn:aws:rds:AWS_Region:123456789012:snapshot:snapshot-name ^
    --s3-bucket-name amzn-s3-demo-bucket ^
    --iam-role-arn iam-role ^
    --kms-key-id my-key
```

Vous trouverez ci-après un exemple de sortie.

```
{
    "Status": "STARTING",
    "IamRoleArn": "iam-role",
    "ExportTime": "2019-08-12T01:23:53.109Z",
    "S3Bucket": "my-export-bucket",
    "PercentProgress": 0,
    "KmsKeyId": "my-key",
    "ExportTaskIdentifier": "my-snapshot-export",
    "TotalExtractedDataInGB": 0,
    "TaskStartTime": "2019-11-13T19:46:00.173Z",
    "SourceArn": "arn:aws:rds:AWS_Region:123456789012:snapshot:snapshot-name"
}
```

Pour fournir un chemin de dossier dans le compartiment S3 pour l'exportation des instantanés, incluez l'`--s3-prefix`option dans la start-export-taskcommande.

API RDS

Pour exporter un instantané de base de données vers Amazon S3 à l'aide de l'API Amazon RDS, utilisez l'StartExportTaskopération avec les paramètres obligatoires suivants :

- `ExportTaskIdentifier`
- `SourceArn`

- `S3BucketName`
- `IamRoleArn`
- `KmsKeyId`

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions avec l'exportation d'instantanés vers S3, consultez Régions et moteurs de base de données pris en charge pour l'exportation d'instantanés vers S3 dans Amazon RDS.

Limites

L'exportation de données d'instantané de bases de données vers Amazon S3 présente les limites suivantes :

- Vous ne pouvez pas exécuter simultanément plusieurs tâches d'exportation pour le même instantané de base de données. Cette règle s'applique aux exportations complètes et partielles.
- L'exportation d'instantanés à partir de bases de données utilisant le stockage magnétique n'est pas prise en charge.
- Les exportations vers S3 ne prennent pas en charge les préfixes S3 contenant deux points (:).
- Les caractères suivants du chemin d'accès au fichier S3 sont convertis en traits de soulignement (_) lors de l'exportation :

```
\ ` " (space)
```

- Si une base de données, un schéma ou une table comporte des caractères autres que les suivants, l'exportation partielle n'est pas prise en charge. Toutefois, vous pouvez exporter l'intégralité de l'instantané de base de données.
  - Lettres latines (A–Z)
  - Chiffres (0–9)
  - Symbole dollar ($)
  - Trait de soulignement (_)

- Les espaces ( ) et certains caractères ne sont pas pris en charge dans les noms de colonnes des tables de base de données. Les tables dont les noms de colonnes contiennent les caractères suivants sont ignorées lors de l'exportation :

  ```
  , ; { } ( ) \n \t = (space)
  ```

- Les tables dont les noms contiennent des barres obliques (/) sont ignorées lors de l'exportation.
- Les tables temporaires et non journalisées de RDS for PostgreSQL sont ignorées lors de l'exportation.
- Si les données contiennent un objet volumineux tel qu'un objet BLOB ou CLOB proche de ou supérieur à 500 Mo, l'exportation échoue.
- Si une table contient une grande ligne proche de ou supérieure à 2 Go, la table est ignorée lors de l'exportation.
- Pour les exportations partielles, la taille maximale de la `ExportOnly` liste est de 200 Ko.
- Nous vous recommandons vivement d'utiliser un nom unique pour chaque tâche d'exportation. Si vous n'utilisez pas un nom de tâche unique, vous risquez de recevoir le message d'erreur suivant :

  ExportTaskAlreadyExistsFault: Une erreur s'est produite (ExportTaskAlreadyExists) lors de l'appel de l' StartExportTaskopération : la tâche d'exportation portant l'ID existe *xxxxx* déjà.
- Vous pouvez supprimer un instantané lors de l'exportation de ses données vers S3, mais les coûts de stockage de cet instantané vous sont tout de même facturés tant que la tâche d'exportation n'est pas terminée.
- Vous ne pouvez pas restaurer les données de capture exportées de S3 vers une nouvelle instance de base de données ou importer des données de capture d'écran de S3 dans une instance de base de données existante.
- Vous pouvez avoir jusqu'à cinq tâches simultanées d'exportation de snapshots de base de données en cours par Compte AWS.
- Pour exporter un instantané de base de données vers un compartiment Amazon S3 multi-comptes, vous devez utiliser l'API AWS CLI ou l'API RDS.
- Une fois qu'Amazon RDS a terminé une tâche d'exportation, vous devrez peut-être attendre un peu avant de démarrer une autre tâche d'exportation à partir du même instantané de base de données.
- Vous ne pouvez pas exporter de vues ou de vues matérialisées.
- RDS Export vers S3 ne prend pas en charge le contrôle d'accès basé sur des balises pour. GuardDuty Malware Protection for S3

Surveillance des exportations de snapshots pour Amazon RDS

Vous pouvez surveiller les exportations de snapshots de base de données à l'aide du AWS Management Console AWS CLI, du ou du RDSAPI.

Console

Pour surveiller les exportations d'instantanés de bases de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Pour afficher la liste des exportations d'instantanés, choisissez l'onglet Exportations dans Amazon S3.
4. Pour afficher des informations sur une exportation d'instantané spécifique, choisissez la tâche d'exportation.

AWS CLI

Pour surveiller les exportations de snapshots de base de données à l'aide de AWS CLI, utilisez la describe-export-taskscommande.

L'exemple suivant montre comment afficher les informations actuelles sur toutes vos exportations d'instantanés.

Example

```
aws rds describe-export-tasks

{
    "ExportTasks": [
        {
            "Status": "CANCELED",
            "TaskEndTime": "2019-11-01T17:36:46.961Z",
            "S3Prefix": "something",
            "ExportTime": "2019-10-24T20:23:48.364Z",
            "S3Bucket": "amzn-s3-demo-bucket",
            "PercentProgress": 0,
            "KmsKeyId": "arn:aws:kms:AWS_Region:123456789012:key/K7MDENG/
bPxRfiCYEXAMPLEKEY",
            "ExportTaskIdentifier": "anewtest",
```

```
            "IamRoleArn": "arn:aws:iam::123456789012:role/export-to-s3",
            "TotalExtractedDataInGB": 0,
            "TaskStartTime": "2019-10-25T19:10:58.885Z",
            "SourceArn": "arn:aws:rds:AWS_Region:123456789012:snapshot:parameter-
groups-test"
        },
{
            "Status": "COMPLETE",
            "TaskEndTime": "2019-10-31T21:37:28.312Z",
            "WarningMessage": "{\"skippedTables\":[],\"skippedObjectives\":[],\"general
\":[{\"reason\":\"FAILED_TO_EXTRACT_TABLES_LIST_FOR_DATABASE\"}]}",
            "S3Prefix": "",
            "ExportTime": "2019-10-31T06:44:53.452Z",
            "S3Bucket": "amzn-s3-demo-bucket1",
            "PercentProgress": 100,
            "KmsKeyId": "arn:aws:kms:AWS_Region:123456789012:key/2Zp9Utk/
h3yCo8nvbEXAMPLEKEY",
            "ExportTaskIdentifier": "thursday-events-test",
            "IamRoleArn": "arn:aws:iam::123456789012:role/export-to-s3",
            "TotalExtractedDataInGB": 263,
            "TaskStartTime": "2019-10-31T20:58:06.998Z",
            "SourceArn":
 "arn:aws:rds:AWS_Region:123456789012:snapshot:rds:example-1-2019-10-31-06-44"
        },
        {
            "Status": "FAILED",
            "TaskEndTime": "2019-10-31T02:12:36.409Z",
            "FailureCause": "The S3 bucket edgcuc-export isn't located in the current
 AWS Region. Please, review your S3 bucket name and retry the export.",
            "S3Prefix": "",
            "ExportTime": "2019-10-30T06:45:04.526Z",
            "S3Bucket": "amzn-s3-demo-bucket2",
            "PercentProgress": 0,
            "KmsKeyId": "arn:aws:kms:AWS_Region:123456789012:key/2Zp9Utk/
h3yCo8nvbEXAMPLEKEY",
            "ExportTaskIdentifier": "wednesday-afternoon-test",
            "IamRoleArn": "arn:aws:iam::123456789012:role/export-to-s3",
            "TotalExtractedDataInGB": 0,
            "TaskStartTime": "2019-10-30T22:43:40.034Z",
            "SourceArn":
 "arn:aws:rds:AWS_Region:123456789012:snapshot:rds:example-1-2019-10-30-06-45"
        }
    ]
```

```
}
```

Pour afficher des informations sur une exportation d'instantané spécifique, incluez l'option `--export-task-identifier` avec la commande `describe-export-tasks`. Pour filtrer la sortie, incluez l'option `--Filters`. Pour plus d'options, consultez la describe-export-taskscommande.

RDS API

Pour afficher des informations sur les exportations de snapshots de base de données à l'aide d'Amazon RDSAPI, utilisez l'DescribeExportTasksopération.

Pour suivre l'achèvement du workflow d'exportation ou pour initier un autre workflow, vous pouvez vous abonner à des rubriques Amazon Simple Notification Service. Pour plus d'informations sur AmazonSNS, consultezUtilisation des notifications d'RDSévénements Amazon.

Annulation d'une tâche d'exportation d'instantanés pour Amazon RDS

Vous pouvez annuler une tâche d'exportation de snapshots de base de données en utilisant le AWS Management Console AWS CLI, le, ou le RDSAPI.

> **Note**
>
> L'annulation d'une tâche d'exportation d'instantané ne supprime aucune des données exportées vers Amazon S3. Pour plus d'informations sur la suppression des données à l'aide de la console, veuillez consulter Comment supprimer des objets d'un compartiment S3 ?. Pour supprimer les données à l'aide deCLI, utilisez la commande delete-object.

Console

Pour annuler une tâche d'importation d'instantané

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'onglet Exports in Amazon S3 (Exportations dans Amazon S3) .
4. Choisissez la tâche d'exportation d'instantané que vous souhaitez annuler.
5. Choisissez Cancel (Annuler).
6. Choisissez Cancel export task (Annuler la tâche d'exportation) sur la page de confirmation.

AWS CLI

Pour annuler une tâche d'exportation d'instantanés à l'aide de AWS CLI, utilisez la cancel-export-taskcommande. La commande requiert l'option `--export-task-identifier`.

Example

```
aws rds cancel-export-task --export-task-identifier my_export
{
    "Status": "CANCELING",
    "S3Prefix": "",
    "ExportTime": "2019-08-12T01:23:53.109Z",
    "S3Bucket": "amzn-s3-demo-bucket",
    "PercentProgress": 0,
    "KmsKeyId": "arn:aws:kms:AWS_Region:123456789012:key/K7MDENG/bPxRfiCYEXAMPLEKEY",
    "ExportTaskIdentifier": "my_export",
    "IamRoleArn": "arn:aws:iam::123456789012:role/export-to-s3",
    "TotalExtractedDataInGB": 0,
    "TaskStartTime": "2019-11-13T19:46:00.173Z",
    "SourceArn": "arn:aws:rds:AWS_Region:123456789012:snapshot:export-example-1"
}
```

RDS API

Pour annuler une tâche d'exportation d'instantanés à l'aide d'Amazon RDSAPI, utilisez l'CancelExportTaskopération avec le `ExportTaskIdentifier` paramètre.

Messages d'échec pour les tâches d'exportation Amazon S3 pour Amazon RDS

Le tableau suivant décrit les messages renvoyés en cas d'échec des tâches d'exportation Amazon S3.

Message d'échec	Description
Une erreur interne inconnue s'est produite.	La tâche a échoué en raison d'une erreur inconnue, d'une exception ou d'un échec.
Une erreur interne inconnue s'est produite lors de l'écriture des	La tâche a échoué en raison d'une erreur inconnue, d'une exception ou d'un échec.

Message d'échec	Description
métadonnées de la tâche d'exporta tion dans le compartiment S3 [nom du compartiment].	
L'RDSexportation n'a pas réussi à écrire les métadonnées de la tâche d'exporta tion car elle ne peut pas assumer le IAM rôle [rôleARN].	La tâche d'exportation assume votre IAM rôle pour valider si elle est autorisée à écrire des métadonnées dans votre compartiment S3. Si la tâche ne peut pas assumer votre IAM rôle, elle échoue.
L'RDSexportation n'a pas réussi à écrire les métadonnées de la tâche d'exporta tion dans le compartiment S3 [nom du compartiment] en utilisant le IAM rôle [rôleARN] avec la KMS clé [ID de clé]. Code d'erreur : [code d'erreur]	Une ou plusieurs autorisations sont manquantes et dès lors, la tâche d'exportation ne peut pas accéder au compartiment S3. Ce message d'échec est généré lors de la réception de l'un des codes d'erreur suivants : • `AWSSecurityTokenServiceException` avec le code d'erreur `AccessDenied` • `AmazonS3Exception` avec le code d'erreur `NoSuchBucket` , `AccessDenied` , `KMS.KMSInvalidStateException` , `403 Forbidden` ou `KMS.DisabledException` Ces codes d'erreur indiquent que les paramètres du IAM rôle, du compartiment S3 ou de la KMS clé sont mal configurés.
Le IAM rôle [rôleARN] n'est pas autorisé à appeler [action S3] sur le compartiment S3 [nom du compartim ent]. Examinez vos autorisations et retentez l'exportation.	La IAM politique est mal configurée. L'autorisation pour l'action S3 spécifique sur le compartiment S3 est manquante, ce qui entraîne l'échec de la tâche d'exportation.
KMSéchec de la vérification des touches. Vérifiez les informations d'identification de votre KMS clé et réessayez.	La vérification KMS des informations d'identification clés a échoué.

Message d'échec	Description
La vérification des informations d'identification S3 a échoué. Vérifiez les autorisations de votre compartiment S3 et de votre IAM politique.	La vérification des informations d'identification S3 a échoué.
Le compartiment S3 [nom du compartiment] n'est pas valide. Il n'est peut-être pas situé dans la région AWS actuelle ou il n'existe pas. Vérifiez le nom de votre compartiment S3 et retentez l'exportation.	Le compartiment S3 n'est pas valide.
Le compartiment S3 [nom du compartiment] ne se trouve pas dans la AWS région actuelle. Vérifiez le nom de votre compartiment S3 et retentez l'exportation.	Le compartiment S3 se trouve dans la mauvaise AWS région.

Résolution RDS des erreurs liées aux SQL autorisations Postgre

Lorsque vous exportez des SQL bases de données Postgre vers Amazon S3, vous pouvez voir une `PERMISSIONS_DO_NOT_EXIST` erreur indiquant que certaines tables ont été ignorées. Cette erreur se produit généralement lorsque le superutilisateur, que vous avez spécifié lors de la création de la base de données, n'est pas autorisé à accéder à ces tables.

Pour corriger cette erreur, exécutez la commande suivante :

```
GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA schema_name TO superuser_name
```

Pour plus d'informations sur les privilèges des superutilisateurs, veuillez consulter Privilèges du compte utilisateur principal.

Conventions de dénomination des fichiers pour les exportations vers Amazon S3 pour Amazon RDS

Les données exportées pour des tables spécifiques sont stockées au format *base_prefix*/*files*, qui utilise le préfixe de base suivant :

```
export_identifier/database_name/schema_name.table_name/
```

Par exemple :

```
export-1234567890123-459/rdststdb/rdststdb.DataInsert_7ADB5D19965123A2/
```

Il existe deux conventions de dénomination des fichiers.

- Convention actuelle :

  ```
  batch_index/part-partition_index-random_uuid.format-based_extension
  ```

  L'index de lot est un numéro de séquence qui représente un lot de données lues dans la table. Si nous ne parvenons pas à partitionner votre table en petits morceaux à exporter en parallèle, il y aura plusieurs index de lots. Il en va de même si votre table est partitionnée en plusieurs tables. Il y aura plusieurs index par lots, un pour chacune des partitions de table de votre table principale.

  Si nous parvenons à partitionner votre table en petits morceaux à lire en parallèle, il n'y aura que le 1 dossier d'index par lots.

  Dans le dossier d'index par lots, un ou plusieurs fichiers Parquet contiennent les données de votre table. Le préfixe du nom du fichier Parquet est`part-`*`partition_index`*. Si votre table est partitionnée, il y aura plusieurs fichiers en commençant par l'index `00000` de partition.

  Il peut y avoir des lacunes dans la séquence d'index de partition. Cela se produit parce que chaque partition est obtenue à partir d'une requête à distance dans votre table. S'il n'y a aucune donnée dans la plage de cette partition, le numéro de séquence est ignoré.

  Supposons, par exemple, que la `id` colonne soit la clé primaire de la table et que ses valeurs minimale et maximale soient `100` et`1000`. Lorsque nous essayons d'exporter cette table avec neuf partitions, nous la lisons avec des requêtes parallèles telles que les suivantes :

  ```
  SELECT * FROM table WHERE id <= 100 AND id < 200
  ```

```
SELECT * FROM table WHERE id <= 200 AND id < 300
```

Cela devrait générer neuf fichiers, de `part-00000-`*`random_uuid`*`.gz.parquet` à`part-00008-`*`random_uuid`*`.gz.parquet`. Toutefois, s'il n'y a aucune ligne IDs entre `200` et`350`, l'une des partitions terminées est vide et aucun fichier n'est créé pour elle. Dans l'exemple précédent, `part-00001-`*`random_uuid`*`.gz.parquet` n'est pas créé.

- Ancienne convention :

```
part-partition_index-random_uuid.format-based_extension
```

C'est la même que la convention actuelle, mais sans le *`batch_index`* préfixe, par exemple :

```
part-00000-c5a881bb-58ff-4ee6-1111-b41ecff340a3-c000.gz.parquet
part-00001-d7a881cc-88cc-5ab7-2222-c41ecab340a4-c000.gz.parquet
part-00002-f5a991ab-59aa-7fa6-3333-d41eccd340a7-c000.gz.parquet
```

La convention de dénomination de fichiers est sujette à modification. Par conséquent, lors de la lecture des tables cibles, nous vous conseillons de lire tout ce qui se trouve à l'intérieur du préfixe de base de la table.

Conversion de données lors de l'exportation vers un compartiment Amazon S3 pour Amazon RDS

Lorsque vous exportez un instantané de base de données vers un compartiment Amazon S3, Amazon RDS convertit les données, les exporte et les stocke au format Parquet. Pour plus d'informations sur Parquet, veuillez consulter le site web Apache Parquet.

Parquet stocke toutes les données sous l'un des types primitifs suivants :

- BOOLEAN
- INT32
- INT64
- INT96
- FLOAT
- DOUBLE

- BYTE_ARRAY – Tableau d'octets de longueur variable, également connu sous le nom de binaire
- FIXED_LEN_BYTE_ARRAY – Tableau d'octets de longueur fixe utilisé lorsque les valeurs ont une taille constante

Les types de données Parquet sont peu nombreux afin de la complexité de la lecture et de l'écriture du format. Parquet fournit des types logiques pour étendre les types primitifs. Un type logique est implémenté sous forme d'annotation avec les données dans un champ de métadonnées `LogicalType`. L'annotation de type logique explique comment interpréter le type primitif.

Lorsque le type logique `STRING` annote un type `BYTE_ARRAY`, il indique que le tableau d'octets doit être interprété comme une chaîne de caractères codée en UTF-8. Une fois la tâche d'exportation terminée, Amazon RDS vous avertit si une conversion de chaîne s'est produite. Les données sous-jacentes exportées sont toujours les mêmes que celles de la source. Cependant, en raison de la différence d'encodage en UTF-8, certains caractères peuvent apparaître différents de la source lorsqu'ils sont lus dans des outils tels que Athena.

Pour plus d'informations, veuillez consulter Parquet Logical Type Definitions dans la documentation Parquet.

Rubriques

- Mappage de type de données MySQL et MariaDB vers Parquet
- Mappage de type de données PostgreSQL vers Parquet

Mappage de type de données MySQL et MariaDB vers Parquet

Le tableau suivant montre le mappage des types de données MySQL et MariaDB aux types de données Parquet lorsque les données sont converties et exportées vers Amazon S3.

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
Types de données numériques			
BIGINT	INT64		
BIGINT UNSIGNED	FIXED_LEN _BYTE_ARRAY(9)	DECIMAL(20,0)	Parquet ne prend en charge que les

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
			types signés, de sorte que le mappage nécessite un octet supplémentaire (8 plus 1) pour stocker le type BIGINT_UNSIGNED.
BIT	BYTE_ARRAY		
DECIMAL	INT32	DECIMAL(p,s)	Si la valeur source est inférieure à 2^{31}, elle est stockée sous la forme INT32.
	INT64	DECIMAL(p,s)	Si la valeur source est supérieure ou égale à 2^{31}, mais inférieure à 2^{63}, elle est stockée sous la formeINT64.
	FIXED_LEN_BYTE_ARRAY(N)	DECIMAL(p,s)	Si la valeur source est supérieure ou égale à 2^{63}, elle est stockée sous la forme FIXED_LEN_BYTE_ARRAY(N).

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
	BYTE_ARRAY	STRING	Parquet ne prend pas en charge une précision décimale supérieure à 38. La valeur décimale est convertie en chaîne de type BYTE_ARRAY et codée sous la forme. UTF8
DOUBLE	DOUBLE		
FLOAT	DOUBLE		
INT	INT32		
INT UNSIGNED	INT64		
MEDIUMINT	INT32		
MEDIUMINT UNSIGNED	INT64		
NUMERIC	INT32	DECIMAL(p,s)	Si la valeur source est inférieure à 2^{31}, elle est stockée sous la formeINT32.
	INT64	DECIMAL(p,s)	Si la valeur source est supérieure ou égale à 2^{31}, mais inférieure à 2^{63}, elle est stockée sous la formeINT64.

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
	FIXED_LEN _ARRAY(N)	DECIMAL(p,s)	Si la valeur source est supérieure ou égale à 2^{63}, elle est stockée sous la forme FIXED_LEN _BYTE_ARRAY(N).
	BYTE_ARRAY	STRING	Parquet ne prend pas en charge la précision numérique supérieure à 38. Cette valeur numérique est convertie en chaîne de type BYTE_ARRA Y et codée sous la forme. UTF8
SMALLINT	INT32		
SMALLINT UNSIGNED	INT32		
TINYINT	INT32		
TINYINT UNSIGNED	INT32	INT (16, vrai)	
Types de données chaîne			
BINARY	BYTE_ARRAY		
BLOB	BYTE_ARRAY		
CHAR	BYTE_ARRAY		
ENUM	BYTE_ARRAY	STRING	

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
LINESTRING	BYTE_ARRAY		
LONGBLOB	BYTE_ARRAY		
LONGTEXT	BYTE_ARRAY	STRING	
MEDIUMBLOB	BYTE_ARRAY		
MEDIUMTEXT	BYTE_ARRAY	STRING	
MULTILINESTRING	BYTE_ARRAY		
SET	BYTE_ARRAY	STRING	
TEXT	BYTE_ARRAY	STRING	
TINYBLOB	BYTE_ARRAY		
TINYTEXT	BYTE_ARRAY	STRING	
VARBINARY	BYTE_ARRAY		
VARCHAR	BYTE_ARRAY	STRING	
Types de données de date et d'heure			
DATE	BYTE_ARRAY	STRING	Une date est convertie en chaîne de type BYTE_ARRAY et encodée sous la forme. UTF8
DATETIME	INT64	TIMESTAMP_MICROS	

Type de données source	Type primitif du format Parquet	Annotation de type logique	Notes de conversion
TIME	BYTE_ARRAY	STRING	Un type TIME est converti en chaîne dans un BYTE_ARRAY et codé sous la forme. UTF8
TIMESTAMP	INT64	TIMESTAMP_MICROS	
YEAR	INT32		
Types de données géométriques			
GEOMETRY	BYTE_ARRAY		
GEOMETRYCOLLECTION	BYTE_ARRAY		
MULTIPOINT	BYTE_ARRAY		
MULTIPOLYGON	BYTE_ARRAY		
POINT	BYTE_ARRAY		
POLYGON	BYTE_ARRAY		
Type de données JSON			
JSON	BYTE_ARRAY	STRING	

Mappage de type de données PostgreSQL vers Parquet

Le tableau suivant montre le mappage des types de données PostgreSQL aux types de données Parquet lorsque les données sont converties et exportées vers Amazon S3.

Type de données PostgreSQL	Type primitif du format Parquet	Annotation de type logique	Notes de mappage
Types de données numériques			
BIGINT	INT64		
BIGSERIAL	INT64		
DECIMAL	BYTE_ARRAY	STRING	Un type DECIMAL est converti en chaîne de type BYTE_ARRAY et codé sous la forme. UTF8 Cette conversio n vise à éviter les complications dues à la précision des données et aux valeurs de données qui ne sont pas un nombre (NaN).
DOUBLE PRECISION	DOUBLE		
INTEGER	INT32		
MONEY	BYTE_ARRAY	STRING	
REAL	FLOAT		
SERIAL	INT32		
SMALLINT	INT32	INT (16, vrai)	
SMALLSERIAL	INT32	INT (16, vrai)	
Types de chaînes et de données associés			

Type de données PostgreSQL	Type primitif du format Parquet	Annotation de type logique	Notes de mappage
ARRAY	BYTE_ARRAY	STRING	Un tableau est converti en chaîne et codé sous la forme BINARY (UTF8). Cette conversion vise à éviter les complications dues à la précision des données, aux valeurs de données qui ne sont pas un nombre (NaN) et aux valeurs de données horaires.
BIT	BYTE_ARRAY	STRING	
BIT VARYING	BYTE_ARRAY	STRING	
BYTEA	BINARY		
CHAR	BYTE_ARRAY	STRING	
CHAR(N)	BYTE_ARRAY	STRING	
ENUM	BYTE_ARRAY	STRING	
NAME	BYTE_ARRAY	STRING	
TEXT	BYTE_ARRAY	STRING	
TEXT SEARCH	BYTE_ARRAY	STRING	
VARCHAR(N)	BYTE_ARRAY	STRING	
xml	BYTE_ARRAY	STRING	

Type de données PostgreSQL	Type primitif du format Parquet	Annotation de type logique	Notes de mappage
Types de données de date et d'heure			
DATE	BYTE_ARRAY	STRING	
INTERVAL	BYTE_ARRAY	STRING	
TIME	BYTE_ARRAY	STRING	
TIME WITH TIME ZONE	BYTE_ARRAY	STRING	
TIMESTAMP	BYTE_ARRAY	STRING	
TIMESTAMP WITH TIME ZONE	BYTE_ARRAY	STRING	
Types de données géométriques			
BOX	BYTE_ARRAY	STRING	
CIRCLE	BYTE_ARRAY	STRING	
LINE	BYTE_ARRAY	STRING	
LINESEGMENT	BYTE_ARRAY	STRING	
PATH	BYTE_ARRAY	STRING	
POINT	BYTE_ARRAY	STRING	
POLYGON	BYTE_ARRAY	STRING	
Types de données JSON			
JSON	BYTE_ARRAY	STRING	
JSONB	BYTE_ARRAY	STRING	
Autres types de données			

Type de données PostgreSQL	Type primitif du format Parquet	Annotation de type logique	Notes de mappage
BOOLEAN	BOOLEAN		
CIDR	BYTE_ARRAY	STRING	Type de données de réseau
COMPOSITE	BYTE_ARRAY	STRING	
DOMAIN	BYTE_ARRAY	STRING	
INET	BYTE_ARRAY	STRING	Type de données de réseau
MACADDR	BYTE_ARRAY	STRING	
OBJECT IDENTIFIER	N/A		
PG_LSN	BYTE_ARRAY	STRING	
RANGE	BYTE_ARRAY	STRING	
UUID	BYTE_ARRAY	STRING	

Utilisation AWS Backup pour gérer les sauvegardes automatisées pour Amazon RDS

AWS Backup est un service de sauvegarde entièrement géré qui facilite la centralisation et l'automatisation de la sauvegarde des données entre les AWS services dans le cloud et sur site. Vous pouvez gérer les sauvegardes de vos RDS bases de données Amazon dans AWS Backup.

> **Note**
>
> Les sauvegardes gérées par AWS Backup sont considérées comme des instantanés de base de données manuels, mais ne sont pas prises en compte dans le quota de clichés de base de données pourRDS. Les sauvegardes créées avec des noms AWS Backup se terminant par`awsbackup:`*`backup-job-number`*.

Pour plus d'informations à ce sujet AWS Backup, consultez le guide du AWS Backup développeur.

Pour afficher les sauvegardes gérées par AWS Backup

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Cliquez sur l'onglet Service de sauvegarde .

   Vos AWS Backup sauvegardes sont répertoriées sous Instantanés du service de sauvegarde.

Surveillance des métriques dans une RDS instance Amazon

Dans les sections suivantes, vous trouverez un aperçu de la RDS surveillance d'Amazon et une explication sur la manière d'accéder aux métriques. Pour savoir comment surveiller les événements, les journaux et les flux d'activité de la base de données, veuillez consulter Surveillance des événements, des journaux et des flux dans un d'instances de base de données RDSAmazon.

Rubriques

- Plan de surveillance
- Référence des performances
- Instructions sur les performances
- Outils de surveillance pour Amazon RDS
- Affichage de l'état
- Recommandations d'Amazon RDS
- Afficher les métriques dans la RDS console Amazon
- Afficher les indicateurs combinés avec le tableau de bord Performance Insights
- Surveillance des métriques Amazon RDS () avec Amazon CloudWatch
- Surveillance des bases de données Amazon RDS avec CloudWatch Database Insights
- Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS
- Analyse des anomalies de performance avec Amazon DevOps Guru pour Amazon RDS
- Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée
- Référence des métriques pour Amazon RDS

Plan de surveillance

Avant de commencer à surveiller Amazon RDS , créez un plan de surveillance. Ce plan doit répondre aux questions suivantes :

- Quels sont les objectifs de la surveillance ?
- Quelles sont les ressources à surveiller ?
- A quelle fréquence les ressources doivent-elles être surveillées ?
- Quels outils de surveillance utiliser ?
- Qui exécute les tâches de surveillance ?

- Qui doit être informé en cas de problème ?

Référence des performances

Pour atteindre vos objectifs de surveillance, vous devez établir une référence. Pour ce faire, mesurez les performances dans différentes conditions de charge à différents moments dans votre environnement RDSAmazon . Vous pouvez surveiller les métriques suivantes :

- Débit réseau
- Connexions client
- I/O pour les opérations de lecture, d'écriture ou de métadonnées
- Soldes de crédit en rafales pour vos instances de base de données

Nous vous recommandons de stocker les données de performance historiques pour Amazon RDS . À l'aide des données stockées, vous pouvez comparer les performances actuelles aux tendances passées. Vous pouvez également faire la distinction entre les modèles normaux de performances et les anomalies, puis concevoir des techniques pour résoudre les problèmes.

Instructions sur les performances

En général, les valeurs acceptables pour les métriques de performances dépendent de l'activité de votre application par rapport à votre référence. Enquêtez sur les écarts cohérents ou tendanciels de vos données de référence. Les métriques suivantes sont souvent à l'origine des problèmes de performances :

- Niveau élevé CPU ou RAM consommation : des valeurs élevées pour CPU ou une RAM consommation élevées peuvent être appropriées si elles correspondent aux objectifs de votre application (comme le débit ou la simultanéité) et si elles sont attendues.
- Utilisation de l'espace disque – Enquêtez sur l'utilisation de l'espace disque si l'espace utilisé est constamment égal ou supérieur à 85 pour cent de l'espace disque total. Voyez s'il est possible de supprimer des données de l'instance ou d'archiver des données sur un système différent pour libérer de l'espace.
- Trafic réseau – Pour le trafic réseau, discutez avec votre administrateur pour connaître le débit attendu pour votre domaine réseau et votre connexion Internet. Enquêtez sur le trafic réseau si le débit est constamment inférieur à vos attentes.

- Connexions de la base de données – Envisagez de limiter les connexions de la base de données si vous constatez un nombre important de connexions utilisateur en même temps qu'une baisse des performances de l'instance et des temps de réponse. Le bon nombre de connexions utilisateur pour votre instance de base de données dépend de votre classe d'instance et de la complexité des opérations exécutées. Pour déterminer le nombre de connexions de la base de données, associez votre instance de base de données à un groupe de paramètres dans lequel le paramètre `User Connections` est configuré sur une autre valeur que 0 (illimité). Vous pouvez utiliser un groupe de paramètres existant ou en créer un nouveau. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).
- IOPSmétriques — Les valeurs attendues pour les IOPS métriques dépendent des spécifications du disque et de la configuration du serveur. Utilisez donc votre référence pour savoir ce qui est typique. Enquêtez si les valeurs sont constamment différentes de vos données de référence. Pour des IOPS performances optimales, assurez-vous que votre ensemble de travail habituel est bien conservé en mémoire afin de minimiser les opérations de lecture et d'écriture.

Lorsque les performances se situent en dehors de votre de base établie, vous devrez peut-être apporter des modifications pour optimiser la disponibilité de votre base de données pour votre charge de travail. Par exemple, vous devrez peut-être modifier la classe d'instance de votre instance de base de données. Ou encore, modifier le nombre d'instances de base de données et de réplicas en lecture disponibles pour les clients.

Outils de surveillance pour Amazon RDS

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances d'Amazon, d'RDS et de vos autres AWS solutions. AWS fournit divers outils de surveillance pour surveiller RDSAmazon , signaler un problème et prendre des mesures automatiques le cas échéant.

Rubriques

- Outils de surveillance automatique
- Outils de surveillance manuelle

Outils de surveillance automatique

Nous vous recommandons d'automatiser le plus possible les tâches de supervision.

Rubriques

- RDSInstance Amazon et recommandations
- CloudWatch Métriques Amazon pour Amazon RDS
- Amazon RDS Performance Insights et surveillance du système d'exploitation
- Services intégrés

RDSInstance Amazon et recommandations

Vous pouvez utiliser les outils automatisés suivants pour surveiller RDSAmazon et signaler un problème :

- État du d'instances RDS Amazon : consultez les informations relatives à l'état actuel de votre d'instances à l'aide de la RDS console Amazon, AWS CLI, ou le RDSAPI.
- Recommandations Amazon RDS — Répondez aux recommandations automatisées concernant les ressources de base de données, telles que les instances de base de données, les les répliques de lecture et les groupes de paramètres de de bases de données. Pour de plus amples informations, veuillez consulter Recommandations d'Amazon RDS.

CloudWatch Métriques Amazon pour Amazon RDS

Amazon RDS s'intègre à Amazon CloudWatch pour des fonctionnalités de surveillance supplémentaires.

- Amazon CloudWatch — Ce service surveille votre AWS les ressources et les applications que vous utilisez AWS en temps réel. Vous pouvez utiliser les CloudWatch fonctionnalités Amazon suivantes avec RDSAmazon :
  - CloudWatch Métriques Amazon — RDSAmazon envoie automatiquement des métriques CloudWatch toutes les minutes pour chaque base de données active. Vous n'avez pas à payer de frais supplémentaires pour RDS les statistiques Amazon CloudWatch. Pour plus d'informations, veuillez consulter Surveillance des métriques Amazon RDS () avec Amazon CloudWatch.
  - CloudWatch Alarmes Amazon : vous pouvez consulter une seule métrique RDSAmazon sur une période donnée. Vous pouvez ensuite effectuer une ou plusieurs actions en fonction de la valeur de la métrique selon le seuil que vous définissez. Pour de plus amples informations, veuillez consulter Surveillance des métriques Amazon RDS () avec Amazon CloudWatch.

Amazon RDS Performance Insights et surveillance du système d'exploitation

Vous pouvez utiliser les outils automatisés suivants pour surveiller les performances d'RDS :

- Amazon RDS Performance Insights — Évaluez la charge de votre base de données et déterminez quand et où prendre des mesures. Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.
- Amazon RDS Enhanced Monitoring : consultez les statistiques du système d'exploitation en temps réel. Pour de plus amples informations, veuillez consulter Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Services intégrés

Procédez comme suit : AWS les services sont intégrés à Amazon RDS :

- Amazon EventBridge est un service de bus d'événements sans serveur qui permet de connecter facilement vos applications à des données provenant de diverses sources. Pour de plus amples informations, veuillez consulter Surveillance des RDS événements Amazon.

- Amazon CloudWatch Logs vous permet de surveiller, de stocker et d'accéder à vos fichiers journaux à partir d'instances RDSAmazon et d'autres sources. CloudTrail Pour de plus amples informations, veuillez consulter Surveillance des fichiers journaux RDSAmazon.
- AWS CloudTrailenregistre API les appels et les événements connexes effectués par ou au nom de votre Compte AWS et envoie les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Pour de plus amples informations, veuillez consulter Surveillance (Amazon RDS API appelle) AWS CloudTrail.
- Database Activity Streams est une fonctionnalité d'RDSAmazon qui fournit un near-real-time flux d'activité dans votre instance de de base de données Oracle. Pour de plus amples informations, veuillez consulter Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données.

Outils de surveillance manuelle

Vous devez surveiller manuellement les éléments non couverts par les CloudWatch alarmes. L'Amazon RDS CloudWatch, AWS Trusted Advisor et autres AWS les tableaux de bord de console fournissent une at-a-glance vue de l'état de votre AWS environnement. Nous recommandons de consulter également les fichiers journaux sur votre instance de base de données.

- Depuis la RDS console Amazon, vous pouvez surveiller les éléments suivants pour vos ressources :
  - Nombre de connexions à une instance de base de données
  - Quantité d'opérations de lecture et d'écriture à une instance de base de données
  - Volume de stockage en cours d'utilisation par une instance de base de données
  - La quantité de mémoire CPU utilisée par une instance de base de données
  - Quantité de trafic réseau en direction et à partir d'une instance de base de données
- À partir du Trusted Advisor tableau de bord, vous pouvez consulter les contrôles suivants d'optimisation des coûts, de sécurité, de tolérance aux pannes et d'amélioration des performances :
  - Instances de base de données Amazon RDS Idle
  - Risque d'accès au groupe de RDS sécurité Amazon
  - RDSSauvegardes Amazon
  - Amazon RDS Multi-AZ

  Pour plus d'informations sur ces vérifications, consultez Bonnes pratiques Trusted Advisor (Checks).

- CloudWatch la page d'accueil montre :
  - Alarmes et statuts en cours
  - Graphiques des alarmes et des ressources
  - Statut d’intégrité du service

  En outre, vous pouvez utiliser CloudWatch pour effectuer les opérations suivantes :
  - Créer des tableaux de bord personnalisés pour surveiller les services qui vous intéressent.
  - Données de métriques de graphiques pour résoudre les problèmes et découvrir les tendances.
  - Recherchez et parcourez tous vos AWS métriques relatives aux ressources.
  - Créer et Modifier des alarmes pour être informé des problèmes.

Affichage de l'état

À l'aide de la console Amazon RDS, vous pouvez accéder rapidement à l'état de votre instance de de base de données.

Rubriques

- Affichage de l'état de l'instance de base de données dans un cluster Aurora

Affichage de l'état de l'instance de base de données dans un cluster Aurora

Le statut d'une instance de base de données indique son état. Vous pouvez utiliser les procédures suivantes pour afficher l'état dans la console Amazon RDS, dans la AWS CLI commande ou dans le fonctionnement de l'API.

> **Note**
>
> Amazon RDS utilise également un autre statut appelé état de la maintenance, qui est affiché dans la colonne Maintenance de la console Amazon RDS. Cette valeur indique l'état de tout correctif de maintenance qui doit être appliqué à une instance de base de données. L'état de la maintenance est indépendant du statut de l'instance de base de données. Pour plus d'informations sur le statut de la maintenance, consultez Appliquer des mises à jour à un d'instances de base de données.

Recherchez les valeurs d'état possibles pour les instances de base de données dans le tableau suivant. Il indique également si vous serez facturé pour l'instance de base de données et son stockage, uniquement pour le stockage, ou si vous ne serez pas facturé. Pour tous les statuts d'instance de base de données, vous êtres toujours facturé pour l'utilisation de la sauvegarde.

Statut d'instanc e de bases de données	Facturé	Description
available	Facturé	L'instance de base de données est saine et disponibl e.
backing-up	Facturé	L'instance de base de données est en cours de sauvegarde.
configuring-enhanced-monitoring	Facturé	La surveillance améliorée est en cours d'activation ou de désactivation pour cette instance de base de données.
configuring-iam-da tabase-auth	Facturé	AWS Identity and Access Management L'authent ification de base de données (IAM) est activée ou désactivée pour cette instance de base de données.

Statut d'instanc e de bases de données	Facturé	Description
configuring-log-exports	Facturé	La publication de fichiers CloudWatch journaux sur Amazon Logs est activée ou désactivée pour cette instance de base de données.
converting-to-vpc	Facturé	L'instance de base de données est en cours de conversion depuis une instance qui ne se trouve pas dans un Amazon Virtual Private Cloud (Amazon VPC) vers une instance qui est dans un Amazon VPC.
creating	Non facturé (non PITR) Facturé (PITR uniquement)	L'instance de base de données est en cours de création. L'instance de base de données n'est pas accessible pendant sa création. Si vous restaurez une base de données pendant la point-in-time restauration (PITR), vous êtes facturé alors que la base de données est en cours de création. C'est le seul scénario dans lequel l'État créateur encourt des frais.
supprimer-prévérifier	Non facturé	Amazon RDS vérifie que les réplicas en lecture sont sains et peuvent être supprimés en toute sécurité.
deleting	Non facturé	L'instance de base de données est en cours de suppression.
failed	Non facturé	L'instance de base de données a échoué et Amazon RDS ne peut pas la récupérer. Effectuez une point-in-time restauration à l'heure de restauration la plus récente de l'instance de base de données pour récupérer les données.
inaccessible-encry ption-credentials	Non facturé	Le AWS KMS key fichier utilisé pour chiffrer ou déchiffrer l'instance de base de données n'est pas accessible ni récupéré.

Statut d'instance de bases de données	Facturé	Description
inaccessible-encryption-credentials-recoverable	Facturé pour stockage	La clé KMS utilisée pour chiffrer ou déchiffrer l'instance de base de données n'est pas accessible. Toutefois, si la clé KMS est active, le redémarrage de l'instance de base de données peut la récupérer. Pour de plus amples informations, veuillez consulter Chiffrement d'une instance de base de données.
incompatible-créer	Non facturé	Amazon RDS tente de créer une instance de base de données mais n'y parvient pas car les ressources sont incompatibles avec votre instance de base de données. Ce statut peut se produire si, par exemple, le profil d'instance de votre instance de base de données ne dispose pas des autorisations appropriées.
incompatible-network	Non facturé	Amazon RDS tente d'effectuer une action de récupération sur une instance de base de données, mais n'y parvient pas, car l'état du VPC empêche l'exécution de l'action. Cette situation peut se produire si, par exemple, toutes les adresses IP disponibles d'un sous-réseau sont en cours d'utilisation et qu'Amazon RDS ne peut pas obtenir d'adresse IP pour l'instance de base de données.
incompatible-option-group	Facturé	Amazon RDS a tenté d'appliquer une modification du groupe d'options, mais n'y parvient pas, et Amazon RDS ne peut pas rétablir l'état antérieur du groupe options. Pour plus d'informations, consultez la liste Événements récents de l'instance de base de données. Cette situation peut se produire si, par exemple, le groupe d'options contient une option telle que TDE et que l'instance de base de données ne comporte pas d'informations chiffrées.

Statut d'instanc e de bases de données	Facturé	Description
incompatible-param eters	Facturé	Amazon RDS ne peut pas démarrer l'instance de base de données, car les paramètres spécifiés dans le groupe de paramètres de base de données de l'instance ne sont pas compatibles avec l'instance. Annulez les modifications des paramètres ou rendez-les compatibles avec l'instance de base de données pour rétablir l'accès à votre instance. Pour en savoir plus sur les paramètres incompatibles, consultez la liste Événements récents correspondant à l'instance de base de données.
incompatible-restore	Non facturé	Amazon RDS ne peut pas effectuer de point-in-time restauration. Parmi les causes courantes de ce statut figurent l'utilisation des tables temporaires, des tables MyISAM avec MySQL ou des tables Aria avec MariaDB.
insufficient-capacity	Non facturé	Amazon RDS ne peut pas créer votre instance car une capacité suffisante n'est actuellement pas disponible. Pour créer votre instance de base de données dans la même AZ avec le même type d'instance, supprimez votre instance de base de données, attendez quelques heures et essayez de la recréer. En variante, vous pouvez créer une nouvelle instance en utilisant une classe d'instances différente ou AZ.
maintenance	Facturé	Amazon RDS applique une mise à jour de maintenan ce à l'instance base de données. Cet état est utilisé pour la maintenance de niveau d'instance que RDS planifie suffisamment à l'avance.

Statut d'instanc e de bases de données	Facturé	Description
modifying	Facturé	L'instance de base de données est en cours de modification en raison d'une demande du client de modification de l'instance.
moving-to-vpc	Facturé	L'instance de base de données est en cours de transfert vers un nouveau Amazon Virtual Private Cloud (Amazon VPC).
rebooting	Facturé	L'instance de base de données est en cours de redémarrage en raison d'une demande du client ou d'un processus Amazon RDS nécessitant le redémarrage de l'instance.
resetting-master-c redentials	Facturé	Les informations d'identification principales de l'instance de base de données sont en cours de réinitialisation en raison d'une demande du client pour les réinitialiser.
renaming	Facturé	L'instance de base de données est en cours de changement de nom en raison d'une demande du client pour la renommer.
restore-error	Facturé	L'instance de base de données a rencontré une erreur lors de la tentative de restauration depuis point-in-time ou vers un instantané.
starting	Facturé pour stockage	L'instance de base de données démarre.
stopped	Facturé pour stockage	L'instance de base de données est arrêtée.
stopping	Facturé pour stockage	L'instance de base de données est en cours d'arrêt.

Statut d'instanc e de bases de données	Facturé	Description
storage-config-upg rade	Facturé	La configuration du système de fichiers de stockage de l'instance de base de données est en cours de mise à niveau. Ce statut s'applique uniquement aux bases de données vertes dans le cadre d'un déploiement bleu/vert, ou aux réplicas en lecture d'instances de base de données.
storage-full	Facturé	L'instance de base de données a atteint sa capacité de stockage allouée. Son statut est critique et nous vous recommandons de corriger ce problème immédiatement. Pour cela, mettez votre stockage à l'échelle en modifiant l'instance de base de données. Pour éviter cette situation, configurez les CloudWatc h alarmes Amazon pour qu'elles vous avertissent lorsque l'espace de stockage est insuffisant.
initialisation du stockage	Facturé	L'instance de base de données charge des blocs de données depuis Amazon S3 afin d'optimiser les performances du volume après avoir été restaurée à partir d'un instantané. Il reste disponible pour les opérations, mais les performances risquent de ne pas être optimales tant que l'initialisation n'est pas terminée.
optimisation du stockage	Facturé	Amazon RDS optimise le stockage de votre instance de base de données. Le processus d'optimisation du stockage est généralement court mais peut parfois prendre jusqu'à 24 heures ou même davantage. Pendant l'optimisation du stockage, l'instance de base de données reste disponible. L'optimisation du stockage est un processus d'arrière-plan qui n'affecte pas la disponibilité de l'instance.

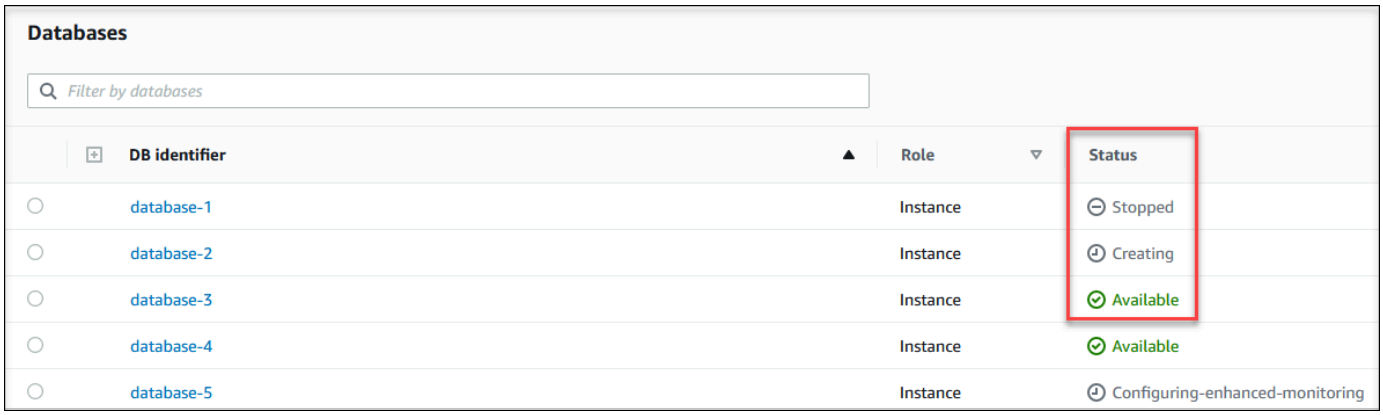
Statut d'instanc e de bases de données	Facturé	Description
upgrading	Facturé	La version du moteur de base de données ou du système d'exploitation est en cours de mise à niveau.

console

Pour afficher le statut d'une l'instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données.

   La Databases page (page Bases de données) apparaît avec la liste des instances de base de données. Pour chaque instance de la base de données r, la valeur du statut est affichée.



INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour afficher l'instance de base de données et ses informations d'état à l'aide de AWS CLI, utilisez la describe-db-instancescommande. Par exemple, la AWS CLI commande suivante répertorie toutes les informations relatives aux instances de base de données.

```
aws rds describe-db-instances
```

Pour afficher une instance de base de données spécifique et son état, appelez la describe-db-instancescommande avec l'option suivante :

- `DBInstanceIdentifier` – Nom de l'instance de base de données.

```
aws rds describe-db-instances --db-instance-identifier mydbinstance
```

Pour afficher uniquement le statut de toutes les instances de base de données, utilisez la requête suivante dans AWS CLI.

```
aws rds describe-db-instances --query 'DBInstances[*].
[DBInstanceIdentifier,DBInstanceStatus]' --output table
```

« Hello, World! »

Pour consulter l'état de l'instance de base de données à l'aide de l'API Amazon RDS, appelez l'DBInstancesopération Describe.

Recommandations d'Amazon RDS

Amazon RDS Amazon fournit des recommandations automatisées pour les ressources de base de données, telles que les instances de base de données, les les répliques de lecture et les groupes de paramètres de base de données. Ces recommandations fournissent des conseils quand aux bonnes pratiques en analysant la la configuration de l'instance de base de données, son utilisation et les données de performances.

Amazon RDS Performance Insights surveille des indicateurs spécifiques et crée automatiquement des seuils en analysant les niveaux considérés comme potentiellement problématiques pour une ressource spécifique. Lorsque de nouvelles valeurs métriques dépassent un seuil prédéfini sur une période donnée, Performance Insights génère une recommandation proactive. Cette recommandation permet d'éviter tout impact futur sur les performances de la base de données. Par exemple, la recommandation « Idle In Transaction » est générée RDSpour les SQL instances Postgre SQL lorsque les sessions connectées à la base de données n'effectuent pas de travail actif, mais peuvent bloquer les ressources de la base de données. Pour recevoir des recommandations proactives, vous devez activer Performance Insights avec une période de rétention payante. Pour plus d'informations sur l'activation de Performance Insights, consultezActiver et désactiver Performance Insights pour Amazon RDS. Pour plus d'informations sur la tarification et la conservation des données pour Performance Insights, consultezTarification et conservation des données pour Performance Insights.

DevOpsGuru for RDS surveille certaines métriques afin de détecter les cas où le comportement de la métrique devient très inhabituel ou anormal. Ces anomalies sont signalées sous forme d'informations réactives accompagnées de recommandations. Par exemple, DevOps Guru for RDS peut vous recommander d'envisager d'augmenter la CPU capacité ou d'étudier les événements d'attente qui contribuent à la charge de la base de données. DevOpsGuru for fournit RDS également des recommandations proactives basées sur des seuils. Pour ces recommandations, vous devez activer DevOps Guru pourRDS. Pour plus d'informations sur l'activation de DevOps Guru pourRDS, consultezActiver DevOps Guru et spécifier la couverture des ressources.

Les recommandations auront l'un des statuts suivants : active, rejetée, en attente ou résolue. Les recommandations résolues sont disponibles pendant 365 jours.

Vous pouvez consulter ou ignorer les recommandations. Vous pouvez appliquer immédiatement une recommandation active basée sur la configuration, la planifier dans la fenêtre de maintenance suivante ou l'ignorer. Pour les recommandations proactives basées sur des seuils et les

recommandations réactives basées sur l'apprentissage automatique, vous devez examiner la cause suggérée du problème, puis exécuter les actions recommandées pour le résoudre.

Les recommandations sont soutenues dans les domaines suivants Régions AWS :

- USA Est (Ohio)
- USA Est (Virginie du Nord)
- USA Ouest (Californie du Nord)
- USA Ouest (Oregon)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Séoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Londres)
- Europe (Paris)
- Europe (Stockholm)
- Amérique du Sud (São Paulo)

Découvrez comment consulter, appliquer, ignorer et modifier les recommandations d'Amazon RDS dans les sections suivantes.

Rubriques

- Consulter les recommandations RDS
- Appliquer les recommandations RDS
- Rejet des recommandations d'RDS
- Modification des recommandations Amazon RDS rejetées en recommandations actives
- Recommandations issues de la référence RDSAmazon

Consulter les recommandations RDS

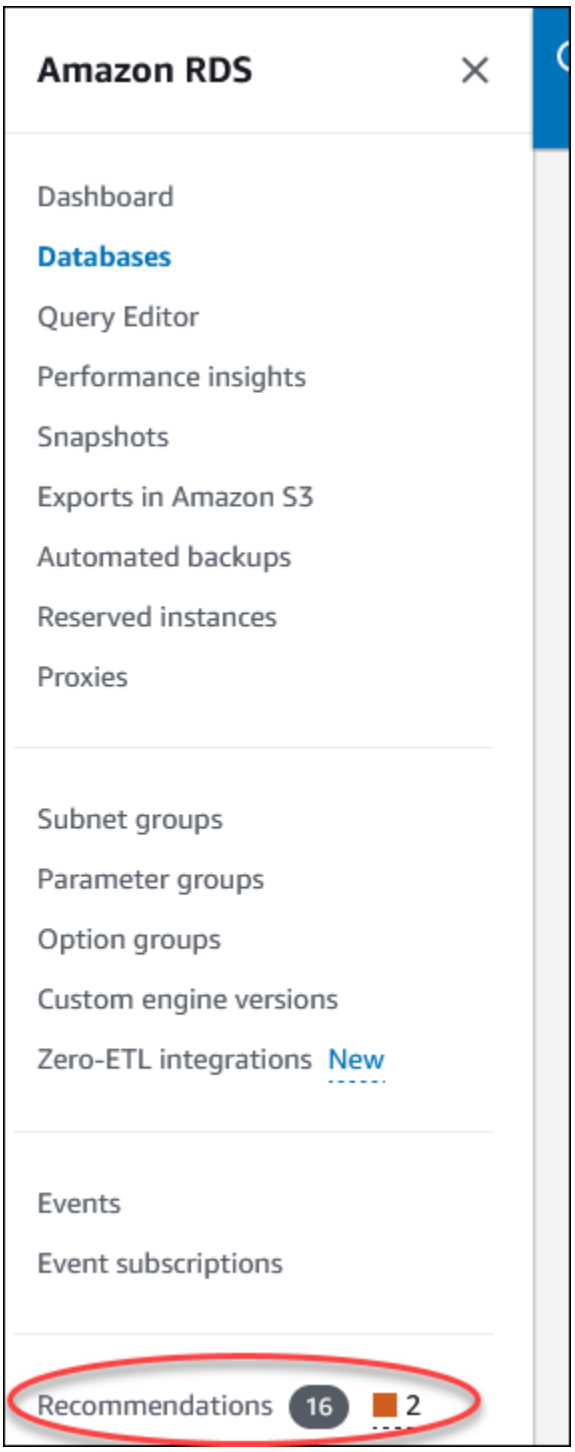
À l'aide de la RDS console Amazon, vous pouvez consulter les recommandations d'RDS concernant les ressources de votre base de données.

Console

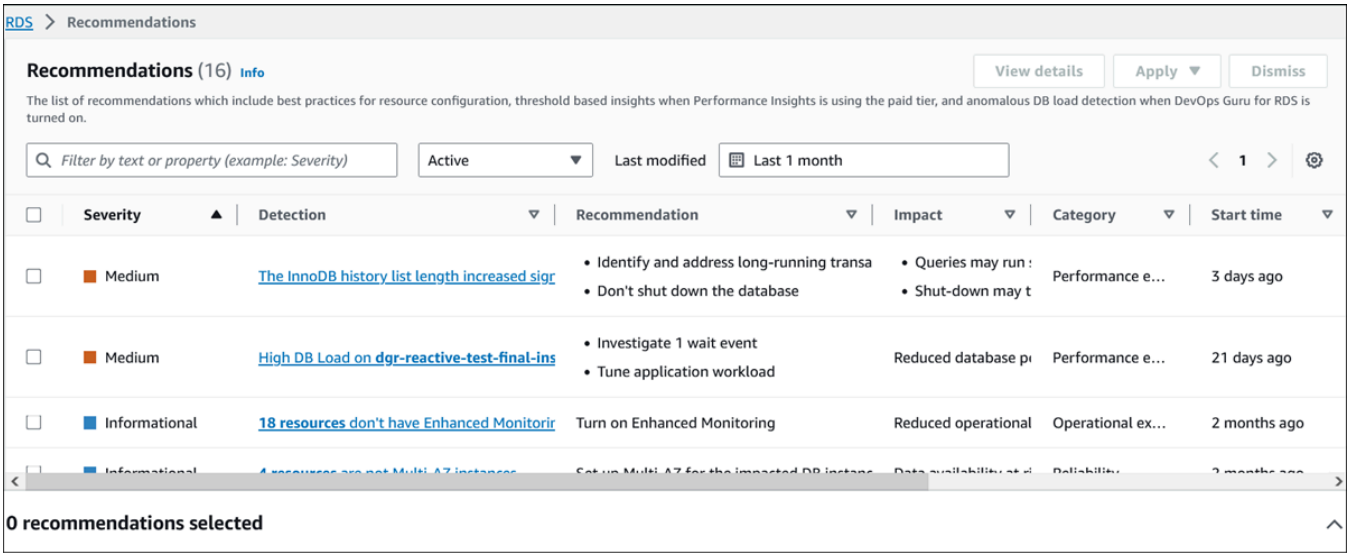
Pour consulter les recommandations d'RDS

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, effectuez l'une des opérations suivantes :

   - Choisissez Recommandations. Le nombre de recommandations actives pour vos ressources et le nombre de recommandations les plus sévères générées le mois dernier sont disponibles à côté de Recommandations. Pour connaître le nombre de recommandations actives pour chaque niveau de gravité, choisissez celui qui indique le niveau de gravité le plus élevé.

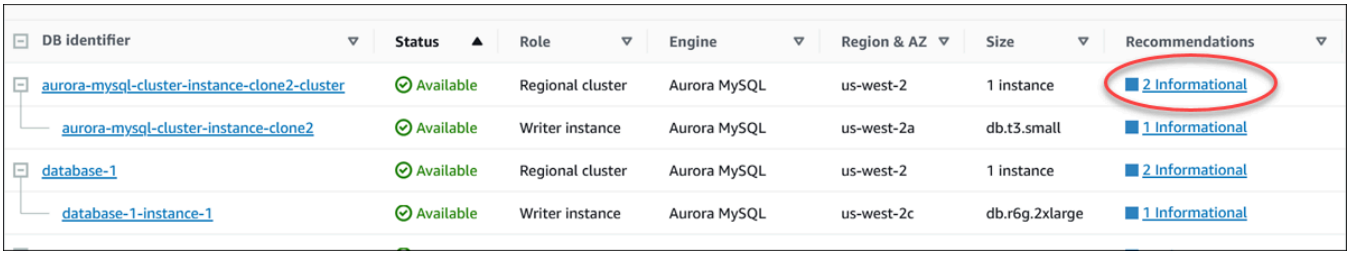


Par défaut, la page Recommandations affiche la liste des nouvelles recommandations du mois dernier. Amazon RDS fournit des recommandations pour toutes les ressources de votre compte et trie les recommandations en fonction de leur gravité.

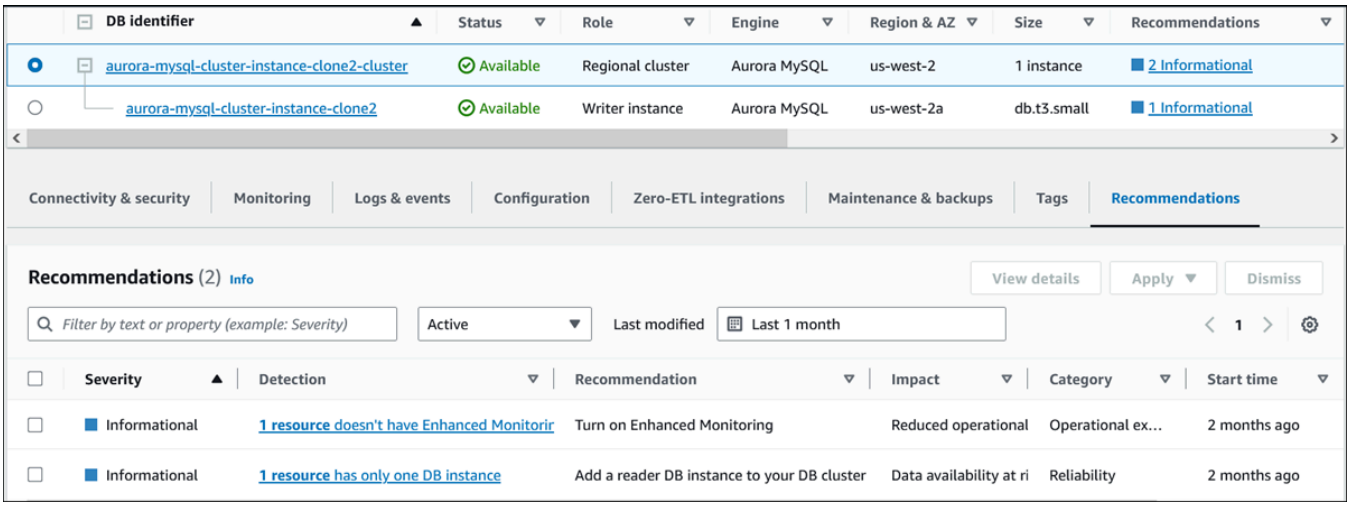


Vous pouvez choisir une recommandation pour consulter une section au bas de la page qui contient les ressources concernées et les détails de la manière dont la recommandation sera appliquée.

- Sur la page Bases de données, sélectionnez Recommandations pour une ressource.



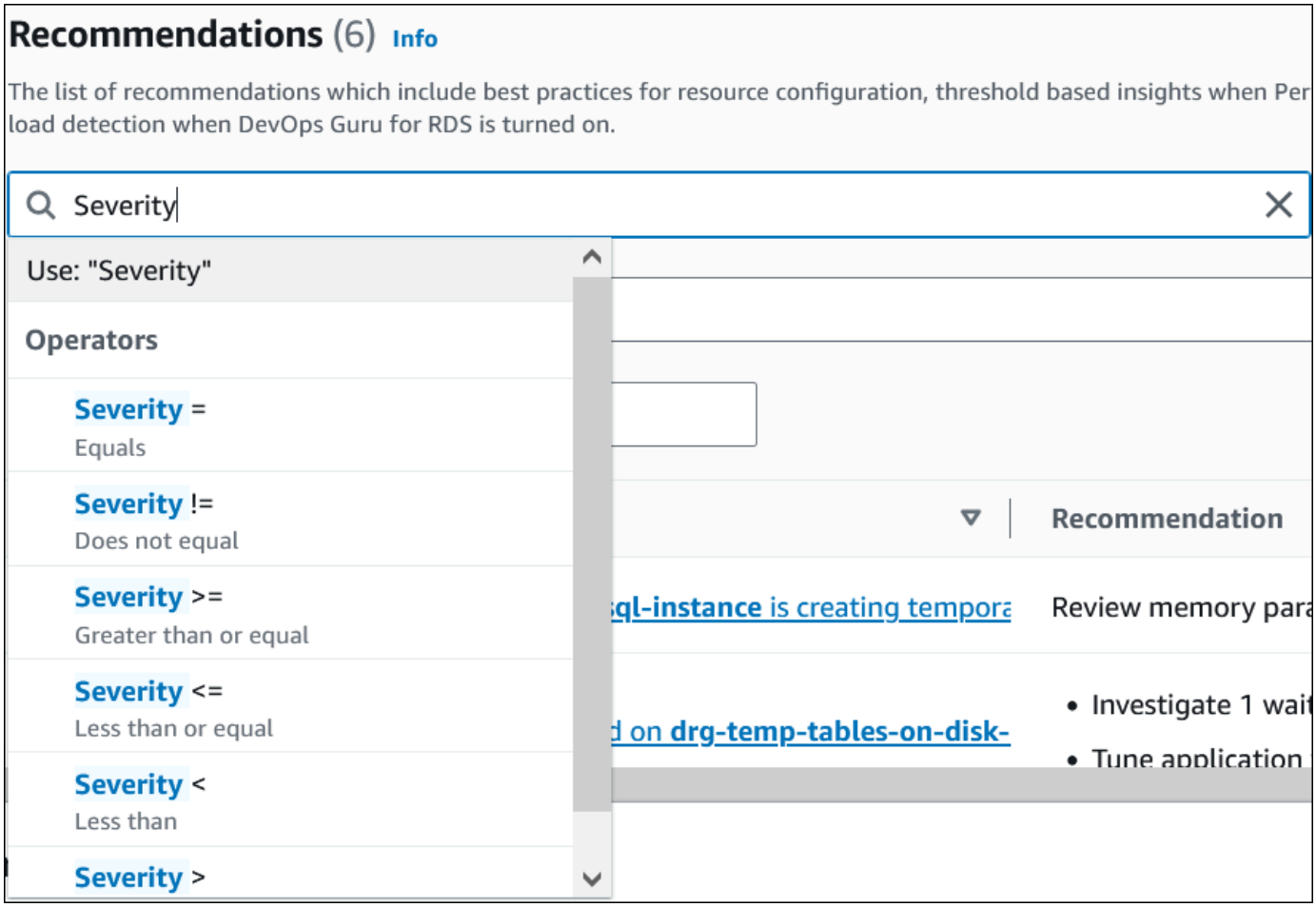
L'onglet Recommandations affiche les recommandations et leurs détails pour la ressource sélectionnée.



Les informations suivantes sont disponibles pour les recommandations :

- Gravité : niveau d'implication du problème. Les niveaux de gravité sont élevés, moyens, faibles et informatifs.
- Détection : nombre de ressources affectées et brève description du problème. Cliquez sur ce lien pour afficher la recommandation et les détails de l'analyse.
- Recommandation — Brève description de l'action recommandée à appliquer.
- Impact : brève description de l'impact possible lorsque la recommandation n'est pas appliquée.
- Catégorie : type de recommandation. Les catégories sont l'efficacité des performances, la sécurité, la fiabilité, l'optimisation des coûts, l'excellence opérationnelle et la durabilité.
- État : statut actuel de la recommandation. Les statuts possibles sont Tous, Actif, Rejeté, Résolu et En attente.
- Heure de début : heure à laquelle le problème a commencé. Par exemple, il y a 18 heures.
- Dernière modification : heure à laquelle la recommandation a été mise à jour pour la dernière fois par le système en raison d'une modification du niveau de gravité, ou heure à laquelle vous avez répondu à la recommandation. Par exemple, il y a 10 heures.
- Heure de fin : heure à laquelle le problème a pris fin. L'heure ne s'affichera pas en cas de problème persistant.
- Identifiant de ressource : nom d'une ou de plusieurs ressources.

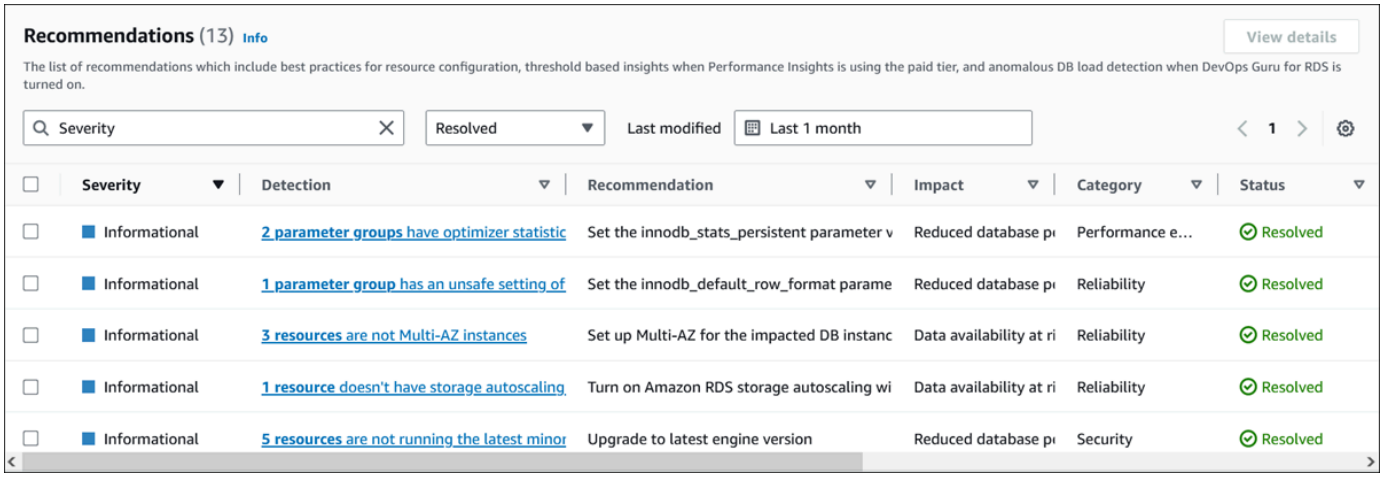
3. (Facultatif) Choisissez les opérateurs de gravité ou de catégorie dans le champ pour filtrer la liste des recommandations.



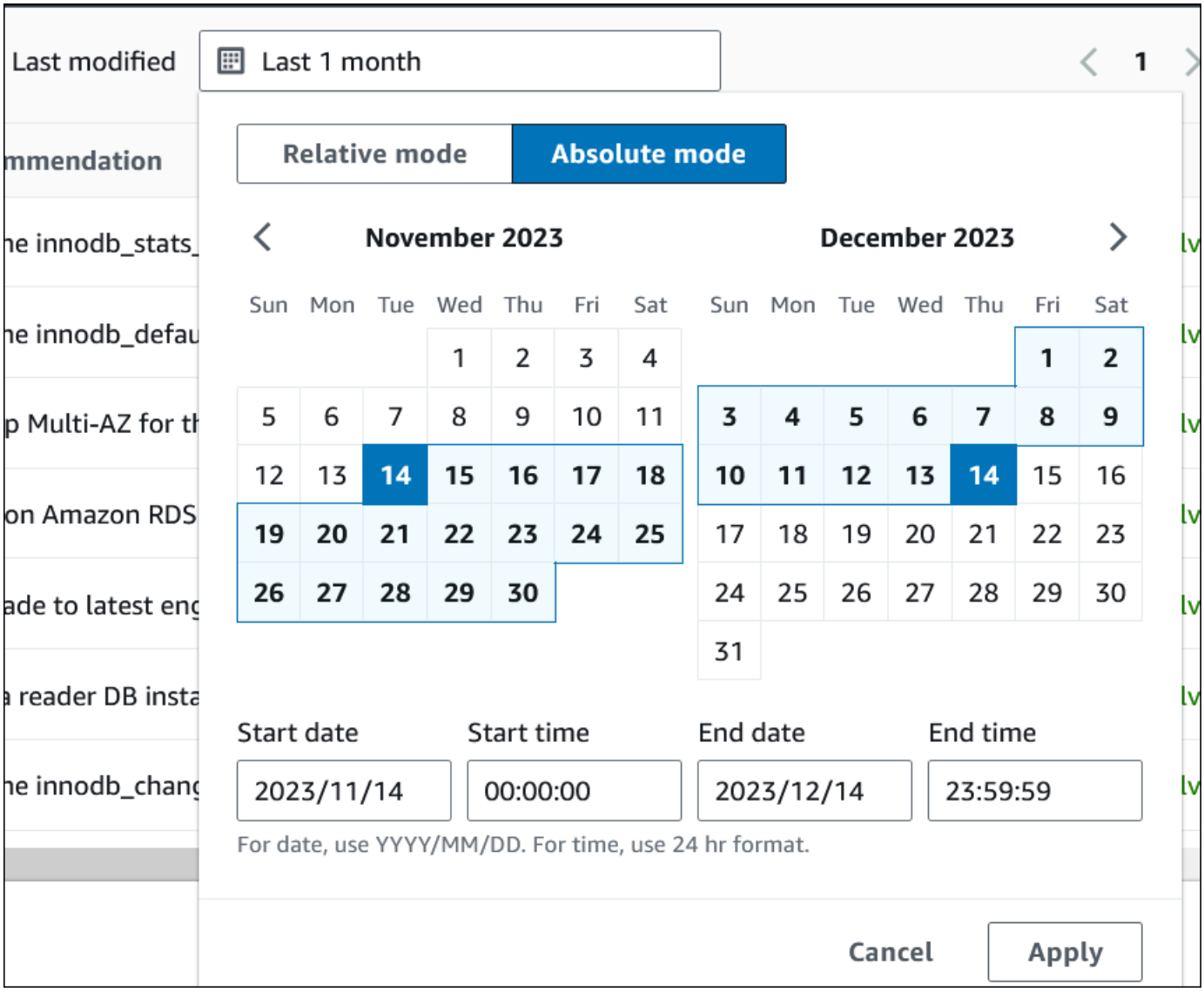
Les recommandations relatives à l'opération sélectionnée apparaissent.

4. (Facultatif) Choisissez l'un des statuts de recommandation suivants :

   - Actif (par défaut) : affiche les recommandations actuelles que vous pouvez appliquer, les planifier pour la prochaine fenêtre de maintenance ou les ignorer.
   - Toutes : affiche toutes les recommandations avec leur état actuel.
   - Rejeté — Affiche les recommandations rejetées.
   - Résolu — Affiche les recommandations qui ont été résolues.
   - En attente : affiche les recommandations dont les actions recommandées sont en cours ou planifiées pour la prochaine fenêtre de maintenance.



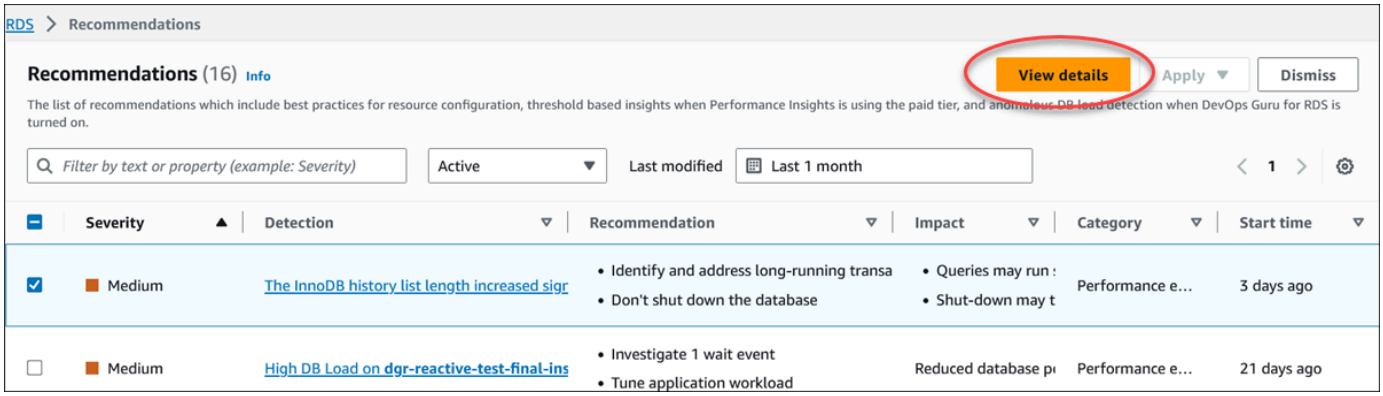
5. (Facultatif) Choisissez le mode relatif ou le mode absolu dans Dernière modification pour modifier la période. La page Recommandations affiche les recommandations générées au cours de la période. La période par défaut est le dernier mois. En mode absolu, vous pouvez choisir la période ou saisir l'heure dans les champs Date de début et Date de fin.



Les recommandations relatives à la période définie s'affichent.

Notez que vous pouvez consulter toutes les recommandations relatives aux ressources de votre compte en définissant la plage sur Toutes.

6. (Facultatif) Choisissez Préférences sur la droite pour personnaliser les détails à afficher. Vous pouvez choisir un format de page, enrouler les lignes du texte et autoriser ou masquer les colonnes.
7. (Facultatif) Choisissez une recommandation, puis sélectionnez Afficher les détails.



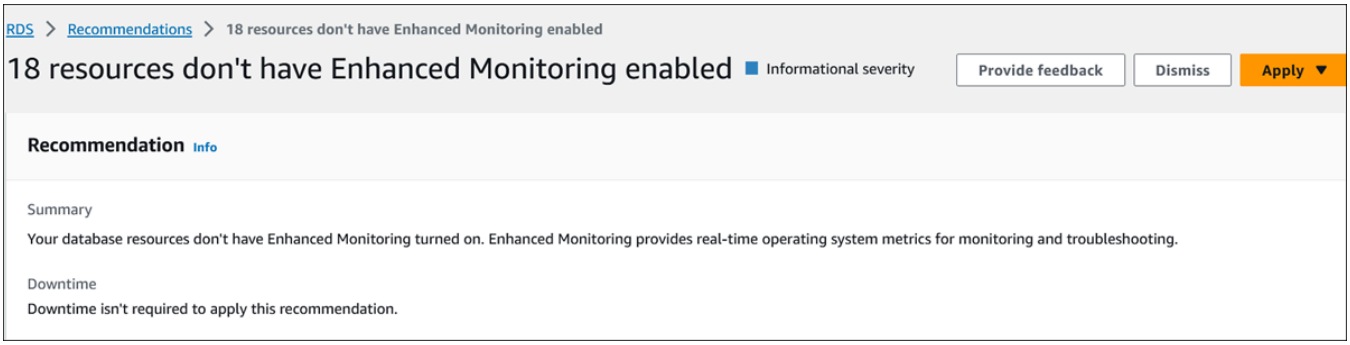
La page de détails des recommandations s'affiche. Le titre indique le nombre total de ressources ainsi que le problème détecté et sa gravité.

Pour plus d'informations sur les composants figurant sur la page de détails d'une recommandation réactive basée sur les anomalies, consultez la section Visualisation des anomalies réactives dans le guide de l'utilisateur Amazon DevOps Guru.

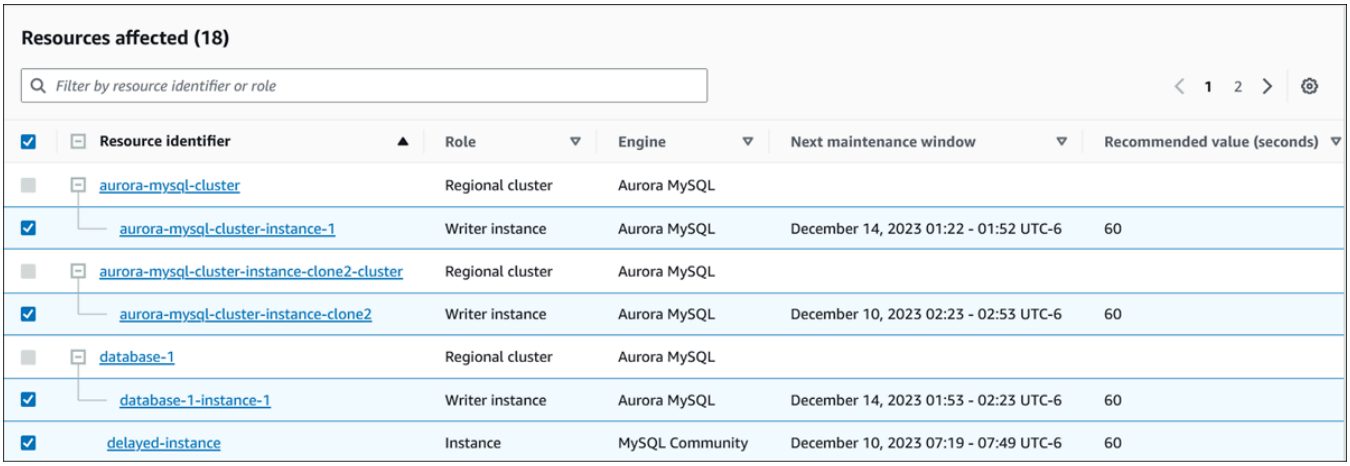
Pour plus d'informations sur les composants figurant sur la page de détails d'une recommandation proactive basée sur un seuil, consultezConsulter les recommandations proactives de Performance Insights.

Les autres recommandations automatisées affichent les composants suivants sur la page détaillée des recommandations :

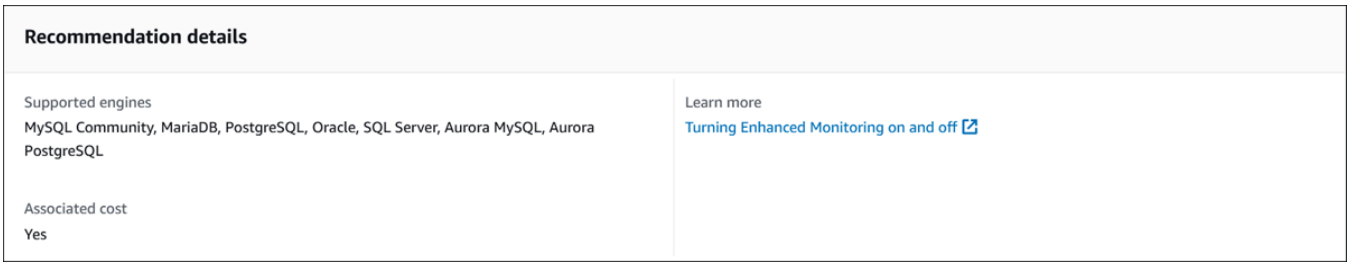
- Recommandation — Un résumé de la recommandation et indiquant si un temps d'arrêt est nécessaire pour appliquer la recommandation.



- Ressources affectées : détails des ressources affectées.



- Détails de la recommandation : informations sur le moteur pris en charge, tout coût associé requis pour appliquer la recommandation et lien vers la documentation pour en savoir plus.



CLI

Pour consulter les RDS recommandations Amazon relatives aux instances de données, utilisez la commande suivante dans AWS CLI.

```
aws rds describe-db-recommendations
```

RDS API

Pour consulter les RDS recommandations d'Amazon à l'aide d'Amazon RDSAPI, utilisez l'escribeDBRecommendationsopération D.

Appliquer les recommandations RDS

Pour appliquer les recommandations RDSAmazon à l'aide de la RDS console Amazon, sélectionnez une recommandation basée sur la configuration ou une ressource affectée sur la page de détails. Choisissez ensuite d'appliquer la recommandation immédiatement ou de la planifier pour la fenêtre de maintenance suivante. Il se peut que la ressource doive redémarrer pour que la modification soit

prise en compte. Pour quelques recommandations relatives aux groupes de paramètres de base de données, vous devrez peut-être redémarrer les ressources.

Les recommandations proactives ou réactives basées sur des seuils ne pourront pas être appliquées et pourraient nécessiter un examen supplémentaire.

Console

Pour appliquer une recommandation basée sur la configuration

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, effectuez l'une des opérations suivantes :

   - Choisissez Recommandations.

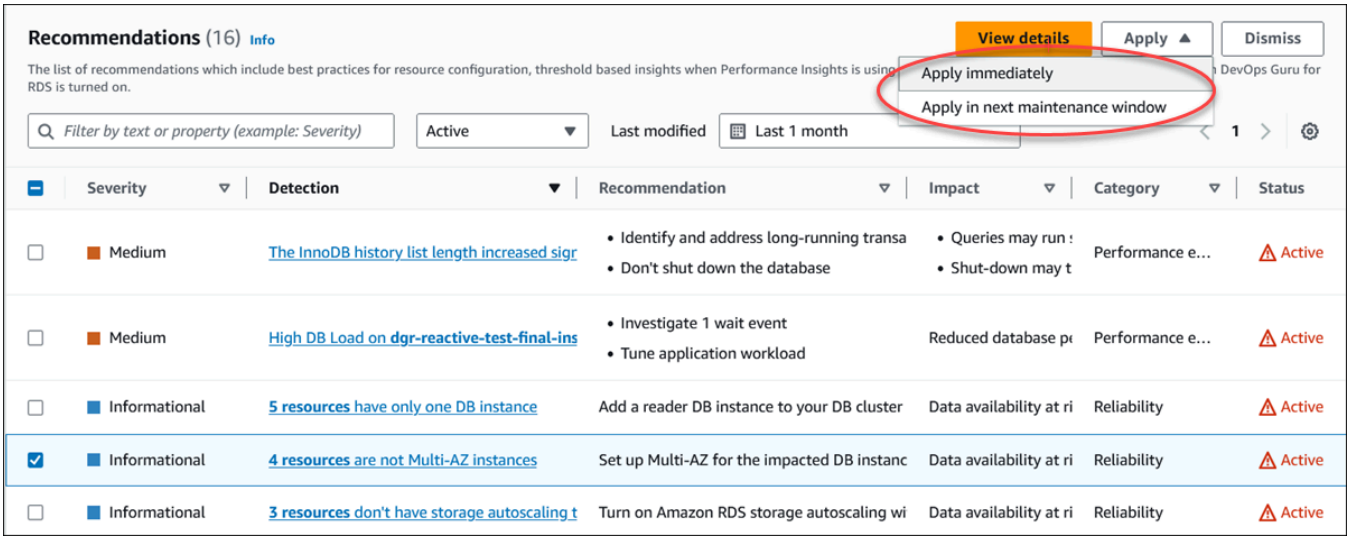
     La page Recommandations apparaît avec la liste de toutes les recommandations.
   - Choisissez Bases de données, puis sélectionnez Recommandations pour une ressource dans la page des bases de données.

     Les détails apparaissent dans l'onglet Recommandations de la recommandation sélectionnée.
   - Choisissez Détection pour une recommandation active sur la page Recommandations ou sur l'onglet Recommandations de la page Bases de données.

     La page de détails des recommandations s'affiche.
3. Choisissez une recommandation ou une ou plusieurs ressources concernées dans la page de détails de la recommandation, puis effectuez l'une des opérations suivantes :

   - Choisissez Appliquer, puis cliquez sur Appliquer immédiatement pour appliquer la recommandation immédiatement.
   - Choisissez Appliquer, puis sélectionnez Appliquer dans la fenêtre de maintenance suivante pour planifier la fenêtre de maintenance suivante.

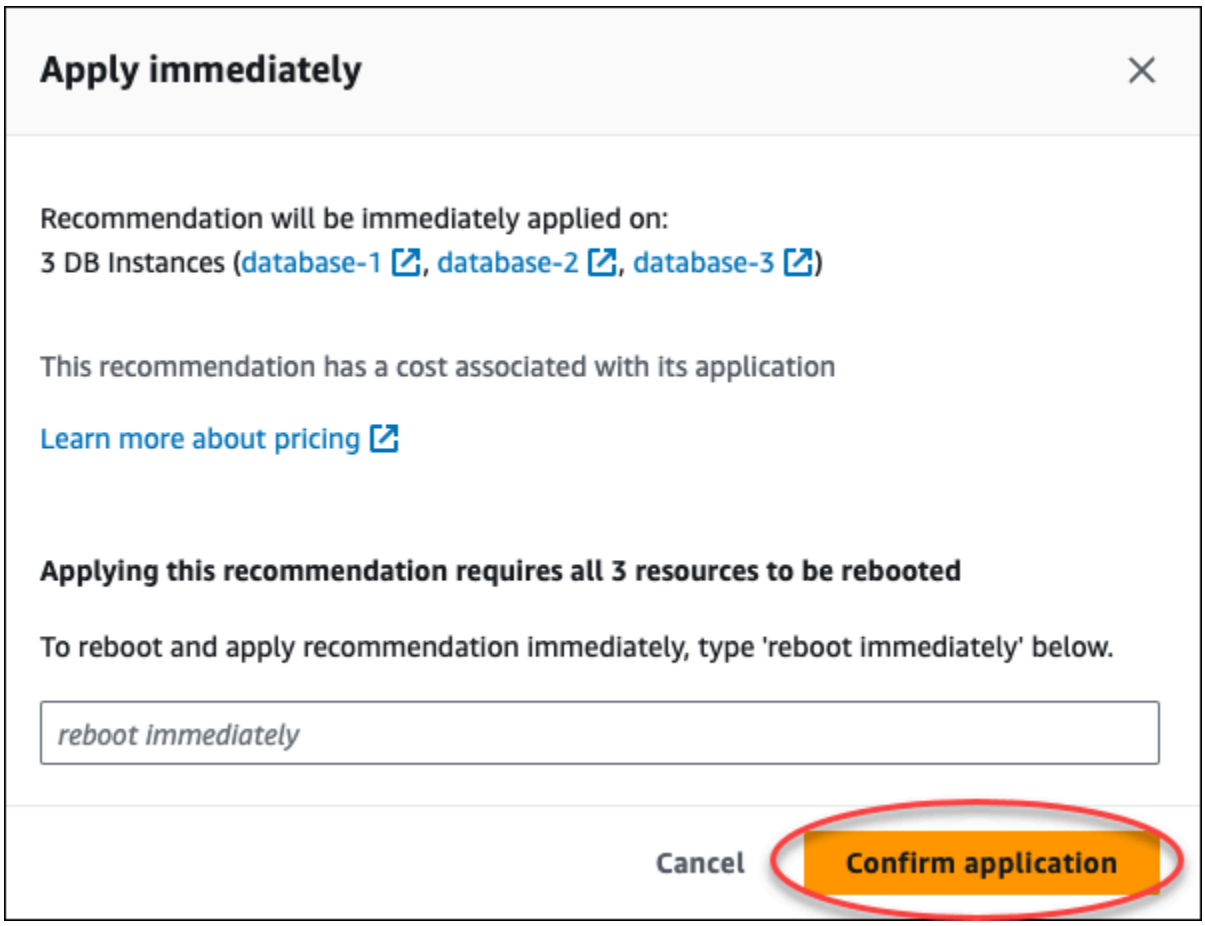
     Le statut de recommandation sélectionné est mis à jour et passe à En attente jusqu'à la prochaine fenêtre de maintenance.



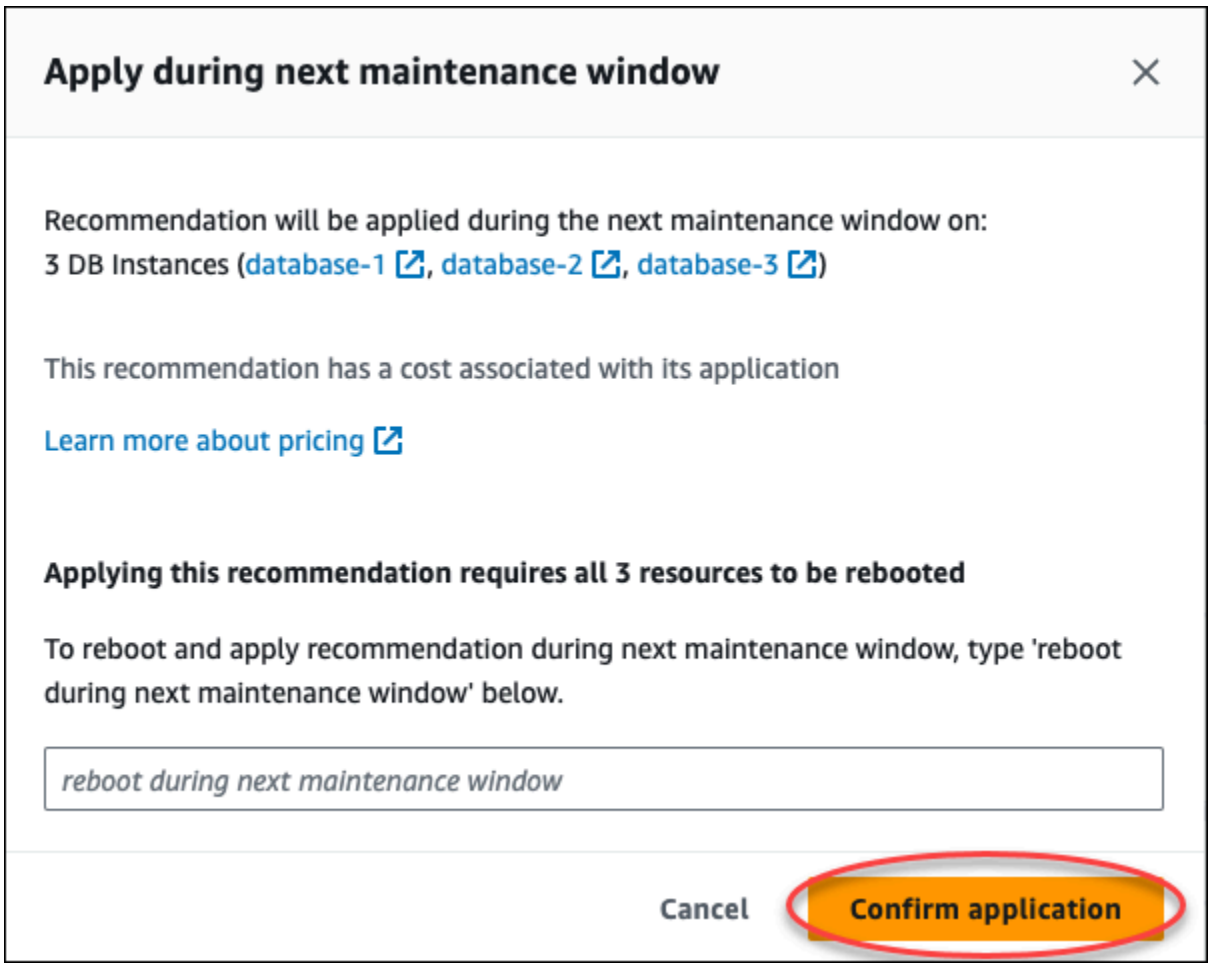
Une fenêtre de confirmation apparaît.

4. Choisissez Confirmer l'application pour appliquer la recommandation. Cette fenêtre confirme si les ressources ont besoin d'un redémarrage automatique ou manuel pour que les modifications prennent effet.

   L'exemple suivant montre la fenêtre de confirmation permettant d'appliquer immédiatement la recommandation.



L'exemple suivant montre la fenêtre de confirmation permettant de planifier l'application de la recommandation dans la fenêtre de maintenance suivante.



Une bannière affiche un message lorsque la recommandation appliquée est réussie ou a échoué.

L'exemple suivant montre la bannière avec le message de réussite.



L'exemple suivant montre la bannière avec le message d'échec.



RDS API

Pour appliquer une recommandation RDS basée sur la configuration à l'aide d'Amazon RDS API

1. Utilisez l'escribeDBRecommendationsopération D. La `RecommendedActions` sortie peut contenir une ou plusieurs actions recommandées.
2. Utilisez l'RecommendedActionobjet pour chaque action recommandée à l'étape 1. La sortie contient `Operation` et`Parameters`.

   L'exemple suivant montre le résultat avec une action recommandée.

```
   "RecommendedActions": [
       {
           "ActionId": "0b19ed15-840f-463c-a200-b10af1b552e3",
           "Title": "Turn on auto backup", // localized
           "Description": "Turn on auto backup for my-mysql-instance-1", //
localized
           "Operation": "ModifyDbInstance",
           "Parameters": [
               {
                   "Key": "DbInstanceIdentifier",
                   "Value": "my-mysql-instance-1"
               },
               {
                   "Key": "BackupRetentionPeriod",
                   "Value": "7"
               }
           ],
           "ApplyModes": ["immediately", "next-maintenance-window"],
           "Status": "applied"
       },
       ... // several others
   ],
```

3. Utilisez le `operation` pour chaque action recommandée à partir de la sortie de l'étape 2 et entrez les `Parameters` valeurs.
4. Une fois l'opération de l'étape 2 réussie, utilisez l'odifyDBRecommendationopération M pour modifier le statut de la recommandation.

Rejet des recommandations d'RDS

Vous pouvez ignorer une ou plusieurs recommandations RDSAmazon à l'aide de la RDS console Amazon ou d'Amazon RDSAPI. AWS CLI

Console

Pour rejeter une ou plusieurs recommandations

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, effectuez l'une des opérations suivantes :

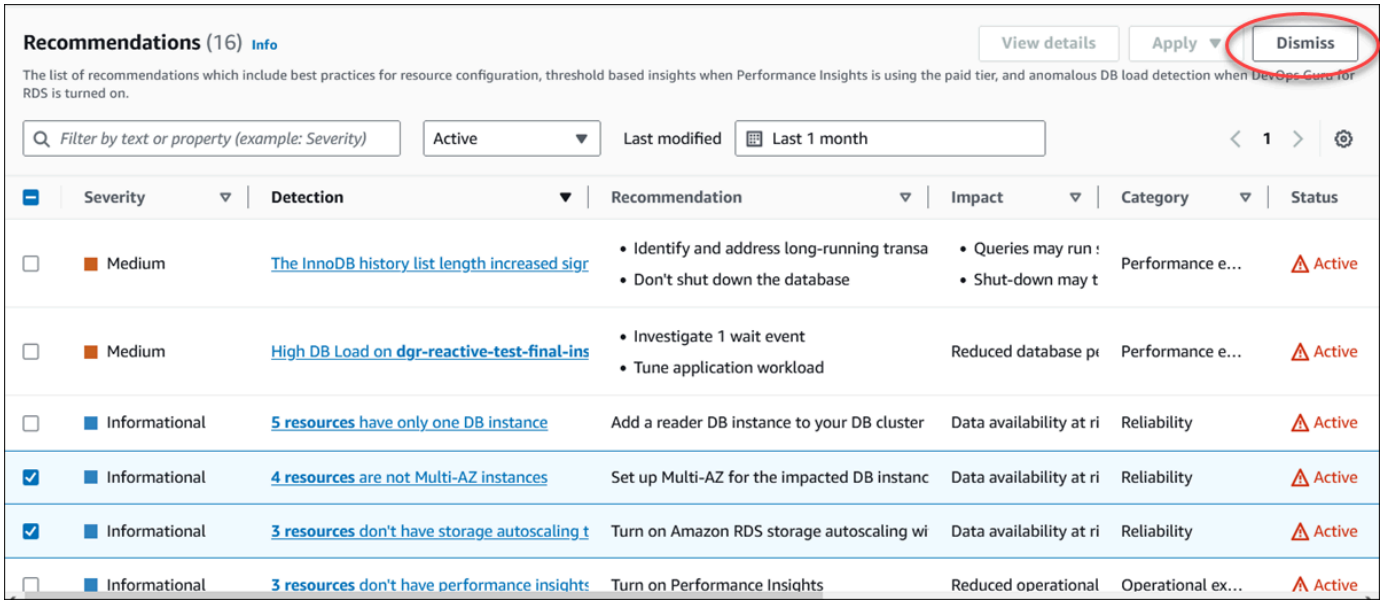
   - Choisissez Recommandations.

     La page Recommandations apparaît avec la liste de toutes les recommandations.
   - Choisissez Bases de données, puis sélectionnez Recommandations pour une ressource dans la page des bases de données.

     Les détails apparaissent dans l'onglet Recommandations de la recommandation sélectionnée.
   - Choisissez Détection pour une recommandation active sur la page Recommandations ou sur l'onglet Recommandations de la page Bases de données.

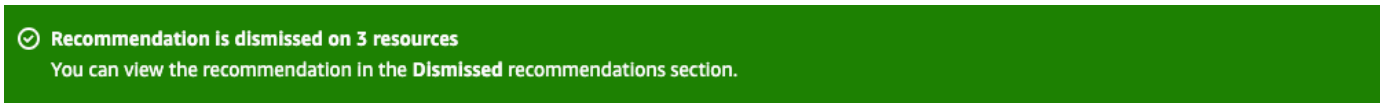
     La page de détails des recommandations affiche la liste des ressources concernées.
3. Choisissez une ou plusieurs recommandations, ou une ou plusieurs ressources concernées dans la page de détails des recommandations, puis choisissez Ignorer.

   L'exemple suivant montre la page Recommandations avec plusieurs recommandations actives sélectionnées pour être ignorées.



Une bannière affiche un message lorsque la ou les recommandations sélectionnées sont rejetées.

L'exemple suivant montre la bannière avec le message de réussite.



L'exemple suivant montre la bannière avec le message d'échec.



CLI

Pour ignorer RDSune à l'aide du AWS CLI

1. Exécutez la commande `aws rds describe-db-recommendations --filters "Name=status,Values=active"`.

   La sortie fournit une liste de recommandations en `active` état.
2. `recommendationId`Recherchez la recommandation que vous souhaitez ignorer à l'étape 1.
3. Exécutez la commande >`aws rds modify-db-recommendation --status dismissed --recommendationId <ID>` à l'aide `recommendationId` de l'étape 2 pour ignorer la recommandation.

RDS API

Pour ignorer RDSune à l'aide d'Amazon RDSAPI, utilisez l'odifyDBRecommendationopération M.

Modification des recommandations Amazon RDS rejetées en recommandations actives

Vous pouvez déplacer une ou plusieurs recommandations RDSAmazon rejetées vers des recommandations actives à l'aide de la RDS console Amazon ou d'Amazon RDSAPI. AWS CLI

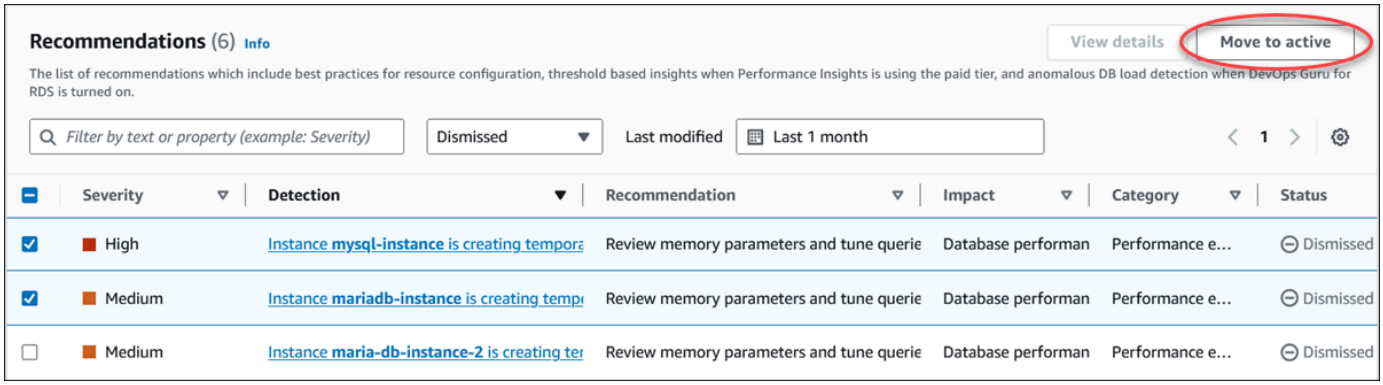
Console

Pour déplacer une ou plusieurs recommandations rejetées vers des recommandations actives

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, effectuez l'une des opérations suivantes :
   - Choisissez Recommandations.

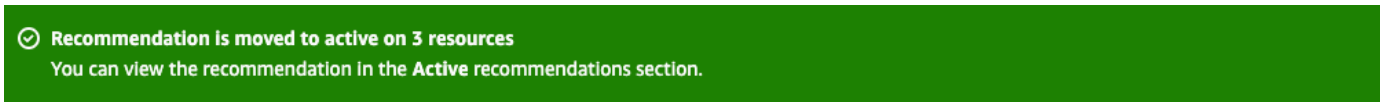
     La page Recommandations affiche une liste de recommandations triées par gravité pour toutes les ressources de votre compte.
   - Choisissez Bases de données, puis sélectionnez Recommandations pour une ressource dans la page des bases de données.

     L'onglet Recommandations affiche les recommandations et leurs détails pour la ressource sélectionnée.
3. Choisissez une ou plusieurs recommandations rejetées dans la liste, puis choisissez Déplacer vers active.



Une bannière affiche un message de réussite ou d'échec lorsque les recommandations sélectionnées passent du statut rejeté à l'état actif.

L'exemple suivant montre la bannière avec le message de réussite.



L'exemple suivant montre la bannière avec le message d'échec.



CLI

Pour remplacer une recommandation RDS rejetée par une recommandation active à l'aide du AWS CLI

1. Exécutez la commande `aws rds describe-db-recommendations --filters "Name=status,Values=dismissed"`.

   La sortie fournit une liste de recommandations en `dismissed` état.
2. `recommendationId`Recherchez la recommandation dont vous souhaitez modifier le statut à partir de l'étape 1.
3. Exécutez la commande `>aws rds modify-db-recommendation --status active --recommendationId <ID>` à `recommendationId` partir de l'étape 2 pour passer à la recommandation active.

RDS API

Pour remplacer une recommandation RDS rejetée par une recommandation active à l'aide d'Amazon RDSAPI, utilisez l'odifyDBRecommendationopération M.

Recommandations issues de la référence RDSAmazon

Amazon RDS génère des recommandations pour une ressource lorsque celle-ci est créée ou modifiée. Vous trouverez des exemples de recommandations d'Amazon RDS dans le tableau suivant.

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Le volume magnétique est utilisé	Vos instances de base de données utilisent le stockage magnétique. Le stockage magnétique n'est pas recommandé pour la plupart des instances de base de données. Choisissez un autre type de stockage : usage général (SSD) ou provisionnéIOPS.	Choisissez un autre type de stockage : usage général (SSD) ou provisionnéIOPS.	Oui	Volumes de la génération précédente dans la EC2 documentation Amazon.
Les sauvegardes automatisées des ressources sont désactivées	Les sauvegardes automatisées ne sont pas activées pour vos instances de base de données. Les sauvegardes automatisées sont recommandées car elles permettent de point-in-time restaurer vos instances de base de données.	Activez les sauvegardes automatisées avec une période de conservation allant jusqu'à 14 jours.	Oui	Activation des sauvegardes automatiques Démystifier les coûts de stockage des sauvegardes RDS Amazon sur AWS le blog de base de données
La mise à niveau de la version mineure du moteur est requise	Les ressources de votre base de données n'exécutent pas la dernière version mineure du moteur de base de	Effectuez une mise à niveau vers la dernière version du moteur.	Oui	Mise à niveau de la version du moteur d'une instance de base de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
	données. La dernière version mineure contient les derniers correctifs de sécurité et d'autres améliorations.			
La surveillance améliorée est désactivée	La surveillance améliorée n'est pas activée sur les ressources de votre base de données. La surveillance améliorée fournit des métriques de système d'exploitation en temps réel pour la surveillance et le dépannage.	Activez la surveillance améliorée.	Non	Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Le chiffrement du stockage est désactivé	Amazon RDS prend en charge le chiffrement au repos pour tous les moteurs de base de données en utilisant les clés que vous gérez dans AWS Key Management Service (AWS KMS). Sur une instance de base de données active avec RDS chiffrement Amazon, les données stockées au repos dans le stockage sont chiffrées, comme dans le cas des sauvegardes automatisées, des répliques de lecture et des instantanés. Si le chiffrement n'est pas activé lors de la création d'une instance de base de données, vous devez créer et restaurer une copie chiffrée de l'instantané déchiffré de l'instance de base de données avant	Activez le chiffrement des données au repos pour votre instance de base de données.	Oui	Sécurité dans Amazon RDS Copier un instantané de base de données pour Amazon RDS

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
	d'activer le chiffrement.			
Performance Insights est désactivé	Performance Insights surveille la charge de votre instance de base de données pour vous aider à analyser et à résoudre les problèmes de performance des bases de données. Nous vous recommandons d'activer Performance Insights.	Activer l'option Performance Insights.	Non	Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS
Le dimensionnement automatique du stockage est désactivé sur les instances de base de données	Le dimensionnement automatique du stockage n'est pas activé pour votre instance de base de données. Lorsque la charge de travail de la base RDS de données augmente, le dimensionnement automatique du stockage adapte automatiquement la capacité de stockage sans interruption de service.	Activez le dimensionnement automatique du RDS stockage Amazon avec un seuil de stockage maximal spécifié	Non	Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSressources : une mise à jour des versions majeures est requise	Les bases de données dotées de la version majeure actuelle du moteur de base de données ne seront pas prises en charge. Nous vous recommandons de passer à la dernière version majeure qui inclut de nouvelles fonctionnalités et améliorations.	Effectuez une mise à niveau vers la dernière version majeure du moteur de base de données.	Oui	Mise à niveau de la version du moteur d'une instance de base de données Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSLa mise à jour de la classe d'instanc e des ressources est requise	Votre instance de base de données exécute une classe d'instance de base de données de générati on antérieure. Nous avons remplacé les classes d'instance de base de données d'une génération précédente par des classes d'instance de base de données offrant un meilleur coût, de meilleure s performances, ou les deux. Nous vous recommand ons d'exécuter votre instance de base de données avec une classe d'instance de base de données de nouvelle génération.	Mettez à niveau la classe d'instance de base de données.	Oui	Moteurs de base de données pris en charge pour les classes d'instance de base de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSressources utilisant l'édition du moteur de fin de support sous licence incluse	Nous vous recommandons de mettre à niveau la version majeure vers la dernière version du moteur prise en charge par Amazon RDS afin de continuer à bénéficier du support de licence actuel. La version du moteur de votre base de données ne sera pas prise en charge avec la licence actuelle.	Nous vous recommandons de mettre à niveau votre base de données vers la dernière version prise en charge par Amazon RDS pour continuer à utiliser le modèle sous licence.	Oui	Mises à niveau des versions majeures d'Oracle

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Instances de base de données n'utilisant pas le déploiement multi-AZ	Nous vous recommandons d'utiliser un déploiement multi-AZ. Les déploiements multi-AZ améliorent la disponibilité et la durabilité de l'instance de base de données.	Configurer le mode Multi-AZ pour les instances de base de données concernées	Non Aucun temps d'arrêt n'a lieu pendant cette modification. Toutefois, il existe un impact possible sur les performances. Pour plus d'informations, consultez Conversion d'une instance de base de	Tarification d'Amazon RDS Multi-AZ

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
			donnée en déploiement multi-AZ pour Amazo RDS.	

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Les paramètres de mémoire de base de données divergent de ceux par défaut	Les paramètres de mémoire des instances de base de données sont significativement différents des valeurs par défaut. Ces paramètres peuvent avoir un impact sur les performances et provoquer des erreurs. Nous vous recommandons de réinitialiser les paramètres de mémoire personnalisés de l'instance de base de données à leurs valeurs par défaut dans le groupe de paramètres de base de données.	Réinitialisez les paramètres de mémoire à leurs valeurs par défaut.	Non	Bonnes pratiques en matière de configuration des paramètres de performance pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`InnoDB_Change_Buffering` paramè utilisant une valeur inférieure à la valeur optimale	La mise en mémoire tampon des modifications permet à une instance My SQL DB de différer quelques écritures, qui sont nécessaires pour maintenir les index secondaires. Cette fonctionnalité était utile dans les environnements où les disques étaient lents. La modification de la configuration de la mise en mémoire tampon a légèrement amélioré les performances de la base de données, mais a retardé la reprise après incident et a prolongé les temps d'arrêt pendant la mise à niveau. Défini `OFF` par défaut dans Ma SQL version 8.4.	Définissez la valeur `InnoDB_Change_Buffering` du paramètre sur `NONE` dans vos groupes de paramètres de base de données.	Non	Bonnes pratiques en matière de configuration des paramètres de performance pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Le paramètre de cache des requêtes est activé	Lorsque les modifications nécessitent la purge de votre cache de requêtes, votre instance de base de données semble bloquée. La plupart des charges de travail ne bénéficient pas d'un cache de requête. Le cache de requêtes a été supprimé de My SQL 8.0 et des versions supérieures. Nous vous recommandons de définir le paramètre query_cache_type sur 0.	Définissez la valeur du `query_cache_type` paramètre sur `0` dans vos groupes de paramètres de base de données.	Oui	Bonnes pratiques en matière de configuration des paramètres de performance pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`log_output` le paramètre est défini sur table	Lorsqu'il `log_output` est défini sur`TABLE`, plus d'espace de stockage est utilisé que lorsqu'il `log_output` est défini sur`FILE`. Nous vous recommandons de définir le paramètre sur`FILE`, afin d'éviter d'atteindre la limite de taille de stockage. `FILE`Paramétré sur par défaut dans MySQL 8.4 et versions supérieures.	Définissez la valeur du `log_output` paramètre sur `FILE` dans vos groupes de paramètres de base de données.	Non	Fichiers journaux de base de données MySQL

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
Groupes de paramètres n'utilisant pas de grandes pages	Les grandes pages peuvent augmenter l'évolutivité de la base de données, mais votre instance de base de données n'utilise pas de grandes pages. Nous vous recommandons de définir la valeur du `use_large_pages` paramètre sur `ONLY` dans le groupe de paramètres de base de données de votre instance de base de données.	Définissez la valeur du `use_large_pages` paramètre sur `ONLY` dans vos groupes de paramètres de base de données.	Oui	Activation de HugePages pour une instance RDS for Oracle

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`autovacuum` le paramètre est désactivé	Le paramètre autovacuum est désactivé pour les de base de données. La désactivation de l'aspirateur automatique augmente le gonflement de la table et de l'index et a un impact sur les performances. Nous vous recommandons d'activer l'autovacuum dans vos groupes de paramètres de base de données.	Activez le paramètre autovacuum dans les groupes de paramètres de votre de base de données.	Non	Comprendre l'autovacuum dans les SQL environnements Amazon RDS pour Postgre sur le blog de base de données AWS

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`synchronous_commit` le paramètre est désactivé	Lorsque `synchronous_commit` le paramètre est désactivé, des données peuvent être perdues lors d'un crash de base de données. La durabilité de la base de données est menacée. Nous vous recommandons d'activer le paramètre `synchronous_commit` .	Activez le `synchronous_commit` paramètre dans vos groupes de paramètres de base de données.	Oui	SQLParamètres Amazon Aurora Postgre : réplication, sécurité et journalisation sur le blog de AWS base de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`track_counts` le paramètre est désactivé	Lorsque le `track_counts` paramètre est désactivé, la base de données ne collecte pas les statistiques d'activité de la base de données. Autovacuum a besoin de ces statistiques pour fonctionner correctement. Nous vous recommandons de définir le paramètre `track_counts` sur 1.	Réglez `track_counts` le paramètre sur`1`.	Non	Statistiques d'exécution pour Postgre SQL
`enable_indexonlyscan` le paramètre est désactivé	Le planificateur ou l'optimiseur de requêtes ne peut pas utiliser le type de plan de scan indexé uniquement lorsqu'il est désactivé. Nous vous recommandons de définir la valeur du `enable_indexonlyscan` paramètre sur`1`.	Définissez la valeur du `enable_indexonlyscan` paramètre sur`1`.	Non	Configuration de la méthode du planificateur pour Postgre SQL

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`enable_indexscan` le paramètre est désactivé	Le planificateur ou l'optimiseur de requêtes ne peut pas utiliser le type de plan d'analyse d'index lorsqu'il est désactivé. Nous vous recommandons de définir la `enable_indexscan` valeur sur1.	Définissez la valeur du `enable_indexscan` paramètre sur1.	Non	Configuration de la méthode du planificateur pour Postgre SQL
`innodb_flush_log_at_trx` le paramètre est désactivé	La valeur du `innodb_flush_log_at_trx` paramètre de votre instance de base de données n'est pas une valeur sûre. Ce paramètre contrôle la persistance des opérations de validation sur le disque. Nous vous recommandons de définir le paramètre `innodb_flush_log_at_trx` sur 1.	Définissez la valeur du `innodb_flush_log_at_trx` paramètre sur1.	Non	Bonnes pratiques en matière de configuration des paramètres de performance pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`sync_binlog` le paramètre est désactivé	La synchronisation du journal binaire avec le disque n'est pas appliquée avant que les validations des transactions ne soient reconnues dans votre instance de base de données. Nous vous recommandons de définir la valeur du `sync_binlog` paramètre sur`1`.	Définissez la valeur du `sync_binlog` paramètre sur`1`.	Non	Bonnes pratiques pour configurer les paramètres de réplication pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`innodb_stats_persistent` le paramètre est désactivé	Votre instance de base de données n'est pas configurée pour conserver les statistiques InnoDB sur le disque. Lorsque les statistiques ne sont pas stockées, elles sont recalculées à chaque redémarrage de l'instance et à chaque accès à la table. Cela entraîne des variations dans le plan d'exécution des requêtes. Vous pouvez modifier la valeur de ce paramètre global au niveau de la table. Nous vous recommandons de définir la valeur du `innodb_stats_persistent` paramètre sur`ON`.	Définissez la valeur du `innodb_stats_persistent` paramètre sur`ON`.	Non	Bonnes pratiques en matière de configuration des paramètres de performance pour Amazon RDS for My SQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`innodb_open_files` le paramètre est faible	Le `innodb_open_files` paramètre contrôle le nombre de fichiers qu'InnoDB peut ouvrir en même temps. InnoDB ouvre tous les fichiers log et tablespace système lorsque mysqld est en cours d'exécution. Votre instance de base de données a une faible valeur pour le nombre maximal de fichiers qu'InnoDB peut ouvrir en même temps. Nous vous recommandons de définir le paramètre `innodb_open_files` sur la valeur minimale 65.	Réglez le `innodb_open_files` paramètre sur une valeur minimale de65.	Oui	InnoDB ouvre des fichiers pour My SQL

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`max_user_connections` le paramètre est faible	Votre instance de base de données a une valeur faible pour le nombre maximal de connexions simultanées pour chaque compte de base de données. Nous vous recommandons de définir le `max_user_connections` paramètre sur un nombre supérieur à`5`.	Augmentez la valeur du `max_user_connections` paramètre à un nombre supérieur à`5`.	Oui	Configuration des limites de ressources du compte pour MySQL
Les répliques de lecture sont ouvertes en mode inscriptible	Votre instance de base de données possède une réplique en lecture en mode inscriptible, qui permet les mises à jour par les clients. Nous vous recommandons de définir le `read_only` paramètre sur de `TrueIfReplica` telle sorte que les répliques lues ne soient pas en mode inscriptible.	Définissez la valeur du `read_only` paramètre sur`TrueIfReplica` .	Non	Bonnes pratiques pour configurer les paramètres de réplication pour Amazon RDS for MySQL on the AWS Database Blog

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`innodb_default_row_format` le réglage des paramètres n'est pas sûr	Votre instance de base de données rencontre un problème connu : une table créée dans une SQL version My inférieure à 8.0.26 avec le paramètre `row_format` défini sur `COMPACT` ou `REDUNDANT` sera inaccessible et irrécupérable lorsque l'index dépasse 767 octets. Nous vous recommandons de définir la valeur du `innodb_default_row_format` paramètre sur`DYNAMIC`.	Définissez la valeur du `innodb_default_row_format` paramètre sur`DYNAMIC`.	Non	Changements dans My SQL 8.0.26

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
`general_logging` le paramètre est activé	La journalisation générale est activée pour votre instance de base de données. Ce paramètre est utile pour résoudre les problèmes liés à la base de données. Cependant, l'activation de la journalisation générale augmente le nombre d'opérations d'E/S et l'espace de stockage alloué, ce qui peut entraîner des conflits et une dégradation des performances. Vérifiez vos exigences en matière d'utilisation générale de la journalisation. Nous vous recommandons de définir la valeur du `general_logging` paramètre sur`0`.	Vérifiez vos exigences en matière d'utilisation générale de la journalisation. Si ce n'est pas obligatoire, nous vous recommandons de définir la valeur du `general_logging` paramètre sur`0`.	Non	Présentation des journaux de base de données RDS for MySQL

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSInstance sous-provisionnée pour la capacité de mémoire du système	Nous vous recommandons de régler vos requêtes de manière à utiliser moins de mémoire ou d'utiliser un type d'instance de base de données avec une plus grande quantité de mémoire allouée. Lorsque la mémoire de l'instance est insuffisante, les performances de la base de données sont affectées.	Utiliser une instance de base de données avec une capacité de mémoire supérieure	Oui	Mise à l'échelle verticale et horizontale de votre RDS instance Amazon sur le blog de AWS base de données Types d'RDSInstances Amazon RDSTarification Amazon

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSInstance sous-provisionnée pour la capacité du système CPU	Nous vous recommandons de régler vos requêtes pour utiliser moins CPU ou de modifier votre instance de base de données pour utiliser une classe d'instance de base de données avec une allocation plus élevéevCPUs. Les performances de la base de données peuvent diminuer lorsqu'une instance de base de données est insuffisanteCPU.	Utiliser une instance de base de données avec une CPU capacité supérieure	Oui	Mise à l'échelle verticale et horizontale de votre RDS instance Amazon sur le blog de AWS base de données Types d'RDSInstances Amazon RDSTarification Amazon

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSles ressources n'utilisent pas correctement le regroupement de connexions	Nous vous recommandons d'activer Amazon RDS Proxy pour regrouper et partager efficacement les connexions de base de données existantes. Si vous utilisez déjà un proxy pour votre base de données, configurez-le correctement pour améliorer le regroupement des connexions et l'équilibrage de charge entre plusieurs instances de base de données. RDS Le proxy peut contribuer à réduire le risque d'épuisement des connexions et d'interruption de service tout en améliorant la disponibilité et l'évolutivité.	Activez le RDS proxy ou modifiez votre configuration de proxy existante	Non	Mise à l'échelle verticale et horizontale de votre RDS instance Amazon sur le blog de AWS base de données Proxy Amazon RDS Tarification d'Amazon RDS Proxy

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSles instances créent un nombre excessif d'objets temporaires	Nous vous recommandons d'ajuster votre charge de travail pour éviter de créer des objets temporaires excessifs, ou de passer à des classes d'RDSinstances prenant en charge des lectures optimisées. RDS Les lectures optimisées améliorent les performances des bases de données pour les charges de travail impliquant un grand nombre d'objets temporaires et/ou de grands objets temporaires. Évaluez votre charge de travail pour déterminer si l'utilisation d'une instance avec des lectures RDS optimisées est bénéfique pour la charge de travail de votre base de données.	Utiliser un type d'instance de base de données avec des lectures RDS optimisées	Oui	Types d'RDSinstances Amazon Améliorer les performances des requêtes RDS pour My SQL grâce à Amazon RDS Optimized Reads Améliorer les performances des requêtes RDS pour MariaDB avec RDS Amazon Optimized Reads Améliorer les performances des requêtes RDS pour Postgre SQL avec Amazon RDS Optimized Reads

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSles instances sont sous-provisionnées pour ce qui est de la capacité du système IOPS	Nous vous recommandons de régler la charge de travail de la base de données pour réduire IOPS ou augmenter l'instance de base de données vers un type avec une IOPS limite par défaut plus élevée. L'instance de base de données actuelle ne peut pas prendre en charge le provisionnéIOPS, ou la charge de travail de la base de données est très utiliséeIOPS.	Utiliser un type d'instance de base de données avec des IOPS limites par défaut plus élevées	Oui	Types d'RDSInstances Amazon Stockage d'instance Amazon RDS DB Charge de base de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSles instances ont sous-provisionné les volumes Amazon EBS	Nous vous recommandons de régler la charge de travail de la base de données afin de réduire IOPS ou d'augmenter le montant provisionné IOPS pour la base de données. Lorsque le IOPS taux d'utilisation approche du niveau ProvisionnéIOPS, les performances de la base de données peuvent diminuer.	Provisionner davantage IOPS pour l'instance de base de données	Oui	Types d'RDSinstances Amazon Stockage d'instance Amazon RDS DB Charge de base de données
RDSles instances sont sous-provisionnées en termes de capacité de débit	Nous recommandons de régler la charge de travail de la base de données afin de réduire le débit ou d'augmenter le débit provisionné pour la base de données. Lorsque l'utilisation du débit approche le débit provisionné, les performances de la base de données peuvent être affectées .	Fournir un débit accru pour l'instance de base de données	Oui	Types d'RDSinstances Amazon Stockage d'instance Amazon RDS DB Charge de base de données

Type	Description	Recommandation	Temps d'arrêt requis	Informations supplémentaires
RDSles instances sont sous-provisionnées pour les E/S EBS	Nous vous recommandons de régler la charge de travail de la base de données afin de réduire les opération s d'E/S ou de modifier l'instance de base de données pour utiliser les volumes Amazon RDS io2 Block Express conçus pour les charges de travail de base de données nécessita nt des performan ces élevées, un débit élevé et une faible latence. Avec la charge de travail actuelle, la base de données risque de ne pas être en mesure de traiter les opérations d'E/ S au rythme requis, ce qui peut entraîner une dégradation des performances.	Utiliser les volumes Amazon RDS io2 Block Express pour l'instance RDS	Non	Stockage d'instance Amazon RDS DB CloudWatchMétriques Amazon pour Amazon RDS IOPSSSDVolumes provisionnés dans le guide de l'EBSutil isateur Amazon

Afficher les métriques dans la RDS console Amazon

Amazon RDS s'intègre CloudWatch à Amazon pour afficher diverses métriques d'instances de base de données du RDS cluster dans la RDS console. Pour obtenir des descriptions de ces métriques, voir Référence des métriques pour Amazon RDS.

Pour votre instance de base de données, les catégories de métriques suivantes sont surveillées :

- CloudWatch— Affiche les CloudWatch métriques Amazon pour auxquelles vous pouvez accéder dans la RDS console. Vous pouvez également accéder à ces métriques dans la CloudWatch console. Chaque métrique inclut un graphique affichant la métrique supervisée sur une période donnée. Pour obtenir la liste des CloudWatch indicateurs, consultez CloudWatch Métriques Amazon pour Amazon RDS.
- Surveillance améliorée : affiche un résumé des métriques du système d'exploitation lorsque votre instance de base de données du cluster de base de RDS données a activé la surveillance améliorée. RDSfournit les statistiques issues de la surveillance améliorée à votre compte Amazon CloudWatch Logs. Chaque métrique du système d'exploitation comprend un graphique montrant la métrique surveillée sur un intervalle spécifique. Pour avoir une présentation, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée. Pour obtenir une liste des métriques de la surveillance améliorée, veuillez consulter Métriques du système d'exploitation dans la surveillance améliorée.
- Liste de processus du système d'exploitation : affiche les détails de chaque processus s'exécutant dans votre instance de base de données.
- Performance Insights — Ouvre le tableau de bord Amazon RDS Performance Insights pour une instance de base de données . Pour une présentation de Performance Insights, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS. Pour obtenir une liste des métriques de Performance Insights, veuillez consulter Statistiques Amazon CloudWatch pour Amazon RDS Performance Insights.

Amazon fournit RDS désormais une vue consolidée des CloudWatch statistiques et statistiques de Performance Insights dans le tableau de bord Performance Insights. Performance Insights doit être activé pour que votre instance de base de données puisse utiliser cette vue. Vous pouvez choisir la nouvelle vue de surveillance dans l'onglet Surveillance ou Performance Insights dans le volet de navigation. Pour consulter les instructions relatives au choix de cette vue, consultez Afficher les indicateurs combinés avec le tableau de bord Performance Insights.

Afficher les indicateurs combinés avec le tableau de bord Performance Insights

> **Important**
>
> AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.
>
> Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).
>
> Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
>
> Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Amazon RDS fournit une vue consolidée des statistiques de performance et des CloudWatch indicateurs de votre instance de base de données dans le tableau de bord Performance Insights. Vous pouvez utiliser le tableau de bord préconfiguré ou créer un tableau de bord personnalisé.
Le tableau de bord préconfiguré fournit les métriques les plus couramment utilisées pour aider à diagnostiquer les problèmes de performances d'un moteur de base de données. Sinon, vous pouvez créer un tableau de bord personnalisé avec les métriques pour un moteur de base de données qui

répond à vos exigences en matière d'analyse. Utilisez ensuite ce tableau de bord pour toutes les instances de base de données de ce type de moteur de base de données dans votre AWS compte.

Vous pouvez choisir la vue de surveillance dans l'onglet Monitoring ou Performance Insights dans le volet de navigation.

Performance Insights doit être activé pour votre instance de base de données pour que vous puissiez afficher les métriques combinées dans le tableau de bord Performance Insights. Pour plus d'informations sur l'activation de Performance Insights, consultez Activer et désactiver Performance Insights pour Amazon RDS.

Dans les sections suivantes, vous apprendrez à afficher les CloudWatch statistiques et les statistiques relatives aux Performances.

Rubriques

- Choix de la nouvelle vue de surveillance dans l'onglet Surveillance
- Choix de la nouvelle vue de surveillance sur la page Performance Insights
- Création d'un tableau de bord personnalisé avec Performance Insights
- Choix du tableau de bord préconfiguré avec Performance Insights

Choix de la nouvelle vue de surveillance dans l'onglet Surveillance

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec

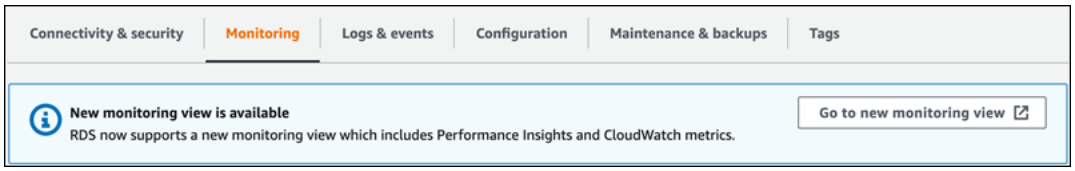
le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Depuis la console Amazon RDS, vous pouvez choisir la nouvelle vue de surveillance pour consulter les Performances Insights et CloudWatch les métriques de votre instance de base de données.

Pour choisir la nouvelle vue de surveillance dans l'onglet Surveillance

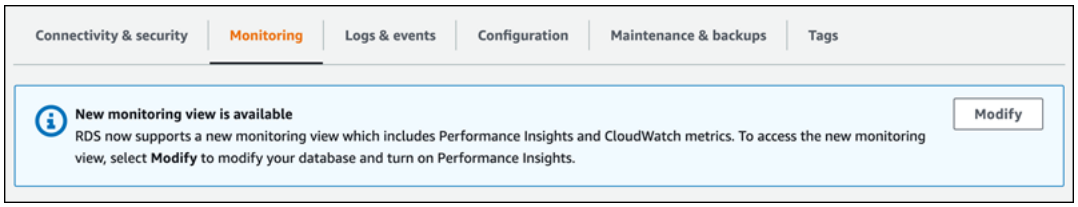
1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation de gauche, sélectionnez Bases de données.
3. Sélectionnez l'instance de base de données que vous souhaitez surveiller.
4. Faites défiler vers le bas et choisissez l'onglet Surveillance.

   Une bannière apparaît avec l'option permettant de choisir la nouvelle vue de surveillance. L'exemple suivant montre la bannière pour choisir la nouvelle vue de surveillance.



5. Choisissez Accéder à une nouvelle vue de surveillance pour ouvrir le tableau de bord Performance Insights contenant des informations sur les performances et des CloudWatch métriques pour votre instance de base de données de de base de données.
6. (Facultatif) Si Performance Insights est désactivé pour votre instance de base de données, une bannière apparaît avec la possibilité de modifier votre cluster de bases de données et d'activer Performance Insights.

L'exemple suivant montre la bannière permettant de modifier le cluster de bases de données dans l'onglet Surveillance.



Choisissez Modifier pour modifier votre cluster de bases de données et activer Performance Insights. Pour plus d'informations sur l'activation de Performance Insights, consultez Activer et désactiver Performance Insights pour Amazon RDS.

Choix de la nouvelle vue de surveillance sur la page Performance Insights

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.

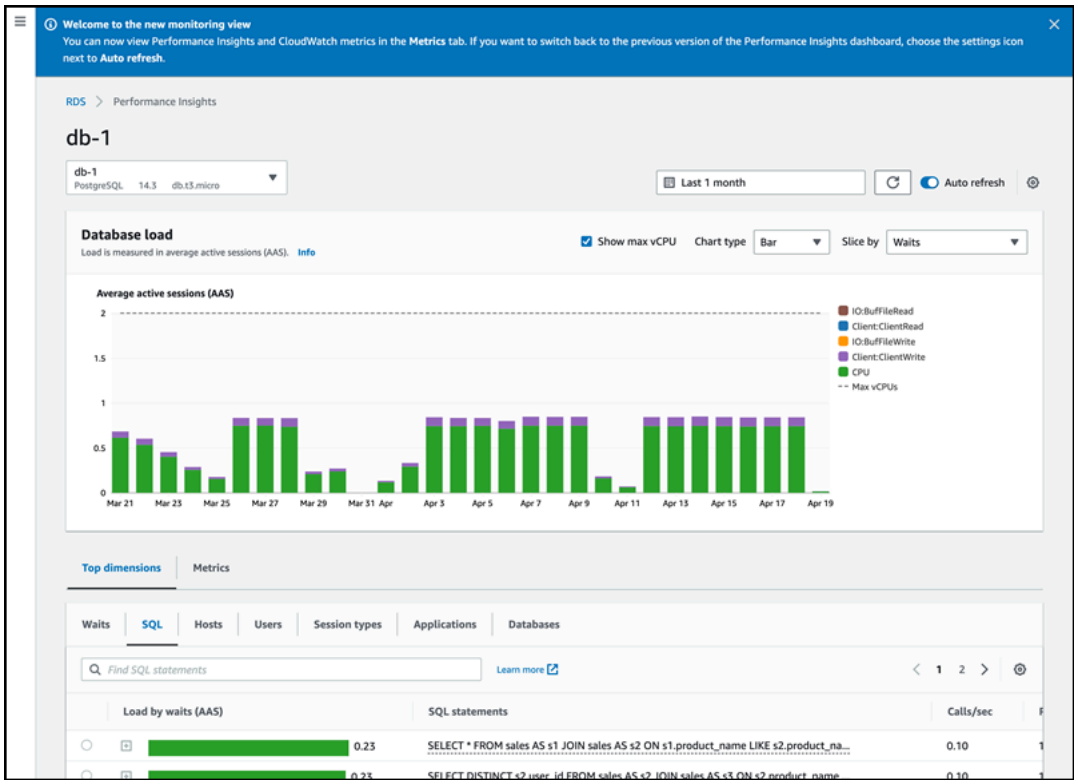
Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance

à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Depuis la console Amazon RDS, vous pouvez choisir la nouvelle vue de surveillance pour consulter les Performances Insights et CloudWatch les métriques de votre instance de base de données.

Pour choisir la nouvelle vue de surveillance avec Performance Insights dans le volet de navigation

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données pour consulter le tableau de bord Performance Insights qui affiche à la fois les statistiques de performance et CloudWatch les métriques de votre instance de base de données.



Création d'un tableau de bord personnalisé avec Performance Insights

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.

Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Dans la nouvelle vue de surveillance, vous pouvez créer un tableau de bord personnalisé avec les métriques dont vous avez besoin pour répondre à vos exigences d'analyse.

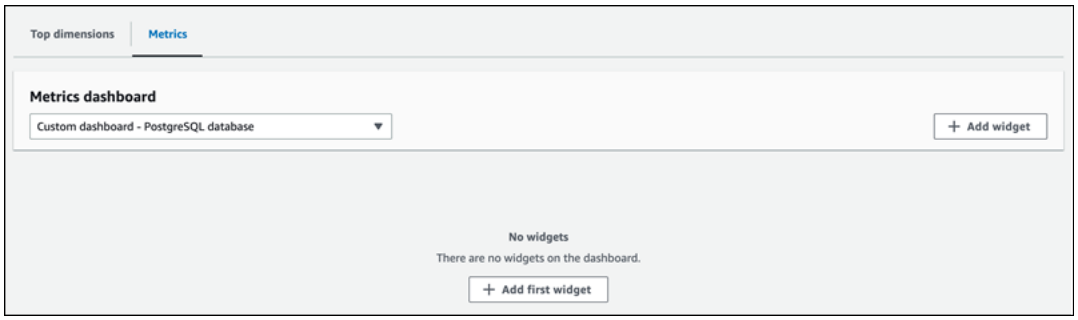
Vous pouvez créer un tableau de bord personnalisé en sélectionnant Performance Insights et CloudWatch les métriques pour votre instance de base de données. Vous pouvez utiliser ce tableau de bord personnalisé pour d'autres instances de base de données du même type de moteur de base de données dans votre AWS compte.

> **Note**
>
> Le tableau de bord personnalisé prend en charge jusqu'à 50 métriques.

Utilisez le menu des paramètres du widget pour modifier ou supprimer le tableau de bord et pour déplacer ou redimensionner la fenêtre du widget.

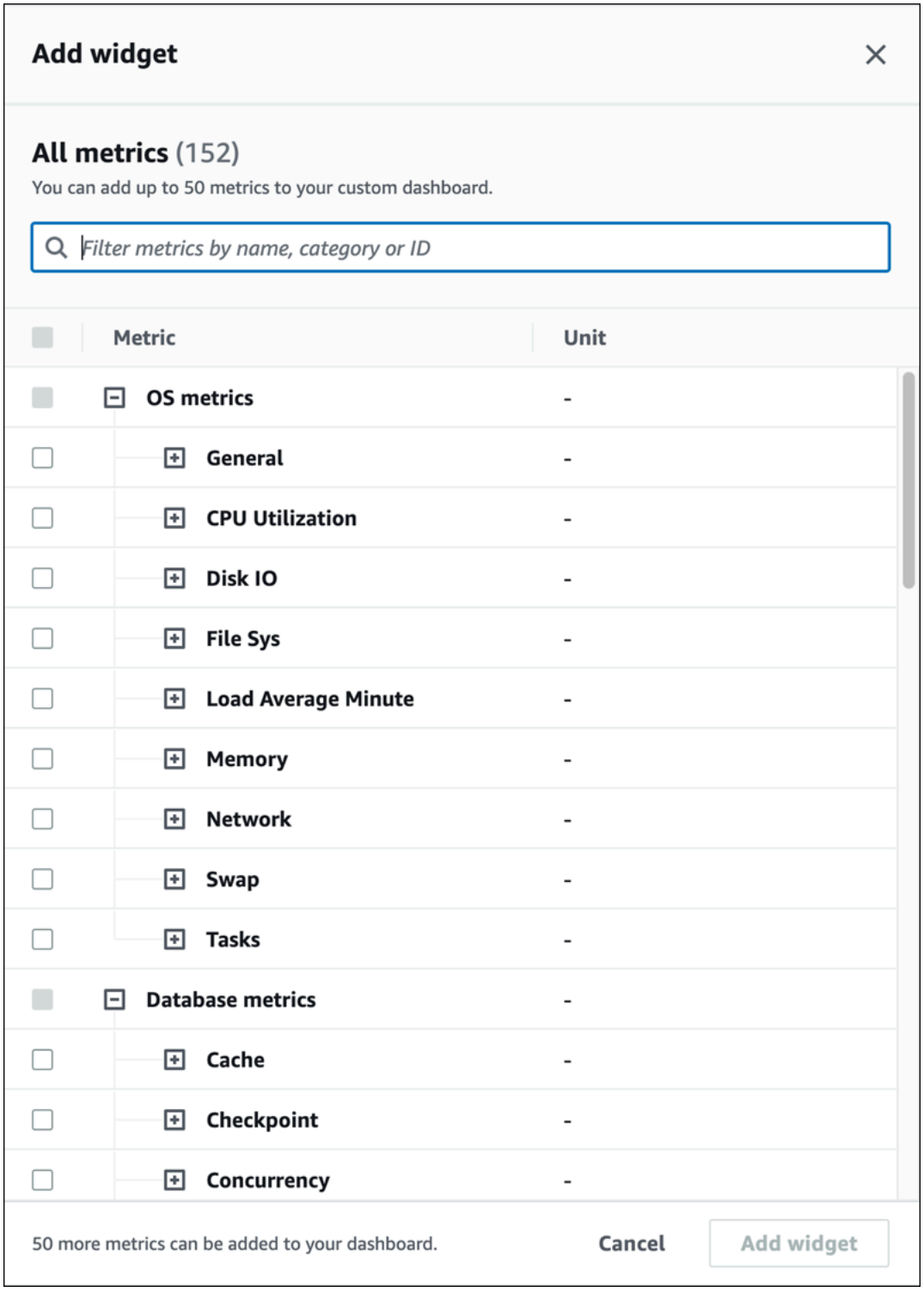
Pour créer un tableau de bord personnalisé avec Performance Insights dans le volet de navigation

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.
4. Faites défiler la fenêtre vers le bas jusqu'à l'onglet Métriques.
5. Sélectionnez le tableau de bord personnalisé dans la liste déroulante. L'exemple suivant montre la création du tableau de bord personnalisé.



6. Choisissez Ajouter un widget pour ouvrir la fenêtre Ajouter un widget. Vous pouvez ouvrir et afficher les métriques du système d'exploitation (OS), les métriques de base de données et CloudWatch les métriques disponibles dans la fenêtre.

   L'exemple suivant montre la fenêtre Ajouter un widget avec les métriques.



7. Sélectionnez les métriques que vous souhaitez afficher dans le tableau de bord et sélectionnez Ajouter un widget. Vous pouvez utiliser le champ de recherche pour trouver une métrique spécifique.

   Les métriques sélectionnées s'affichent dans votre tableau de bord.

8. (Facultatif) Si vous souhaitez modifier ou supprimer votre tableau de bord, choisissez l'icône des paramètres en haut à droite du widget, puis sélectionnez l'une des actions suivantes dans le menu.

   - Modifier : modifiez la liste des métriques dans la fenêtre. Sélectionnez Mettre à jour le widget après avoir sélectionné les métriques pour votre tableau de bord.
   - Supprimer : supprime le widget. Sélectionnez Supprimer dans la fenêtre de confirmation.

Choix du tableau de bord préconfiguré avec Performance Insights

> **Important**
>
> AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.
> Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).
> Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
> Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

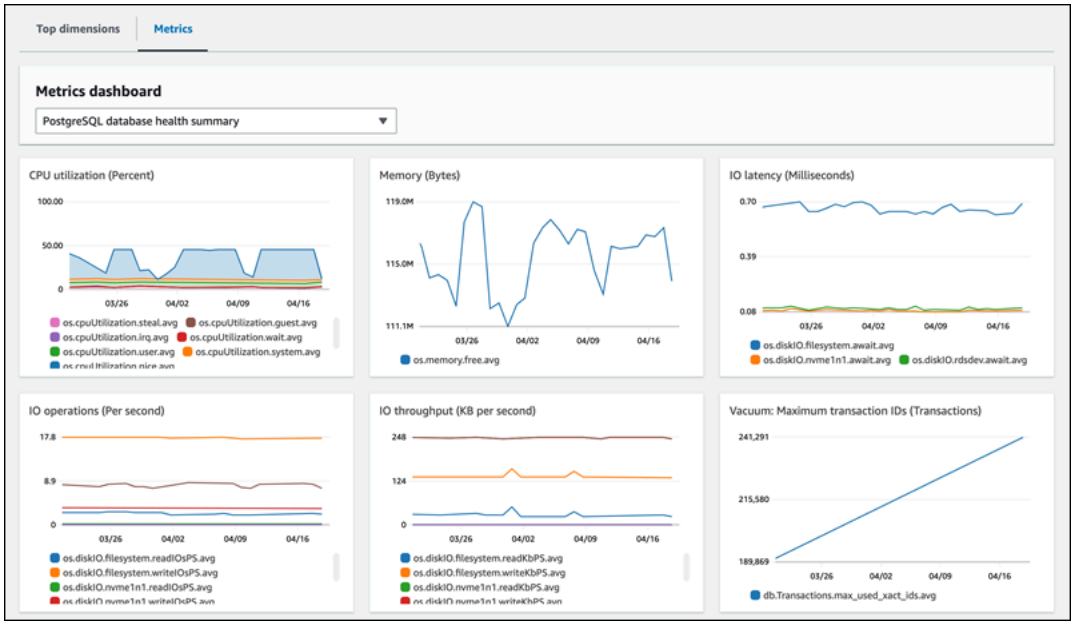
Le tableau de bord préconfiguré vous permet d'afficher les métriques les plus couramment utilisées. Ce tableau de bord permet de diagnostiquer les problèmes de performances à l'aide d'un moteur de base de données et de réduire le temps de restauration moyen de quelques heures à quelques minutes.

> **Note**
>
> Ce tableau de bord ne peut pas être modifié.

Pour choisir le tableau de bord préconfiguré avec Performance Insights dans le volet de navigation

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.
4. Faites défiler la fenêtre vers le bas jusqu'à l'onglet Métriques.
5. Sélectionnez un tableau de bord préconfiguré dans la liste déroulante.

   Vous pouvez afficher les métriques pour l'instance de base de données dans le tableau de bord. L'exemple suivant présente un tableau de bord de métriques préconfiguré.



Surveillance des métriques Amazon RDS () avec Amazon CloudWatch

Amazon CloudWatch est un référentiel de statistiques. Le référentiel collecte et traite les données brutes de Amazon RDS en métriques lisibles et disponibles presque en temps réel. Pour obtenir la liste complète des métriques Amazon RDS envoyées à .

Pour analyser et résoudre les problèmes de performances de vos bases de données à grande échelle, utilisez CloudWatch Database Insights .

Rubriques

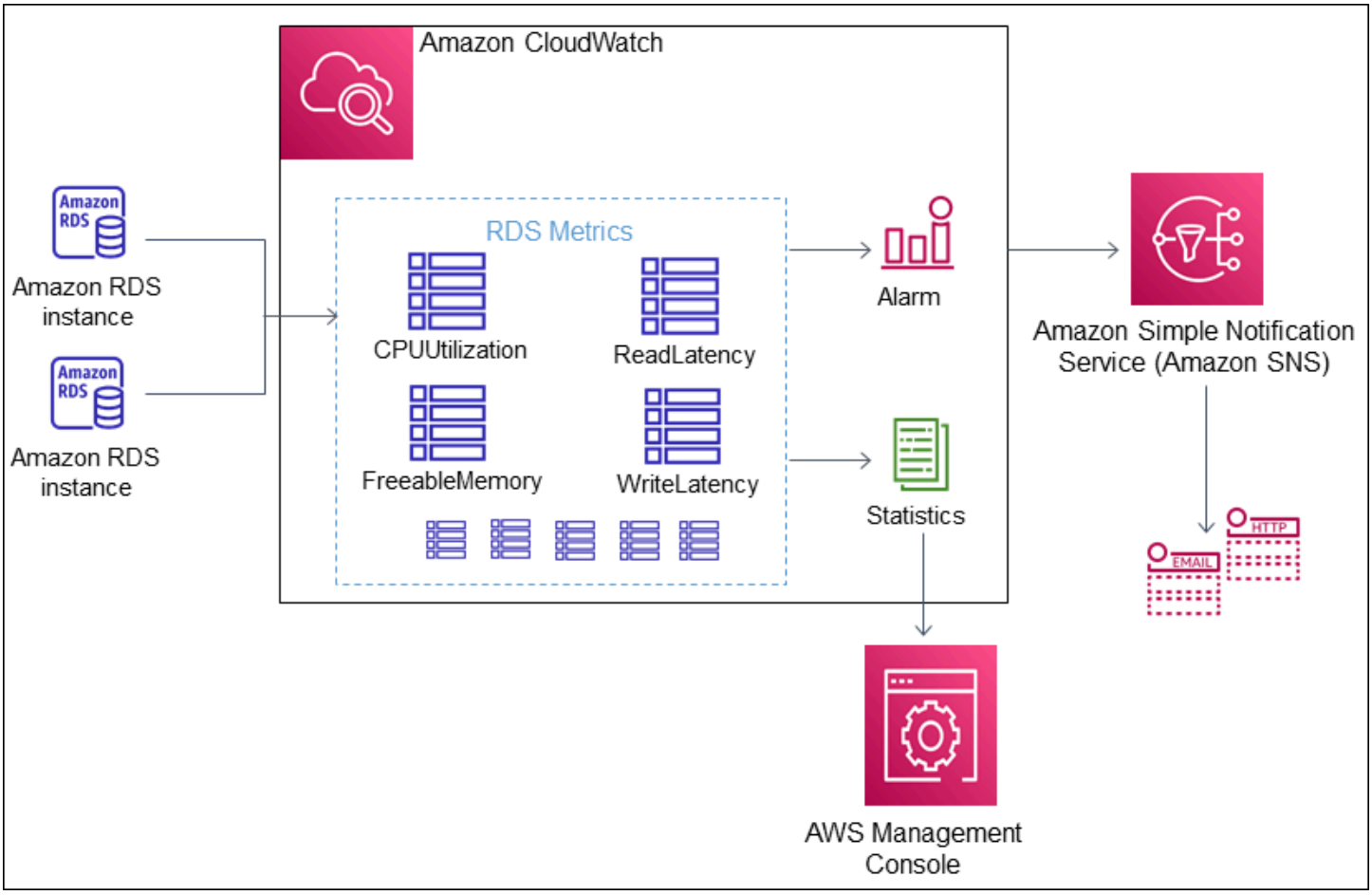
- Présentation d'Amazon RDS, d' et d'Amazon CloudWatch
- Affichage des métriques instances de base de données dans la CloudWatch console et AWS CLI
- Exportation des indicateurs de Performance Insights vers CloudWatch
- Création d' CloudWatch alarmes pour surveiller Amazon RDS
- Tutoriel : Création d'une CloudWatch alarme Amazon pour le retard de réplication d'un cluster de bases de données multi-AZ pour Amazon RDS

Présentation d'Amazon RDS, d' et d'Amazon CloudWatch

Par défaut, Amazon RDS () envoie automatiquement des données métriques par intervalles CloudWatch d'une minute. Par exemple, la métrique `CPUUtilization` enregistre le pourcentage d'utilisation du CPU pour une instance de base de données au fil du temps. Les points de données d'une durée de 60 secondes (1 minute) sont disponibles pendant 15 jours Cela signifie que pouvez accéder aux informations historiques et voir la façon dont votre service ou application web s'exécute.

Vous pouvez désormais exporter les tableaux de bord des métriques Performance Insights d'Amazon RDS vers Amazon. CloudWatch Vous pouvez exporter les tableaux de bord de métriques préconfigurés ou personnalisés sous forme de nouveau tableau de bord ou les ajouter à un tableau de bord existant CloudWatch . Le tableau de bord exporté peut être consulté dans la CloudWatch console. Pour plus d'informations sur la manière d'exporter les tableaux de bord des métriques Performance Insights vers CloudWatch, consultezExportation des indicateurs de Performance Insights vers CloudWatch.

Comme le montre le schéma suivant, vous pouvez configurer des alarmes pour vos CloudWatch métriques. Par exemple, vous pouvez créer une alarme qui signale que l'utilisation du CPU d'une instance est supérieure à 70 %. Vous pouvez configurer Amazon Simple Notification Service pour vous envoyer un e-mail lorsque le seuil est dépassé.



Amazon RDS publie les types de statistiques suivants sur Amazon CloudWatch :

- Métriques pour vos instances de base de données RDS

  Pour obtenir un tableau de ces métriques, consultez CloudWatch Métriques Amazon pour Amazon RDS.

- Métriques de Performance Insights

  Pour obtenir un tableau de ces métriques, consultez Statistiques Amazon CloudWatch pour Amazon RDS Performance Insights et Métrique de compteur de Performance Insights.

- Mesures de surveillance améliorées (publiées sur Amazon CloudWatch Logs)

  Pour obtenir un tableau de ces métriques, consultez Métriques du système d'exploitation dans la surveillance améliorée.

- Mesures d'utilisation pour les quotas de service Amazon RDS dans votre Compte AWS

Pour obtenir un tableau de ces métriques, consultez Mesures CloudWatch d'utilisation d'Amazon pour Amazon RDS (Aurora). Pour plus d'informations sur les quotas Amazon RDS, consultez Quotas et contraintes pour Amazon RDS.

Pour plus d'informations CloudWatch, consultez Qu'est-ce qu'Amazon CloudWatch ? dans le guide de CloudWatch l'utilisateur Amazon. Pour plus d'informations sur la rétention CloudWatch des métriques, consultez la section Conservation des métriques.

Affichage des métriques instances de base de données dans la CloudWatch console et AWS CLI

Vous trouverez ci-dessous des informations sur la manière d'afficher les métriques de votre instance de base de données à l'aide de CloudWatch. Pour plus d'informations sur la surveillance des métriques du système d'exploitation de votre instance de base de données en temps réel à l'aide CloudWatch des journaux, consultezSurveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Lorsque vous utilisez les ressources RDS , RDSAmazon envoie des métriques et des dimensions à Amazon CloudWatch toutes les minutes.

Vous pouvez désormais exporter les tableaux de bord des métriques Performance Insights d'Amazon RDS vers Amazon CloudWatch et consulter ces statistiques dans la CloudWatch console. Pour plus d'informations sur la manière d'exporter les tableaux de bord des métriques Performance Insights vers CloudWatch, consultezExportation des indicateurs de Performance Insights vers CloudWatch.

Utilisez les procédures suivantes pour afficher les métriques d'Amazon RDS dans la CloudWatch console etCLI.

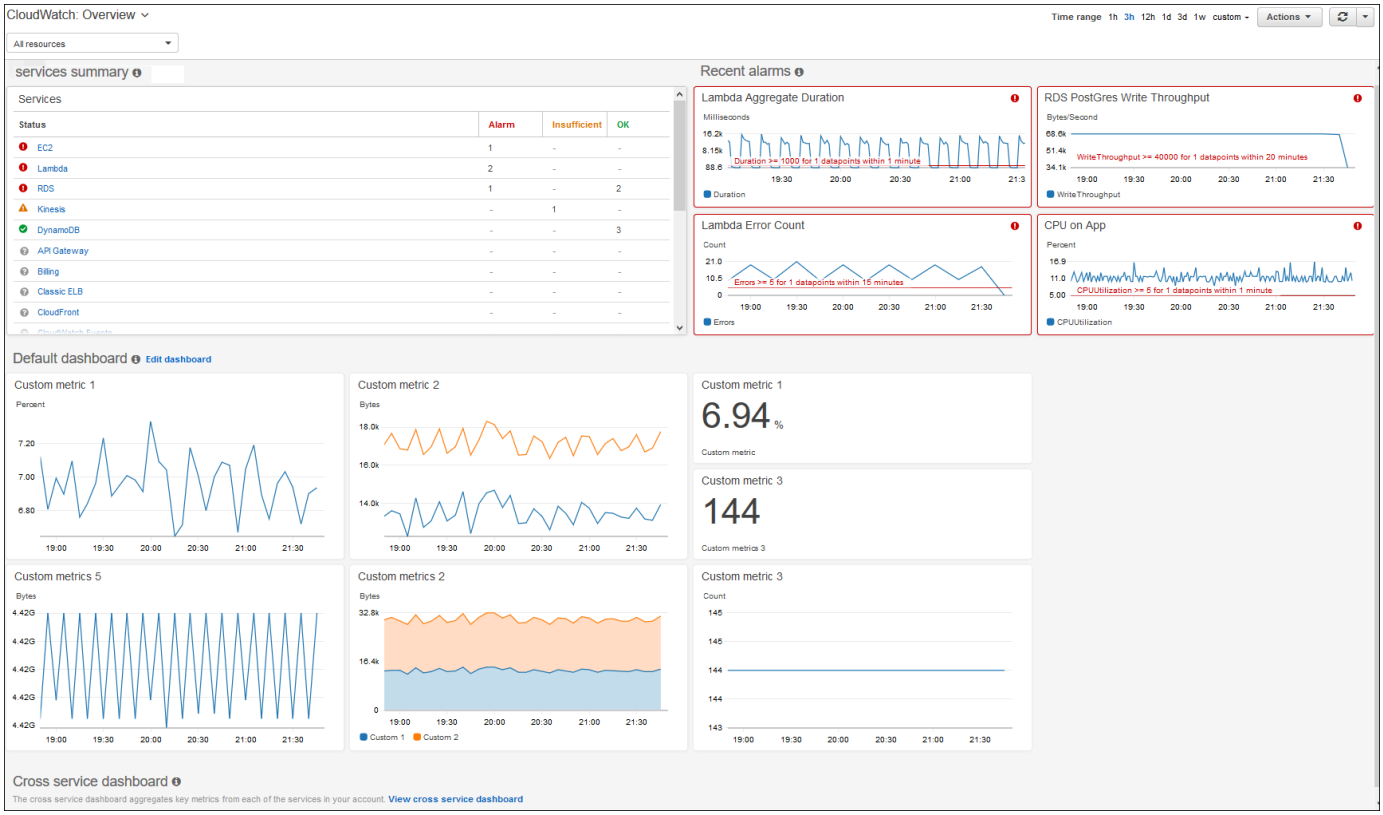
Console

Pour consulter les métriques à l'aide de la CloudWatch console Amazon

Les métriques sont d'abord regroupées par espace de noms de service, puis par les différentes combinaisons de dimension au sein de chaque espace de noms.

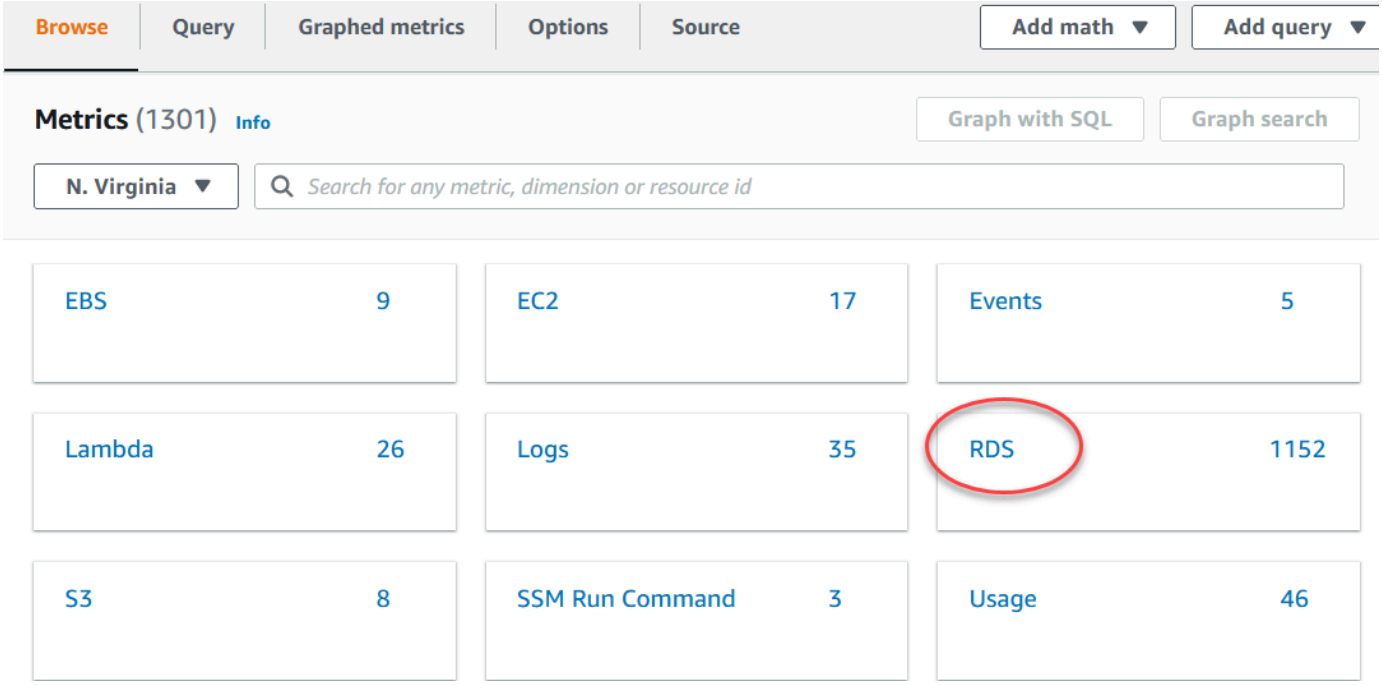
1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.

   La page CloudWatch d'accueil de la vue d'ensemble apparaît.



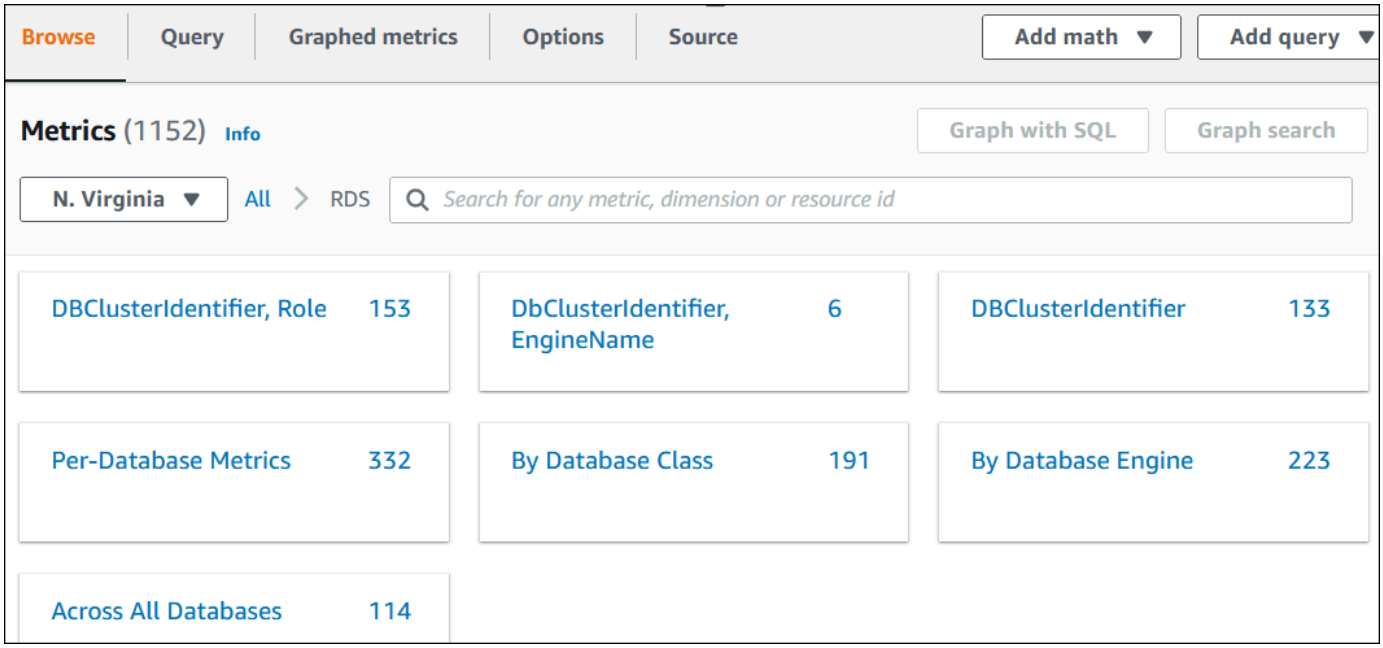
2. Si nécessaire, changez la Région AWS. Dans la barre de navigation, choisissez Région AWS où se trouvent vos AWS ressources. Pour de plus amples informations, veuillez consulter Régions et points de terminaison.

3. Dans le panneau de navigation, choisissez Metrics (Métriques), All metrics (Toutes les métriques).

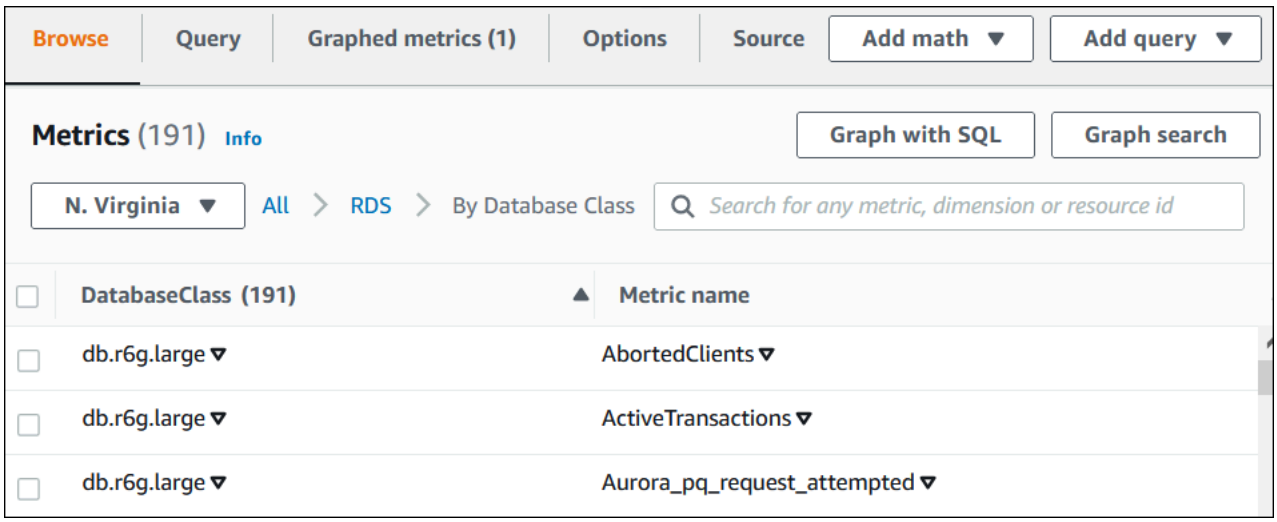


4. Faites défiler la page vers le bas et choisissez l'espace de noms des RDSmétriques.

   La page affiche les dimensions d'RDS. Pour une liste complète de ces dimensions, veuillez consulter CloudWatch Dimensions Amazon pour Amazon RDS.



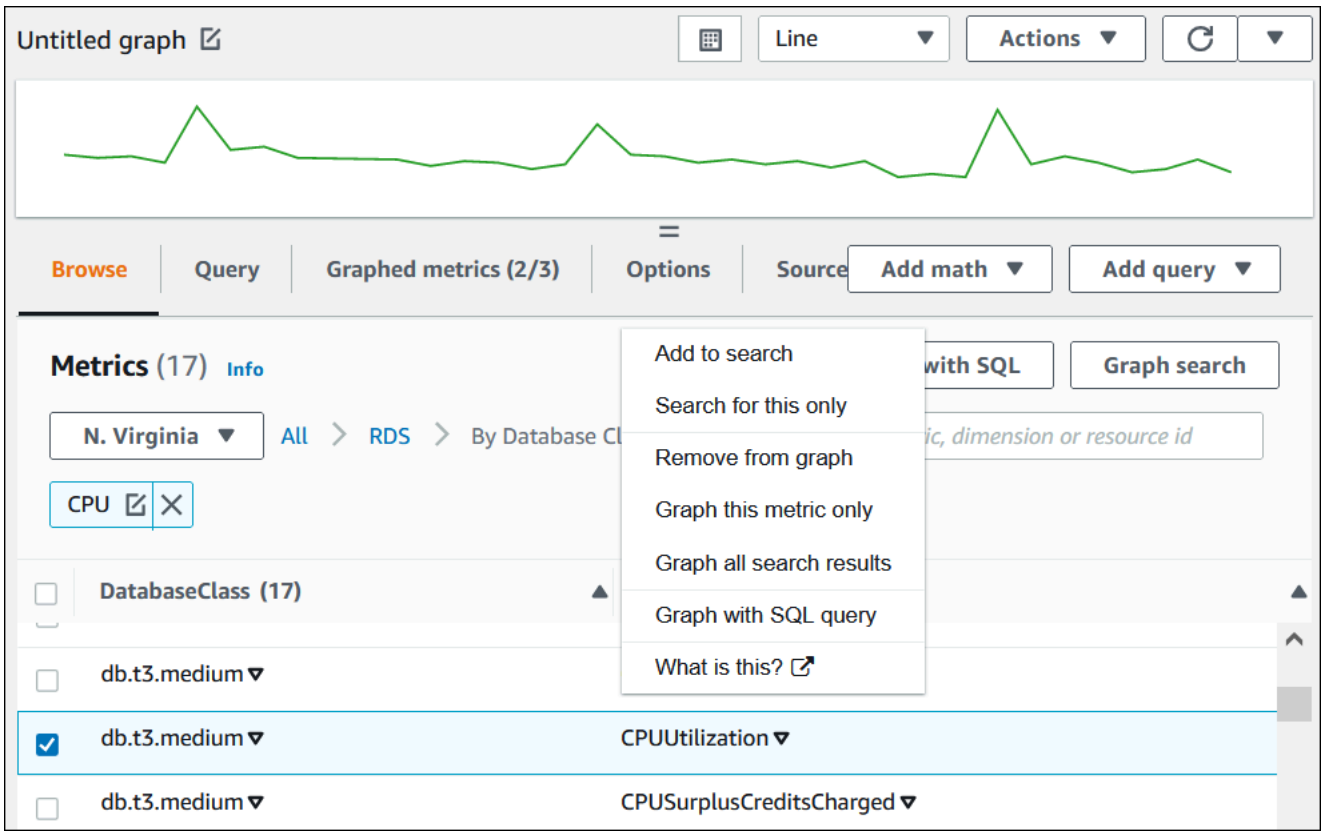
5. Sélectionnez une dimension de métrique, par exemple By Database Class (Par classe de base de données).



6. Effectuez l'une des actions suivantes :

   - Pour trier les métriques, utilisez l’en-tête de colonne.
   - Pour représenter graphiquement une métrique, cochez la case en regard de la métrique.
   - Pour filtrer par ressource, sélectionnez l'ID de ressource, puis Add to search (Ajouter à la recherche).
   - Pour filtrer par métrique, choisissez le nom de la métrique, puis Ajouter à la recherche.

   L'exemple suivant filtre la classe db.t3.medium et représente graphiquement la métrique. CPUUtilization



AWS CLI

Pour obtenir des informations métriques à l'aide de AWS CLI, utilisez la CloudWatch commande `list-metrics`. Dans l'exemple indiqué ci-dessous, vous répertoriez toutes les métriques dans l'espace de noms `AWS/RDS`.

```
aws cloudwatch list-metrics --namespace AWS/RDS
```

Pour obtenir des données métriques, utilisez la commande `get-metric-data`.

L'exemple suivant permet d'obtenir `CPUUtilization` des statistiques par exemple `my-instance` sur une période spécifique de 24 heures, avec une granularité de 5 minutes.

Créez un JSON fichier `CPU_metric.json` avec le contenu suivant.

```
{
   "StartTime" : "2023-12-25T00:00:00Z",
   "EndTime" : "2023-12-26T00:00:00Z",
   "MetricDataQueries" : [{
     "Id" : "cpu",
```

```
     "MetricStat" : {
    "Metric" : {
        "Namespace" : "AWS/RDS",
        "MetricName" : "CPUUtilization",
        "Dimensions" : [{ "Name" : "DBInstanceIdentifier" , "Value" : my-instance}]
    },
       "Period" : 360,
       "Stat" : "Minimum"
     }
   }]
}
```

Example

Pour LinuxmacOS, ou Unix :

```
aws cloudwatch get-metric-data \
    --cli-input-json file://CPU_metric.json
```

Dans Windows :

```
aws cloudwatch get-metric-data ^
     --cli-input-json file://CPU_metric.json
```

L'exemple de sortie apparaît comme suit :

```
{
    "MetricDataResults": [
        {
            "Id": "cpu",
            "Label": "CPUUtilization",
            "Timestamps": [
                "2023-12-15T23:48:00+00:00",
                "2023-12-15T23:42:00+00:00",
                "2023-12-15T23:30:00+00:00",
                "2023-12-15T23:24:00+00:00",
                ...
            ],
            "Values": [
                13.299778337027714,
                13.677507543049558,
                14.24976250395827,
```

```
                13.02521708695145,
                ...
            ],
            "StatusCode": "Complete"
        }
    ],
    "Messages": []
}
```

Pour plus d'informations, consultez la section Obtenir des statistiques pour une métrique dans le guide de CloudWatch l'utilisateur Amazon.

Exportation des indicateurs de Performance Insights vers CloudWatch

Performance Insights vous permet d'exporter le tableau de bord des métriques préconfiguré ou personnalisé de votre instance de base de données vers Amazon CloudWatch. Vous pouvez exporter le tableau de bord des métriques en tant que nouveau tableau de bord ou l'ajouter à un CloudWatch tableau de bord existant. Lorsque vous choisissez d'ajouter le tableau de bord à un tableau de CloudWatch bord existant, vous pouvez créer une étiquette d'en-tête afin que les statistiques apparaissent dans une section distincte du CloudWatch tableau de bord.

Vous pouvez consulter le tableau de bord des métriques exportées dans la CloudWatch console. Si vous ajoutez de nouvelles mesures à un tableau de bord de statistiques Performance Insights après l'avoir exporté, vous devez à nouveau exporter ce tableau de bord pour afficher les nouvelles mesures dans la CloudWatch console.

Vous pouvez également sélectionner un widget de mesures dans le tableau de bord Performance Insights et consulter les données de mesures dans la CloudWatch console.

Pour plus d'informations sur l'affichage des métriques dans la CloudWatch console, consultezAffichage des métriques instances de base de données dans la CloudWatch console et AWS CLI.

Dans les sections suivantes, exportez les métriques Performance Insights vers CloudWatch un tableau de bord nouveau ou existant et affichez les métriques Performance Insights dans CloudWatch.

Rubriques

- Exportation des métriques Performance Insights sous forme de nouveau tableau de bord vers CloudWatch

- Ajouter des métriques Performance Insights à un CloudWatch tableau de bord existant
- Affichage d'un widget de mesure Performance Insights dans CloudWatch

Exportation des métriques Performance Insights sous forme de nouveau tableau de bord vers CloudWatch

Choisissez un tableau de bord de métriques préconfiguré ou personnalisé dans le tableau de bord Performance Insights et exportez-le en tant que nouveau tableau de bord vers CloudWatch. Vous pouvez consulter le tableau de bord exporté dans la CloudWatch console.

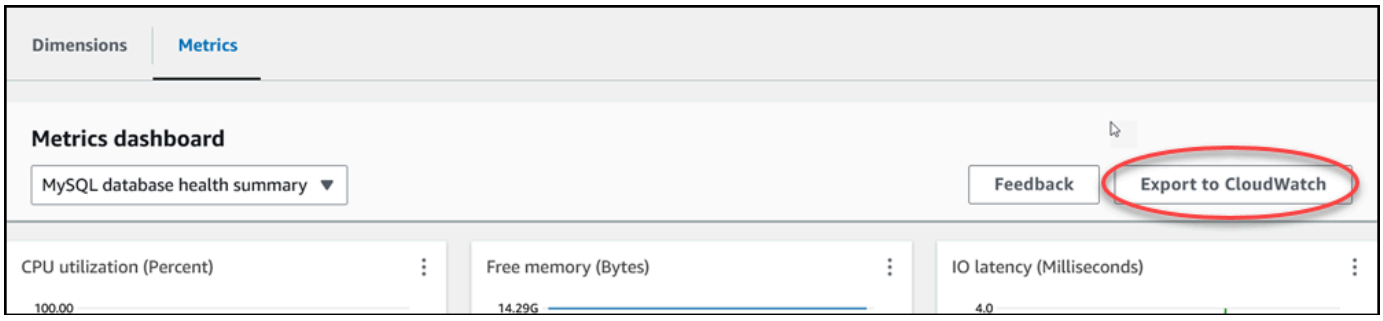
Pour exporter un tableau de bord métrique Performance Insights en tant que nouveau tableau de bord vers CloudWatch

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

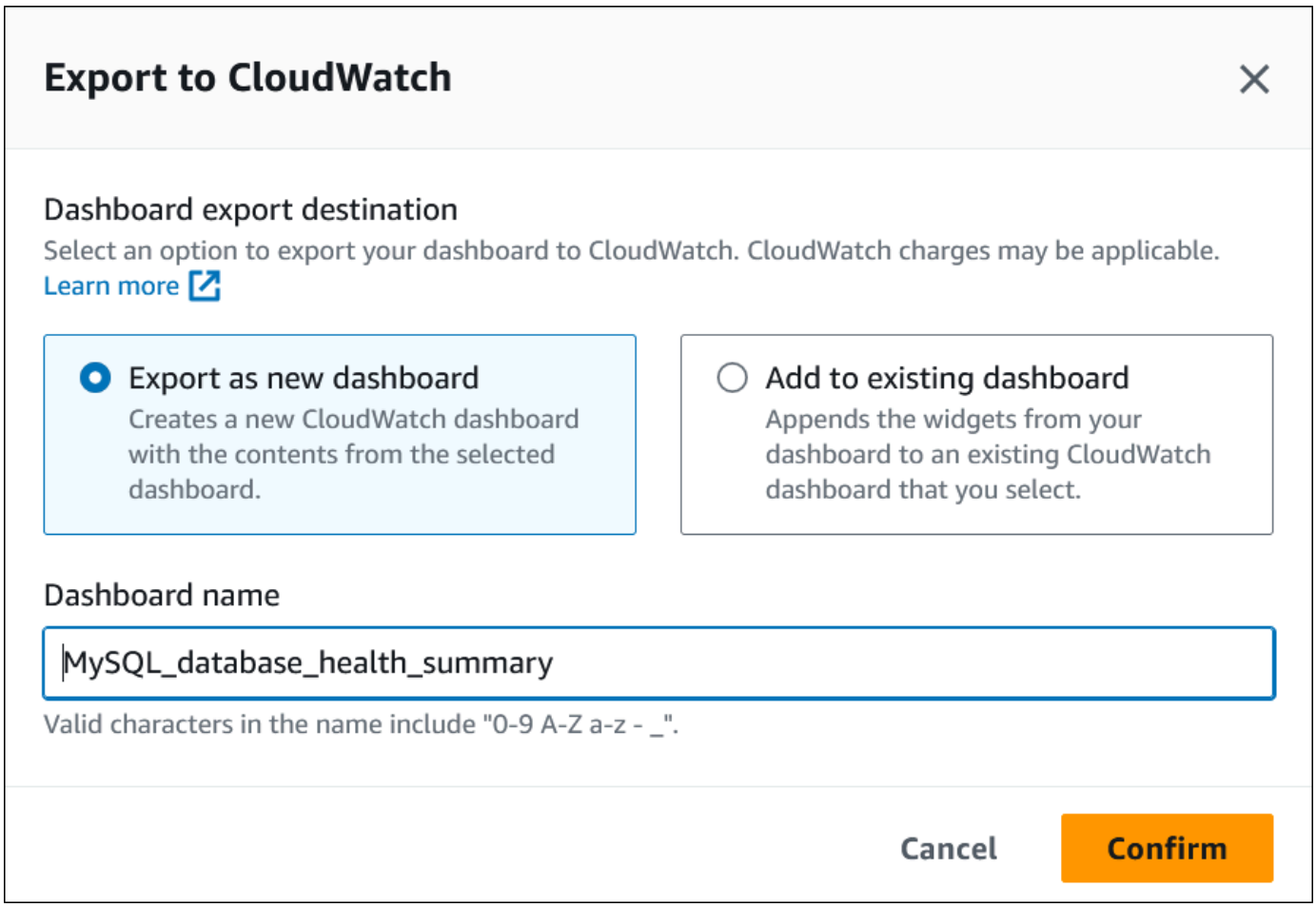
   Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.
4. Faites défiler l'écran vers le bas et choisissez Métriques.

   Par défaut, le tableau de bord préconfiguré avec les métriques Performance Insights s'affiche.
5. Choisissez un tableau de bord préconfiguré ou personnalisé, puis sélectionnez Exporter vers CloudWatch.

   La CloudWatch fenêtre Exporter vers apparaît.

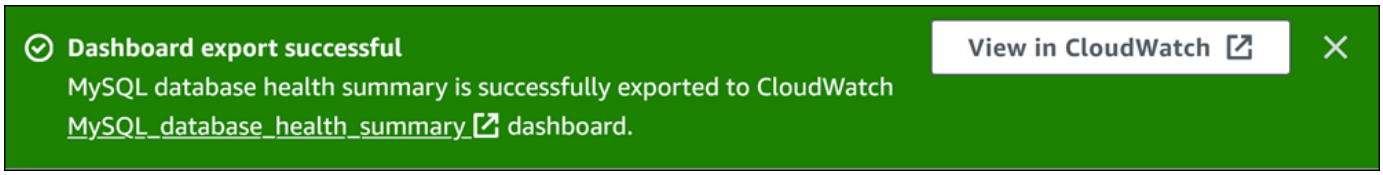


6. Choisissez Exporter en tant que nouveau tableau de bord.



7. Entrez le nom du nouveau tableau de bord dans le champ Nom du tableau de bord et choisissez Confirmer.

   Une bannière affiche un message une fois l'exportation du tableau de bord réussie.



8. Cliquez sur le lien ou sur Afficher CloudWatch dans la bannière pour afficher le tableau de bord des statistiques dans la CloudWatch console.

Ajouter des métriques Performance Insights à un CloudWatch tableau de bord existant

Ajoutez un tableau de bord de métriques préconfiguré ou personnalisé à un tableau de CloudWatch bord existant. Vous pouvez ajouter une étiquette au tableau de bord des statistiques pour qu'elle apparaisse dans une section distincte du CloudWatch tableau de bord.

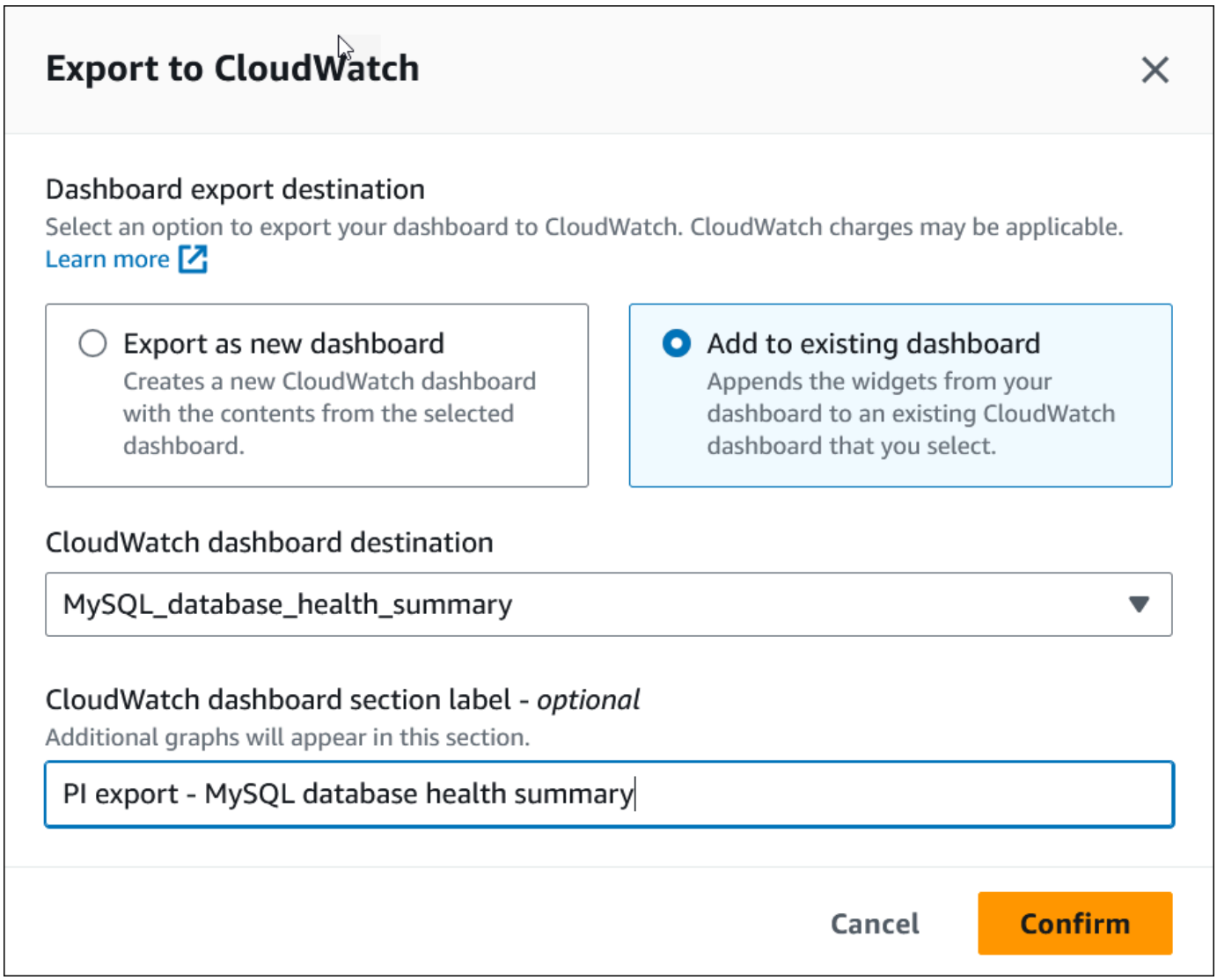
Pour exporter les métriques vers un tableau de CloudWatch bord existant

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

   Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.
4. Faites défiler l'écran vers le bas et choisissez Métriques.

   Par défaut, le tableau de bord préconfiguré avec les métriques Performance Insights s'affiche.
5. Choisissez le tableau de bord préconfiguré ou personnalisé, puis sélectionnez Exporter vers CloudWatch.

   La CloudWatch fenêtre Exporter vers apparaît.
6. Choisissez Ajouter au tableau de bord existant.



7. Spécifiez la destination et l'étiquette du tableau de bord, puis choisissez Confirmer.

   - CloudWatch destination du tableau de bord : choisissez un CloudWatch tableau de bord existant.
   - CloudWatch étiquette de la section du tableau de bord - facultatif - Entrez un nom pour les métriques Performance Insights qui apparaîtront dans cette section du CloudWatch tableau de bord.

   Une bannière affiche un message une fois l'exportation du tableau de bord réussie.

8. Cliquez sur le lien ou sur Afficher CloudWatch dans la bannière pour afficher le tableau de bord des statistiques dans la CloudWatch console.

Affichage d'un widget de mesure Performance Insights dans CloudWatch

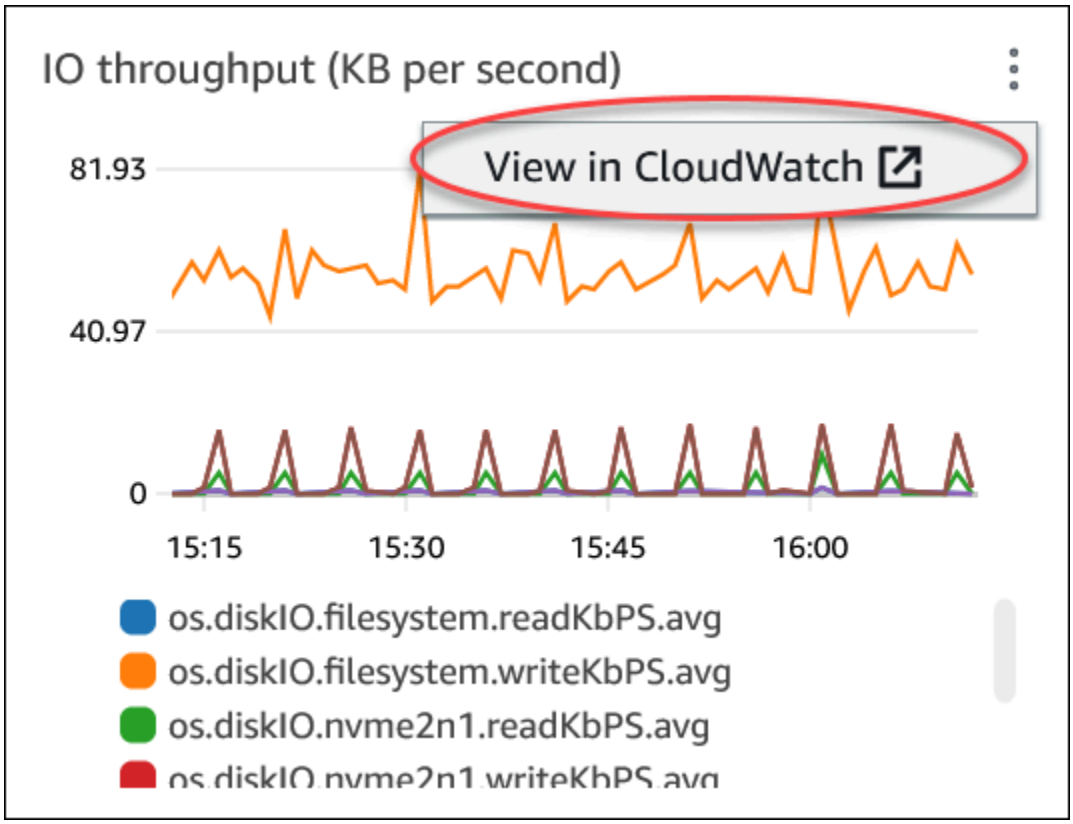
Sélectionnez un widget de mesure Performance Insights dans le tableau de bord Amazon RDS Performance Insights et consultez les données métriques dans la CloudWatch console.

Pour exporter un widget de mesures et afficher les données de mesures dans la CloudWatch console

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

   Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.
4. Faites défiler la page vers le bas jusqu'à Métriques.

   Par défaut, le tableau de bord préconfiguré avec les métriques Performance Insights s'affiche.
5. Choisissez un widget métrique, puis choisissez Afficher CloudWatch dans le menu.



   Les données métriques apparaissent dans la CloudWatch console.

Création d' CloudWatch alarmes pour surveiller Amazon RDS

Vous pouvez créer une CloudWatch alarme qui envoie un SNS message Amazon lorsque l'alarme change d'état. Une alarme surveille une seule métrique pendant la période que vous spécifiez. Elle peut également réaliser une ou plusieurs actions en fonction de la valeur de la métrique par rapport à un seuil donné sur un certain nombre de périodes. L'action est une notification envoyée à une SNS rubrique Amazon ou à une politique Amazon EC2 Auto Scaling.

Les alarmes déclenchent des actions uniquement pour les changements d'état prolongés. CloudWatch les alarmes n'invoquent pas d'actions simplement parce qu'elles sont dans un état particulier. L'état doit avoir changé et avoir été maintenu pendant un nombre de périodes spécifié.

Vous pouvez utiliser la fonction mathématique DB_ PERF _ INSIGHTS metric de la CloudWatch console pour interroger les indicateurs de compteur d'Amazon RDS pour Performance Insights. La INSIGHTS fonction DB_ PERF _ inclut également la DBLoad métrique à des intervalles inférieurs à une minute. Vous pouvez définir des CloudWatch alarmes sur ces métriques.

Pour plus de détails sur la création d'une alarme, voir Créer une alarme sur les indicateurs de compteur Performance Insights à partir d'une AWS base de données.

Pour régler une alarme à l'aide du AWS CLI

- Appelez `put-metric-alarm`. Pour plus d'informations, consultez la référence de la commande AWS CLI.

Pour régler une alarme à l'aide du CloudWatch API

- Appelez `PutMetricAlarm`. Pour plus d'informations, consultez Amazon CloudWatch API Reference

Pour plus d'informations sur la configuration des SNS rubriques Amazon et la création d'alarmes, consultez la section Utilisation des CloudWatch alarmes Amazon.

Tutoriel : Création d'une CloudWatch alarme Amazon pour le retard de réplication d'un cluster de bases de données multi-AZ pour Amazon RDS

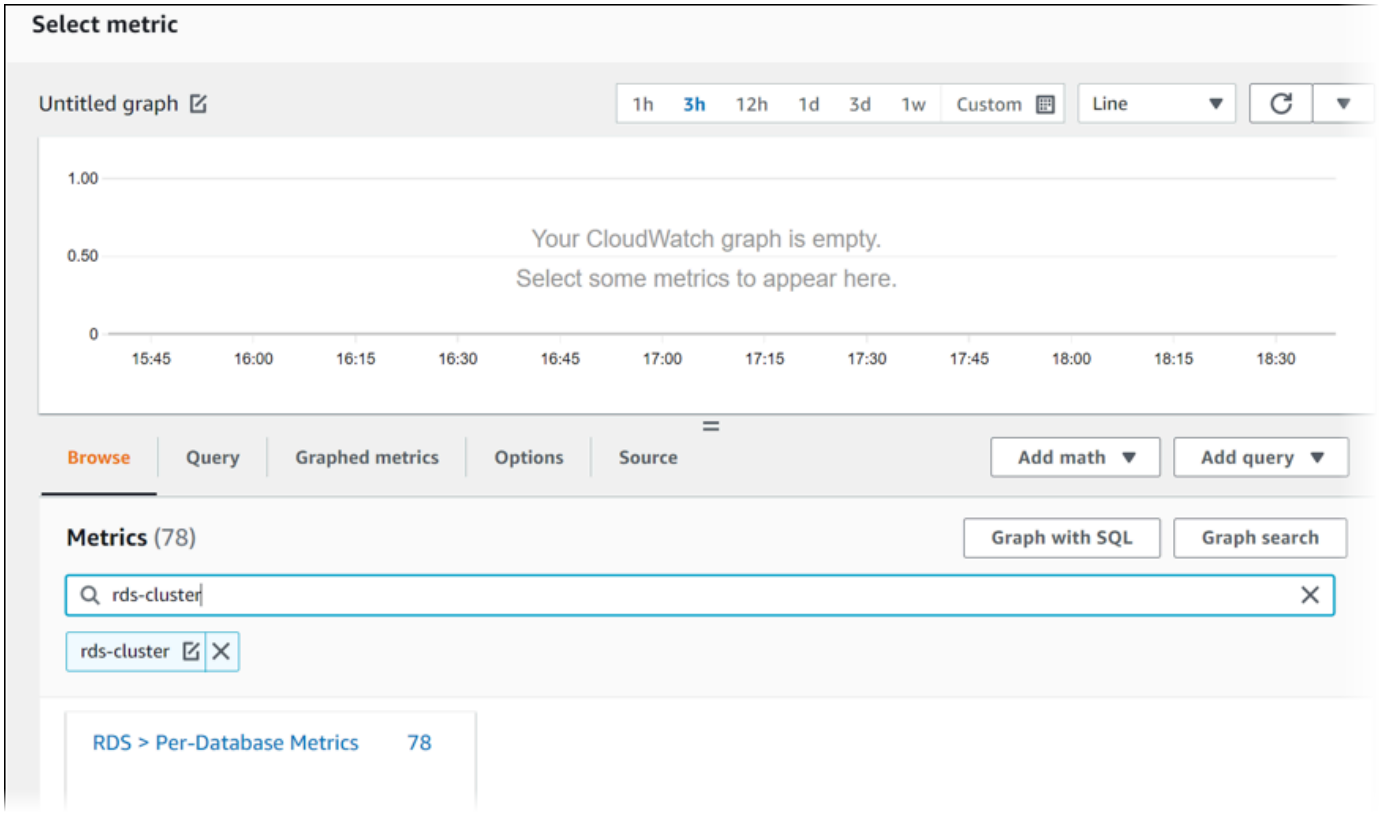
Vous pouvez créer une CloudWatch alarme Amazon qui envoie un SNS message Amazon lorsque le délai de réplication d'un cluster de bases de données multi-AZ dépasse un seuil. Une alarme surveille

la métrique `ReplicaLag` sur la période de temps que vous spécifiez. L'action est une notification envoyée à une SNS rubrique Amazon ou à une politique Amazon EC2 Auto Scaling.

Pour définir une CloudWatch alarme en cas de retard de réplication d'un cluster de bases de données multi-AZ

1. Connectez-vous à la CloudWatch console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, choisissez Alarms (alertes), All alarms (Toutes les alertes).
3. Choisissez Create alarm (Créer une alarme).
4. Sur la page Specify metric and conditions (Spécifier une métrique et des conditions), sélectionnez Select metric (Sélectionner une métrique).
5. Dans la zone de recherche, saisissez le nom de votre cluster de base de données Multi-AZ et appuyez sur Entrée.

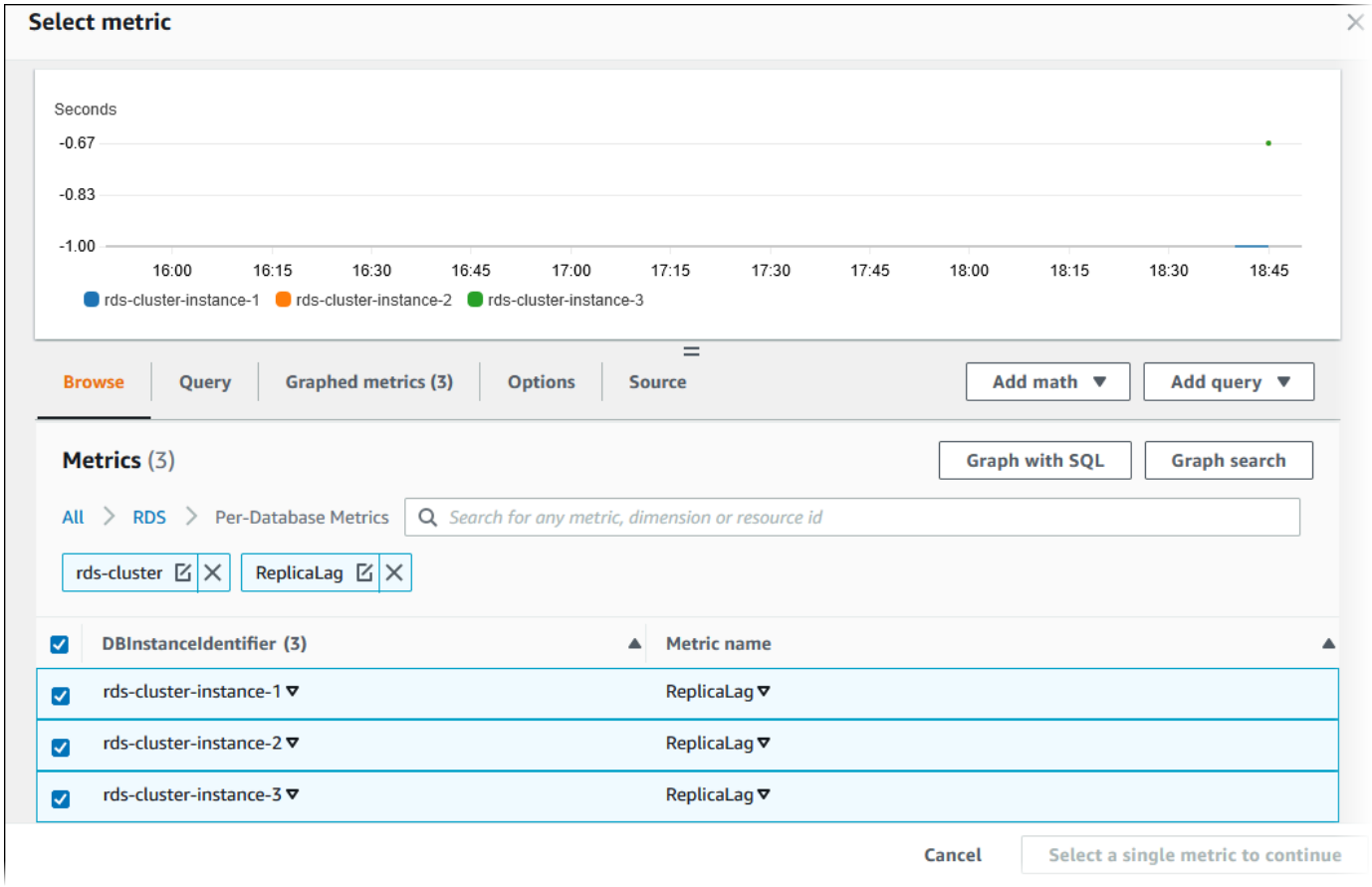
   L'image suivante illustre la page Select metric (Sélectionner la métrique) avec un cluster de base de données Multi-AZ nommé `rds-cluster` saisi.



6. Choisissez RDS« Métriques par base de données ».

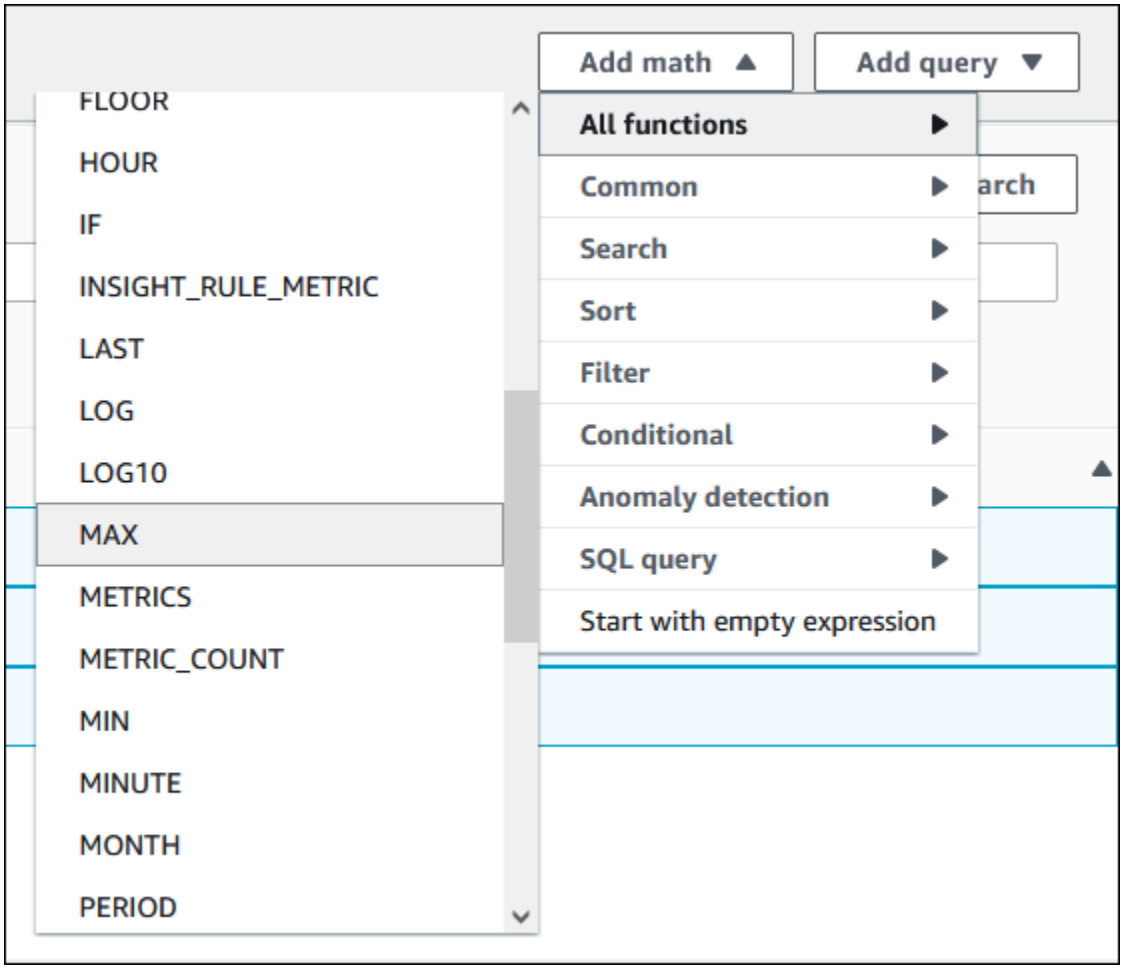
7. Dans la zone de recherche, saisissez **`ReplicaLag`** et appuyez sur Entrée, puis sélectionnez chaque instance de base de données dans le cluster de base de données.

   L'image suivante montre la page Sélectionner une métrique avec les instances de base de données sélectionnées pour la ReplicaLagmétrique.



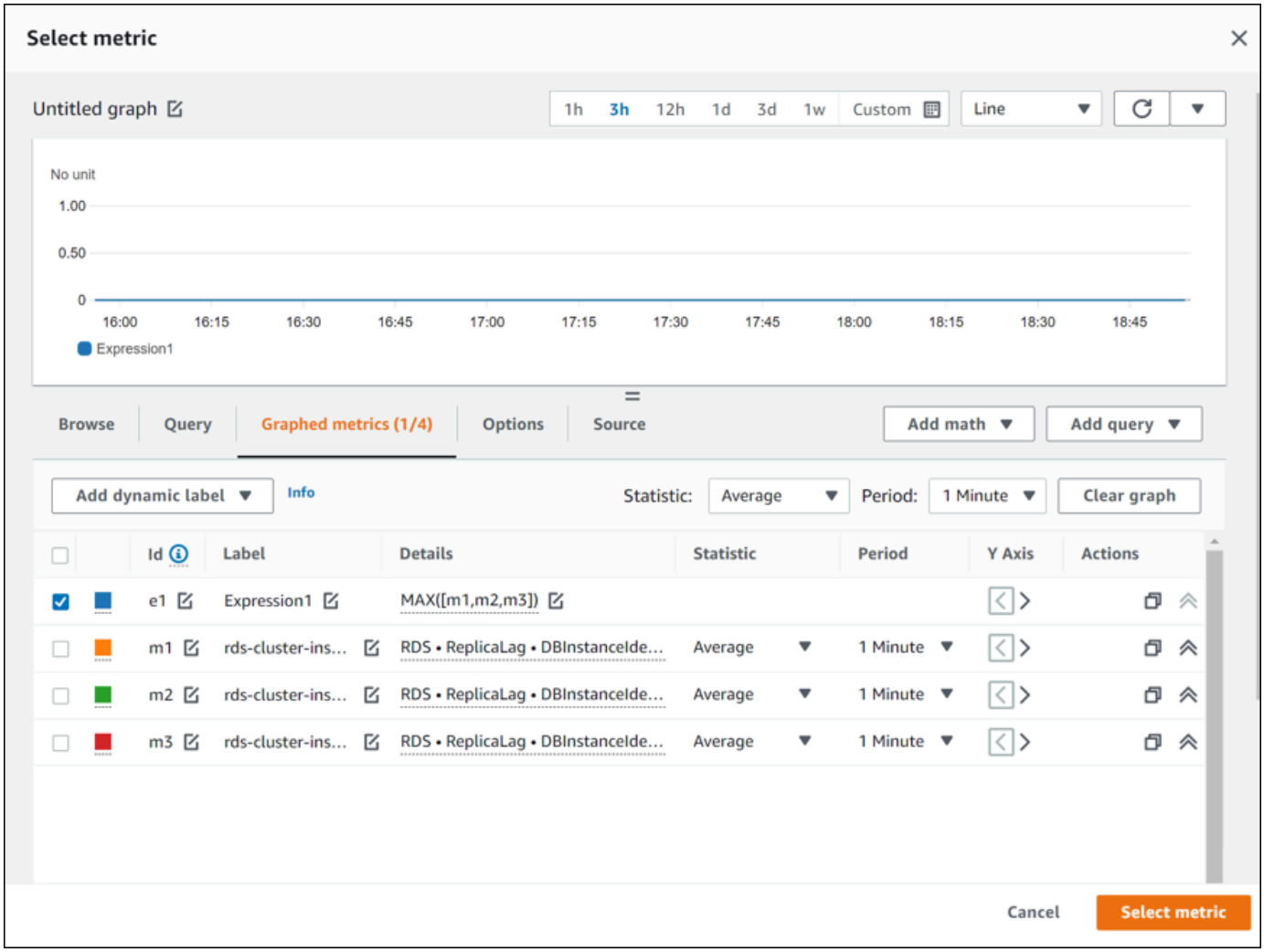
   Cette alarme prend en compte le décalage de réplica pour les trois instances de base de données dans le cluster de base de données Multi-AZ. L'alarme réagit lorsqu'une instance de base de données dépasse le seuil. Elle utilise une expression mathématique qui renvoie la valeur maximale des trois métriques. Commencez par trier par nom de métrique, puis choisissez les trois ReplicaLagmétriques.

8. Dans Ajouter des mathématiques, sélectionnez Toutes les fonctions, MAX.



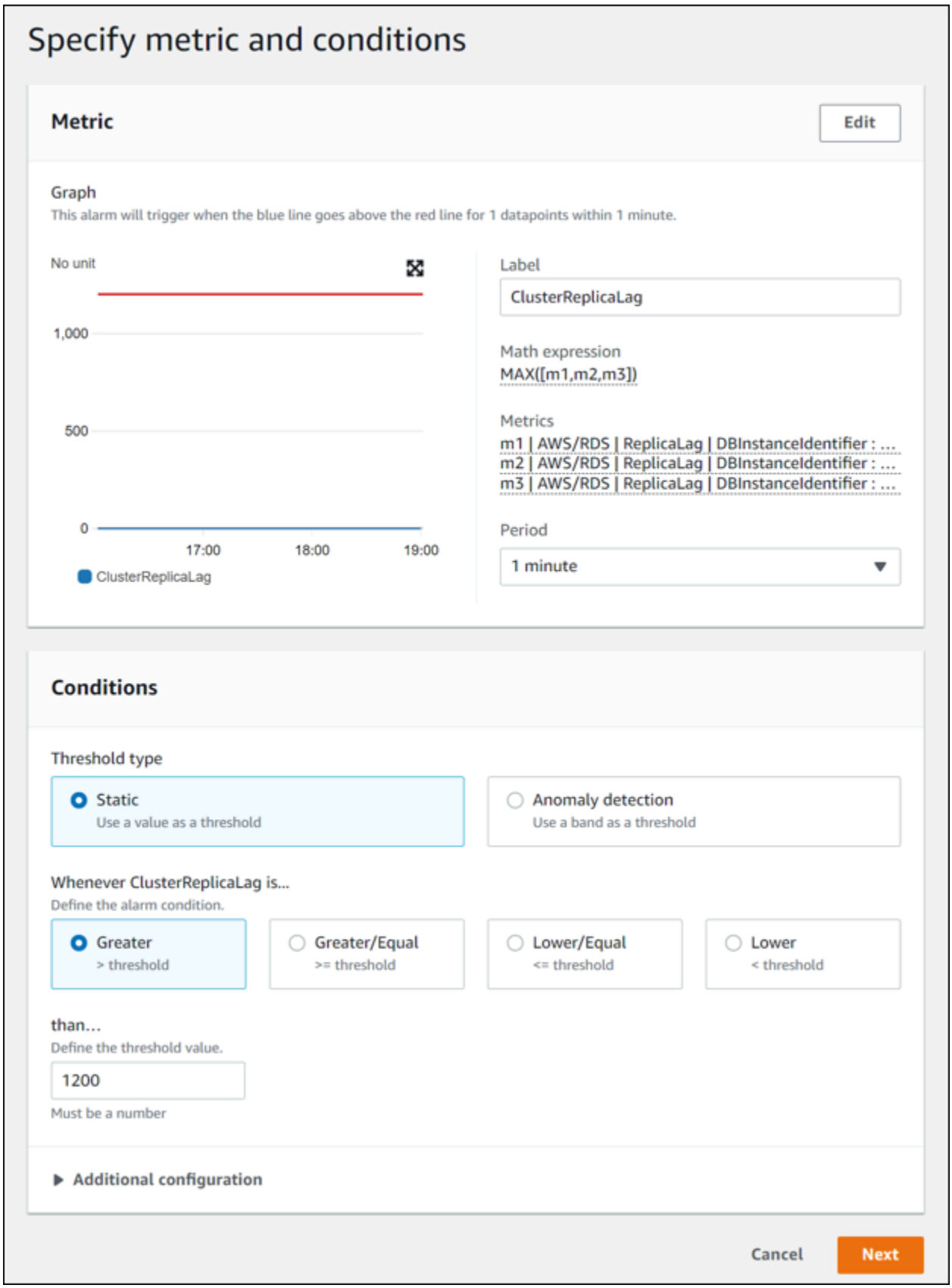
9. Choisissez l'onglet Graphed metrics (Métriques graphiques) et modifiez les détails de Expression1 pour **`MAX([m1,m2,m3])`**.
10. Pour les trois ReplicaLagindicateurs, remplacez la période par 1 minute.
11. Effacer la sélection parmi toutes les métriques, sauf Expression1.

    La page Select metric (Sélectionner la métrique) devrait ressembler à l'image suivante.



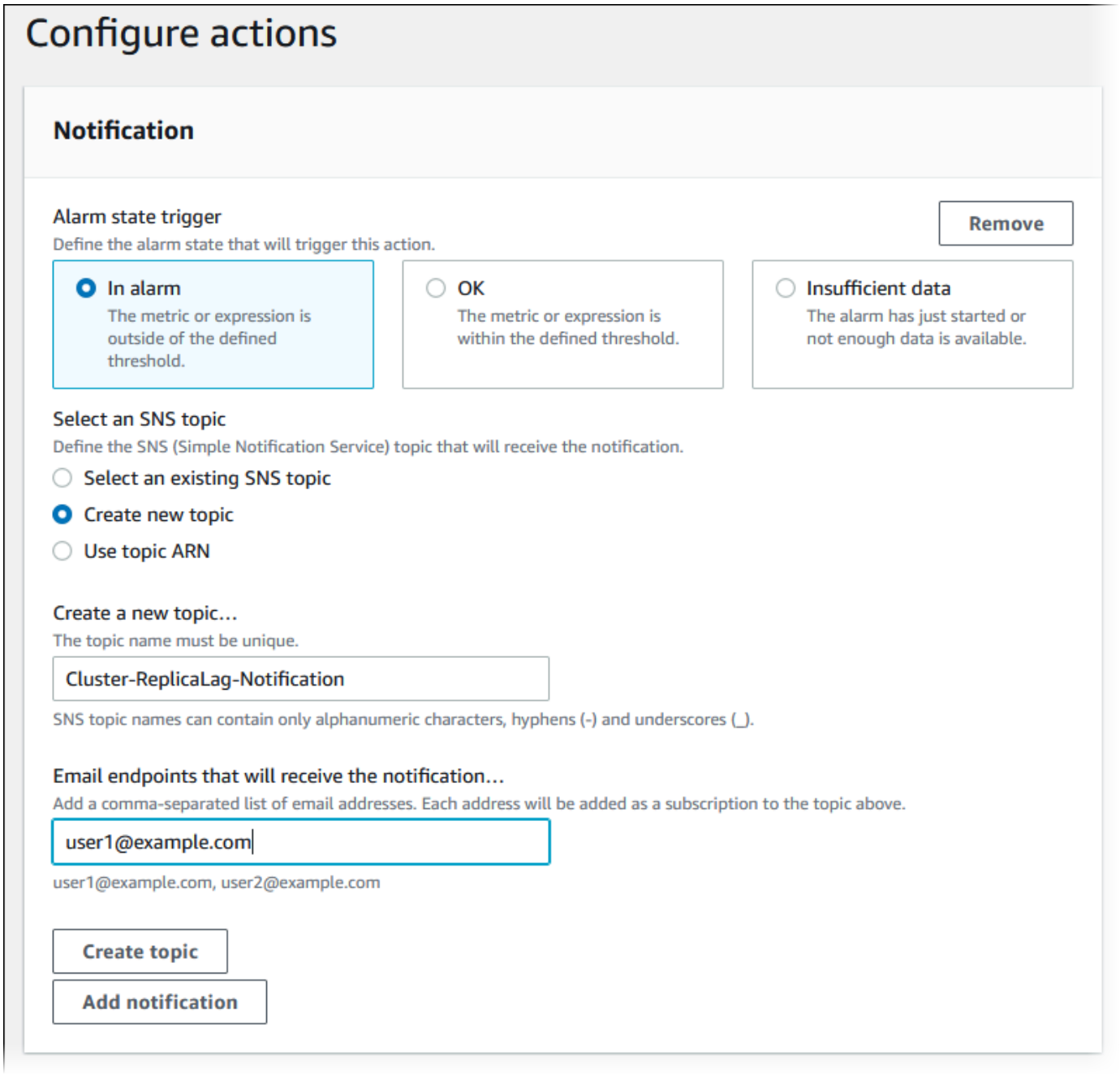
12. Choisissez Select metric (Sélectionner une métrique).
13. Dans la page Specify metric and conditions (Spécifier les métriques et les conditions), remplacez l'étiquette par un nom significatif tel que **`ClusterReplicaLag`**, puis saisissez un nombre de secondes dans Define the threshold value (Définition de la valeur de seuil). Pour ce didacticiel, saisissez **`1200`** secondes (20 minutes). Vous pouvez ajuster cette valeur en fonction des exigences de votre charge de travail.

    La page Specify metric and conditions (Spécifier les métriques et les conditions) devrait ressembler à l'image suivante.

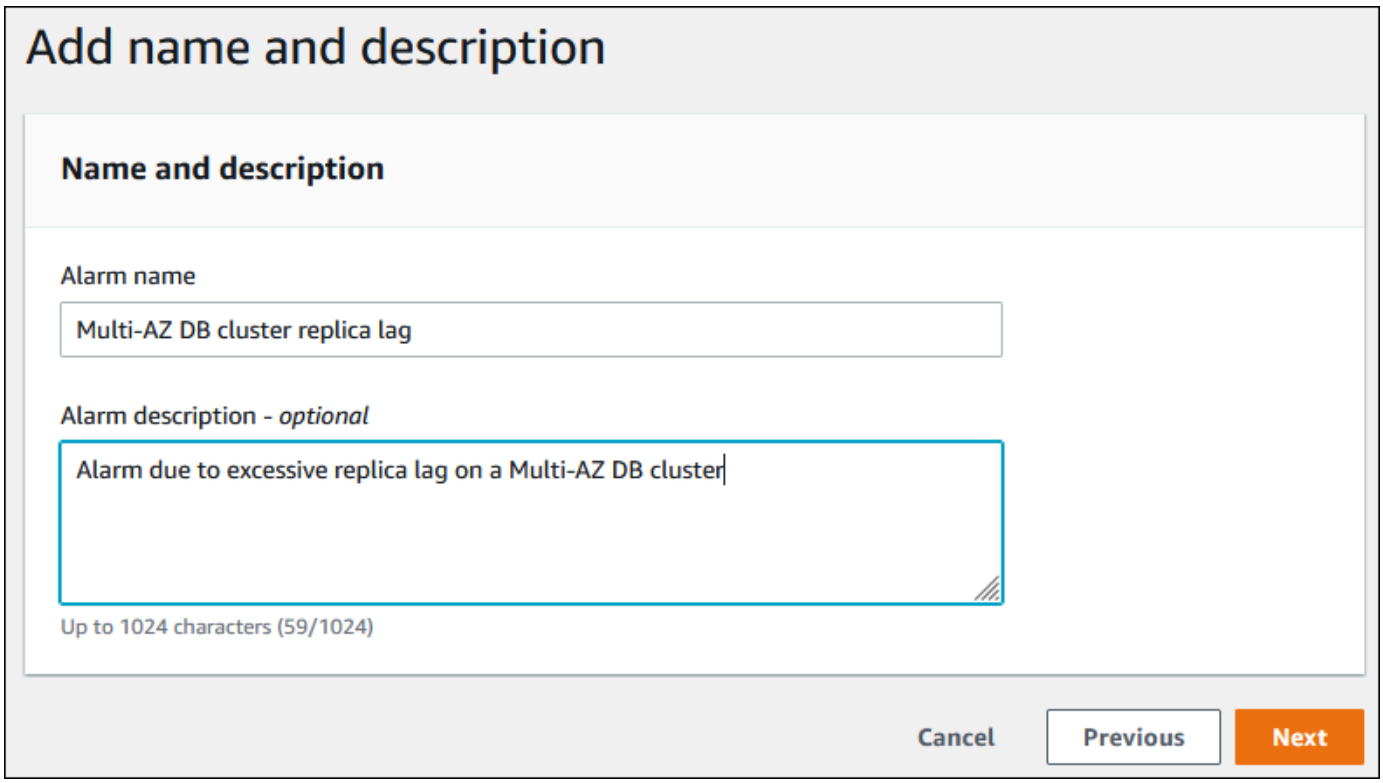


14. Choisissez Next (Suivant), et la page Configure actions (Configurer des actions) s'affiche.

15. Gardez In alarm (En alarme) sélectionné, choisissez Create new topic (Créer une rubrique), puis saisissez le nom de la rubrique et une adresse e-mail valide.



16. Choisissez Create topic (Créer une rubrique), puis choisissez Next (Suivant).
17. Dans le page Add a name and description (Ajouter un nom et une description), saisissez un Alarm name (Nom d'alarme) et une Alarm description (Description de l'alarme), puis choisissez Next (Suivant).



18. Prévisualisez l'alarme que vous êtes sur le point de créer dans la page Preview and create (Prévisualiser et créer), puis choisissez Create alarm (Créer une alarme).

Surveillance des bases de données Amazon RDS avec CloudWatch Database Insights

Surveillez la charge de base de données (charge de base de données) de votre parc d'instances de base de données Amazon RDS avec Database Insights. DB Load mesure le niveau d'activité des sessions dans votre base de données. Vous pouvez utiliser Database Insights pour analyser et résoudre les problèmes de performances de vos bases de données Amazon RDS à grande échelle.

Avec Database Insights, vous pouvez visualiser la charge de base de données sur votre parc de bases de données et filtrer la charge en fonction des temps d'attente, des instructions SQL, des hôtes ou des utilisateurs.

Par défaut, RDS active le mode standard de Database Insights pour vos bases de données Amazon RDS.

Pour plus d'informations sur l'utilisation de Database Insights dans la CloudWatch console Amazon, consultez CloudWatch Database Insights dans le guide de CloudWatch l'utilisateur Amazon.

Tarification

Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Rubriques

- Support du moteur de base de données Amazon RDS pour Database Insights
- Activation du mode avancé de Database Insights pour Amazon RDS (Aurora)
- Activation du mode standard de Database Insights pour Amazon RDS (Aurora)
- Configuration de votre base de données pour surveiller les requêtes SQL lentes avec Database Insights pour (Amazon RDS)
- Considérations relatives aux informations sur les bases de données pour Amazon RDS (Aurora)

Support du moteur de base de données Amazon RDS pour Database Insights

Le tableau suivant présente les moteurs de base de données Amazon RDS compatibles avec Database Insights.

Moteur de base de données Amazon RDS	Versions et régions soumises à la gestion des versions du moteur	Restrictions de classe d'instance
Amazon RDS for MariaDB	Pour plus d'informations sur la disponibilité des versions et des régions de Database Insights with RDS pour MariaDB, consultez . Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS	Database Insights n'est pas pris en charge pour les classes d'instances suivantes : • db.t2.micro • db.t2.small • db.t3.micro • db.t3.small • db.t4g.micro • db.t4g.small
RDS for MySQL	Pour plus d'informations sur la disponibilité des versions et des régions de Database Insights with RDS for MySQL, consultezRégions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	Database Insights n'est pas pris en charge pour les classes d'instances suivantes : • db.t2.micro • db.t2.small • db.t3.micro • db.t3.small • db.t4g.micro • db.t4g.small

Moteur de base de données Amazon RDS	Versions et régions soumises à la gestion des versions du moteur	Restrictions de classe d'instance
Amazon RDS for Microsoft SQL Server	Pour plus d'informations sur la disponibilité des versions et des régions de Database Insights with RDS for SQL Server, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	N/A
Amazon RDS for PostgreSQL	Pour plus d'informations sur la disponibilité des versions et des régions de Database Insights with RDS pour PostgreSQL, consultez. Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS	N/A
Amazon RDS for Oracle	Pour plus d'informations sur la disponibilité des versions et des régions de Database Insights with RDS pour Oracle, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	N/A

Support du moteur de base de données Amazon RDS pour les fonctionnalités de Database Insights

Le tableau suivant présente les moteurs de base de données Amazon RDS qui prennent en charge les fonctionnalités de Database Insights.

Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
Statistiques SQL pour Performance Insights	Tous	Tous	Tous	Tous
Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord Performance Insights pour Amazon RDS	Tous	Tous	RDS for Oracle	Tous
Analyse des performances de base de données pour une période donnée	Niveau payant uniquement	Tous	• RDS for MariaDB • RDS for MySQL • RDS for PostgreSQL	Tous
Consulter les recommandations proactives de Performance Insights	Niveau payant uniquement	• USA Est (Ohio) • USA Est (Virginie du Nord) • USA Ouest (Californie du Nord) • USA Ouest (Oregon) • Asie-Pacifique (Mumbai)	Tous	Tous

Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
		• Asie-Pacifique (Séoul) • Asie-Pacifique (Singapour) • Asie-Pacifique (Sydney) • Asie-Pacifique (Tokyo) • Canada (Centre) • Europe (Francfort) • Europe (Irlande) • Europe (Londres) • Europe (Paris) • Europe (Stockholm) • Amérique du Sud (São Paulo)		

Support d'Amazon RDS pour la région pour Database Insights

Amazon RDS prend en charge Database Insights dans les domaines suivants Régions AWS.

- USA Est (Virginie du Nord)
- USA Est (Ohio)
- USA Ouest (Californie du Nord)

- USA Ouest (Oregon)
- Afrique (Le Cap)
- Asie-Pacifique (Hong Kong)
- Asie-Pacifique (Hyderabad)
- Asie-Pacifique (Jakarta)
- Asie-Pacifique (Malaisie)
- Asie-Pacifique (Melbourne)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Osaka)
- Asia Pacific (Seoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Canada-Ouest (Calgary)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Londres)
- Europe (Milan)
- Europe (Paris)
- Europe (Espagne)
- Europe (Stockholm)
- Europe (Zurich)
- Israël (Tel Aviv)
- Moyen-Orient (Bahreïn)
- Moyen-Orient (EAU)
- Amérique du Sud (São Paulo)
- AWS GovCloud (USA Est)
- AWS GovCloud (US-Ouest)

Activation du mode avancé de Database Insights pour Amazon RDS (Aurora)

Pour activer le mode avancé de Database Insights pour Amazon RDS , suivez les procédures suivantes.

Activation du mode avancé de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ (de base de données)

Activez le mode avancé de Database Insights lors de la création d'une base de données pour Amazon RDS .

Console

Dans la console, vous pouvez activer le mode avancé de Database Insights lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ.

Pour activer le mode avancé de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données).
3. Choisissez Créer une base de données.
4. Dans la section Database Insights, sélectionnez le mode avancé. Choisissez ensuite les options suivantes :
   - Conservation – Durée de conservation des données de Performance Insights. La période de conservation doit être de 15 mois pour le mode avancé de Database Insights.
   - AWS KMS key – Spécifiez votre clé KMS. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.
5. Choisissez Créer une base de données.

AWS CLI

Pour activer le mode avancé de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ, appelez la create-db-cluster AWS CLI commande create-db-instanceor et fournissez les valeurs suivantes :

- `--database-insights-mode advanced`pour activer le mode avancé de Database Insights.
- `--engine`— Le moteur de base de données pour le de base de données.
- `--db-instance-identifier`— L'identifiant de l'instance de base de données ou `--db-cluster-identifier` — L'identifiant du cluster de base de données multi-AZ.
- `--enable-performance-insights`pour activer Performance Insights pour Database Insights.
- `--performance-insights-retention-period`— La période de conservation des données de votre instance de base de données ou de votre cluster de base de données multi-AZ. Pour activer Database Insights, la période de conservation doit être d'au moins 465 jours.

L'exemple suivant active le mode avancé de Database Insights lors de la création d'un de base de données.

Pour Linux, macOS ou Unix :

```
aws rds create-db-instance \
    --database-insights-mode advanced \
    --engine postgresql \
    --db-instance-identifier sample-db-identifier \
    --enable-performance-insights \
    --performance-insights-retention-period 465
```

Pour Windows :

```
aws rds create-db-instance ^
    --database-insights-mode advanced ^
    --engine postgresql ^
    --db-instance-identifier sample-db-identifier ^
    --enable-performance-insights ^
    --performance-insights-retention-period 465
```

RDS API

Pour activer le mode avancé de Database Insights lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ, spécifiez les paramètres suivants pour votre opération d'API Create DBInstance or Create DBCluster Amazon RDS.

- `DatabaseInsightsMode` sur `advanced`
- `EnablePerformanceInsights` sur `True`
- `PerformanceInsightsRetentionPeriod`jusqu'à au moins 465 jours

Activation du mode avancé de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ (de base de données)

Activez Database Insights lorsque vous modifiez une base de données pour Amazon RDS . La modification d'une de bases de données pour activer le mode avancé de Database Insights ne provoque pas de temps d'arrêt.

> **Note**
>
> Pour activer Database Insights, chaque instance de base de données d'un de base de données multi-AZ doit avoir les mêmes paramètres Performance Insights et Enhanced Monitoring.

Console

Dans la console, vous pouvez activer le mode avancé de Database Insights lorsque vous modifiez une instance de base de données ou un cluster de base de données multi-AZ.

Pour activer le mode avancé de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données).
3. Choisissez une instance de base de données ou un cluster de base de données Multi-AZ, et choisissez Modifiy (Modifier).

4. Dans la section Database Insights, sélectionnez le mode avancé. Choisissez ensuite les options suivantes :

   - Conservation – Durée de conservation des données de Performance Insights. La période de conservation doit être de 15 mois pour le mode avancé de Database Insights.
   - AWS KMS key – Spécifiez votre clé KMS. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour plus d'informations, consultez Chiffrement des ressources Amazon RDS.

5. Choisissez Continuer.
6. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement. Si vous choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée, votre base de données ignore ce paramètre et active immédiatement le mode avancé de Database Insights.
7. Choisissez Modifier l'instance de .

AWS CLI

Pour activer le mode avancé de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ, appelez la modify-db-cluster AWS CLI commande modify-db-instanceor et fournissez les valeurs suivantes :

- `--database-insights-mode advanced`pour activer le mode avancé de Database Insights.
- `--db-instance-identifier`— L'identifiant de l'instance de base de données ou `--db-cluster-identifier` — L'identifiant du cluster de base de données multi-AZ.
- `--enable-performance-insights`pour activer Performance Insights pour Database Insights.
- `--performance-insights-retention-period`— Période de conservation des données pour votre instance de base de données de de base de données. Pour activer le mode avancé de Database Insights, la période de conservation doit être d'au moins 465 jours.

L'exemple suivant active le mode avancé de Database Insights lors de la modification d'une instance de base de données de de bases de données.

Pour Linux, macOS ou Unix :

```
aws rds modify-db-instance \
```

```
    --database-insights-mode advanced \
    --db-instance-identifier sample-db-identifier \
    --enable-performance-insights \
    --performance-insights-retention-period 465
```

Pour Windows :

```
aws rds modify-db-instance ^
    --database-insights-mode advanced ^
    --db-instance-identifier sample-db-identifier ^
    --enable-performance-insights ^
    --performance-insights-retention-period 465
```

RDS API

Pour activer le mode avancé de Database Insights lorsque vous modifiez une instance de base de données ou un de base de données multi-AZ, spécifiez les paramètres suivants pour votre opération d'API DBInstanceModify ou DBCluster Modify Amazon RDS.

- `DatabaseInsightsMode` sur `advanced`
- `EnablePerformanceInsights` sur `True`
- `PerformanceInsightsRetentionPeriod`jusqu'à au moins 465 jours

Activation du mode standard de Database Insights pour Amazon RDS (Aurora)

Pour activer le mode standard de Database Insights pour Amazon RDS , suivez les procédures suivantes.

Activation du mode standard de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ (de base de données)

Activez le mode standard de Database Insights lors de la création d'une base de données pour Amazon RDS .

Console

Dans la console, vous pouvez activer le mode standard de Database Insights lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ.

Pour activer le mode standard de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données).
3. Choisissez Créer une base de données.
4. Dans la section Database Insights, sélectionnez le mode standard. Choisissez ensuite l'une des options suivantes pour activer ou désactiver Performance Insights :
   - Pour désactiver Performance Insights, désélectionnez Enable Performance Insights.
   - Pour activer Performance Insights, sélectionnez Enable Performance Insights. Pour configurer Performance Insights, spécifiez les options suivantes :
     - Conservation – Durée de conservation des données de Performance Insights. La période de conservation doit être d'au moins 7 jours.
     - AWS KMS key – Spécifiez votre clé KMS. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.
5. Choisissez Créer une base de données.

AWS CLI

Pour activer le mode standard de Database Insights lors de la création d'une instance de base de données ou d'un cluster de base de données multi-AZ, appelez la create-db-cluster AWS CLI commande create-db-instanceor et fournissez les valeurs suivantes :

- `--database-insights-mode standard`pour activer le mode standard de Database Insights.
- `--engine`— Le moteur de base de données pour le de base de données.
- `--db-instance-identifier`— L'identifiant de l'instance de base de données ou `--db-cluster-identifier` — L'identifiant du cluster de base de données multi-AZ.

- `--enable-performance-insights`ou `--no-enable-performance-insights` pour activer ou désactiver Performance Insights. Si vous le spécifiez`--enable-performance-insights`, vous devez également spécifier le `--performance-insights-retention-period` — La période de conservation des données pour votre instance de base de données de de bases de données. La période de conservation doit être d'au moins 7 jours.

L'exemple suivant active le mode Standard de Database Insights and Performance Insights lors de la création d'une instance de base de données de de base de données.

Pour Linux, macOS ou Unix :

```
aws rds create-db-instance \
    --database-insights-mode standard \
    --engine postgresql \
    --db-instance-identifier sample-db-identifier \
    --enable-performance-insights \
    --performance-insights-retention-period 7
```

Pour Windows :

```
aws rds create-db-instance ^
    --database-insights-mode standard ^
    --engine postgresql ^
    --db-instance-identifier sample-db-identifier ^
    --enable-performance-insights ^
    --performance-insights-retention-period 7
```

L'exemple suivant active le mode Standard de Database Insights et désactive Performance Insights lors de la création d'une instance de base de données de de base de données.

Pour Linux, macOS ou Unix :

```
aws rds create-db-instance \
    --database-insights-mode standard \
    --engine postgresql \
    --db-instance-identifier sample-db-identifier \
    --no-enable-performance-insights
```

Pour Windows :

```
aws rds create-db-instance ^
    --database-insights-mode standard ^
    --engine postgresql ^
    --db-instance-identifier sample-db-identifier ^
    --no-enable-performance-insights
```

RDS API

Pour activer le mode standard de Database Insights lorsque vous créez une instance de base de données ou un de base de données multi-AZ, spécifiez les paramètres suivants pour votre opération d'API Create DBInstance or Create DBCluster Amazon RDS.

- `DatabaseInsightsMode` sur `standard`
- `EnablePerformanceInsights`à `True` ou`False`. Si vous définissez sur `EnablePerformanceInsights``True`, vous devez définir une `PerformanceInsightsRetentionPeriod` durée d'au moins 7 jours.

Activation du mode standard de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ (de base de données)

Activez le mode standard de Database Insights lorsque vous modifiez une base de données pour Amazon RDS . La modification d'une de bases de données pour activer le mode standard de Database Insights ne provoque pas de temps d'arrêt.

> **Note**
>
> Pour activer Database Insights, chaque instance de base de données d'un de base de données multi-AZ doit avoir les mêmes paramètres Performance Insights et Enhanced Monitoring.

Console

Dans la console, vous pouvez activer le mode standard de Database Insights lorsque vous modifiez une instance de base de données ou un cluster de base de données multi-AZ.

Pour activer le mode standard de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données).
3. Choisissez une instance de base de données ou un cluster de base de données Multi-AZ, et choisissez Modifiy (Modifier).
4. Dans la section Database Insights, sélectionnez le mode standard. Choisissez ensuite l'une des options suivantes :
   - Pour désactiver Performance Insights, désélectionnez Enable Performance Insights.
   - Pour activer Performance Insights, sélectionnez Enable Performance Insights. Pour configurer Performance Insights, spécifiez les options suivantes :
     - Conservation – Durée de conservation des données de Performance Insights. La période de conservation doit être d'au moins 7 jours.
     - AWS KMS key – Spécifiez votre clé KMS. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour plus d'informations, consultez Chiffrement des ressources Amazon RDS.
5. Choisissez Continuer.
6. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement. Si vous choisissez Appliquer lors de la prochaine fenêtre de maintenance planifiée, votre base de données ignore ce paramètre et active immédiatement le mode standard de Database Insights.
7. Choisissez Modifier l'instance de .

AWS CLI

Pour activer le mode standard de Database Insights lors de la modification d'une instance de base de données ou d'un cluster de base de données multi-AZ, appelez la modify-db-cluster AWS CLI commande modify-db-instanceor et fournissez les valeurs suivantes :

- `--database-insights-mode standard`pour activer le mode standard de Database Insights.

- `--db-instance-identifier`— L'identifiant de l'instance de base de données ou `--db-cluster-identifier` — L'identifiant du cluster de base de données multi-AZ.
- `--enable-performance-insights`ou `--no-enable-performance-insights` pour activer ou désactiver Performance Insights. Si vous le spécifiez`--enable-performance-insights`, vous devez également spécifier le `--performance-insights-retention-period` — La période de conservation des données pour votre instance de base de données ou votre cluster de base de données multi-AZ. La période de conservation doit être d'au moins 7 jours.

L'exemple suivant active le mode Standard de Database Insights et active Performance Insights lors de la modification d'une instance de base de données de de base de données.

Pour Linux, macOS ou Unix :

```
aws rds modify-db-instance \
    --database-insights-mode standard \
    --db-instance-identifier sample-db-identifier \
    --enable-performance-insights \
    --performance-insights-retention-period 7
```

Pour Windows :

```
aws rds modify-db-instance ^
    --database-insights-mode standard ^
    --db-instance-identifier sample-db-identifier ^
    --enable-performance-insights ^
    --performance-insights-retention-period 7
```

L'exemple suivant active le mode Standard de Database Insights et désactive Performance Insights lors de la modification d'une instance de base de données de de base de données.

Pour Linux, macOS ou Unix :

```
aws rds modify-db-instance \
    --database-insights-mode standard \
    --db-instance-identifier sample-db-identifier \
    --no-enable-performance-insights
```

Pour Windows :

```
aws rds modify-db-instance ^
    --database-insights-mode standard ^
    --db-instance-identifier sample-db-identifier ^
    --no-enable-performance-insights
```

RDS API

Pour activer le mode standard de Database Insights lorsque vous modifiez une instance de base de données ou un de base de données multi-AZ, spécifiez les paramètres suivants pour votre opération d'API DBInstanceModify ou DBCluster Modify Amazon RDS.

- `DatabaseInsightsMode` sur `standard`
- `EnablePerformanceInsights`à `True` ou`False`. Si vous définissez sur `EnablePerformanceInsights``True`, vous devez définir une `PerformanceInsightsRetentionPeriod` durée d'au moins 7 jours.

Configuration de votre base de données pour surveiller les requêtes SQL lentes avec Database Insights pour (Amazon RDS)

Pour surveiller les requêtes SQL lentes pour votre base de données, vous pouvez utiliser la section Requêtes SQL lentes du tableau de bord Database Insights. Avant de configurer votre base de données pour surveiller les requêtes SQL lentes, la section Requêtes SQL lentes est vide.

Pour plus d'informations sur la surveillance des requêtes SQL lentes dans le tableau de bord Database Insights, consultez la section Affichage du tableau de bord des instances de base de CloudWatch données pour Database Insights dans le guide de CloudWatch l'utilisateur Amazon.

Pour configurer votre base de données afin de surveiller les requêtes SQL lentes avec Database Insights, procédez comme suit :

1. Activez les exportations de CloudWatch journaux vers Logs.
2. Créez ou modifiez le groupe de de paramètres de base de données pour votre instance de base de données.

Pour plus d'informations sur la configuration des exportations de journaux, consultez la section Publication de journaux de base de données sur Amazon CloudWatch Logs dans le guide de l'utilisateur Amazon RDS .

Pour créer ou modifier votre groupe de paramètres de base de données, consultez les rubriques suivantes.

- Création d'un groupe de paramètres de base de données dans RDSAmazon
- Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora)

RDS for MariaDB

Pour configurer votre instance de base de données RDS pour MariaDB afin de surveiller les requêtes SQL lentes, vous pouvez utiliser la combinaison de paramètres suivante comme exemple :

- `log_slow_query`— réglé sur `1`
- `log_slow_query_time`— réglé sur `1.0`
- `log_output`— réglé sur `FILE`

Il s'agit d'une configuration possible. Pour un guide complet sur les paramètres du journal des requêtes lentes de MariaDB et les options de configuration supplémentaires, consultez la documentation de MariaDB sur le journal des requêtes lentes.

RDS for MySQL

Pour configurer votre instance de base de données RDS pour MySQL afin de surveiller les requêtes SQL lentes, vous pouvez utiliser la combinaison de paramètres suivante comme exemple :

- `slow_query_log`— réglé sur `1`
- `long_query_time`— réglé sur `1.0`
- `log_output`— réglé sur `FILE`

Il s'agit d'une configuration possible. Pour un guide complet sur les paramètres du journal des requêtes lentes de MySQL et les options de configuration supplémentaires, consultez la documentation MySQL relative au journal des requêtes lentes.

RDS for PostgreSQL

Pour configurer votre instance de base de données RDS pour PostgreSQL afin de surveiller les requêtes SQL lentes, vous pouvez utiliser la combinaison de paramètres suivante comme exemple. Notez que la définition de ces paramètres peut réduire les performances de votre instance de base de données.

- `log_min_duration_statement`— réglé sur `1000`
- `log_statement`— réglé sur `none`
- `log_destination`— réglé sur `stderr`

Il s'agit d'une configuration possible. Pour un guide complet sur les paramètres de journalisation de PostgreSQL et les options de configuration supplémentaires, consultez la documentation PostgreSQL relative à la configuration de journalisation.

> **Note**
>
> Pour RDS for MySQL , vous pouvez configurer le paramètre `long_query_time` avec une granularité d'une microseconde. Par exemple, vous pouvez définir ce paramètre sur`0.000001`. En fonction du nombre de requêtes sur l'instance de base de données, la valeur du paramètre `long_query_time` peut réduire les performances. Commencez par la valeur`1.0`, puis ajustez-la en fonction de votre charge de travail. Lorsque vous définissez ce paramètre sur`0`, Database Insights enregistre toutes les requêtes.

Pour plus d'informations sur les journaux RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL, consultez ce qui suit.

- Fichiers journaux de base de données MariaDB
- Fichiers journaux de base de données MySQL
- Fichiers journaux de base de données RDS for PostgreSQL

Considérations relatives aux informations sur les bases de données pour Amazon RDS (Aurora)

Vous trouverez ci-dessous des considérations relatives à Database Insights pour Amazon RDS ().

- Vous ne pouvez pas gérer Database Insights pour une instance de base de données dans un de base de données Multi-AZ.
- Pour activer le mode avancé de Database Insights, vous devez activer Performance Insights et définir la période de conservation de Performance Insights sur au moins 465 jours (15 mois). Il n'y a aucun coût supplémentaire pour fixer la période de conservation de Performance Insights à 15 mois en plus du coût de Database Insights. Pour plus d'informations sur la tarification de Database Insights, consultez Amazon CloudWatch Pricing.
- Pour activer Database Insights, chaque instance de base de données d'un de base de données multi-AZ doit avoir les mêmes paramètres Performance Insights et Enhanced Monitoring.
- La modification d'une de bases de données pour activer l'un ou l'autre mode de Database Insights ne provoque pas de temps d'arrêt.

Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.

Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Performance Insights développe les fonctions de surveillance existantes d'Amazon RDS pour illustrer et vous aider à analyser les performances de votre base de données. Avec le tableau de bord Performance Insights, vous pouvez visualiser la charge de la base de données de votre Charge d'instance de base de données Amazon RDS et filtrer la charge par attentes, instructions SQL, hôtes ou utilisateurs. Pour plus d'informations sur l'utilisation de Performance Insights avec Amazon DocumentDB, consultez le Guide du développeur Amazon DocumentDB.

Rubriques

- Présentation de Performance Insights sur Amazon RDS
- Activer et désactiver Performance Insights pour Amazon RDS
- Présentation du schéma de performance pour Performance Insights sur , Amazon RDS for MariaDB ou MySQL
- Configuration des politiques d'accès pour Performance Insights
- Analyse des métriques à l'aide du tableau de bord de Performance Insights
- Consulter les recommandations proactives de Performance Insights
- Récupération de métriques avec Performance Insights API for Amazon RDS
- Journalisation des appels Performance Insights à l'aide de AWS CloudTrail
- Performance Insights API et VPC points de terminaison d'interface ()AWS PrivateLink

Présentation de Performance Insights sur Amazon RDS

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.

Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Par défaut, RDS active Performance Insights dans l'assistant de création de la console pour tous les moteurs Amazon RDS. Si vous disposez de plusieurs bases de données sur une instance de base de données, Performance Insights regroupe les données de performance.

Vous trouverez un aperçu de Performance Insights pour Amazon RDS dans la vidéo suivante.

Utilisation de Performance Insights pour analyser les performances de Amazon Aurora PostgreSQL

Important

Les rubriques suivantes expliquent comment utiliser Amazon RDS Performance Insights avec des moteurs de base de données non Aurora. Pour de plus amples informations sur l'utilisation d'Amazon RDS Performance Insights avec Amazon Aurora, veuillez consulter Utilisation d'Amazon RDS Performance Insights dans le Guide de l'utilisateur pour Amazon Aurora.

Rubriques

- Charge de la base de données
- Maximum CPU
- Prise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour Performance Insights
- Tarification et conservation des données pour Performance Insights

Charge de la base de données

La charge de base de données (charge de base de données) mesure le niveau d'activité de session dans votre base de données. `DBLoad`est l'indicateur clé de Performance Insights, et Performance Insights collecte la charge de base de données chaque seconde.

Rubriques

- Sessions actives
- Sessions actives en moyenne
- Exécutions actives moyennes
- Dimensions

Sessions actives

Une session de base de données représente le dialogue d'une application avec une base de données relationnelle. Une session active est une connexion qui a transmis du travail au moteur de base de données et qui attend une réponse.

Une session est active lorsqu'elle s'exécute sur le processeur (CPU) ou attend qu'une ressource devienne disponible pour pouvoir continuer. Par exemple, une session active peut attendre qu'une page (ou un bloc) soit lue en mémoire avant d'utiliser le processeur pendant la lecture des données de la page.

Sessions actives en moyenne

Les sessions actives en moyenne (AAS) représentent l'unité de la métrique `DBLoad` de Performance Insights. Elle mesure le nombre de sessions actives simultanément sur la base de données.

Toutes les secondes, Performance Insights échantillonne le nombre de sessions exécutant simultanément une requête. Pour chaque session active, Performance Insights collecte les données suivantes :

- Instruction SQL
- État de la session (en cours d'exécution sur le processeur ou en attente)
- Host (Hôte)
- Utilisateur exécutant le SQL

Performance Insights calcule les AAS en divisant le nombre total de sessions par le nombre d'échantillons pour une période déterminée. Par exemple, la table suivante présente 5 échantillons consécutifs d'une requête en cours d'exécution, prélevés à des intervalles d'une seconde.

Exemple	Nombre de sessions exécutant la requête	AAS	Calcul
1	2	2	2 sessions au total/1 échantillon
2	0	1	2 sessions au total/2 échantillons
3	4	2	6 sessions au total/3 échantillons
4	0	1.5	6 sessions au total/4 échantillons
5	4	2	10 sessions au total/5 échantillons

Dans l'exemple précédent, la charge de la base de données pour l'intervalle de temps était de 2 AAS. Cette mesure signifie qu'en moyenne, deux sessions étaient actives à la fois à n'importe quel moment au cours de la période où les cinq échantillons ont été prélevés.

Exécutions actives moyennes

La moyenne des exécutions actives (AAE) par seconde est liée à l'AAS. Pour calculer l'AAE, Performance Insights divise la durée totale d'exécution d'une requête par l'intervalle de temps. Le tableau suivant présente le calcul de l'AAE pour la même requête que dans le tableau précédent.

Temps écoulé (en secondes)	Durée totale d'exécution (en secondes)	AAE	Calcul
60	120	2	120 secondes d'exécution/60 secondes écoulées
120	120	1	120 secondes d'exécution/120 secondes écoulées

Temps écoulé (en secondes)	Durée totale d'exécution (en secondes)	AAE	Calcul
180	380	2.11	380 secondes d'exécution/180 secondes écoulées
240	380	1.58	380 secondes d'exécution/240 secondes écoulées
300	600	2	600 secondes d'exécution/300 secondes écoulées

Dans la plupart des cas, l'AAS et l'AAE d'une requête sont approximativement identiques. Cela dit, comme les données utilisées pour les calculs proviennent de sources différentes, les calculs varient souvent légèrement.

Dimensions

La métrique `db.load` est différente des autres métriques de série chronologique, car vous pouvez la décomposer en sous-composants appelés dimensions. Vous pouvez considérer les dimensions comme des catégories de « tranches » pour les différentes caractéristiques de la métrique `DBLoad`.

Lorsque vous diagnostiquez des problèmes de performances, les dimensions suivantes sont souvent les plus utiles :

Rubriques

- Événements d'attente
- Principaux éléments SQL
- Plans

Pour obtenir la liste complète des dimensions des moteurs Amazon RDS, veuillez consulter Charge de base de données tranchée par dimensions.

Événements d'attente

Un événement d'attente fait qu'une instruction SQL attend qu'un événement spécifique se produise avant de pouvoir continuer à s'exécuter. Les événements d'attente constituent une dimension (ou catégorie) importante pour la charge de la base de données, car ils indiquent les points de blocage du travail.

Chaque session active est soit en cours d'exécution au niveau du processeur soit en attente. Par exemple, les sessions sollicitent le processeur lorsqu'elles recherchent un tampon dans la mémoire, effectuent un calcul ou exécutent du code procédural. Lorsque les sessions ne sollicitent pas le processeur, c'est peut-être qu'elles attendent qu'un tampon de mémoire se libère, qu'un fichier de données soit lu ou qu'un journal soit écrit. Le temps que passe une session à attendre des ressources est autant de temps en moins qu'elle passe à s'exécuter au niveau du processeur.

Lorsque vous réglez une base de données, vous cherchez souvent à identifier les ressources que les sessions attendent. Par exemple, deux ou trois événements d'attente peuvent représenter 90 % de la charge de la base de données. Cette mesure signifie qu'en moyenne, les sessions actives passent la majeure partie de leur temps à attendre un petit nombre de ressources. Si vous trouvez la cause de ces attentes, vous pouvez tenter une solution.

Les événements d'attente varient en fonction du moteur de base de données :

- Pour plus d'informations sur tous les événements d'attente MariaDB et MySQL, veuillez consulter Wait Event Summary Tables dans la documentation MySQL.
- Pour plus d'informations sur tous les événements d'attente PostgreSQL, consultez The Statistics Collector > Wait Event tables dans la documentation de PostgreSQL.
- Pour plus d'informations sur tous les événements d'attente Oracle, veuillez consulter Descriptions of Wait Events dans la documentation Oracle.
- Pour plus d'informations sur tous les événements d'attente SQL Server, veuillez consulter Types of Waits dans la documentation SQL Server.

> **Note**
>
> Pour Oracle, les processus en arrière-plan s'exécutent parfois sans instruction SQL associée. Dans ce cas, Performance Insights communique le type de processus en arrière-plan concaténé avec le signe deux-points, et la classe d'attente associée à ce processus en arrière-plan. Parmi les types de processus en arrière-plan figurent `LGWR`, `ARC0`, `PMON`, etc.

Par exemple, lorsque le programme d'archivage effectue des opérations d'I/O, le rapport Performance Insights correspondant est similaire à `ARC1:System I/O`. Parfois, le type du processus en arrière-plan est également omis et Performance Insights communique uniquement la classe d'attente, par exemple `:System I/O`.

Principaux éléments SQL

Là où les événements d'attente présentent des goulots d'étranglement, les principaux éléments SQL indiquent quelles requêtes contribuent le plus à la charge de la base de données. Par exemple, de nombreuses requêtes peuvent être en cours d'exécution sur la base de données, mais une seule d'entre elles peut consommer 99 % de la charge de la base de données. Dans ce cas, la charge élevée peut indiquer un problème avec la requête.

Par défaut, la console Performance Insights affiche les principales requêtes SQL qui contribuent à la charge de la base de données. La console affiche également des statistiques pertinentes pour chaque instruction. Pour diagnostiquer les problèmes de performances d'une instruction spécifique, vous pouvez examiner son plan d'exécution.

Plans

Un plan d'exécution, également appelé simplement un plan, est une séquence d'étapes qui accèdent aux données. Par exemple, un plan de jonction de tables `t1` et `t2` peut parcourir toutes les lignes de `t1` et comparer chaque ligne à une ligne de `t2`. Dans une base de données relationnelle, un optimiseur est un code intégré qui détermine le plan le plus efficace pour une requête SQL.

Pour les instances de base de données, Performance Insights collecte automatiquement les plans d'exécution. Pour diagnostiquer les problèmes de performances SQL, examinez les plans capturés pour les requêtes SQL gourmandes en ressources. Les plans montrent comment la base de données a analysé et exécuté les requêtes.

Pour savoir comment analyser la charge de base de données à l'aide de plans, voir :

- Oracle: Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord Performance Insights pour Amazon RDS
- Serveur SQL : Analyse des plans d'exécution des SQL serveurs à l'aide du tableau de bord Performance Insights pour Amazon RDS

Capture du plan

Toutes les cinq minutes, Performance Insights identifie les requêtes les plus gourmandes en ressources et enregistre leurs plans. Ainsi, vous n'avez pas besoin de collecter et de gérer manuellement un grand nombre de plans. Au lieu de cela, vous pouvez utiliser l'onglet Top SQL (Principaux éléments SQL) pour vous concentrer sur les plans des requêtes les plus problématiques.

> **Note**
>
> Performance Insights ne capture pas de plans pour les requêtes dont le texte dépasse la limite maximale de texte de requête collectable. Pour de plus amples informations, veuillez consulter Accès à plus de texte SQL dans le tableau de bord Performance Insights.

La période de conservation des plans d'exécution est la même que pour vos données Performance Insights. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.

Requêtes récapitulatives

L'onglet Top SQL (Principaux éléments SQL) affiche les requêtes récapitulatives par défaut. Une requête récapitulative n'a pas elle-même de plan, mais toutes les requêtes utilisant des valeurs littérales ont des plans. Par exemple, une requête récapitulative peut inclure le texte `WHERE \`email\`=?`. Le récapitulatif peut contenir deux requêtes, l'une avec le texte `WHERE email=user1@example.com` et l'autre avec `WHERE email=user2@example.com`. Chacune de ces requêtes littérales peut inclure plusieurs plans.

Lorsque vous sélectionnez une requête de résumé, la console affiche tous les plans relatifs aux instructions secondaires du résumé sélectionné. Par conséquent, vous n'avez pas besoin de parcourir toutes les instructions enfant pour trouver le plan. Vous pouvez voir des plans qui ne figurent pas dans la liste affichée des 10 premières instructions enfant. La console affiche les plans pour toutes les requêtes enfant pour lesquelles des plans ont été collectés, que les requêtes figurent ou non dans les 10 premières requêtes.

Maximum CPU

Dans le tableau de bord, le graphique Database load (Charge de base de données)collecte, regroupe et affiche les informations de session. Pour savoir si les sessions actives dépassent le maximumCPU, examinez leur relation avec la CPU ligne Max v. Performance Insights détermine la

CPU valeur maximale de v en fonction du nombre de cœurs v CPU (virtuelsCPU) pour votre instance de base de données.

Un processus peut être exécuté sur un v CPU à la fois. Si le nombre de processus dépasse le nombre devCPUs, les processus commencent à être mis en file d'attente. Lorsque le nombre de files d'attente augmente, les performances de la base de données diminuent. Si la charge de la base de données est souvent supérieure à la CPU ligne Max v et que l'état d'attente principal est le mêmeCPU, la charge CPU est surchargée. Dans ce cas, vous souhaiterez peut-être limiter les connexions à l'instance, régler les SQL requêtes présentant une CPU charge élevée ou envisager une classe d'instance plus importante. Quel que soit leur état d'attente, les instances élevées et régulières indiquent que des problèmes de goulots d'étranglement ou de conflits de ressources devront peut-être être résolus. Cela peut être vrai même si la charge de base de données ne franchit pas la CPU ligne Max v.

Prise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour Performance Insights

Important

AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.

Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).

Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.

Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Le tableau suivant fournit les moteurs de base de données Amazon RDS qui prennent en charge l'analyse des performances.

> **Note**
>
> Pour Amazon Aurora, consultez Prise en charge de Performance Insights par les moteurs de base de données Amazon Aurora dans le Guide de l'utilisateur Amazon Aurora.

Moteur de base de données Amazon RDS	Versions et régions soumises à la gestion des versions du moteur	Restrictions de classe d'instance
Amazon RDS for MariaDB	Pour obtenir plus d'informations sur la disponibilité des versions et des régions de Performance Insights avec RDS for MariaDB, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	Performance Insights n'est pas pris en charge pour les classes d'instances suivantes : • db.t2.micro • db.t2.small • db.t3.micro • db.t3.small • db.t4g.micro • db.t4g.small

Moteur de base de données Amazon RDS	Versions et régions soumises à la gestion des versions du moteur	Restrictions de classe d'instance
RDS for MySQL	Pour obtenir plus d'informations sur la disponibilité des versions et des régions de Performance Insights avec RDS for MySQL, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	Performance Insights n'est pas pris en charge pour les classes d'instances suivantes : • db.t2.micro • db.t2.small • db.t3.micro • db.t3.small • db.t4g.micro • db.t4g.small
Amazon RDS for Microsoft SQL Server	Pour obtenir plus d'informations sur la disponibilité des versions et des régions de Performance Insights avec RDS for SQL Server, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	N/A
Amazon RDS for PostgreSQL	Pour obtenir plus d'informations sur la disponibilité des versions et des régions de Performance Insights avec RDS for PostgreSQL Server, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	N/A

Moteur de base de données Amazon RDS	Versions et régions soumises à la gestion des versions du moteur	Restrictions de classe d'instance
Amazon RDS for Oracle	Pour obtenir plus d'informations sur la disponibilité des versions et des régions de Performance Insights avec RDS for Oracle, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.	N/A

Prise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances

Le tableau suivant fournit les moteurs de base de données Amazon RDS qui prennent en charge les fonctionnalités d'analyse des performances.

Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
Statistiques SQL pour Performance Insights	Tous	Tous	Tous	Tous
Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord Performance Insights pour Amazon RDS	Tous	Tous	RDS for Oracle	Tous
Analyse des performances de	Niveau payant uniquement	Tous	RDS for PostgreSQL	Tous

Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
base de données pour une période donnée				

Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
Consulter les recommandations proactives de Performance Insights	Niveau payant uniquement	• USA Est (Ohio) • USA Est (Virginie du Nord) • USA Ouest (Californie du Nord) • USA Ouest (Oregon) • Asie-Pacifique (Mumbai) • Asie-Pacifique (Séoul) • Asie-Pacifique (Singapour) • Asie-Pacifique (Sydney) • Asie-Pacifique (Tokyo) • Canada (Centre) • Europe (Francfort) • Europe (Irlande) • Europe (Londres) • Europe (Paris)	Tous	Tous

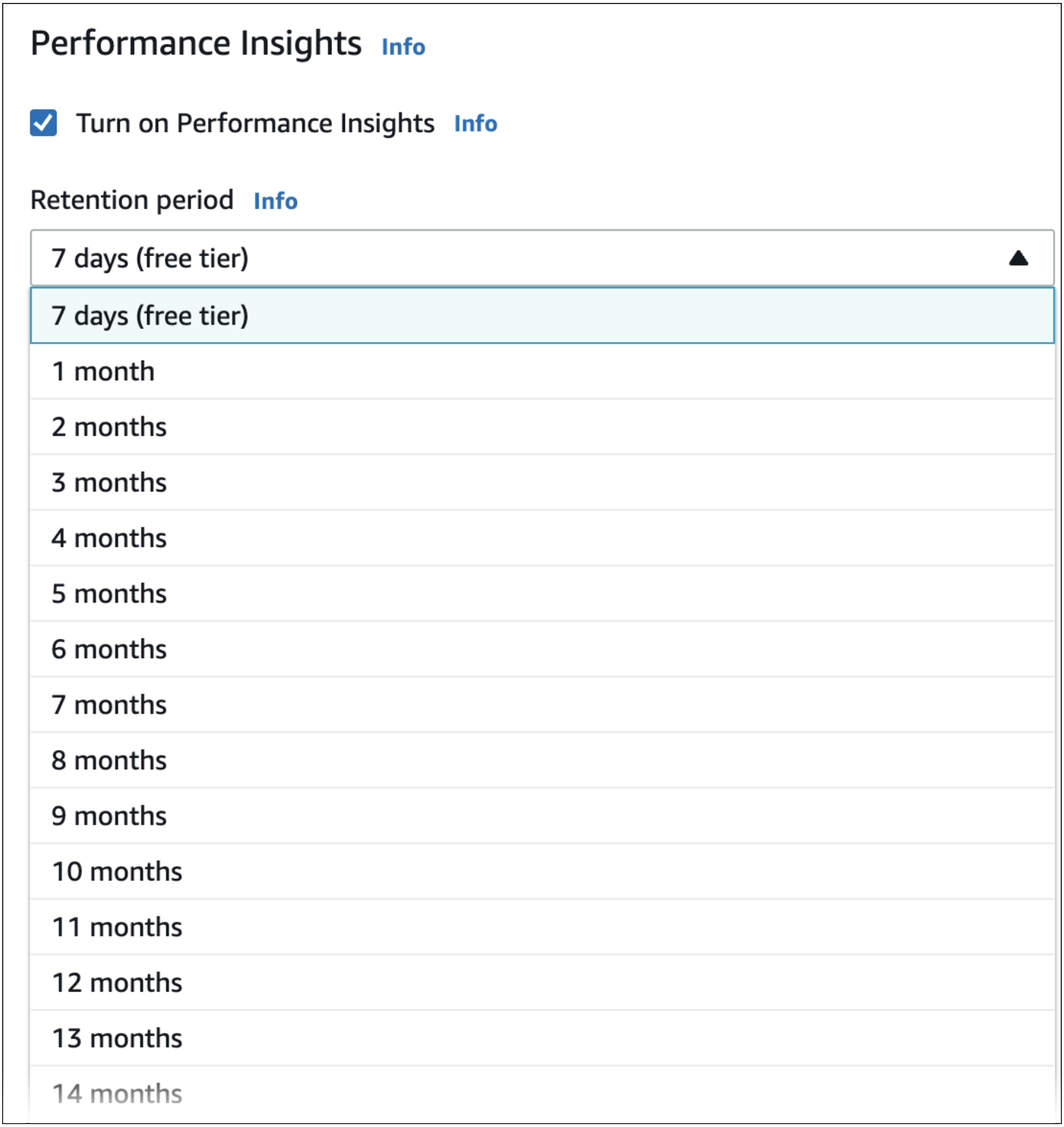
Fonctionnalité	Niveau de tarification	Régions prises en charge	Moteurs de base de données pris en charge	Classes d'instances prises en charge
		• Europe (Stockholm) • Amérique du Sud (São Paulo)		

Tarification et conservation des données pour Performance Insights

Par défaut, Performance Insights inclut 7 jours d'historique des données de performance et 1 million de demandes d'API par mois. Vous pouvez également acheter des périodes de conservation plus longues. Pour des informations complètes sur les prix, consultez la section Performance Insights Pricing (Tarification de Performance Insights).

Dans la console RDS, vous pouvez choisir l'une des périodes de conservation suivantes pour vos données Performance Insights :

- Par défaut (7 jours)
- ***n***mois, où ***n***est un nombre compris entre 1 et 24



Pour savoir comment définir une période de conservation à l'aide du AWS CLI, voirActiver et désactiver Performance Insights pour Amazon RDS.

> **Note**
>
> L'arrêt d'une instance de base de données ou d'un de base de données multi-AZ alors que Performance Insights est activé n'a aucune incidence sur la conservation des données. Lorsqu'une instance de base de données ou un de base de données multi-AZ est arrêté, Performance Insights ne collecte aucune donnée.

Activer et désactiver Performance Insights pour Amazon RDS

> **Important**
>
> AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.
>
> Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).
>
> Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
>
> Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

Vous pouvez activer Performance Insights pour votre instance de de base de données ou votre cluster de base de données multi-AZ lorsque vous le créez. Si nécessaire, vous pouvez le désactiver ultérieurement en modifiant l'instance de base de données de votre de bases de données depuis la console. L'activation ou la désactivation de Performance Insights ne provoquent pas de temps d'arrêt, de redémarrage ou de basculement.

> **Note**
>
> Le schéma de performance est un outil de performance facultatif utilisé par Amazon RDS for MariaDB ou MySQL. Si vous activez ou désactivez le schéma de performance, un redémarrage est requis. Toutefois, si vous activez ou désactivez Performance Insights, aucun redémarrage n'est requis. Pour de plus amples informations, veuillez consulter Présentation du schéma de performance pour Performance Insights sur , Amazon RDS for MariaDB ou MySQL.

L'agent Performance Insights consomme une quantité limitée d'UC et de mémoire sur l'hôte de base de données. Lorsque la charge de base de données est élevée, l'agent limite l'impact sur les performances en collectant des données moins fréquemment.

Console

Dans la console, vous pouvez activer ou désactiver Performance Insights lorsque vous créez ou modifiez une instance de base de données ou un cluster de base de données Multi-AZ.

Activation ou désactivation de Performance Insights lors de la création d'une instance de base de données ou d'un de base de données multi-AZ

Après avoir créé une nouvelle instance de base de données ou un de base de données multi-AZ, Amazon RDS active Performance Insights par défaut. Pour désactiver Performance Insights de bases de données, choisissez l'option Database Insights — Standard et désélectionnez l'option Enable Performance Insights.

Pour plus d’informations, consultez les rubriques suivantes.

- Pour créer une instance de base de données, suivez les instructions relatives à votre moteur de base de données dans Création d'une instance de base de données Amazon RDS.

- Pour créer un cluster de base de données Multi-AZ, suivez les instructions pour votre moteur de base de données présentées dans Création d'un cluster de base de données multi-AZ pour Amazon RDS.

L'image suivante représente la section Performance Insights.



Vous disposez des options suivantes lorsque vous choisissez Activer Performance Insights :

- Conservation (pour le mode standard de Database Insights uniquement) : durée de conservation des données Performance Insights. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.
- AWS KMS key— Spécifiez votre AWS KMS key. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour de plus amples informations, veuillez consulter Modification d'une AWS KMS politique pour Performance Insights.

Activation ou désactivation de Performance Insights lors de la modification d'un cluster de bases de données, d'une de base de données multi-AZ

Dans la console, vous pouvez modifier de base de données ou un cluster de base de données multi-AZ pour gérer Performance Insights.

Pour gérer Performance Insights pour une instance de base de données ou un cluster de base de données multi-AZ (de base de données) à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données).
3. Choisissez une de base de données ou un cluster de base de données multi-AZ, puis choisissez Modifier.
4. Pour activer Performance Insights, sélectionnez Enable Performance Insights. Pour désactiver Performance Insights de bases de données, choisissez l'option Database Insights — Standard et désélectionnez l'option Enable Performance Insights.

   Vous disposez des options suivantes lorsque vous choisissez Activer Performance Insights :

   - Conservation (pour le mode standard de Database Insights uniquement) : durée de conservation des données Performance Insights. Le paramètre de conservation est défini par défaut (7 jours). Pour conserver vos données de performance plus longtemps, indiquez 1 à 24 mois. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.
   - AWS KMS key – Spécifiez votre clé KMS. Performance Insights chiffre toutes les données potentiellement sensibles à l'aide votre clé KMS. Les données sont chiffrées en transit et au repos. Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.
5. Choisissez Continuer.
6. Pour Scheduling of Modifications (Planification des modifications), choisissez Apply immediately (Appliquer immédiatement). Si vous choisissez Apply during the next scheduled maintenance window (Appliquer lors de la prochaine fenêtre de maintenance planifiée), votre instance ignore ce paramètre et active immédiatement Performance Insights.
7. Choisissez Modify instance (Modifier l'instance).

AWS CLI

Lorsque vous utilisez la create-db-instance AWS CLI commande, activez Performance Insights en spécifiant `--enable-performance-insights` et en `--database-insights-mode` réglant sur `advanced` ou`standard`. Pour désactiver Performance Insights, spécifiez `--no-enable-performance-insights` et réglez `database-insights-mode` sur`standard`.

Vous pouvez également spécifier ces valeurs à l'aide des AWS CLI commandes suivantes :

- create-db-cluster
- modify-db-cluster
- create-db-instance-read-réplique
- modify-db-instance
- restore-db-instance-from-s3

Lorsque vous activez Performance Insights dans l'interface CLI, vous pouvez éventuellement spécifier le nombre de jours pour conserver les données de Performance Insights avec l'option `--performance-insights-retention-period`. Vous pouvez spécifier 7 *`month`* * 31 (où *`month`* est un nombre compris entre 1 et 23) ou`731`. Par exemple, si vous souhaitez conserver vos données de performance pendant 3 mois, indiquez 93, soit 3 * 31. La durée par défaut est de 7 jours. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.

L'exemple suivant active Performance Insights pour `sample-db-cluster` et spécifie que les données de Performance Insights sont conservées pendant 93 jours (3 mois).

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-cluster \
 --database-insights-mode standard \
    --db-cluster-identifier sample-db-instance \
    --enable-performance-insights \
    --performance-insights-retention-period 93
```

Dans Windows :

```
aws rds modify-db-cluster ^
 --database-insights-mode standard ^
    --db-cluster-identifier sample-db-instance ^
    --enable-performance-insights ^
    --performance-insights-retention-period 93
```

Si vous spécifiez une période de conservation telle que 94 jours, qui n'est pas une valeur valide, RDS émet une erreur.

```
An error occurred (InvalidParameterValue) when calling the CreateDBInstance
 operation:
Invalid Performance Insights retention period. Valid values are: [7, 31, 62, 93,
 124, 155, 186, 217,
248, 279, 310, 341, 372, 403, 434, 465, 496, 527, 558, 589, 620, 651, 682, 713, 731]
```

> **Note**
>
> Vous ne pouvez activer Performance Insights que pour une instance d'un cluster de base de données où Performance Insights n'est pas géré au niveau du cluster.

RDS API

Lorsque vous créez une nouvelle instance de base de données de bases de données à l'aide de l'DBInstanceopération Create operation Amazon RDS API, activez Performance Insights en définissant `EnablePerformanceInsights` sur. `True` Pour désactiver Performance Insights, définissez `EnablePerformanceInsights` sur `False` et réglez `DatabaseInsightsMode` sur`standard`.

Vous pouvez également spécifier la valeur `EnablePerformanceInsights` à l'aide des opérations d'API suivantes :

- Créer DBCluster (cluster de base de données multi-AZ)
- Modifier DBCluster (cluster de bases de données multi-AZ)
- ModifyDBInstance
- CréerDBInstanceReadReplica
- Restaurer à DBInstance partir de S3

Lorsque vous activez Performance Insights, vous pouvez facultativement spécifier la durée, en jours, de conservation des données de Performance Insights avec le paramètre `PerformanceInsightsRetentionPeriod`. Vous pouvez spécifier 7 *`month`* * 31 (où *`month`* est un nombre compris entre 1 et 23) ou`731`. Par exemple, si vous souhaitez conserver vos données de performance pendant 3 mois, indiquez 93, soit 3 * 31. La durée par défaut est de 7 jours. Pour obtenir plus d'informations sur les périodes de conservation, consultez Tarification et conservation des données pour Performance Insights.

Présentation du schéma de performance pour Performance Insights sur , Amazon RDS for MariaDB ou MySQL

Le schéma de performance est une fonctionnalité facultative pour la surveillance des performances d'exécution d'Amazon RDS for MariaDB ou MySQL à un faible niveau de détails. Le schéma de performance est conçu pour avoir un impact minimal sur les performances de base de données. Performance Insights est une fonctionnalité distincte que vous pouvez utiliser avec ou sans le schéma de performance.

Rubriques

- Présentation du schéma de performance
- Performance Insights et le schéma de performance
- Gestion automatique du schéma de performance par Performance Insights
- Effet d'un redémarrage sur le schéma de performance
- Déterminer si Performance Insights gère le schéma de performance
- Activez le schéma de performance pour , Amazon RDS pour MariaDB ou MySQL.

Présentation du schéma de performance

Le schéma de performance surveille les événements dans les bases de données MariaDB et MySQL. Un événement est une action du serveur de base de données qui consomme du temps et qui a été instrumentée de manière à ce que des informations temporelles puissent être collectées. Voici quelques exemples d'événements :

- Appels de fonction
- Attend le système d'exploitation
- Étapes de l'exécution SQL
- Groupes d'instructions SQL

Le moteur de stockage `PERFORMANCE_SCHEMA` est un mécanisme de mise en œuvre de la fonctionnalité de schéma de performance. Ce moteur collecte les données d'événement à l'aide d'une instrumentation dans le code source de la base de données. Le moteur stocke les événements dans des tables à mémoire uniquement dans la base de données `performance_schema`. Vous pouvez interroger `performance_schema` tout comme vous pouvez interroger d'autres tables. Pour plus

d'informations, consultez la section MySQL Performance Schema (Schéma de performance MySQL) dans le Manuel de référence de MySQL.

Performance Insights et le schéma de performance

Performance Insights et Performance Schema sont des fonctionnalités distinctes, mais elles sont liées. Le comportement de Performance Insights pour Amazon RDS for MariaDB ou MySQL dépend de l'activation ou non du schéma de performance, et si oui, si Performance Insights gère automatiquement le schéma de performance. Le tableau suivant décrit le comportement.

Schéma de performance activé	Mode de gestion de Performance Insights	Comportement de Performance Insights
Oui	Automatique	• Collecte des informations de surveillance détaillées et de bas niveau • Collecte des métriques de la session active toutes les secondes • Affiche la charge de la base de données classée par événements d'attente détaillés, que vous pouvez utiliser pour identifier les goulots d'étranglement
Oui	Manuelle	• Collecte les événements d'attente et les métriques par SQL • Collecte des métriques de la session active toutes les secondes • Les rapports sur les états des utilisateurs, tels que l'insertion et l'envoi, ne vous aident pas à identifier les goulots d'étranglement
Non	N/A	• Il ne collecte pas les événements d'attente, les métriques par SQL ou d'autres informations de surveillance détaillées et de bas niveau

Schéma de performance activé	Mode de gestion de Performance Insights	Comportement de Performance Insights
		• Il collecte des métriques de la session active toutes les cinq secondes au lieu de toutes les secondes • Les rapports sur les états des utilisateurs, tels que l'insertion et l'envoi, ne vous aident pas à identifier les goulots d'étranglement

Gestion automatique du schéma de performance par Performance Insights

Le schéma de performance est également activé lorsque vous créez une instance de base de données Amazon RDS for MariaDB ou MySQL avec Performance Insights activé. Le cas échéant, Performance Insights gère automatiquement vos paramètres de schéma de performance. Il s'agit de la configuration recommandée.

Lorsque Performance Insights gère automatiquement le schéma de performance, sa source `performance_schema` est`System default`.

> **Note**
>
> La gestion automatique du schéma de performance n'est pas prise en charge pour la classe d'instance t4g.medium.

Si vous modifiez la valeur du paramètre `performance_schema` manuellement et que vous souhaitez ensuite passer à la gestion automatique, consultez Activez le schéma de performance pour , Amazon RDS pour MariaDB ou MySQL..

> **Important**
>
> Lorsque Performance Insights active le schéma de performance, il ne modifie pas les valeurs du groupe de paramètres. Toutefois, les valeurs sont modifiées sur les instances de base de données en cours d'exécution. La seule façon de voir les valeurs modifiées est d'exécuter la commande `SHOW GLOBAL VARIABLES`.

Effet d'un redémarrage sur le schéma de performance

Performance Insights et le schéma de performance diffèrent dans leurs exigences en matière de redémarrage des instances de base de données :

Schéma de performance

Pour activer ou désactiver cette fonction, vous devez redémarrer l'instance de base de données.

Performance Insights

Pour activer ou désactiver cette fonction, il n'est pas nécessaire de redémarrer l'instance de base de données.

Si le schéma de performance n'est pas activé et que vous activez Performance Insights sans redémarrer l'instance de base de données, le schéma de performance ne sera pas activé.

Déterminer si Performance Insights gère le schéma de performance

Pour savoir si Performance Insights gère actuellement le schéma de performance pour toutes les versions principales du moteur prises en charge, consultez le tableau suivant.

Définition du paramètre performance_schema	Paramétrage de la colonne Source	Performance Insights gère le schéma de performance
`0`	`System default`	Oui
`0` ou `1`	`Modified`	Non

Dans la procédure suivante, vous déterminez si Performance Insights gère automatiquement le schéma de performance.

Pour déterminer si Performance Insights gère automatiquement le schéma de performance

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez Groupes de paramètres.
3. Sélectionnez le nom du groupe de paramètres pour votre instance de base de données.

4. Entrez **`performance_schema`** dans la barre de recherche.
5. Vérifiez si Source est la valeur par défaut du système et si la valeur est 0. Si c'est le cas, Performance Insights gère automatiquement le schéma de performance.

   Dans l'exemple présenté ici, Performance Insights ne gère pas automatiquement le schéma de performance.



Activez le schéma de performance pour , Amazon RDS pour MariaDB ou MySQL.

Supposons que Performance Insights soit activé pour votre instance de base de données ou votre cluster de base de données Multi-AZ mais qu'il ne gère pas actuellement le schéma de performance. Si vous voulez permettre à Performance Insights de gérer automatiquement le schéma de performance, suivez les étapes suivantes.

Pour configurer le schéma de performance pour une gestion automatique

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Groupes de paramètres.
3. Sélectionnez le nom du groupe de paramètres pour votre instance de base de données ou votre cluster de base de données Multi-AZ.
4. Choisissez Modifier.
5. Entrez **`performance_schema`** dans la barre de recherche.
6. Sélectionnez le paramètre `performance_schema`.
7. Choisissez Définir à la valeur par défaut.
8. Confirmez en choisissant Définir les valeurs par défaut.
9. Choisissez Save Changes (Enregistrer les modifications).
10. Redémarrage de l'instance de base de données ou du cluster de base de données Multi-AZ.

> **Important**
>
> Chaque fois que vous activez ou désactivez le schéma de performance, veillez à redémarrer l'instance de base de données ou le cluster de base de données Multi-AZ.

Pour plus d'informations sur la modification des paramètres d'instance de base de données, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora). Pour de plus amples informations sur le tableau de bord, veuillez consulter Analyse des métriques à l'aide du tableau de bord de Performance Insights. Pour plus d'informations sur le schéma de performance MySQL, consultez MySQL Performance Schema (pour 8.0) et MySQL Performance Schema (pour 8.4) dans la documentation MySQL.

Configuration des politiques d'accès pour Performance Insights

Pour accéder à Performance Insights, un directeur doit disposer des autorisations appropriées auprès de AWS Identity and Access Management (IAM). Vous pouvez accorder l'accès des manières suivantes :

- Attachez la politique gérée par `AmazonRDSPerformanceInsightsReadOnly` à un ensemble d'autorisations ou à un rôle permettant d'accéder à toutes les opérations en lecture seule de l'API Performance Insights. Joignez les CloudWatch autorisations suivantes : `GetMetricStatisticsListMetrics`, et. `GetMetricData` Pour plus d'informations sur CloudWatch les autorisations, consultez la référence CloudWatch des autorisations Amazon.
- Attachez la politique gérée par `AmazonRDSPerformanceInsightsFullAccess` à un ensemble d'autorisations ou à un rôle permettant d'accéder à toutes les opérations de l'API Performance Insights. Joignez les CloudWatch autorisations suivantes : `GetMetricStatisticsListMetrics`, et. `GetMetricData` Pour plus d'informations sur CloudWatch les autorisations, consultez la référence CloudWatch des autorisations Amazon.
- Créez une politique IAM personnalisée et attachez-la à un jeu d'autorisations ou à un rôle.

Si vous avez spécifié une clé gérée par le client lorsque vous avez activé Performance Insights, assurez-vous que les utilisateurs de votre compte disposent des `kms:GenerateDataKey` autorisations `kms:Decrypt` et sur le AWS KMS key.

Dans les sections suivantes, associez une politique AWS gérée à un principal IAM, créez une stratégie IAM personnalisée, modifiez une AWS KMS politique et accordez un accès détaillé à Performance Insights.

Rubriques

- Fixer le AmazonRDSPerformanceInsightsReadOnly politique à l'égard d'un directeur IAM
- Fixer le AmazonRDSPerformanceInsightsFullAccess politique à l'égard d'un directeur IAM
- Création d'une IAM politique personnalisée pour Performance Insights
- Modification d'une AWS KMS politique pour Performance Insights
- Octroi d'un accès détaillé à Performance Insights

Fixer le AmazonRDSPerformanceInsightsReadOnly politique à l'égard d'un directeur IAM

`AmazonRDSPerformanceInsightsReadOnly`est une politique AWS gérée qui donne accès à toutes les opérations en lecture seule de l'API Amazon RDS Performance Insights.

Si vous attachez `AmazonRDSPerformanceInsightsReadOnly` à un jeu d'autorisations ou à un rôle, le destinataire peut utiliser Performance Insights avec d'autres fonctions de la console.

Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSPerformance InsightsReadOnly.

Fixer le AmazonRDSPerformanceInsightsFullAccess politique à l'égard d'un directeur IAM

`AmazonRDSPerformanceInsightsFullAccess`est une politique AWS gérée qui donne accès à toutes les opérations de l'API Amazon RDS Performance Insights.

Si vous attachez `AmazonRDSPerformanceInsightsFullAccess` à un jeu d'autorisations ou à un rôle, le destinataire peut utiliser Performance Insights avec d'autres fonctions de la console.

Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSPerformance InsightsFullAccess.

Création d'une IAM politique personnalisée pour Performance Insights

Pour les utilisateurs qui ne disposent pas de la `AmazonRDSPerformanceInsightsFullAccess` politique `AmazonRDSPerformanceInsightsReadOnly` OR, vous pouvez accorder l'accès à

Performance Insights en créant ou en modifiant une IAM politique gérée par l'utilisateur. Lorsque vous associez la politique à un ensemble d'IAMautorisations ou à un rôle, le destinataire peut utiliser Performance Insights.

Créer une stratégie personnalisée

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Politiques.
3. Sélectionnez Create policy (Créer une politique).
4. Sur la page Créer une politique, choisissez l'JSONoption.
5. Copiez et collez le texte fourni dans la section du document de JSON politique du Guide de référence des politiques AWS gérées pour AmazonRDSPerformanceInsightsReadOnly ou AmazonRDSPerformanceInsightsFullAccesspolitique.
6. Choisissez Examiner une stratégie.
7. Indiquez un nom pour la stratégie et éventuellement une description, puis choisissez Créer une stratégie.

Vous pouvez désormais attacher la politique à un jeu d'autorisations ou à un rôle. La procédure suivante suppose que vous disposez déjà d'un utilisateur à cette fin.

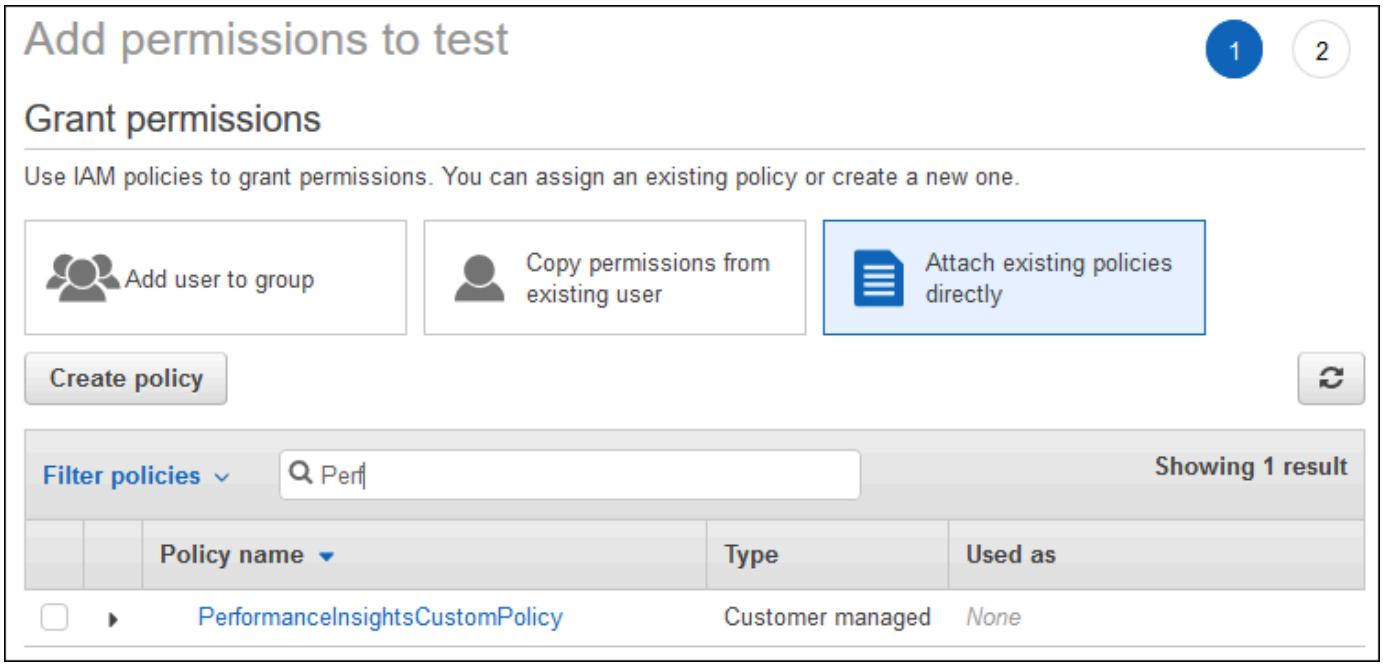
Pour attacher la politique à un utilisateur

1. Ouvrez la console IAM à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le volet de navigation, sélectionnez Users.
3. Choisissez un utilisateur existant dans la liste.

> **Important**
>
> Pour utiliser Performance Insights, assurez-vous d'avoir accès à Amazon RDS en plus de respecter la politique personnalisée. Par exemple, la politique `AmazonRDSPerformanceInsightsReadOnly` prédéfinie fournit un accès en lecture seule à Amazon. RDS Pour de plus amples informations, veuillez consulter Gestion des accès à l'aide de politiques.

4. Sur la page Récapitulatif, choisissez Ajouter des autorisations.

5. Choisissez Attacher directement les stratégies existantes. Dans le champ Rechercher, saisissez les premiers caractères du nom de votre police, comme indiqué dans l'image suivante.



6. Choisissez votre stratégie, puis sélectionnez Suivant : Vérification.
7. Choisissez Add permissions.

Modification d'une AWS KMS politique pour Performance Insights

Performance Insights utilise un AWS KMS key pour chiffrer les données sensibles. Lorsque vous activez Performance Insights via l'API ou la console, vous pouvez effectuer l'une ou l'autre des opérations suivantes :

- Choisissez la valeur par défaut Clé gérée par AWS.

  Amazon RDS utilise le Clé gérée par AWS pour votre nouvelle instance de base de données. Amazon RDS crée une Clé gérée par AWS pour votre Compte AWS. Vous Compte AWS avez un Amazon RDS différent Clé gérée par AWS pour chacun Région AWS d'entre eux.

- Choisissez une clé gérée par le client.

  Si vous spécifiez une clé gérée par le client, les utilisateurs de votre compte qui appellent l'API Performance Insights ont besoin des autorisations `kms:Decrypt` et `kms:GenerateDataKey` sur la clé KMS. Vous pouvez configurer ces autorisations via des politiques IAM. Toutefois, nous vous recommandons de gérer ces autorisations via votre politique de clé KMS. Pour plus d'informations,

consultez Politiques de clé dans AWS KMS dans le Guide du développeur AWS Key Management Service .

Example

L'exemple suivant montre comment ajouter des instructions à votre politique KMS. Ces instructions permettent d'accéder à Performance Insights. Selon la façon dont vous utilisez la clé KMS, vous pouvez modifier certaines restrictions. Avant d'ajouter des instructions à votre politique, supprimez tous les commentaires.

JSON

```
{
    "Version" : "2012-10-17",
    "Id" : "your-policy",
    "Statement" : [
        {
            "Sid" : "AllowViewingRDSPerformanceInsights",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::444455556666:role/Role1"
                ]
                },
             "Action": [
                "kms:Decrypt",
                "kms:GenerateDataKey"
                ],
            "Resource": "*",
            "Condition" : {
            "StringEquals" : {
                "kms:ViaService" : "rds.us-east-1.amazonaws.com"
                },
            "ForAnyValue:StringEquals": {
                "kms:EncryptionContext:aws:pi:service": "rds",
                "kms:EncryptionContext:service": "pi",
                "kms:EncryptionContext:aws:rds:db-id": "db-
AAAAABBBBBCCCCDDDDDEEEEE"
                }
            }
        }
```

```
    ]
}
```

Comment Performance Insights utilise les clés gérées par le AWS KMS client

L'analyse des performances utilise les clés gérées par le client pour chiffrer les données sensibles. Lorsque vous activez l'analyse des performances, vous pouvez fournir une clé AWS KMS via l'API. L'analyse des performances crée des autorisations KMS sur cette clé. Il utilise la clé et effectue les opérations nécessaires au traitement des données sensibles. Les données sensibles incluent des champs tels que l'utilisateur, la base de données, l'application et le texte de requête SQL. L'analyse des performances garantit que les données restent chiffrées à la fois au repos et en transit.

Comment fonctionne Performance Insights IAM AWS KMS

IAM accorde des autorisations à des personnes spécifiques APIs. Performance Insights possède le public suivant APIs, que vous pouvez restreindre à l'aide de politiques IAM :

- `DescribeDimensionKeys`
- `GetDimensionKeyDetails`
- `GetResourceMetadata`
- `GetResourceMetrics`
- `ListAvailableResourceDimensions`
- `ListAvailableResourceMetrics`

Vous pouvez utiliser les demandes d'API suivantes pour obtenir des données sensibles.

- `DescribeDimensionKeys`
- `GetDimensionKeyDetails`
- `GetResourceMetrics`

Lorsque vous utilisez l'API pour obtenir des données sensibles, l'analyse des performances exploite les informations d'identification de l'appelant. Cette vérification garantit que l'accès aux données sensibles est limité aux personnes ayant accès à la clé KMS.

Lorsque vous les appelez APIs, vous avez besoin d'autorisations pour appeler l'API via la politique IAM et d'autorisations pour invoquer l'`kms:decrypt`action via la politique AWS KMS clé.

L'API `GetResourceMetrics` peut renvoyer des données sensibles et non sensibles. Les paramètres de demande déterminent si la réponse doit inclure des données sensibles. L'API renvoie des données sensibles lorsque la demande inclut une dimension sensible dans les paramètres de filtre ou de regroupement.

Pour plus d'informations sur les dimensions que vous pouvez utiliser avec l'`GetResourceMetrics`API, consultez DimensionGroup.

Example Exemples

L'exemple suivant demande les données sensibles pour le groupe `db.user` :

```
POST / HTTP/1.1
Host: <Hostname>
Accept-Encoding: identity
X-Amz-Target: PerformanceInsightsv20180227.GetResourceMetrics
Content-Type: application/x-amz-json-1.1
User-Agent: <UserAgentString>
X-Amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>,
 Signature=<Signature>
Content-Length: <PayloadSizeBytes>
{
  "ServiceType": "RDS",
  "Identifier": "db-ABC1DEFGHIJKL2MNOPQRSTUV3W",
  "MetricQueries": [
    {
      "Metric": "db.load.avg",
      "GroupBy": {
        "Group": "db.user",
        "Limit": 2
      }
    }
  ],
  "StartTime": 1693872000,
  "EndTime": 1694044800,
  "PeriodInSeconds": 86400
}
```

Example

L'exemple suivant demande les données non sensibles pour la métrique `db.load.avg` :

```
POST / HTTP/1.1
Host: <Hostname>
Accept-Encoding: identity
X-Amz-Target: PerformanceInsightsv20180227.GetResourceMetrics
Content-Type: application/x-amz-json-1.1
User-Agent: <UserAgentString>
X-Amz-Date: <Date>
Authorization: AWS4-HMAC-SHA256 Credential=<Credential>, SignedHeaders=<Headers>,
 Signature=<Signature>
Content-Length: <PayloadSizeBytes>
{
    "ServiceType": "RDS",
    "Identifier": "db-ABC1DEFGHIJKL2MNOPQRSTUV3W",
    "MetricQueries": [
        {
            "Metric": "db.load.avg"
        }
    ],
    "StartTime": 1693872000,
    "EndTime": 1694044800,
    "PeriodInSeconds": 86400
}
```

Octroi d'un accès détaillé à Performance Insights

Le contrôle d'accès précis offre des moyens supplémentaires de contrôler l'accès à Performance Insights. Ce contrôle d'accès permet d'autoriser ou de refuser l'accès à des dimensions individuelles pour `GetResourceMetrics` `DescribeDimensionKeys` les actions `GetDimensionKeyDetails` Performance Insights. Pour utiliser un accès détaillé, spécifiez les dimensions dans la politique IAM à l'aide de clés de condition. L'évaluation de l'accès suit la logique d'évaluation de la politique IAM. Pour de plus amples informations, consultez Logique d’évaluation des politiques dans le Guide de l’utilisateur IAM. Si la déclaration de politique IAM ne spécifie aucune dimension, elle contrôle l'accès à toutes les dimensions pour l'action spécifiée. Pour la liste des dimensions disponibles, voir DimensionGroup.

Pour connaître les dimensions auxquelles vos informations d'identification sont autorisées à accéder, utilisez le `AuthorizedActions` paramètre dans `ListAvailableResourceDimensions` et spécifiez l'action. Les valeurs autorisées pour `AuthorizedActions` sont les suivantes :

- `GetResourceMetrics`
- `DescribeDimensionKeys`
- `GetDimensionKeyDetails`

Par exemple, si vous spécifiez `GetResourceMetrics` le `AuthorizedActions` paramètre, `ListAvailableResourceDimensions` renvoie la liste des dimensions auxquelles l'`GetResourceMetrics`action est autorisée à accéder. Si vous spécifiez plusieurs actions dans le `AuthorizedActions` paramètre, il `ListAvailableResourceDimensions` renvoie une intersection de dimensions auxquelles ces actions sont autorisées à accéder.

Example

L'exemple suivant fournit l'accès aux dimensions `GetResourceMetrics` et aux `DescribeDimensionKeys` actions spécifiées.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowToDiscoverDimensions",
            "Effect": "Allow",
            "Action": [
                "pi:ListAvailableResourceDimensions"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ]
        },
        {
            "Sid": "SingleAllow",
            "Effect": "Allow",
            "Action": [
                "pi:GetResourceMetrics",
```

```
                "pi:DescribeDimensionKeys"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ],
            "Condition": {
                "ForAllValues:StringEquals": {
                    "pi:Dimensions": [
                        "db.sql_tokenized.id",
                        "db.sql_tokenized.statement"
                    ]
                }
            }
        }


    ]
}
```

Voici la réponse pour la dimension demandée :

```
 // ListAvailableResourceDimensions API
// Request
{
    "ServiceType": "RDS",
    "Identifier": "db-ABC1DEFGHIJKL2MNOPQRSTUV3W",
    "Metrics": [ "db.load" ],
    "AuthorizedActions": ["DescribeDimensionKeys"]
}

// Response
{
    "MetricDimensions": [ {
        "Metric": "db.load",
        "Groups": [
            {
                "Group": "db.sql_tokenized",
                "Dimensions": [
                    { "Identifier": "db.sql_tokenized.id" },
```

```
                //  { "Identifier": "db.sql_tokenized.db_id" }, // not included
 because not allows in the IAM Policy
                  { "Identifier": "db.sql_tokenized.statement" }
              ]
           }

       ] }
    ]
}
```

L'exemple suivant indique une autorisation et deux refus d'accès pour les dimensions.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
          {
            "Sid": "AllowToDiscoverDimensions",
            "Effect": "Allow",
            "Action": [
                "pi:ListAvailableResourceDimensions"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ]
          },

          {
            "Sid": "O01AllowAllWithoutSpecifyingDimensions",
            "Effect": "Allow",
            "Action": [
                "pi:GetResourceMetrics",
                "pi:DescribeDimensionKeys"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ]
        },

        {
```

```
            "Sid": "O01DenyAppDimensionForAll",
            "Effect": "Deny",
            "Action": [
                "pi:GetResourceMetrics",
                "pi:DescribeDimensionKeys"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "pi:Dimensions": [
                        "db.application.name"
                    ]
                }
            }
        },

        {
            "Sid": "O01DenySQLForGetResourceMetrics",
            "Effect": "Deny",
            "Action": [
                "pi:GetResourceMetrics"
            ],
            "Resource": [
                "arn:aws:pi:us-east-1:123456789012:metrics/rds/db-
ABC1DEFGHIJKL2MNOPQRSTUV3W"
            ],
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "pi:Dimensions": [
                        "db.sql_tokenized.statement"
                    ]
                }
            }
        }
    ]
}
```

Les réponses aux dimensions demandées sont les suivantes :

```
   // ListAvailableResourceDimensions API
// Request
{
    "ServiceType": "RDS",
    "Identifier": "db-ABC1DEFGHIJKL2MNOPQRSTUV3W",
    "Metrics": [ "db.load" ],
    "AuthorizedActions": ["GetResourceMetrics"]
}

// Response
{
    "MetricDimensions": [ {
        "Metric": "db.load",
        "Groups": [
            {
                "Group": "db.application",
                "Dimensions": [

                  // removed from response because denied by the IAM Policy
                  //  { "Identifier": "db.application.name" }
                ]
            },
            {
                "Group": "db.sql_tokenized",
                "Dimensions": [
                    { "Identifier": "db.sql_tokenized.id" },
                    { "Identifier": "db.sql_tokenized.db_id" },

                  // removed from response because denied by the IAM Policy
                  //  { "Identifier": "db.sql_tokenized.statement" }
                ]
            },
            ...
        ] }
    ]
}
```

```
// ListAvailableResourceDimensions API
// Request
{
    "ServiceType": "RDS",
```

```
    "Identifier": "db-ABC1DEFGHIJKL2MNOPQRSTUV3W",
    "Metrics": [ "db.load" ],
    "AuthorizedActions": ["DescribeDimensionKeys"]
}

// Response
{
    "MetricDimensions": [ {
        "Metric": "db.load",
        "Groups": [
            {
                "Group": "db.application",
                "Dimensions": [
                  // removed from response because denied by the IAM Policy
                  //  { "Identifier": "db.application.name" }
                ]
            },
            {
                "Group": "db.sql_tokenized",
                "Dimensions": [
                    { "Identifier": "db.sql_tokenized.id" },
                    { "Identifier": "db.sql_tokenized.db_id" },

                  // allowed for DescribeDimensionKeys because our IAM Policy
                  // denies it only for GetResourceMetrics
                    { "Identifier": "db.sql_tokenized.statement" }
                ]
            },
            ...
        ] }
    ]
}
```

Analyse des métriques à l'aide du tableau de bord de Performance Insights

> **Important**
>
> AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. L'API Performance Insights continuera d'exister sans modification de prix. Les coûts de l'API

Performance Insights apparaîtront sur votre AWS facture avec le coût de CloudWatch Database Insights.
Nous vous recommandons de mettre à niveau toutes les instances de base de données de de bases de données utilisant le niveau payant de Performance Insights vers le mode avancé de Database Insights avant le 30 novembre 2025. Pour plus d'informations sur la mise à niveau vers le mode avancé de Database Insights, consultezActivation du mode avancé de Database Insights pour Amazon RDS (Aurora).
Si vous n'effectuez aucune action, les instances de de base de données utilisant Performance Insights utiliseront par défaut le mode Standard de Database Insights. Avec le mode standard de Database Insights, vous risquez de perdre l'accès à l'historique des données de performance au-delà de 7 jours et de ne pas être en mesure d'utiliser les plans d'exécution et les fonctionnalités d'analyse à la demande dans la console Amazon RDS. Après le 30 novembre 2025, seul le mode avancé de Database Insights prendra en charge les plans d'exécution et les analyses à la demande.
Avec CloudWatch Database Insights, vous pouvez surveiller la charge de base de données de votre parc de bases de données et analyser et résoudre les problèmes de performance à grande échelle. Pour plus d'informations sur Database Insights, consultezSurveillance des bases de données Amazon RDS avec CloudWatch Database Insights. Pour plus d'informations sur les tarifs, consultez Amazon CloudWatch Pricing.

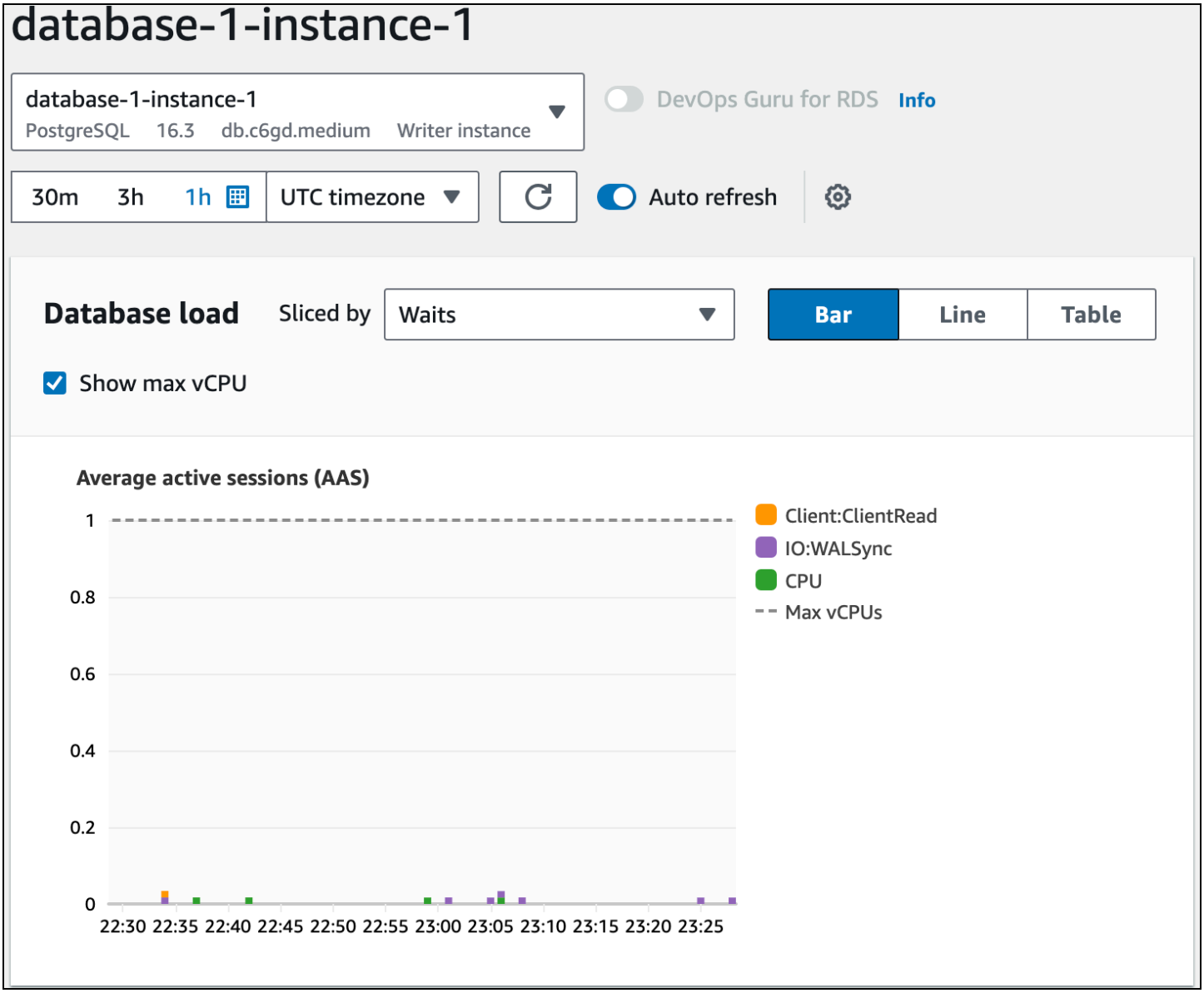
Le tableau de bord de Performance Insights contient des informations sur les performances des bases de données qui vous aideront à analyser et à résoudre les problèmes de performances. Sur la page de tableau de bord principale, vous pouvez afficher des informations concernant la charge de la base de données. Vous pouvez « trancher » la charge de base de données en différentes dimensions, telles que les événements d'attente ou SQL.

Tableau de bord Performance Insights

- Présentation du tableau de bord Performance Insights
- Accès au tableau de bord Performance Insights
- Analyse de la charge de base de données par événements d'attente
- Analyse des performances de base de données pour une période donnée
- Analyse des requêtes à l'aide de SQL l'onglet Top de Performance Insights
- Analyse de la PDB charge Oracle la plus élevée
- Analyse des plans d'exécution à l'aide du tableau de bord Performance Insights pour Amazon RDS

Présentation du tableau de bord Performance Insights

Le tableau de bord est le moyen le plus simple d'interagir avec Performance Insights. L'exemple suivant montre le tableau de bord d'une SQL instance de base de données Postgre.

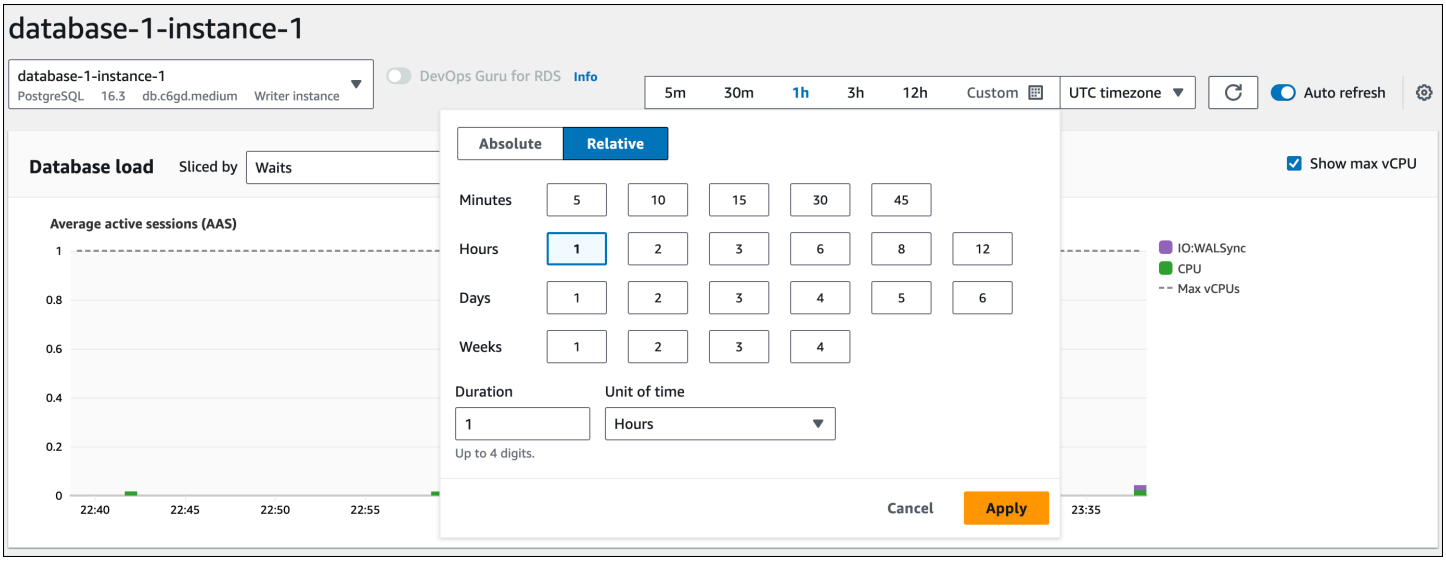


Rubriques

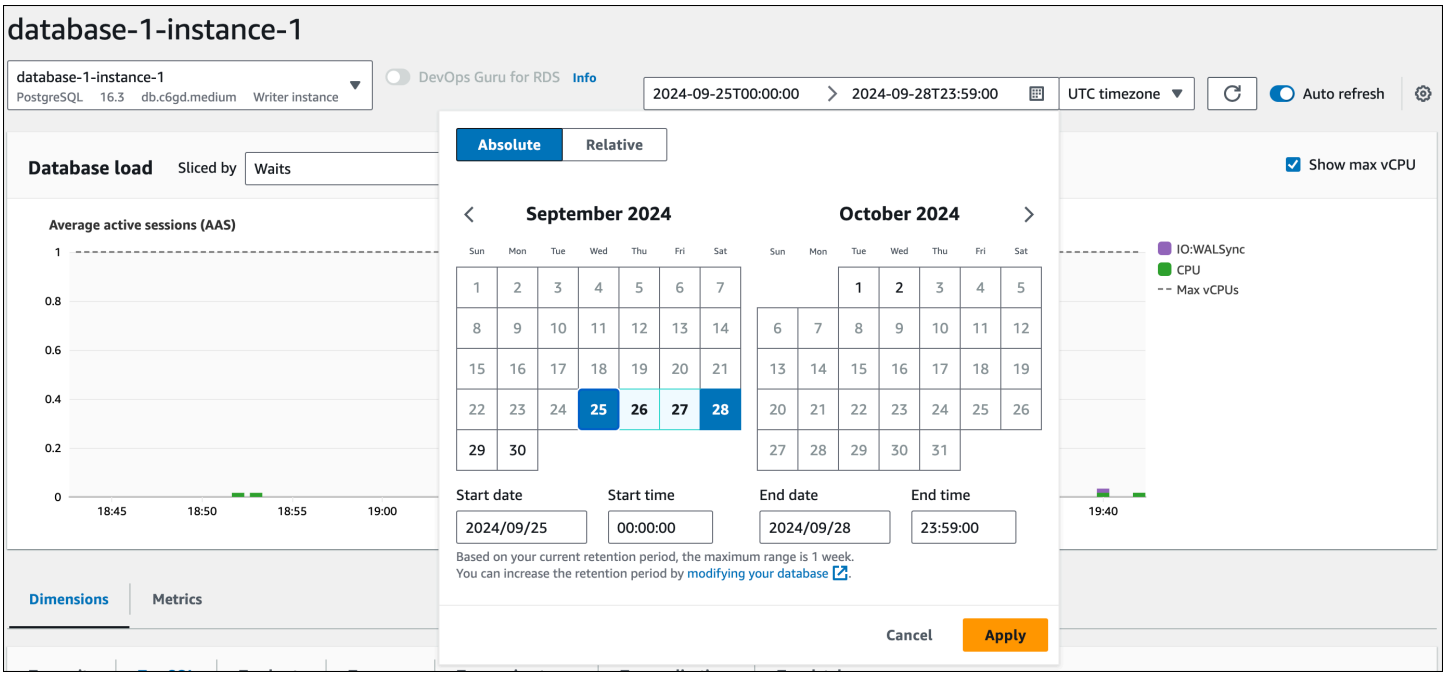
- Filtre de plage de temps
- Graphique Counter Metrics (Métriques de compteur)
- Graphique Database Load (Charge de la base de données)
- Tableau des dimensions principales

Filtre de plage de temps

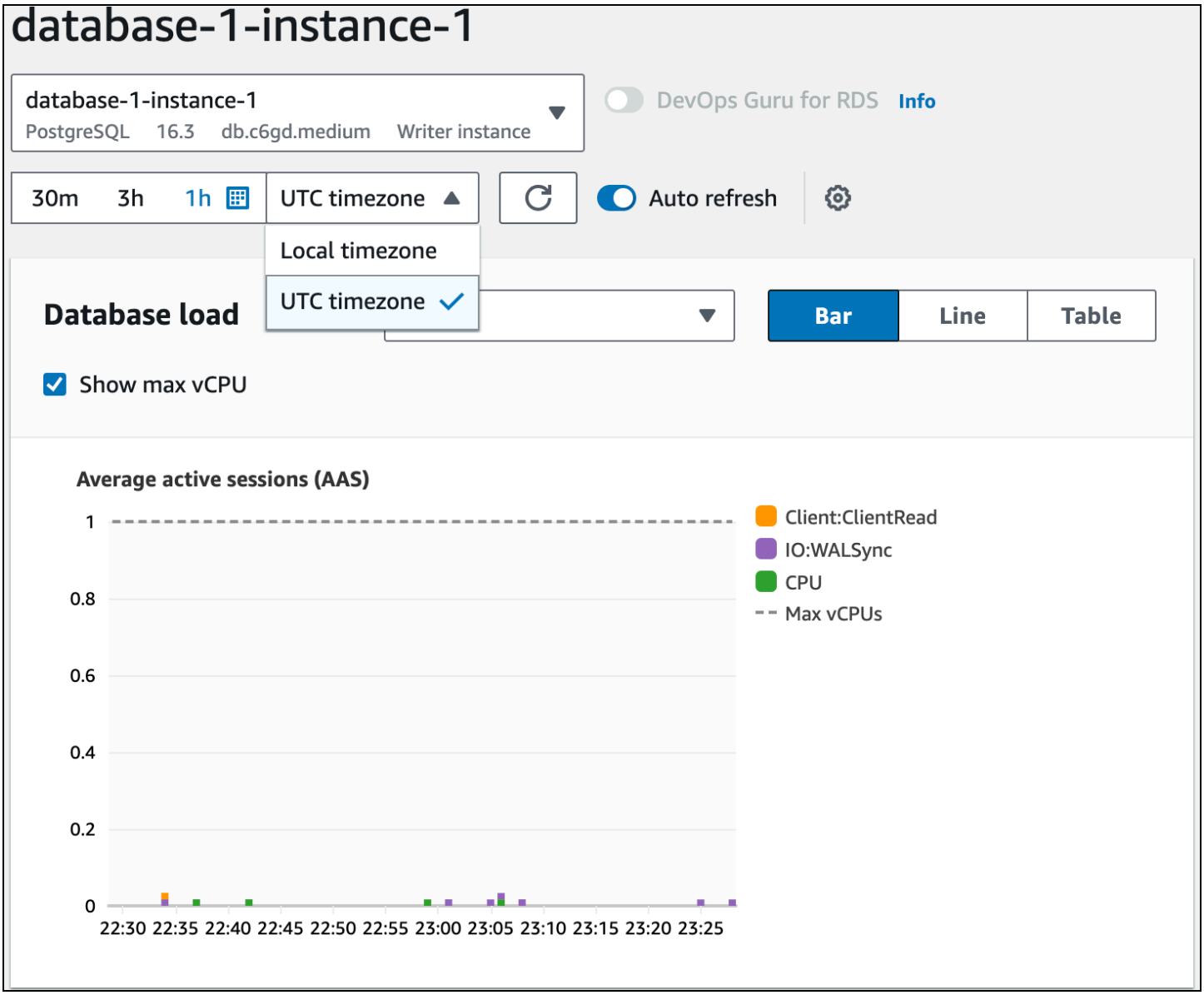
Par défaut, le tableau de bord de Performance Insights affiche la charge de la base de données pour la dernière heure. Vous pouvez régler cette plage pour qu'elle soit aussi courte que 5 minutes ou aussi longue que 2 ans. Vous pouvez également sélectionner une plage relative personnalisée.



Vous pouvez sélectionner une plage absolue avec une date et une heure de début et de fin. L'exemple suivant montre la plage horaire commençant à minuit le 25/09/24 et se terminant à 23 h 59 le 28/09/24.



Par défaut, le fuseau horaire du tableau de bord Performance Insights est Coordinated Universal Time (UTC). Vous pouvez également choisir le fuseau horaire local.

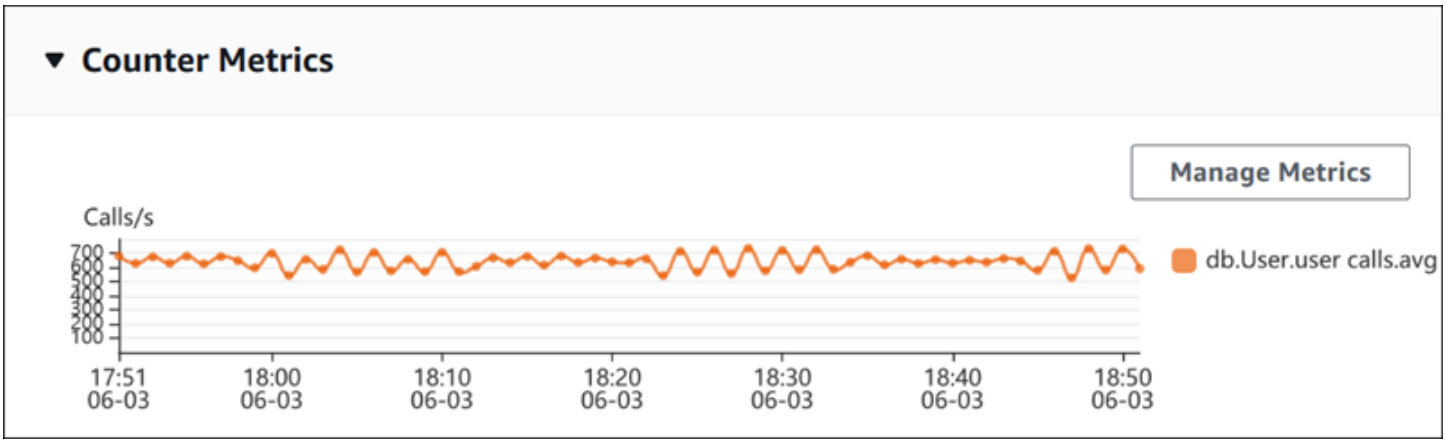


Graphique Counter Metrics (Métriques de compteur)

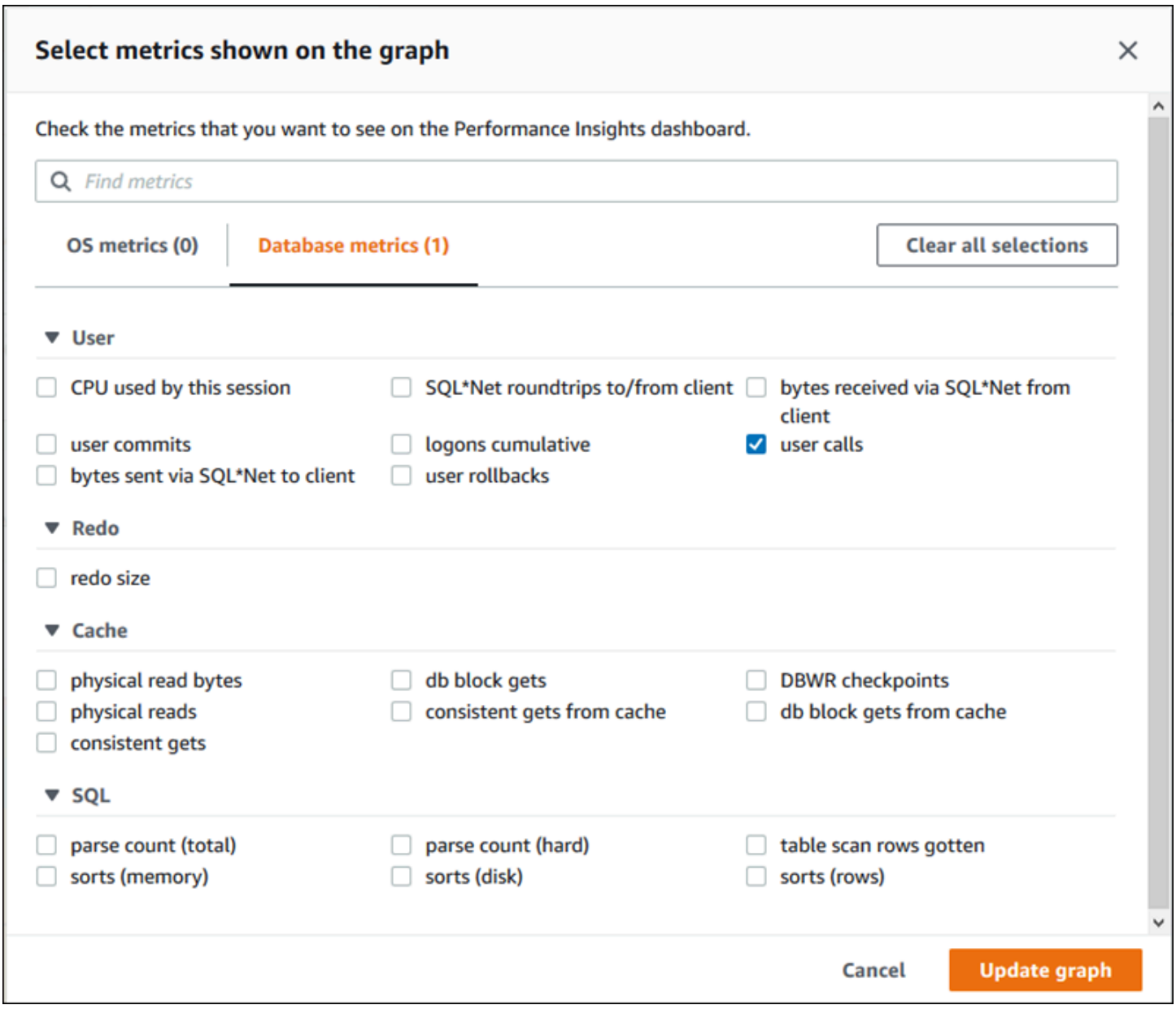
Grâce aux métriques de compteur, vous pouvez personnaliser le tableau de bord de Performance Insights de sorte à inclure jusqu'à 10 graphiques supplémentaires. Ces graphiques présentent une dizaine de métriques de performances de base de données et de système d'exploitation. Vous pouvez établir des corrélations entre ces informations et la charge de la base de données pour identifier et analyser les problèmes de performances.

Le graphique Counter Metrics (Métriques de compteur) affiche les données des compteurs de performances. Les métriques par défaut varient en fonction du moteur de base de données :

- MariaDB SQL et moi — `db.SQL.Innodb_rows_read.avg`
- Oracle – `db.User.user calls.avg`
- Microsoft SQL Server — `db.Databases.Active Transactions(_Total).avg`
- Poster — SQL `db.Transactions.xact_commit.avg`



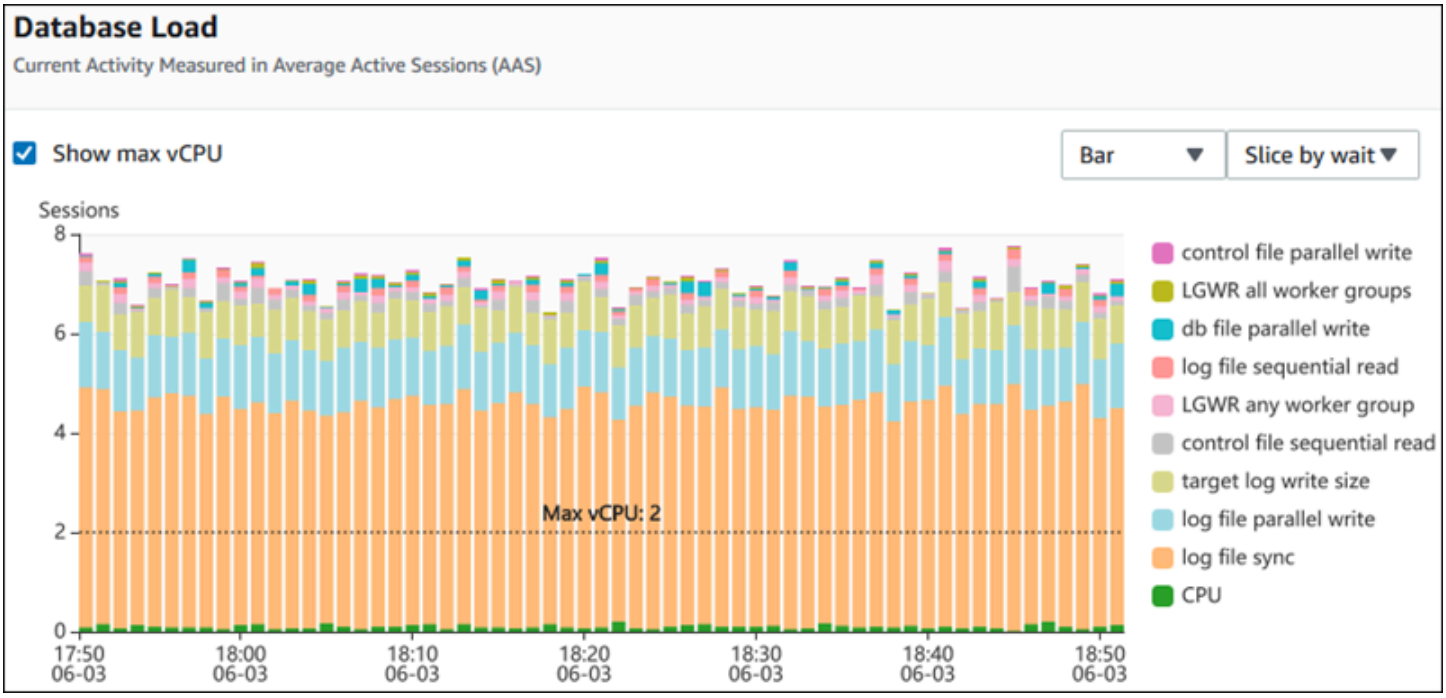
Pour changer de compteurs de performance, choisissez Manage Metrics (Gérer les métriques). Vous pouvez sélectionner plusieurs métriques de système d'exploitation ou métriques de base de données, comme illustré dans la capture d'écran suivante. Pour afficher les détails relatifs à une métrique, passez la souris sur le nom de la métrique.



Pour obtenir une description des métriques de compteur que vous pouvez ajouter pour chaque moteur de base de données, voir Métrique de compteur de Performance Insights.

Graphique Database Load (Charge de la base de données)

Le graphique de charge de base de données montre comment l'activité de la base de données se compare à la capacité de l'instance de base de données telle que représentée par la CPU ligne Max v. Par défaut, le graphique en courbes empilées représente la charge de la base de données sous forme de sessions actives en moyenne par unité de temps. La charge de la base de données est découpée (groupée) par états d'attente.

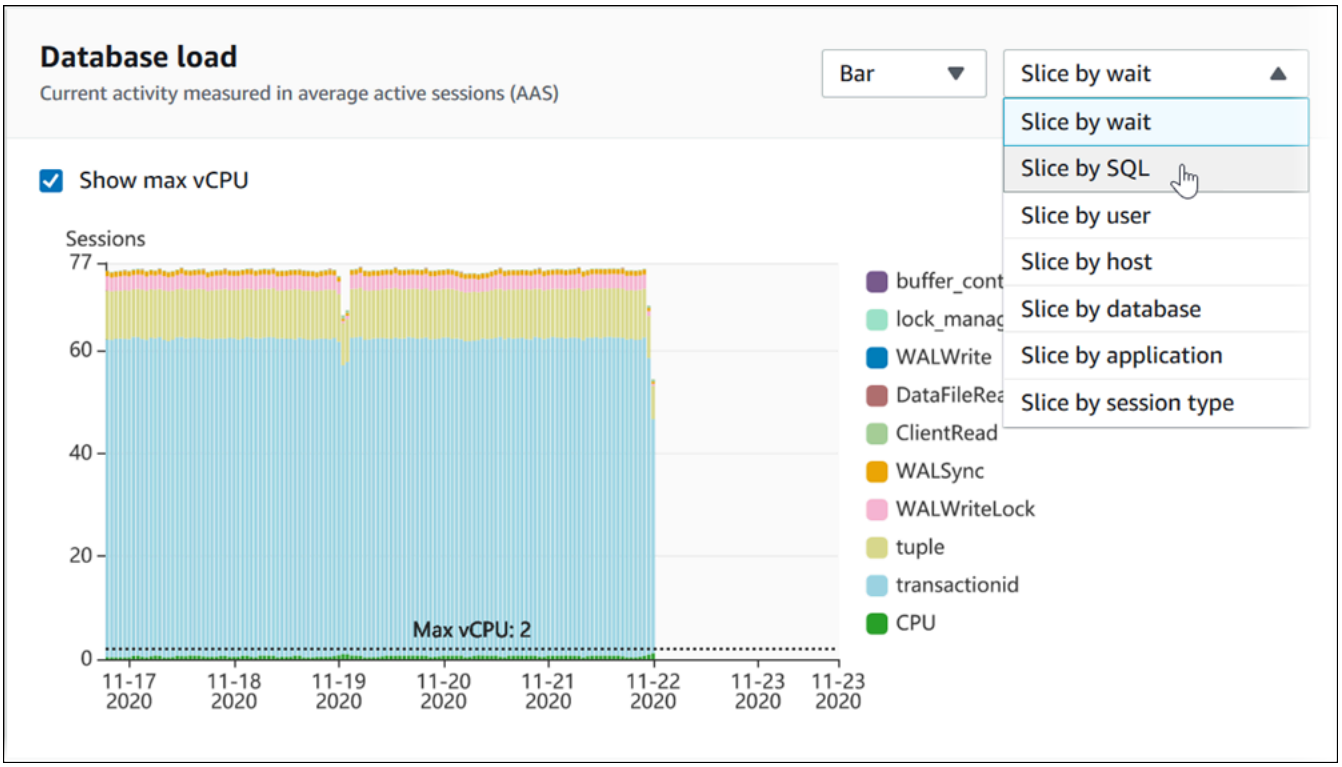


Charge de base de données tranchée par dimensions

Vous pouvez afficher la charge sous la forme de sessions actives regroupées par dimensions prises en charge. Le tableau suivant montre les dimensions prises en charge pour les différents moteurs.

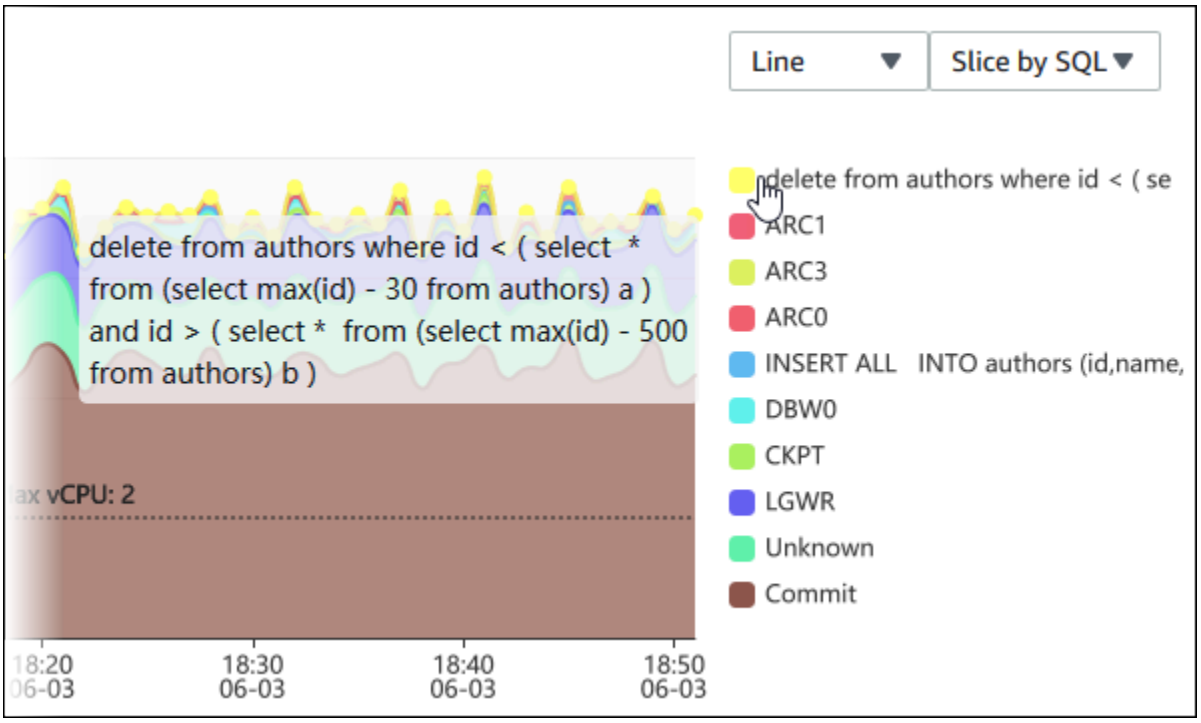
Dimension	Oracle	Serveur SQL	Poster SQL	Mon SQL
Host (Hôte)	Oui	Oui	Oui	Oui
SQL	Oui	Oui	Oui	Oui
Utilisateur	Oui	Oui	Oui	Oui
Éléments d'attente	Oui	Oui	Oui	Oui
Plans	Oui	Non	Non	Non
Application	Non	Non	Oui	Non
Base de données	Non	Non	Oui	Oui
Type de session	Non	Non	Oui	Non

L'image suivante montre les dimensions d'une SQL instance de base de données Postgre.



Détails de charge de base de données pour un élément de dimension

Pour afficher les détails d'un élément de charge de base de données dans une dimension, passez la souris sur le nom d'élément. L'image suivante montre les détails d'une SQL déclaration.



Pour afficher les détails d'un élément pour la période sélectionnée dans la légende, survolez cet élément.

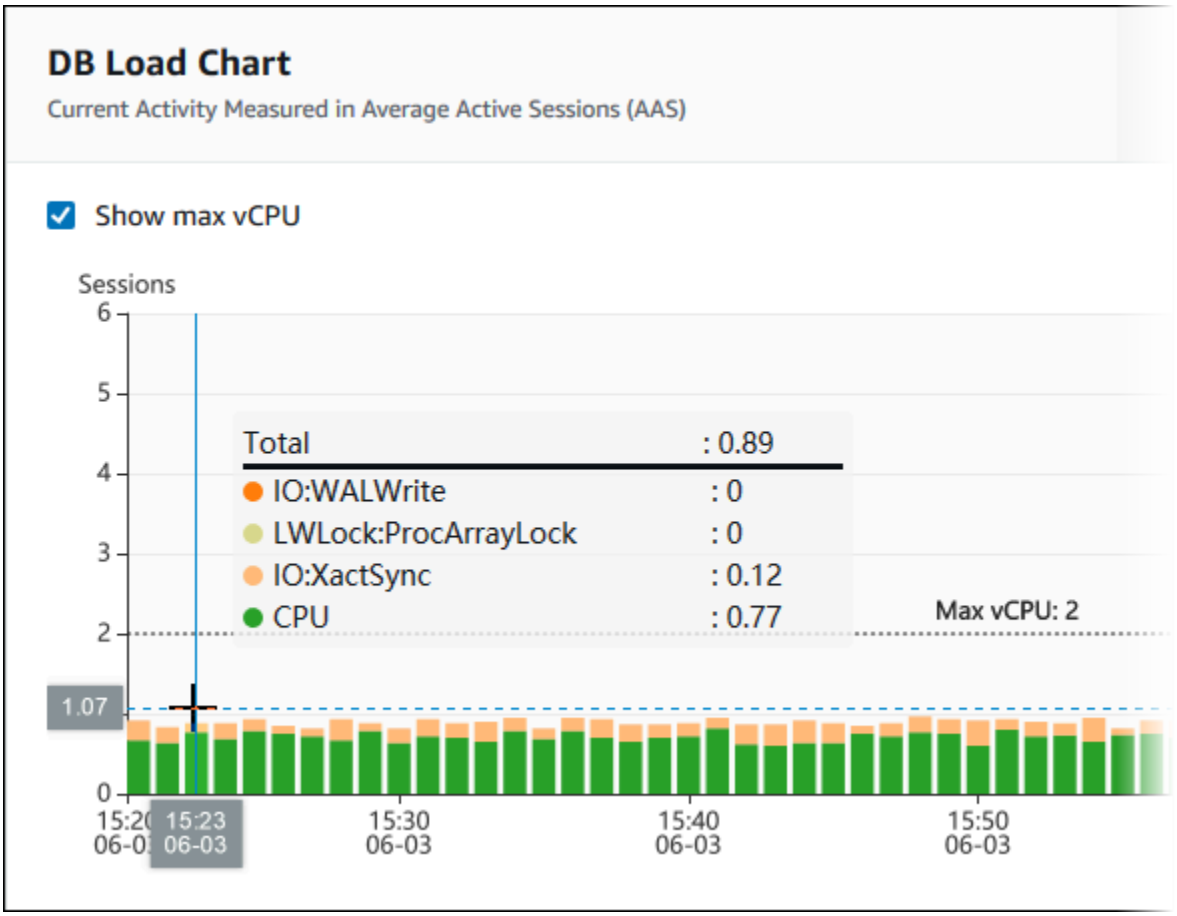
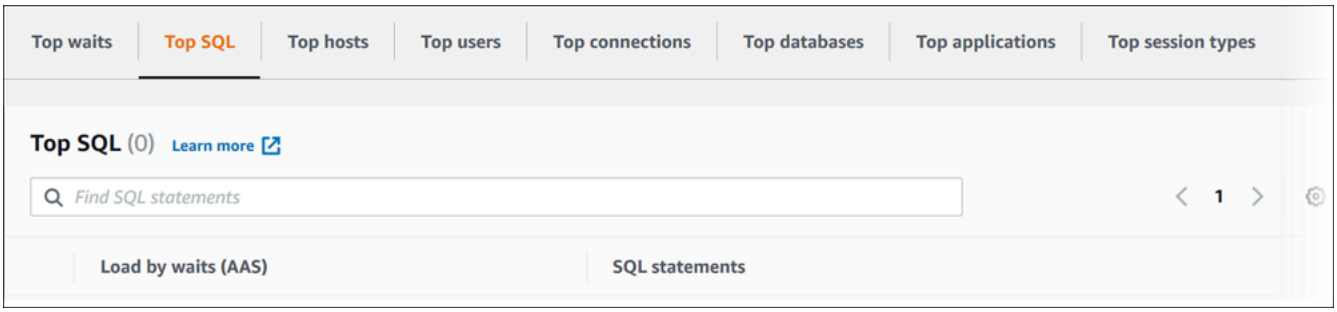


Tableau des dimensions principales

Le tableau des dimensions principales découpe la charge de la base de données selon différentes dimensions. Une dimension est une catégorie ou « tranche » qui représente l'une des différentes caractéristiques de la charge de la base de données. Si la dimension estSQL, Top SQL affiche les SQL instructions qui contribuent le plus à la charge de la base de données.



Choisissez l'un des onglets de dimension suivants.

Onglet	Description	Moteurs pris en charge
Haut SQL	Les SQL instructions en cours d'exécution	Tous
Principaux éléments d'attente	Événement pour lequel le backend de la base de données attend	Tous
Principaux hôtes	Nom d'hôte du client connecté	Tous
Principaux utilisateurs	Utilisateur connecté à la base de données	Tous
Principales bases de données	Nom de la base de données à laquelle le client est connecté	PostgreSQL, MySQL, MariaDB et Server uniquemen t SQL
Principales applications	Nom de l'application connectée à la base de données	Postgre SQL et SQL Server uniquement
Principaux types de session	Type de la session en cours	Postgre uniquement SQL

Pour savoir comment analyser des requêtes à l'aide de l'SQLonglet Haut, consultezVue d'ensemble de l'SQLonglet Top.

Accès au tableau de bord Performance Insights

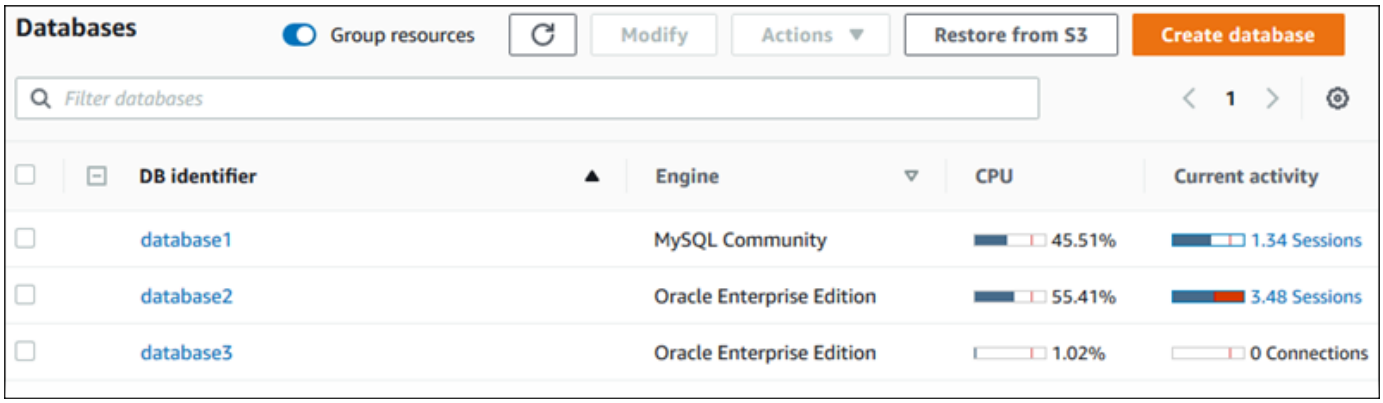
Amazon RDS fournit une vue consolidée des CloudWatch statistiques et des statistiques relatives aux Performances dans le tableau de bord Performance Insights.

Pour accéder au tableau de bord de Performance Insights, procédez comme suit.

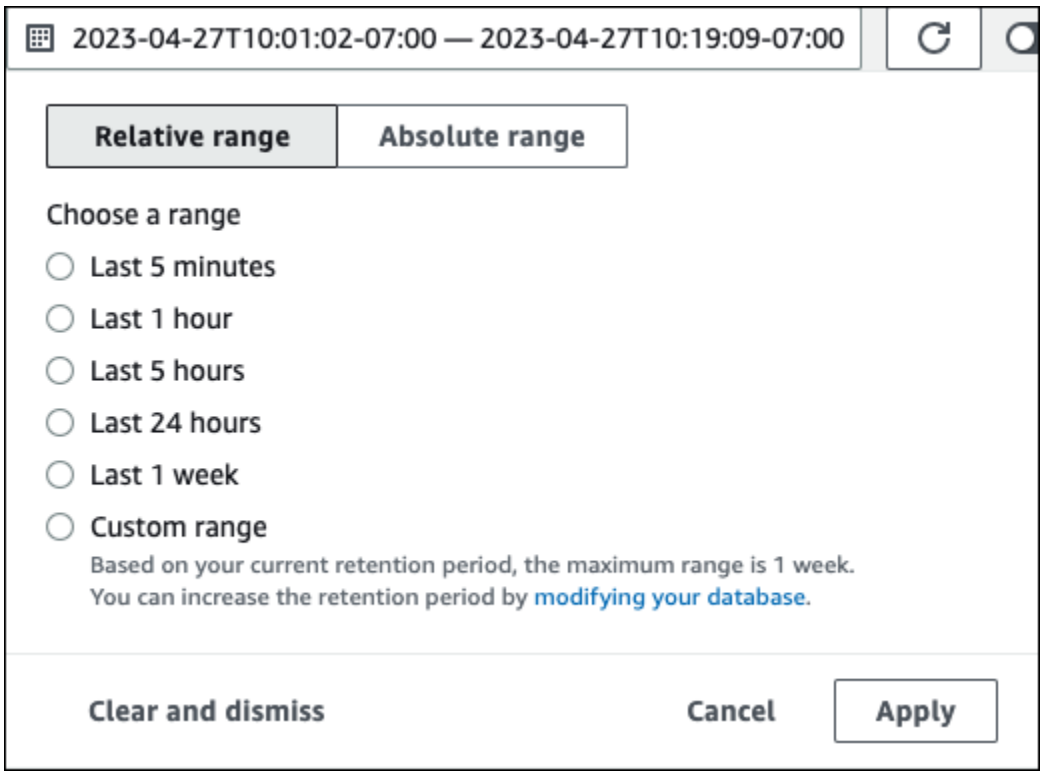
Pour consulter le tableau de bord Performance Insights dans la console AWS de gestion

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

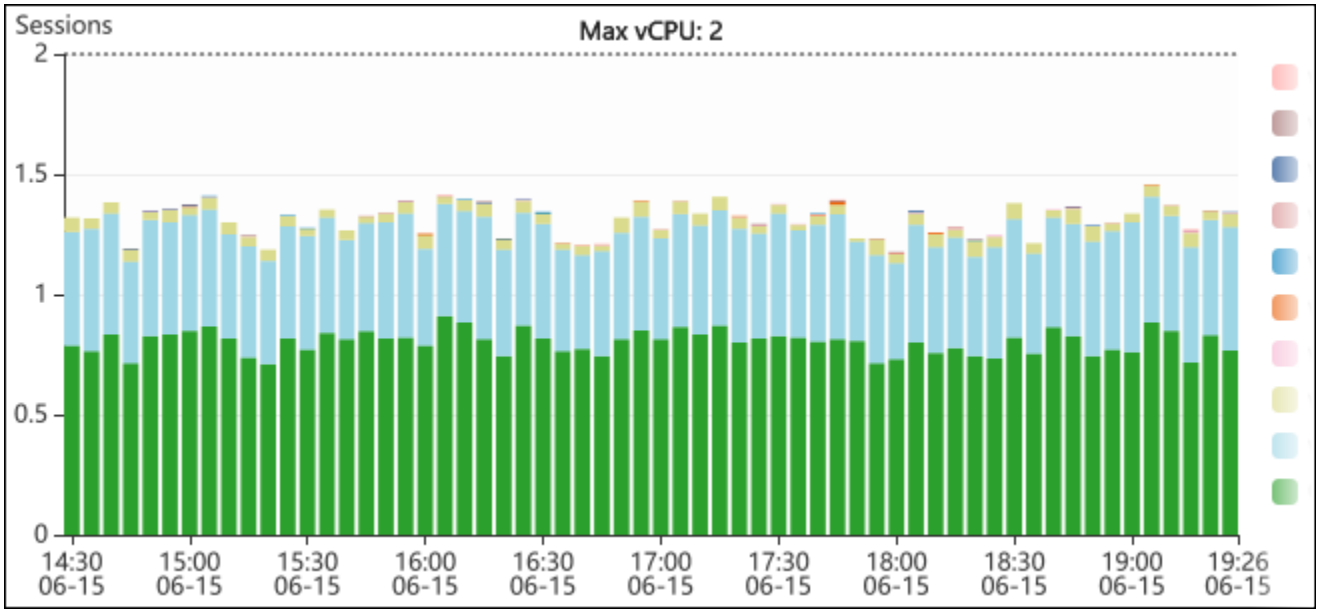
Pour les instances de base de données pour lesquelles Performance Insights est activé, vous pouvez également accéder au tableau de bord Performance Insights en choisissant l'élément Sessions dans la liste des instances de base de données. Sous Activité actuelle, l'élément Sessions affiche la charge de base de données dans les sessions actives moyennes lors des cinq dernières minutes. La barre affiche visuellement le chargement. Votre instance de base de données est à l'arrêt lorsque la barre est vide. La barre se remplit de bleu à mesure que le chargement augmente. Lorsque la charge dépasse le nombre de virtual CPUs (vCPUs) sur la classe d'instance de base de données, la barre devient rouge, ce qui indique un goulot d'étranglement potentiel.



4. (Facultatif) Choisissez la plage de dates ou de temps en haut à droite et spécifiez un autre intervalle de temps relatif ou absolu. Vous pouvez désormais spécifier une période et générer un rapport d'analyse des performances de base de données. Ce rapport fournit les informations et recommandations identifiées. Pour de plus amples informations, veuillez consulter Création d'un rapport d'analyse des performances dans Performance Insights.

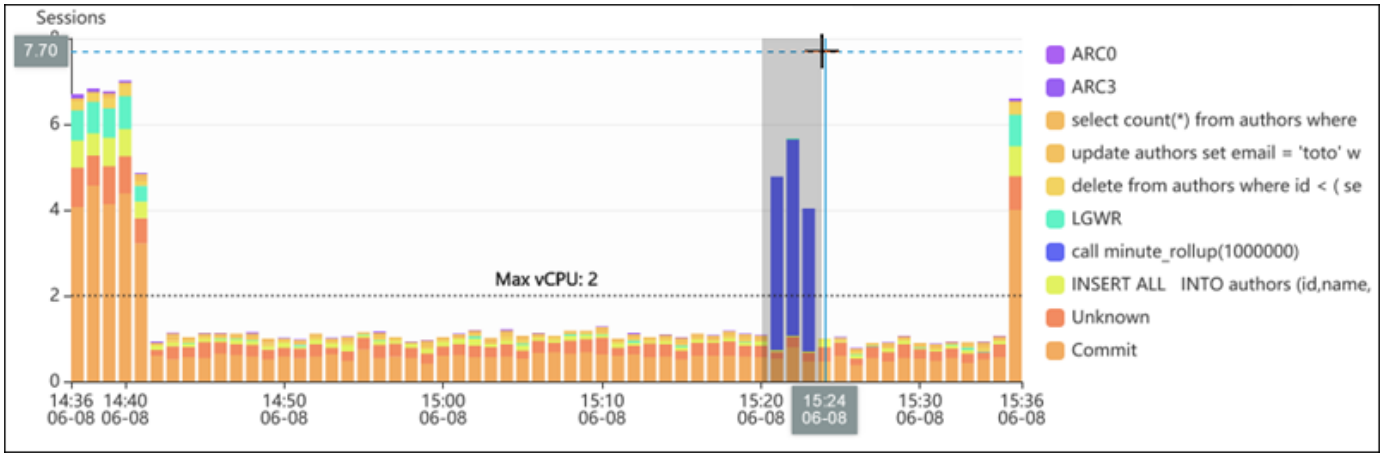


Dans la capture d'écran suivante, l'intervalle de charge de base de données est de 5 heures.

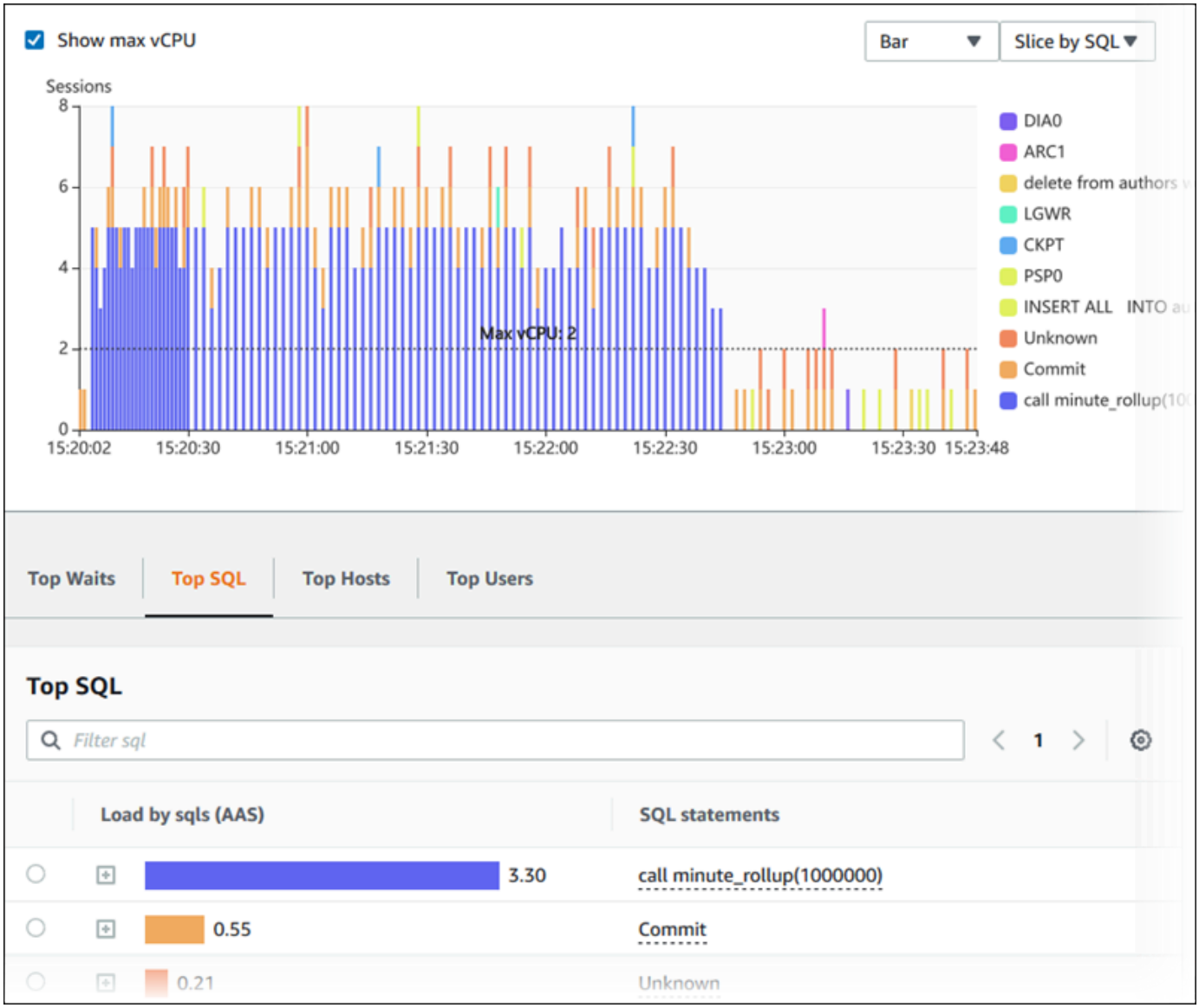


5. (Facultatif) Pour effectuer un zoom avant sur une partie du graphique de charge de la base de données, choisissez l'heure de début et faites glisser jusqu'à la fin de la période souhaitée.

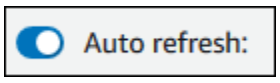
   La zone sélectionnée est mise en évidence dans le tableau de charge de la base de données.



Lorsque vous relâchez la souris, le graphique de charge de la base de données zoome sur la AWS région sélectionnée et le tableau des dimensions supérieures est recalculé.



6. (Facultatif) Pour actualiser automatiquement vos données, sélectionnez Actualisation automatique.



Le tableau de bord Performance Insights s'actualise automatiquement avec de nouvelles données. Le taux de rafraîchissement dépend de la quantité de données affichées :

- Pour 5 minutes, les données seront actualisées toutes les 10 secondes.
- Pour 1 heure, les données seront actualisées toutes les 5 minutes.
- Pour 5 heures, les données seront actualisées toutes les 5 minutes.
- Pour 24 heures, les données seront actualisées toutes les 30 minutes.

- Pour 1 semaine, les données seront actualisées tous les jours.
- Pour 1 mois, les données seront actualisées tous les jours.

Analyse de la charge de base de données par événements d'attente

Si le graphique Database load (Charge de la base de données) présente un goulot d'étranglement, vous pouvez déterminer la provenance de la charge. Pour ce faire, examinez le tableau des principaux éléments de charge en dessous du graphique Database load (Charge de la base de données). Choisissez un élément en particulier, comme une SQL requête ou un utilisateur, pour accéder à cet élément et en voir les détails.

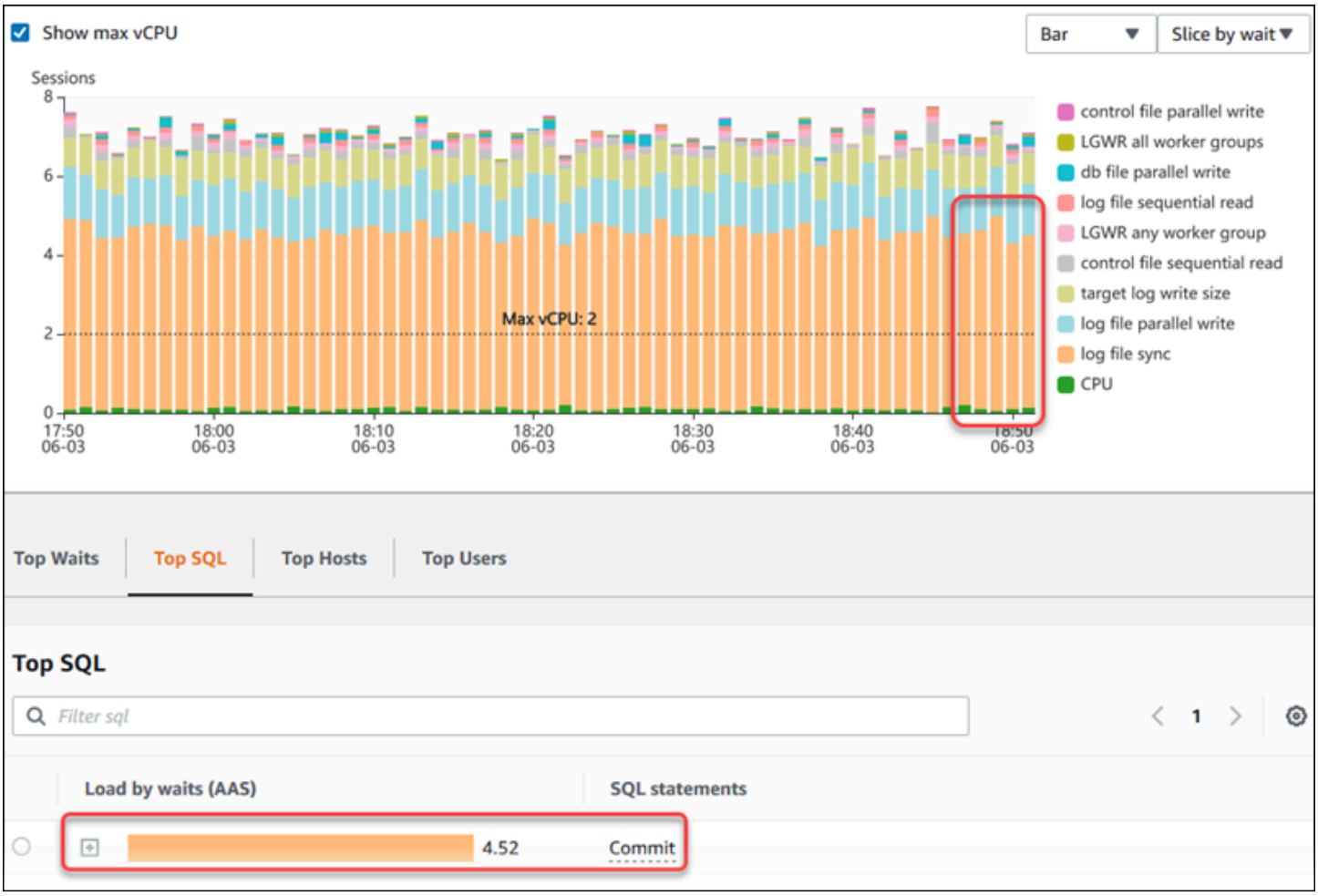
La charge de base de données regroupée par temps d'attente et par SQL requêtes les plus fréquentes est la vue par défaut du tableau de bord Performance Insights. Cette combinaison fournit en général la meilleure compréhension des problèmes de performances. L'affichage de la charge de la base de données en fonction de l'attente indique s'il existe des goulots d'étranglement liés aux ressources ou à des actions simultanées dans la base de données. Dans ce cas, l'SQLonglet du tableau des éléments à charger par le haut indique les requêtes à l'origine de cette charge.

Votre flux de travail standard pour diagnostiquer les problèmes de performances se présente comme suit :

1. Consultez le graphique de charge de la base de données pour voir s'il y a des incidents de charge de base de données dépassant la CPU ligne maximale.
2. Si c'est le cas, observez le graphique Database load (Charge de la base de données) et identifiez le ou les états d'attente qui sont les principaux responsables.
3. Identifiez les requêtes de résumé à l'origine du chargement en identifiant les requêtes figurant dans l'SQLonglet du tableau des éléments de chargement les plus importants qui contribuent le plus à ces états d'attente. Vous pouvez les identifier dans la colonne DB Load by Wait (Charge de base de données par attente).
4. Choisissez l'une de ces requêtes de synthèse dans l'SQLonglet pour la développer et voir les requêtes enfants qui la composent.

Par exemple, dans le tableau de bord suivant, les attentes log file sync (synchronisation de fichier journal) constituent la majeure partie de la charge de base de données. L'attente de LGWRtous les groupes de travailleurs est également élevée. Le SQL graphique du haut montre la cause des temps

d'attente pour la synchronisation des fichiers journaux : les `COMMIT` requêtes fréquentes. Dans ce cas, une validation moins fréquente permet de réduire la charge de base de données.



Analyse des performances de base de données pour une période donnée

Analysez les performances des bases de données à l'aide d'une analyse à la demande en créant un rapport d'analyse des performances pour une période donnée. Consultez les rapports d'analyse des performances pour détecter les problèmes de performances, tels que les goulots d'étranglement des ressources ou les modifications apportées à une requête dans votre instance de base de données. Le tableau de bord d'analyse des performances vous permet de sélectionner une période et de créer un rapport d'analyse des performances. Vous pouvez également ajouter une ou plusieurs balises au rapport.

Pour utiliser cette fonctionnalité, vous devez utiliser la période de conservation du niveau payant. Pour plus d'informations, consultez Tarification et conservation des données pour Performance Insights.

Le rapport est disponible dans l'onglet Rapports d'analyse des performances – nouveau pour être sélectionné et affiché. Ce rapport contient les informations, les métriques associées et les recommandations permettant de résoudre le problème de performances. Le rapport peut être consulté pendant toute la durée de conservation de l'analyse des performances.

Le rapport est supprimé si l'heure de début de la période d'analyse du rapport se situe en dehors de la période de rétention. Vous pouvez également supprimer le rapport avant la fin de la période de conservation.

Pour détecter les problèmes de performances et générer le rapport d'analyse pour votre instance de base de données, vous devez activer l'analyse des performances. Pour plus d'informations sur l'activation de Performance Insights, consultez Activer et désactiver Performance Insights pour Amazon RDS.

Pour obtenir des informations de prise en charge de la région, du moteur de base de données et des classes d'instances pour cette fonctionnalité, consultez Prise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

Dans les sections suivantes, vous pouvez créer, afficher, ajouter des balises et supprimer un rapport d'analyse des performances.

Rubriques

- Création d'un rapport d'analyse des performances dans Performance Insights
- Afficher un rapport d'analyse des performances dans Performance Insights
- Ajouter des balises à un rapport d'analyse des performances dans Performance Insights
- Supprimer un rapport d'analyse des performances dans Performance Insights

Création d'un rapport d'analyse des performances dans Performance Insights

Vous pouvez créer un rapport d'analyse des performances pour une période spécifique dans le tableau de bord d'analyse des performances. Vous pouvez sélectionner une période et ajouter une ou plusieurs balises au rapport d'analyse.

La période d'analyse peut aller de 5 minutes à 6 jours. Il doit y avoir au moins 24 heures de données de performance avant le début de l'analyse.

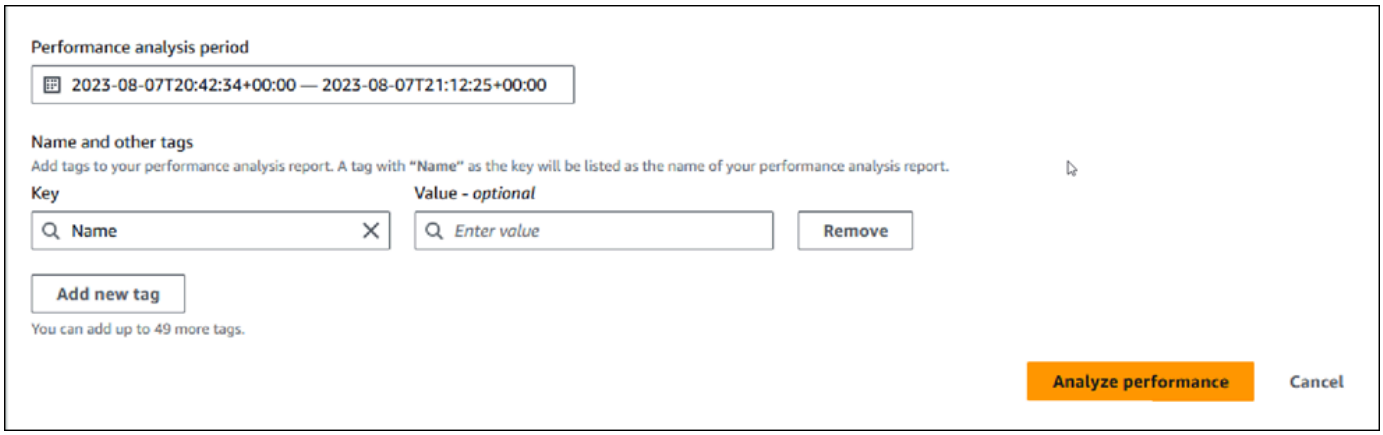
Pour obtenir des informations de prise en charge de la région, du moteur de base de données et des classes d'instances pour cette fonctionnalité, consultez Prise en charge de la classe d'instances,

de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

Pour créer un rapport d'analyse des performances pour une période donnée

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.
4. Choisissez Analyser les performances dans la section Chargement de la base de données du tableau de bord Performance Insights.

   Les champs permettant de définir la période et d'ajouter une ou plusieurs balises au rapport d'analyse des performances s'affichent.

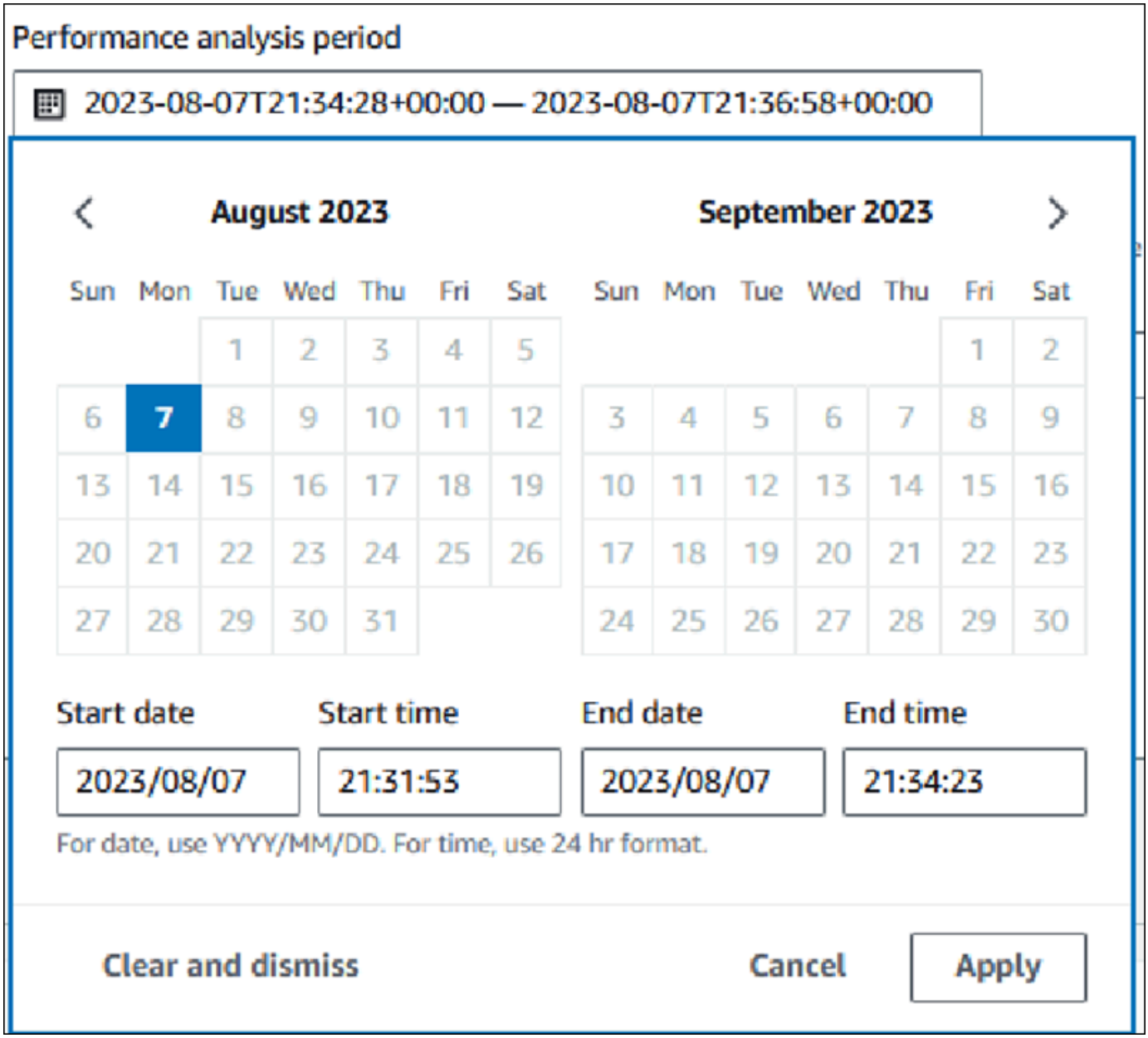


5. Choisissez une période. Si vous définissez une période dans la Plage relative ou dans la Plage absolue en haut à droite, vous pouvez uniquement saisir ou sélectionner la date et l'heure du rapport d'analyse au cours de cette période. Si vous sélectionnez la période d'analyse en dehors de cette période, un message d'erreur s'affiche.

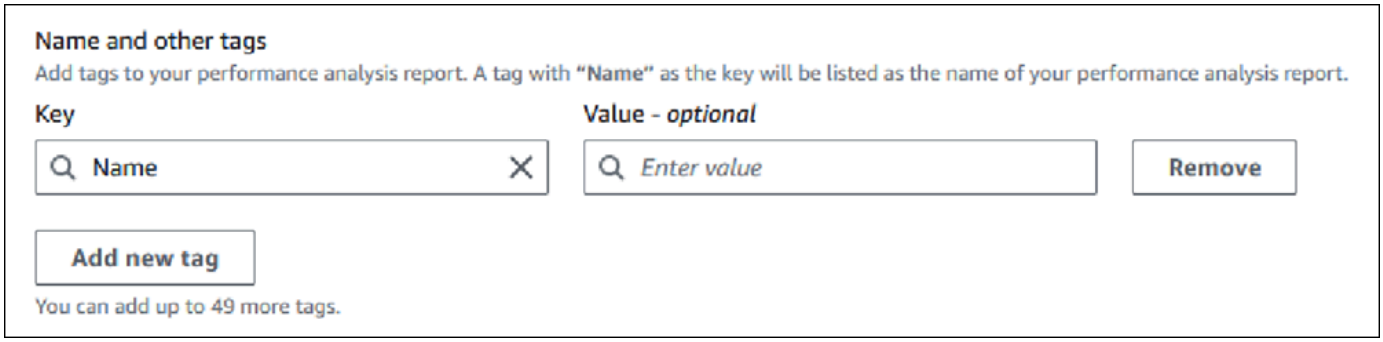
   Pour définir la période, vous pouvez effectuer l'une des opérations suivantes :

   - Appuyez sur l'un des curseurs du graphique de charge de base de données et faites-le glisser.

     La zone Période d'analyse des performances affiche la période sélectionnée et le graphique de charge de base de données met en évidence la période sélectionnée.
   - Choisissez les paramètres Date de début, Heure de début,Date de fin et Heure de fin dans la zone Période d'analyse des performances.



6. (Facultatif) Entrez Clé et Valeur-facultatif pour ajouter une balise pour le rapport.



7. Choisissez Analyser les performances.

   Une bannière affiche un message indiquant si la génération du rapport est réussie ou a échoué. Le message fournit également le lien permettant de consulter le rapport.

   L'exemple suivant montre la bannière avec le message de réussite de création du rapport.

   Le rapport peut être consulté dans l'onglet Rapports d'analyse des performances – nouveau.

Vous pouvez créer un rapport d'analyse des performances à l'aide de l'interface AWS CLI. Pour un exemple expliquant comment créer un rapport à l'aide de AWS CLI, voirCréation d'un rapport d'analyse des performances pour une période donnée.

Afficher un rapport d'analyse des performances dans Performance Insights

L'onglet Rapports d'analyse des performances – nouveau répertorie tous les rapports créés pour l'instance de base de données. Pour chaque test, les résultats des tests suivants sont affichés :

- ID : identifiant unique du rapport.
- Nom : clé de balise ajoutée au rapport.
- Heure de création du rapport : heure à laquelle vous avez créé le rapport.
- Heure de début de l'analyse : heure de début de l'analyse dans le rapport.
- Heure de fin de l'analyse : heure de fin de l'analyse dans le rapport.

Pour afficher un rapport d'analyse des performances

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données pour laquelle vous souhaitez consulter le rapport d'analyse.
4. Faites défiler la page vers le bas et sélectionnez Rapports d'analyse des performances : nouvel onglet dans le tableau de bord Performance Insights.

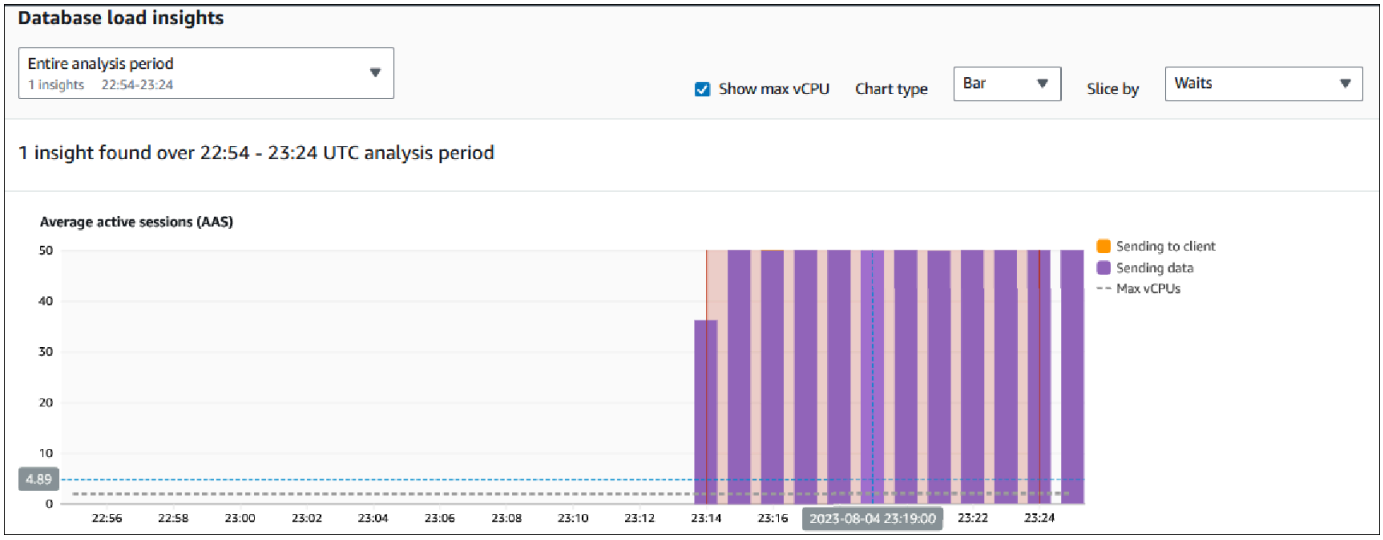
   Tous les rapports d'analyse pour les différentes périodes sont affichés.

5. Choisissez ID du rapport que vous souhaitez consulter.

   Le graphique de charge de base de données affiche la période d'analyse complète par défaut si plusieurs informations sont identifiées. Si le rapport a identifié une information, le graphique de charge de base de données affiche par défaut cette information.

   Le tableau de bord répertorie également les balises du rapport dans la section Balises.

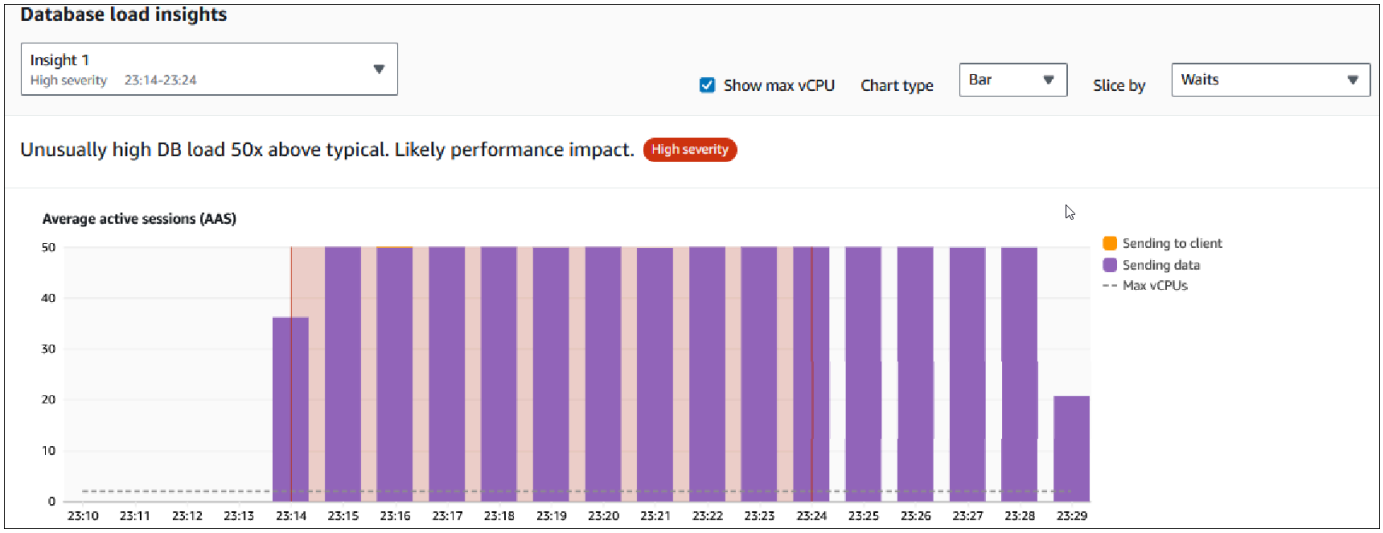
   L'exemple suivant montre l'ensemble de la période d'analyse du rapport.

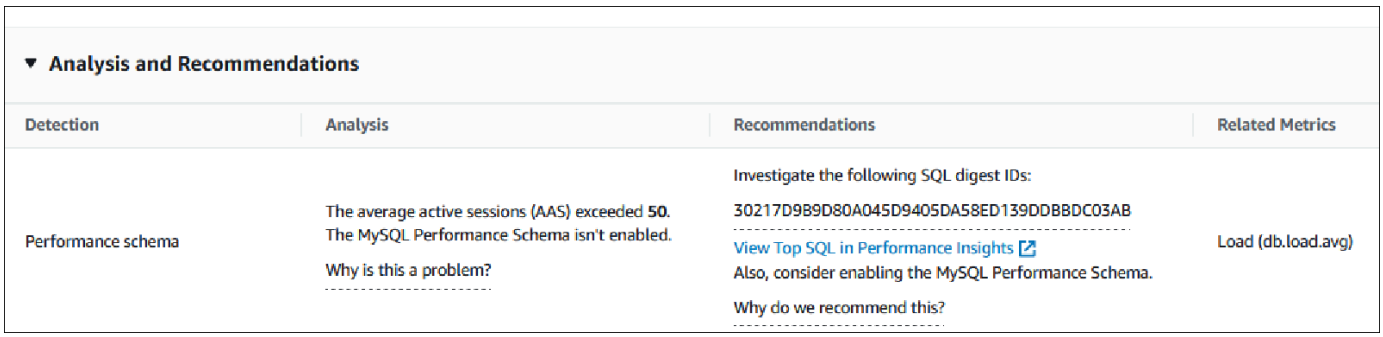


6. Choisissez l'information dans la liste Informations de charge de base de données que vous souhaitez consulter si plusieurs informations sont identifiées dans le rapport.

   Le tableau de bord affiche le message d'information, le graphique de charge de base de données mettant en évidence la période couverte par les informations, l'analyse et les recommandations, ainsi que la liste des balises de rapport.

   L'exemple suivant montre l'information de charge de base de données dans le rapport.





Ajouter des balises à un rapport d'analyse des performances dans Performance Insights

Vous pouvez ajouter une balise lorsque vous créez ou consultez un rapport. Vous pouvez ajouter jusqu'à 50 balises par rapport.

Vous avez besoin d'autorisations pour ajouter les balises. Pour plus d'informations sur les stratégies d'accès pour l'analyse des performances, consultez Configuration des politiques d'accès pour Performance Insights.

Pour ajouter une ou plusieurs balises lors de la création d'un rapport, consultez l'étape 6 de la procédure Création d'un rapport d'analyse des performances dans Performance Insights.

Pour ajouter une ou plusieurs balises lors de la consultation d'un rapport

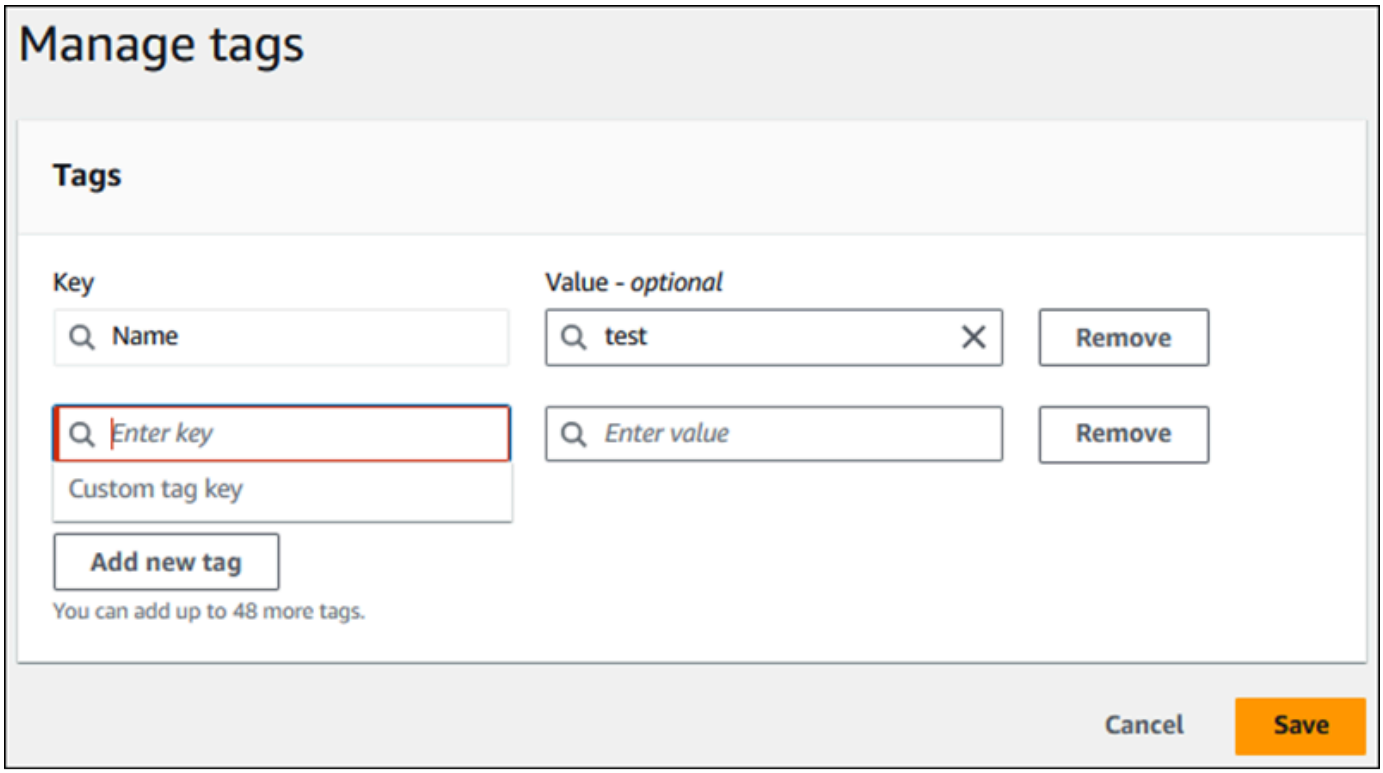
1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.

4. Faites défiler la page vers le bas et choisissez Rapports d'analyse des performances – nouveau.
5. Choisissez le rapport pour lequel vous souhaitez ajouter les balises.

   Le tableau de bord affiche le rapport.
6. Faites défiler vers le bas jusqu'à Balises et choisissez Gérer les balises.
7. Sélectionnez Ajouter une nouvelle balise.
8. Entrez la Clé et la Valeur – facultatif, puis choisissez Ajouter une nouvelle balise.

   L'exemple suivant fournit la possibilité d'ajouter une nouvelle balise pour le rapport sélectionné.



Une nouvelle balise est créée pour le rapport.

La liste des balises du rapport est affichée dans la section Balises du tableau de bord. Si vous souhaitez supprimer une balise du rapport, choisissez Supprimer à côté de la balise.

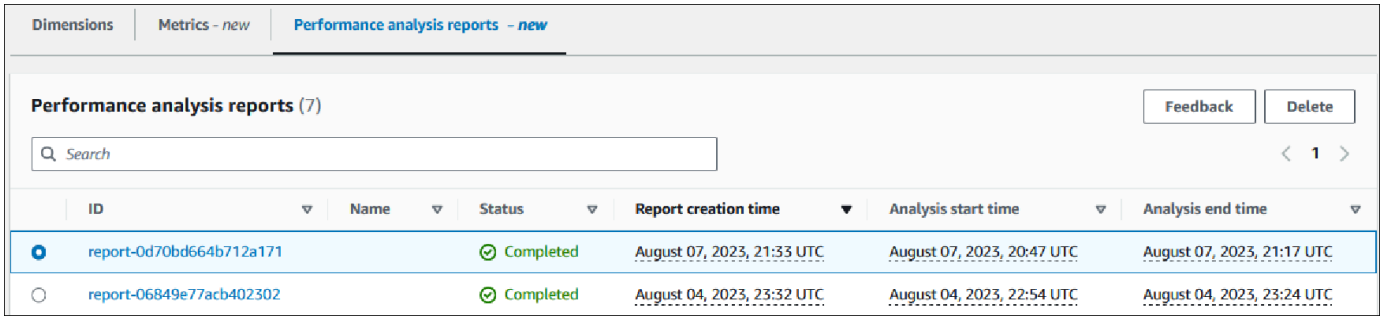
Supprimer un rapport d'analyse des performances dans Performance Insights

Vous pouvez supprimer un rapport de la liste des rapports affichée dans l'onglet Rapports d'analyse des performances ou lors de l'affichage d'un rapport.

Pour supprimer un rapport

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. Choisissez une instance de base de données.

   Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.
4. Faites défiler la page vers le bas et choisissez Rapports d'analyse des performances – nouveau.
5. Sélectionnez le rapport que vous souhaitez supprimer et choisissez Supprimer en haut à droite.



   Une fenêtre de confirmation s'affiche. Le rapport est supprimé une fois que vous avez choisi de confirmer.
6. (Facultatif) Choisissez l'ID du rapport que vous souhaitez supprimer.

   Dans le coin supérieur droit de la page du rapport, choisissez Supprimer.

   Une fenêtre de confirmation s'affiche. Le rapport est supprimé une fois que vous avez choisi de confirmer.

Analyse des requêtes à l'aide de SQL l'onglet Top de Performance Insights

Dans le tableau de bord Amazon RDS Performance Insights, vous pouvez trouver des informations sur les requêtes en cours et récentes dans l'SQLonglet Haut du tableau des principales dimensions. Vous pouvez utiliser ces informations pour régler vos requêtes.

Rubriques

- Vue d'ensemble de l'SQLonglet Top
- Accès à plus de texte SQL dans le tableau de bord Performance Insights
- Afficher SQL les statistiques dans le tableau de bord Performance Insights

Vue d'ensemble de l'SQLonglet Top

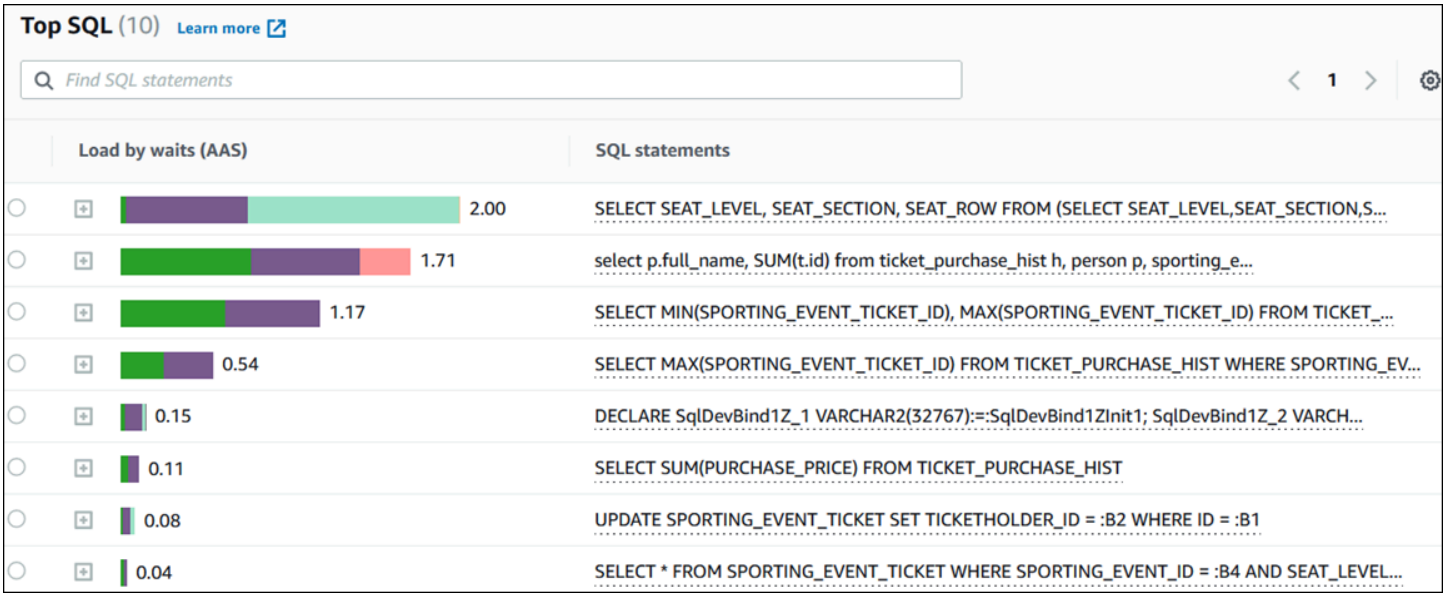
Par défaut, l'SQLonglet Top affiche les 25 requêtes qui contribuent le plus à la charge de la base de données. Pour affiner vos requêtes, vous pouvez analyser des informations telles que le texte de la requête et les SQL statistiques. Vous pouvez également choisir les statistiques que vous souhaitez voir apparaître dans l'SQLonglet Top.

Rubriques

- SQLtexte
- Statistiques SQL
- Charger par temps d'attente () AAS
- Afficher les informations d’SQL
- Choisissez vos préférences en matière de statistiques

SQLtexte

Par défaut, chaque ligne du SQL tableau supérieur affiche 500 octets de texte pour chaque instruction.

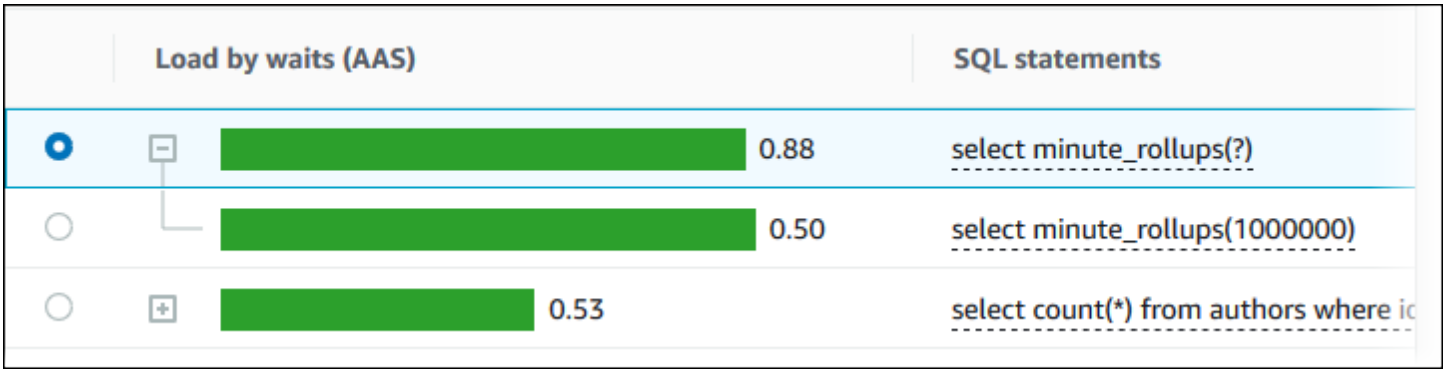


Pour savoir comment afficher plus que les 500 octets de SQL texte par défaut, consultezAccès à plus de texte SQL dans le tableau de bord Performance Insights.

Un SQLcondensé est un composite de plusieurs requêtes réelles qui sont structurellement similaires mais qui peuvent avoir des valeurs littérales différentes. Le récapitulatif remplace les valeurs codées en dur par un point d'interrogation. Ainsi, `SELECT * FROM emp WHERE lname= ?` est un exemple de récapitulatif. Ce récapitulatif peut inclure les requêtes enfant suivantes :

```
SELECT * FROM emp WHERE lname = 'Sanchez'
SELECT * FROM emp WHERE lname = 'Olagappan'
SELECT * FROM emp WHERE lname = 'Wu'
```

Pour voir les SQL déclarations littérales dans un résumé, sélectionnez la requête, puis le symbole plus (+). Dans l'exemple suivant, la requête sélectionnée est un récapitulatif.



> **Note**
>
> Un SQL résumé regroupe des SQL déclarations similaires, mais ne supprime pas les informations sensibles.

Performance Insights peut afficher le SQL texte Oracle sous la forme Inconnu. Le texte a ce statut dans les situations suivantes :

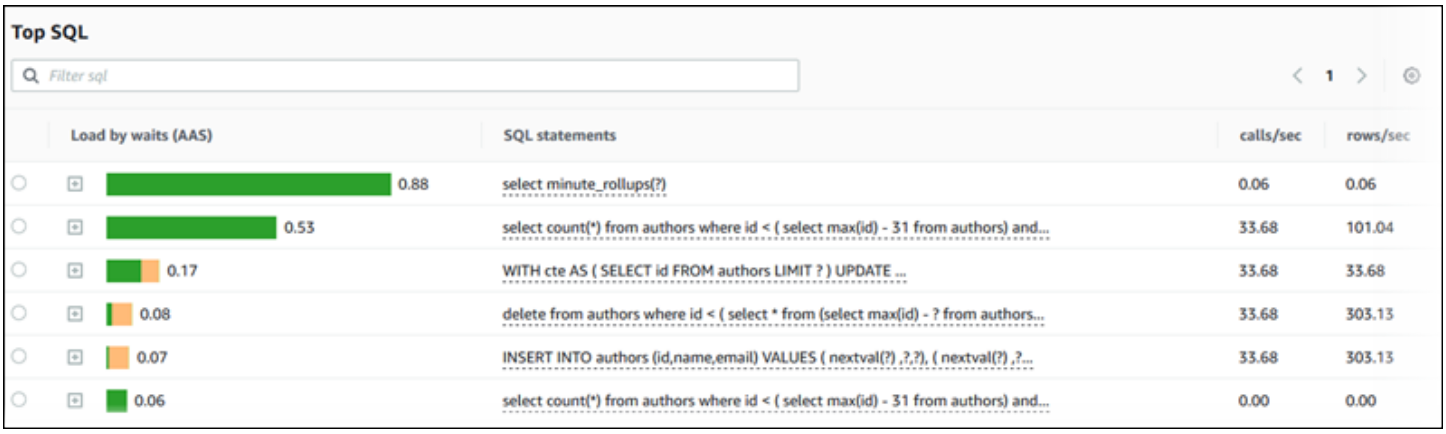
- Un autre utilisateur de base de données Oracle SYS est actif mais n'est pas en cours d'exécutionSQL. Par exemple, lorsqu'une requête parallèle se termine, le coordinateur de la requête attend que les processus d'assistance envoient leurs statistiques de session. Pendant la durée de l'attente, le texte de la requête affiche Unknown (Inconnu).

- RDSPour une instance Oracle sur Standard Edition 2, Oracle Resource Manager limite le nombre de threads parallèles. Le processus en arrière-plan qui effectue ce travail fait que le texte de la requête s'affiche comme Unknown (Inconnu).

Statistiques SQL

SQLles statistiques sont des mesures liées aux performances relatives aux requêtes. SQL Par exemple, Performance Insights peut montrer le nombre d'exécutions par seconde ou le nombre de lignes traitées par seconde. Performance Insights collecte des statistiques uniquement pour les requêtes les plus courantes. Généralement, celles-ci correspondent aux requêtes les plus importantes par charge affichées dans le tableau de bord Performance Insights.

Chaque ligne du SQL tableau supérieur affiche les statistiques pertinentes pour le SQL relevé ou le résumé, comme indiqué dans l'exemple suivant.



Performance Insights peut fournir `0.00` des rapports et - (inconnu) des SQL statistiques. Cette situation se produit dans les conditions suivantes :

- Il n'existe qu'un seul exemple. Par exemple, Performance Insights calcule les taux de variation des SQL requêtes Postgre sur la base de plusieurs échantillons issus de la `pg_stat_statements` vue. Lorsqu'une charge de travail est exécutée pendant une courte période, Performance Insights peut ne collecter qu'un seul exemple, ce qui signifie qu'il ne peut pas calculer le taux de modification. La valeur inconnue est représentée par un tiret (-).
- Deux exemples ont les mêmes valeurs. Performance Insights ne peut pas calculer un taux de modification car aucun changement n'a eu lieu, il rapporte donc le taux comme `0.00`.
- Une SQL instruction Postgre ne contient pas d'identifiant valide. Postgre SQL crée un identifiant pour une instruction uniquement après analyse et analyse. Ainsi, une instruction peut exister dans les structures SQL internes en mémoire de Postgre sans identifiant. Comme Performance

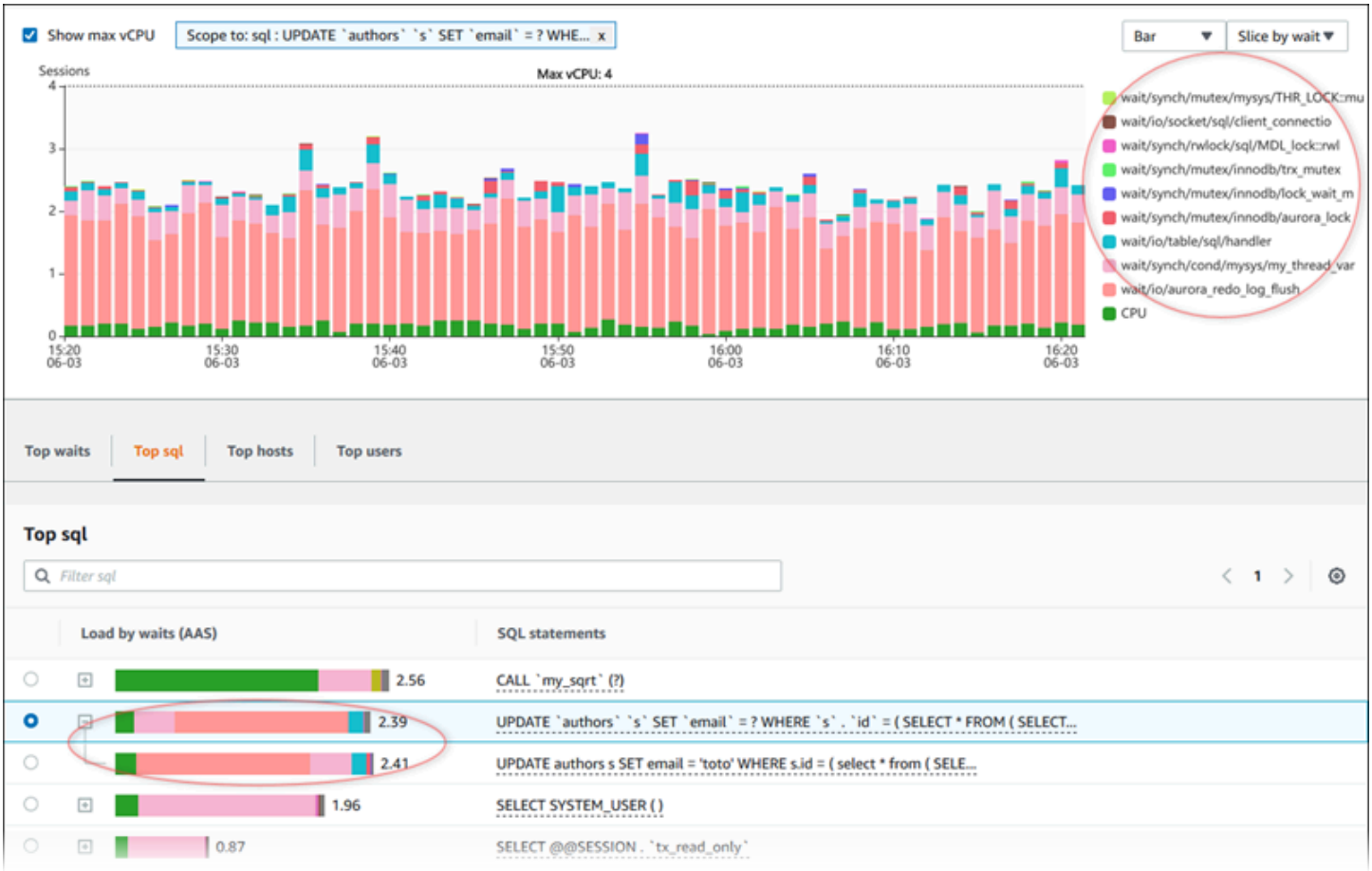
Insights échantillonne les structures internes en mémoire une fois par seconde, les requêtes à faible latence peuvent n'apparaître que pour un seul exemple. Si l'identifiant de la requête n'est pas disponible pour cet exemple, Performance Insights ne peut pas associer cette déclaration à ses statistiques. La valeur inconnue est représentée par un tiret (-).

Pour une description des SQL statistiques relatives aux moteurs Amazon RDS , consultezStatistiques SQL pour Performance Insights.

Charger par temps d'attente () AAS

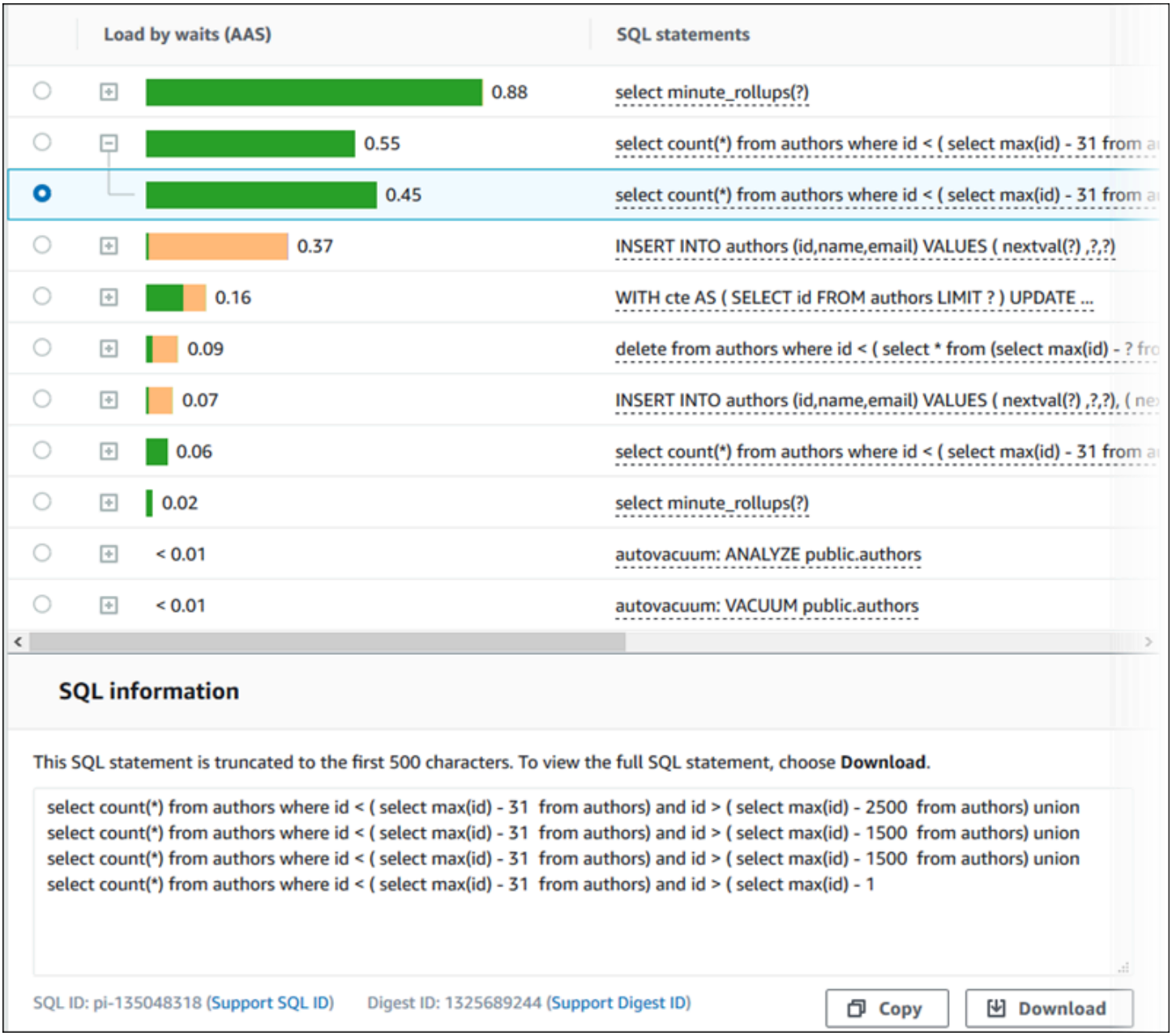
En haut SQL, la colonne Load by waits (AAS) illustre le pourcentage de charge de base de données associé à chaque élément de chargement supérieur. Cette colonne reflète la charge pour cet élément selon le regroupement actuellement sélectionné dans DB Load Chart (Graphique de charge de base de données). Pour plus d'informations sur le nombre moyen de sessions actives (AAS), consultezSessions actives en moyenne.

Par exemple, vous pouvez regrouper le graphique DB Load (Charge de la base de données) par états d'attente. Vous examinez les SQL requêtes dans le tableau des éléments les plus chargés. Dans ce cas, la dimension, la segmentation et le code de couleurs de la barre DB Load by Waits (Charge de base de données par attente) représentent la proportion du temps d'un état d'attente donné auquel cette requête contribue. Cette barre indique également les états d'attente qui affectent la requête sélectionnée.



Afficher les informations d'SQL

Dans le SQL tableau supérieur, vous pouvez ouvrir un relevé pour consulter ses informations. Les informations s'affichent dans le volet inférieur.

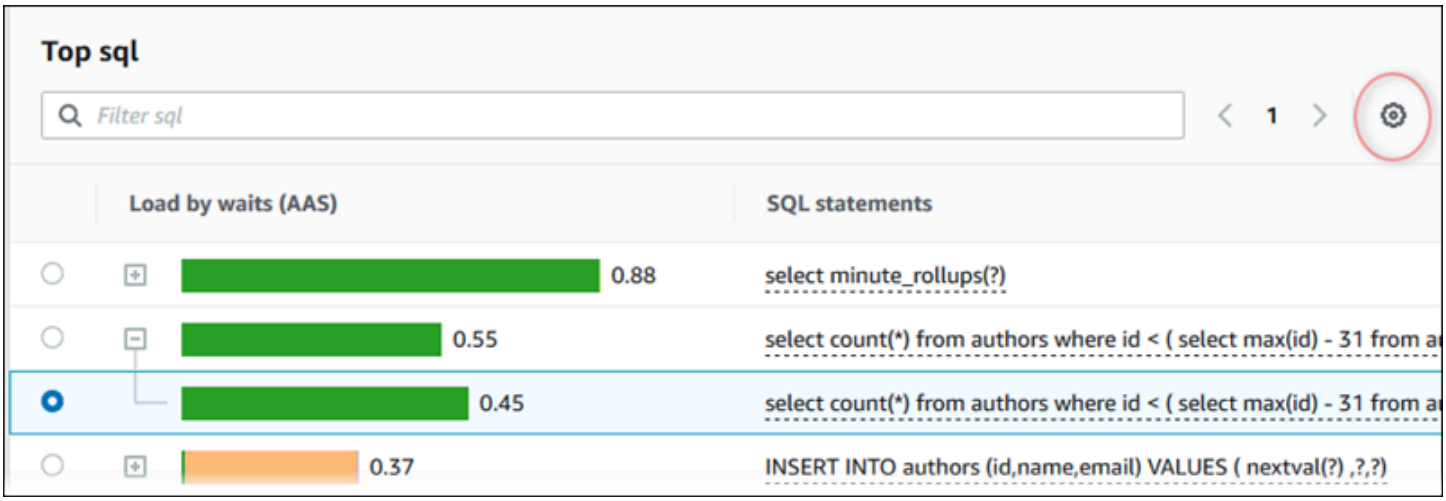


Les types d'identificateurs (IDs) suivants associés aux SQL instructions :

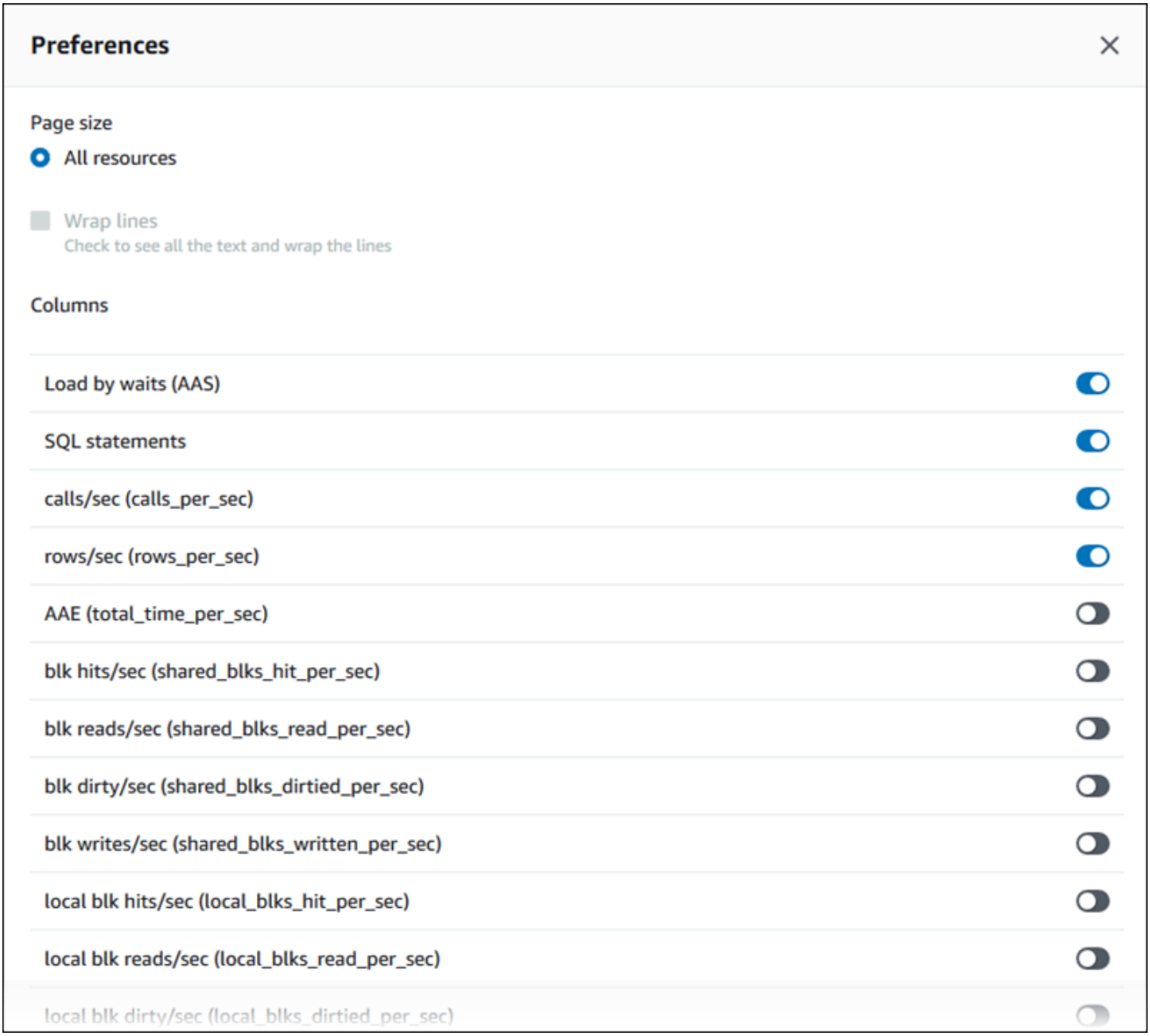
- SQLID de support : valeur de hachage de l'SQLidentifiant. Cette valeur sert uniquement à référencer un SQL identifiant lorsque vous travaillez avec le AWS Support. AWS Support n'a pas accès à vos informations réelles SQL IDs et à vos SQL SMS.
- Support Digest ID (ID digest de support) – Valeur de hachage de l'ID digest. Cette valeur sert uniquement à référencer un identifiant de résumé lorsque vous travaillez avec AWS Support. AWS Support n'a pas accès à votre résumé IDs ni à votre SQL texte.

Choisissez vos préférences en matière de statistiques

Vous pouvez contrôler les statistiques affichées dans l'SQLonglet supérieur en cliquant sur l'icône Préférences.



Lorsque vous choisissez l'icône Préférences, la fenêtre Préférences s'ouvre. La capture d'écran suivante est un exemple de la fenêtre Preferences (Préférences).



Pour activer les statistiques que vous souhaitez voir apparaître dans l'SQLonglet supérieur, utilisez votre souris pour faire défiler la fenêtre vers le bas, puis choisissez Continuer.

Pour plus d'informations sur les statistiques par seconde ou par appel pour les moteurs Amazon RDS, consultez la section sur les SQL statistiques spécifiques au moteur dans Statistiques SQL pour Performance Insights

Accès à plus de texte SQL dans le tableau de bord Performance Insights

Par défaut, chaque ligne du tableau Top SQL (Principaux éléments SQL) affiche 500 octets de texte SQL pour chaque instruction SQL.



Lorsqu'une instruction SQL dépasse 500 octets, vous pouvez afficher davantage de texte dans la section SQL text (Texte SQL) située sous le tableau Top SQL (Top SQL). Dans ce cas, la longueur maximale du texte affiché dans SQL text (Texte SQL) est de 4 Ko. Cette limite est imposée par la console et est soumise aux limites fixées par le moteur de base de données. Pour enregistrer le texte affiché dans SQL text (Texte SQL), sélectionnez Download (Télécharger).

Rubriques

- Limites de taille de texte pour les moteurs Amazon RDS
- Définition de la limite de SQL texte pour les instances de base de données Amazon RDS pour Postgre SQL
- Affichage et téléchargement de SQL texte dans le tableau de bord Performance Insights

Limites de taille de texte pour les moteurs Amazon RDS

Lorsque vous téléchargez du texte SQL, le moteur de la base de données détermine sa longueur maximale. Vous pouvez télécharger du texte SQL jusqu'aux limites suivantes par moteur.

Moteur de base de données	Longueur maximale du texte téléchargé
Amazon RDS pour MySQL et MariaDB	La longueur est fixée à 4 096 octets lorsque le schéma de performance est activé. Si le schéma de performance n'est pas activé, la longueur est fixée à 65 535 octets.
Amazon RDS for Microsoft SQL Server	4,096 caractères
Amazon RDS for Oracle	1 000 octets

La section SQL text (Texte SQL) de la console Performance Insights affiche jusqu'au la taille maximum renvoyée par le moteur. Par exemple, si MySQL renvoie au plus 1 Ko à Performance

Insights, celui-ci ne peut collecter et afficher que 1 Ko, même si la requête d'origine est plus volumineuse. Ainsi, lorsque vous visualisez la requête en SQL text (Texte SQL) ou que vous la téléchargez, Performance Insights renvoie le même nombre d'octets.

Si vous utilisez l'API AWS CLI or, Performance Insights n'applique pas la limite de 4 Ko imposée par la console. `DescribeDimensionKeyset` `GetResourceMetrics` renvoient au maximum 500 octets.

> **Note**
>
> `GetDimensionKeyDetails`renvoie la requête complète, mais la taille dépend de la limite du moteur.

Définition de la limite de SQL texte pour les instances de base de données Amazon RDS pour Postgre SQL

Amazon RDS for Postgre SQL texte différemment. Vous pouvez définir la limite de taille du texte avec le paramètre `track_activity_query_size` de l'instance de base de données. Ce paramètre possède les caractéristiques suivantes :

Taille de texte par défaut

Sur Amazon RDS pour Postgre SQL version 9.6, le `track_activity_query_size` paramètre par défaut est de 1 024 octets. Sur Amazon RDS pour Postgre SQL version 10 ou ultérieure, la valeur par défaut est de 4 096 octets.

Taille maximale du text

La limite `track_activity_query_size` est de 102 400 octets pour Amazon RDS pour Postgre SQL Postgre version 12 ou inférieure. Le maximum est de 1 Mo pour la version 13 et versions ultérieures.

Si le moteur renvoie 1 Mo à Performance Insights, la console affiche uniquement les 4 premiers Ko. Si vous téléchargez la requête, vous obtenez la totalité des 1 Mo. Dans ce cas, l'affichage et le téléchargement renvoient des quantités différentes d'octets. Pour plus d'informations sur le paramètre d'`track_activity_query_size`instance de base de données, consultez les statistiques d'exécution dans la documentation SQL Postgre.

Pour augmenter la taille SQL du texte, augmentez la `track_activity_query_size` limite. Pour modifier le paramètre, modifiez le paramètre dans le groupe de paramètres associé à l'instance de SQL base de données Amazon RDS for Postgre SQL .

Pour modifier le paramètre lorsque l'instance utilise le groupe de paramètres par défaut

1. Créez un nouveau groupe de paramètres pour l'instance de base de données, associé au moteur de base de données et à sa version appropriés.
2. Définissez le paramètre dans le nouveau groupe de paramètres.
3. Associez le nouveau groupe de paramètres à l'instance de base de données.

Pour de plus amples informations sur la définition d'un paramètre d'instance de base de données, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

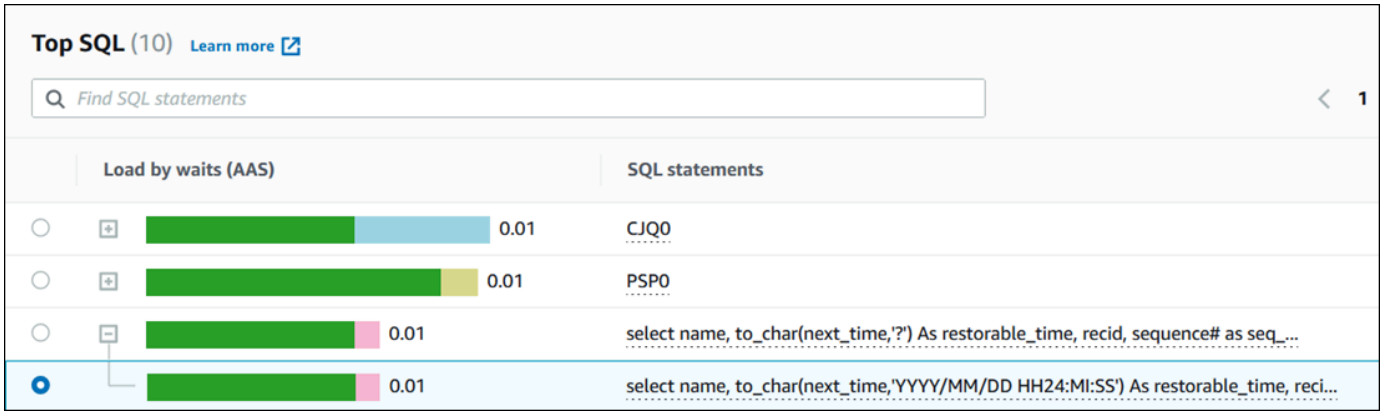
Affichage et téléchargement de SQL texte dans le tableau de bord Performance Insights

Dans le tableau de bord Performance Insights, vous pouvez consulter ou télécharger SQL du texte.

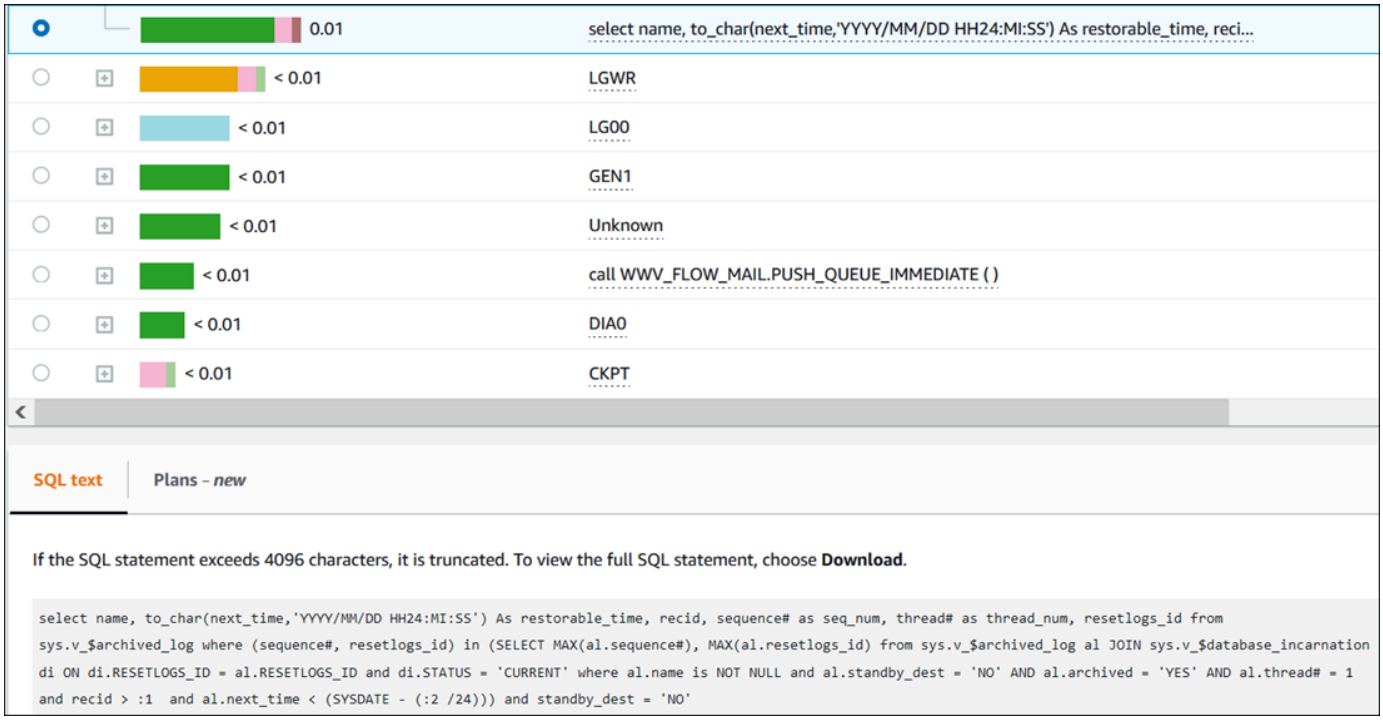
Pour afficher plus de SQL texte dans le tableau de bord Performance Insights

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Performance Insights.
3. Choisissez une instance de base de données.
4. Faites défiler l'écran vers le bas jusqu'à SQL l'onglet Top du tableau de bord Performance Insights.
5. Choisissez le signe plus pour développer un SQL résumé et choisissez l'une des requêtes secondaires du résumé.

   SQLles instructions dont le texte est supérieur à 500 octets ressemblent à l'image suivante.



6. Faites défiler l'écran jusqu'à l'onglet de SQLtexte.



Le tableau de bord Performance Insights peut afficher jusqu'à 4 096 octets pour chaque SQL instruction.

7. (Facultatif) Choisissez Copier pour copier l'SQLinstruction affichée, ou choisissez Télécharger pour télécharger l'SQLinstruction afin d'afficher le SQL texte jusqu'à la limite du moteur de base de données.

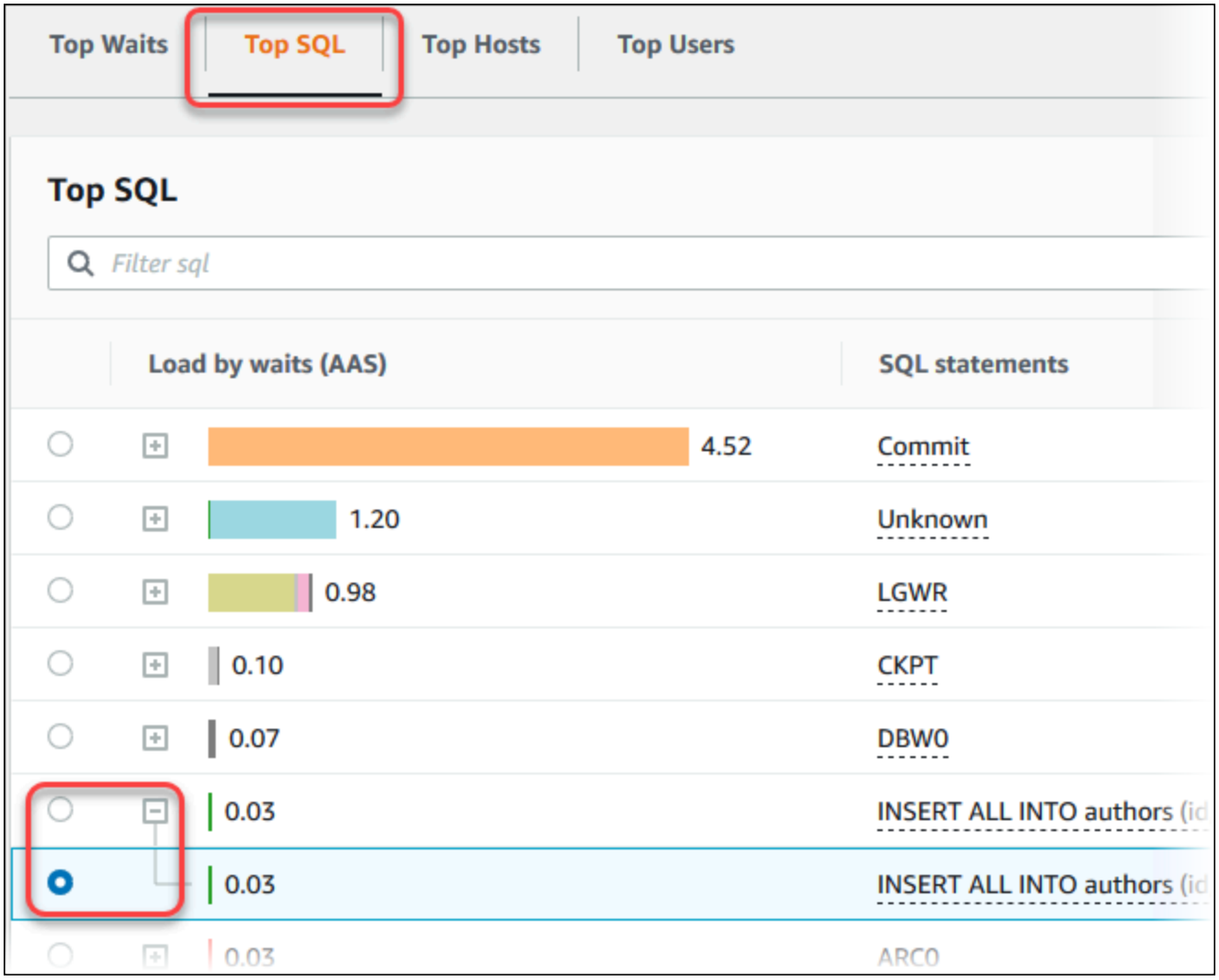
> **Note**
>
> Pour copier ou télécharger la SQL déclaration, désactivez les bloqueurs de fenêtres contextuelles.

Afficher SQL les statistiques dans le tableau de bord Performance Insights

Dans le tableau de bord Performance Insights, les SQL statistiques sont disponibles dans l'SQLonglet supérieur du graphique de charge de la base de données.

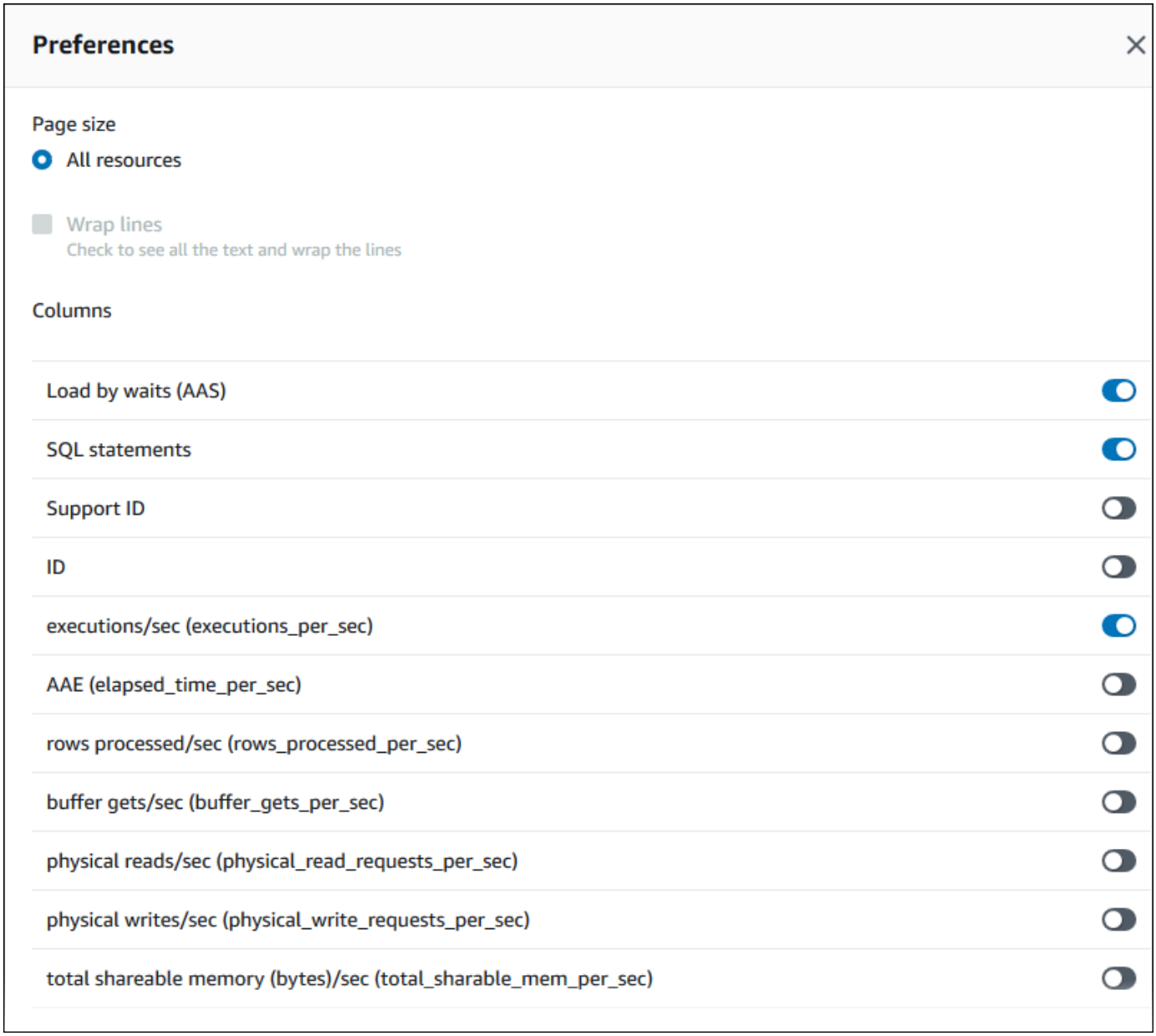
Pour afficher les statistiques SQL

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation gauche, choisissez Performance Insights.
3. En haut de la page, choisissez la base de données dont vous souhaitez consulter les SQL statistiques.
4. Faites défiler la page vers le bas et choisissez l'SQLonglet Top.
5. Choisissez une instruction individuelle ou une requête de synthèse.

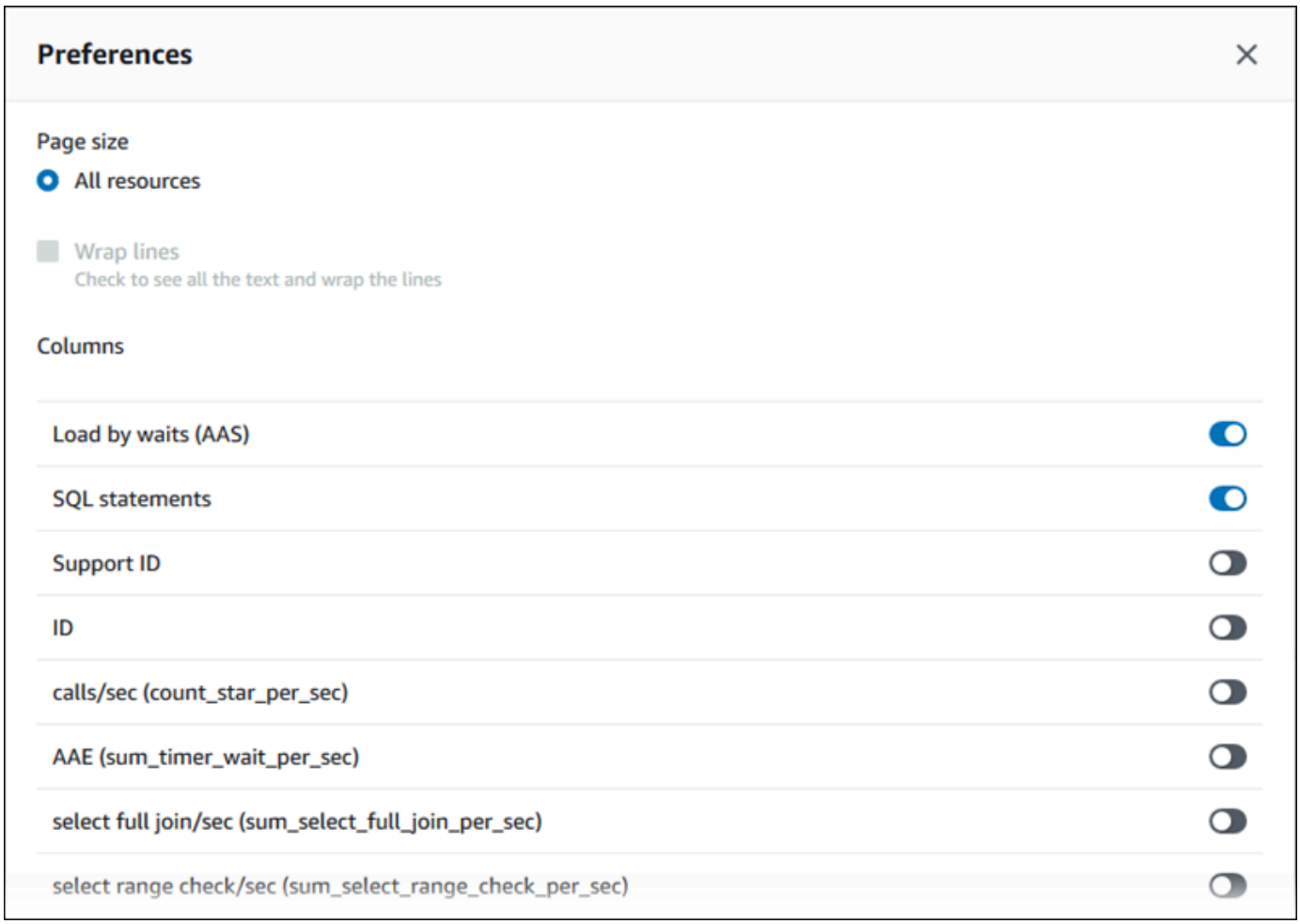


6. Choisissez les statistiques à afficher en sélectionnant l'icône en forme d'engrenage dans le coin supérieur droit du graphique. Pour une description des SQL statistiques relatives aux moteurs Amazon RDS , consultezStatistiques SQL pour Performance Insights.

   L'exemple suivant montre les préférences de statistiques pour les instances de base de données Oracle.



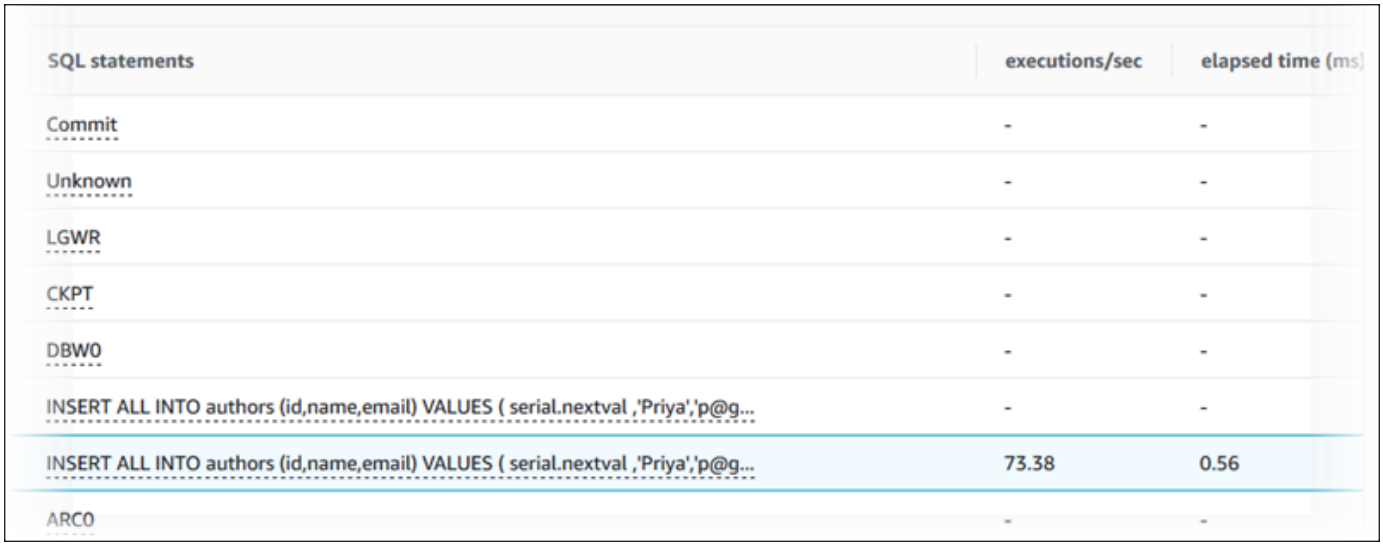
   L'exemple suivant montre les préférences pour les instances MariaDB et My SQL My DB.



7. Choisissez Save (Enregistrer) pour enregistrer vos préférences.

   Le SQL tableau supérieur est actualisé.

   L'exemple suivant montre les statistiques d'une SQL requête Oracle.



SQL statements	executions/sec	elapsed time (ms)
Commit	-	-
Unknown	-	-
LGWR	-	-
CKPT	-	-
DBW0	-	-
INSERT ALL INTO authors (id,name,email) VALUES (serial.nextval ,'Priya','p@g...	-	-
INSERT ALL INTO authors (id,name,email) VALUES (serial.nextval ,'Priya','p@g...	73.38	0.56
ARC0	-	-

Analyse de la PDB charge Oracle la plus élevée

Lorsque vous analysez la charge sur une base de données de conteneur Oracle (CDB), vous souhaiterez peut-être identifier les bases de données enfichables (PDBs) qui contribuent le plus à la charge de la base de données. Vous souhaiterez peut-être également comparer les performances de personnes PDBs exécutant des requêtes similaires afin d'affiner les performances. Pour plus d'informations sur OracleCDBs, consultezArchitecture de base de données RDS for Oracle.

Dans le tableau de bord Amazon RDS Performance Insights, vous trouverez des informations sur les bases de données enfichables (PDBs) sous PDB l'onglet Top de l'onglet Dimensions.

Pour obtenir des informations sur la région, le moteur de base de données et les classes d'instance compatibles avec cette fonctionnalité, consultezPrise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

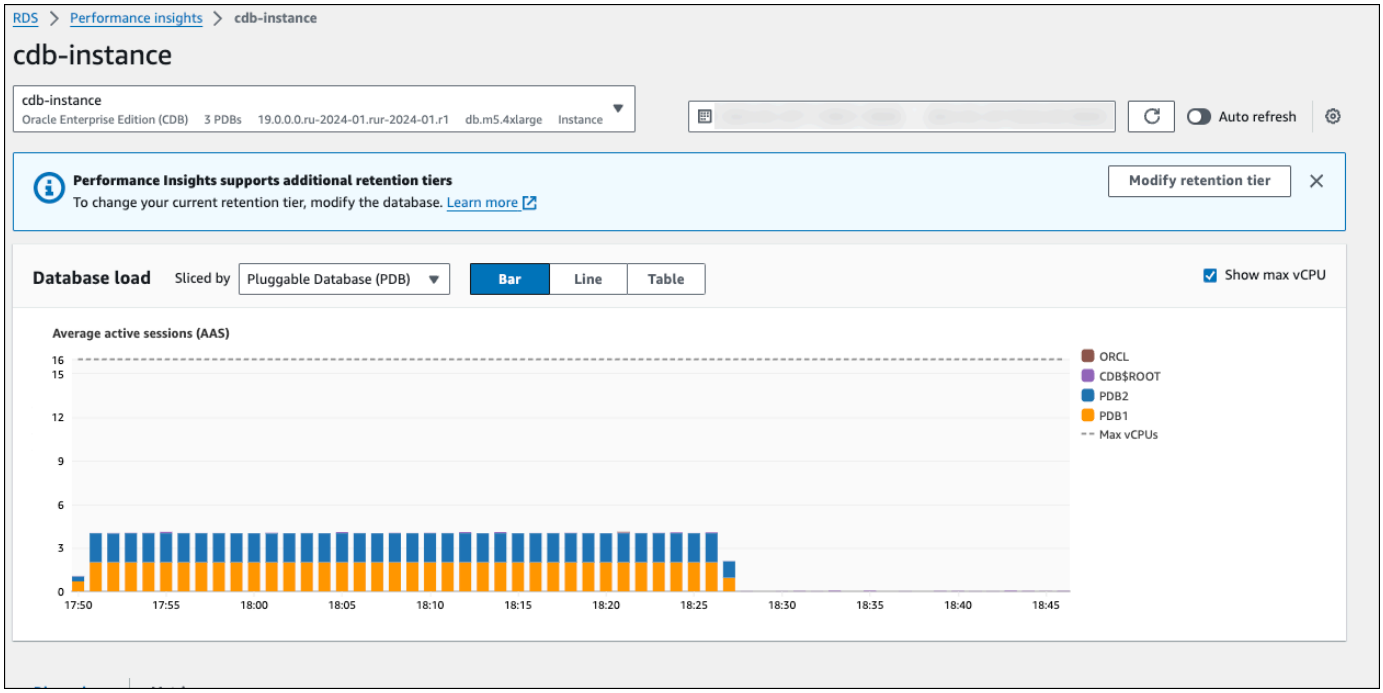
Pour analyser la PDB charge maximale dans un Oracle CDB

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation de gauche, sélectionnez Performance Insights.
3. Choisissez une CDB instance Oracle.

   Le tableau de bord Performance Insights correspondant à cette instance de base de données s'affiche.

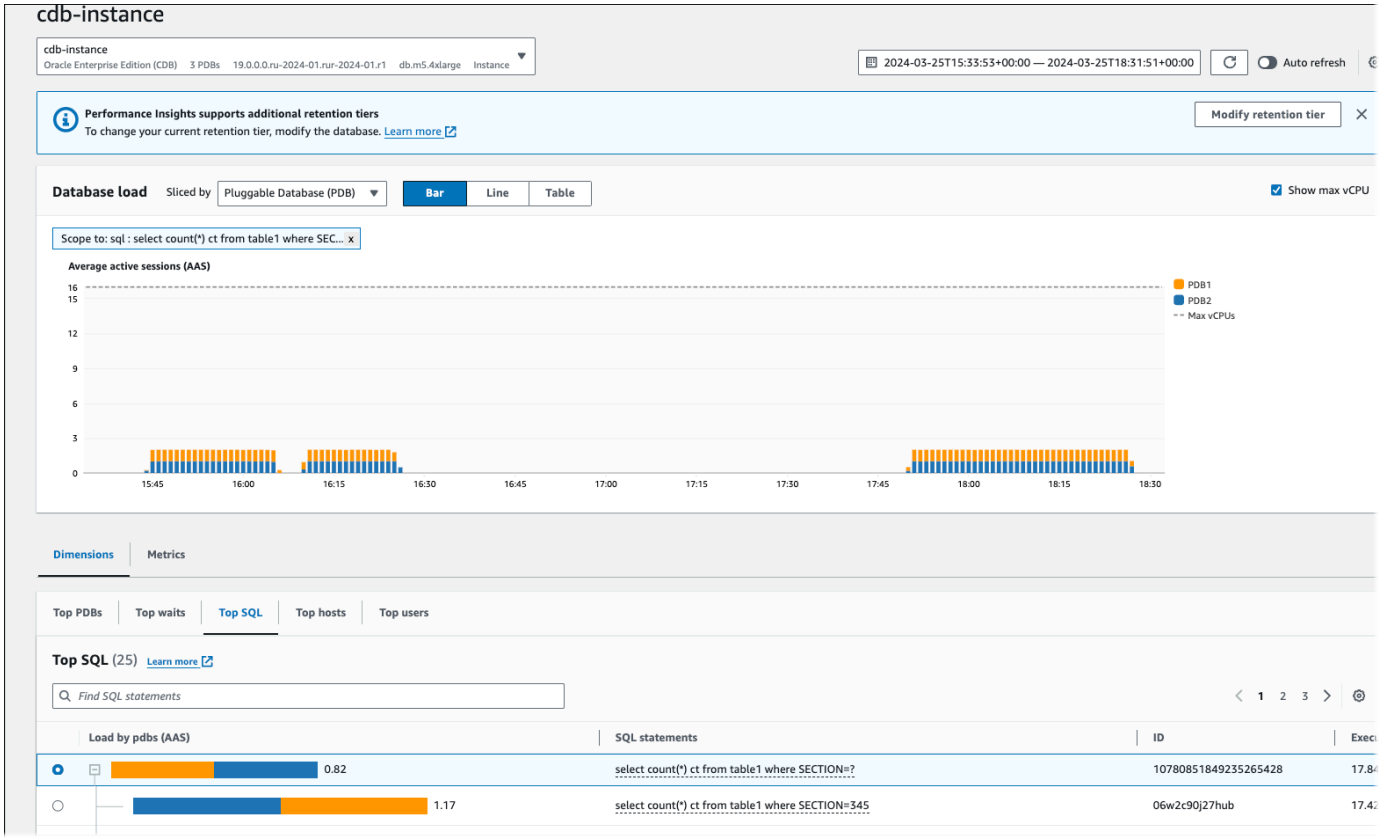
4. Dans la section Chargement de base de données (chargement de base de données), choisissez Pluggable database (PDB) à côté de Slice by.

   Le graphique des sessions actives moyennes indique PDB les sessions les plus chargées. Les PDB identifiants apparaissent à droite des carrés codés par couleur. Chaque identifiant identifie de manière unique unPDB.

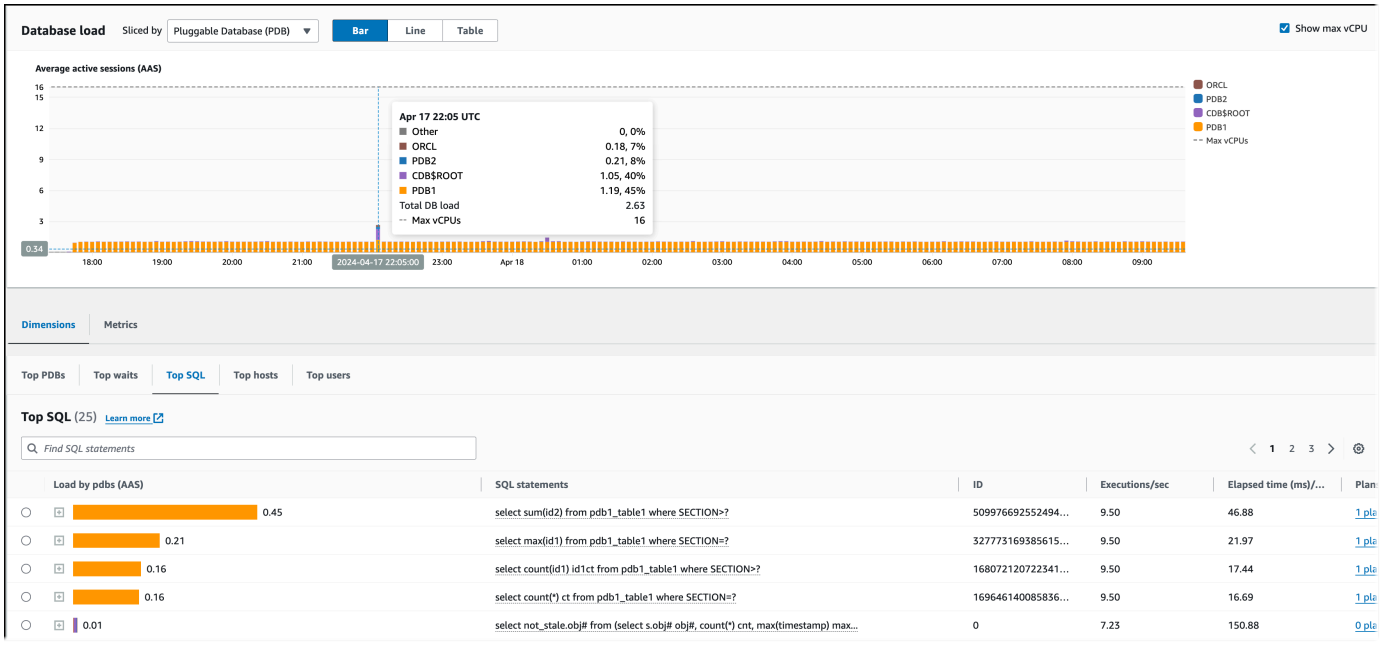


5. Faites défiler la page vers le bas jusqu'à SQL l'onglet Haut.

   Dans l'exemple suivant, vous pouvez voir la même SQL requête et la charge qu'elle entraîne pour plusieursPDBs.



Dans l'exemple suivant, un appareil supporte PDB une charge plus élevée que les autres PDBs dans leCDB.



Pour plus d'informations sur OracleCDBs, reportez-vous aux CDBssections et PDBs.

Analyse des plans d'exécution à l'aide du tableau de bord Performance Insights pour Amazon RDS

Dans le tableau de bord Amazon RDS Performance Insights, vous trouverez des informations sur les plans d'exécution pour les instances de base de données Oracle et SQL Server. Vous pouvez utiliser ces informations pour savoir quels plans contribuent le plus à la charge de base de données.

Pour analyser Oracle or SQL Server plans d'exécution, consultez les rubriques suivantes.

Analyse des plans d'exécution

- Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord Performance Insights pour Amazon RDS
- Analyse des plans d'exécution des SQL serveurs à l'aide du tableau de bord Performance Insights pour Amazon RDS

Présentation de l'analyse des plans d'exécution pour Amazon RDS

Vous pouvez utiliser le tableau de bord Amazon RDS Performance Insights pour savoir quels plans contribuent le plus à la charge de base de données pour les instances de base de données Oracle et SQL Server.

Par exemple, les principales SQL déclarations à un moment donné peuvent utiliser les plans présentés dans le tableau suivant.

Haut SQL	Plan
SELECTSUM(montant_vendu) prod_id des FROM ventes = 10 WHERE	Plan A
SELECTSUM(montant_vendu) prod_id des FROM ventes = 521 WHERE	Plan B
SELECTSUM(s_total) WHERE région FROM de vente = 10	Plan A
SELECT* FROM emp WHERE emp_id = 1000	Plan C
SELECTSUM(montant_vendu) prod_id des FROM ventes = 72 WHERE	Plan A

Avec la fonction de plan de Performance Insights, vous pouvez effectuer les opérations suivantes :

- Découvrez quels plans sont utilisés par les principales SQL requêtes.

  Par exemple, vous pouvez découvrir que la majeure partie de la charge de base de données est générée par des requêtes utilisant le plan A et le plan B, avec seulement un faible pourcentage utilisant le plan C.
- Comparez différents plans pour la même requête.

  Dans l'exemple précédent, trois requêtes sont identiques à l'exception de l'ID du produit. Deux requêtes utilisent le plan A, mais une requête utilise le plan B. Pour voir la différence entre les deux plans, vous pouvez utiliser Performance Insights.
- Découvrez quand une requête est passée à un nouveau plan.

  Vous pouvez voir qu'une requête a utilisé le plan A, puis est passée au plan B à un moment donné. Y a-t-il eu un changement dans la base de données à ce moment ? Par exemple, si une table est vide, l'optimiseur peut choisir une analyse complète de table. Si la table est chargée avec un million de lignes, l'optimiseur peut passer à une analyse de plage d'index.
- Examinez les étapes spécifiques d'un plan présentant le coût le plus élevé.

  Par exemple, une requête de longue durée peut indiquer l'absence d'une condition de jointure dans une équijointure. Cette condition manquante mène à une jointure cartésienne, qui joint toutes les lignes de deux tables.

Vous pouvez effectuer les tâches précédentes à l'aide de la fonction de capture de plan de Performance Insights. Tout comme vous pouvez découper les requêtes en fonction des événements d'attente et du SQL début, vous pouvez les découper en fonction de la dimension du plan.

Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord Performance Insights pour Amazon RDS

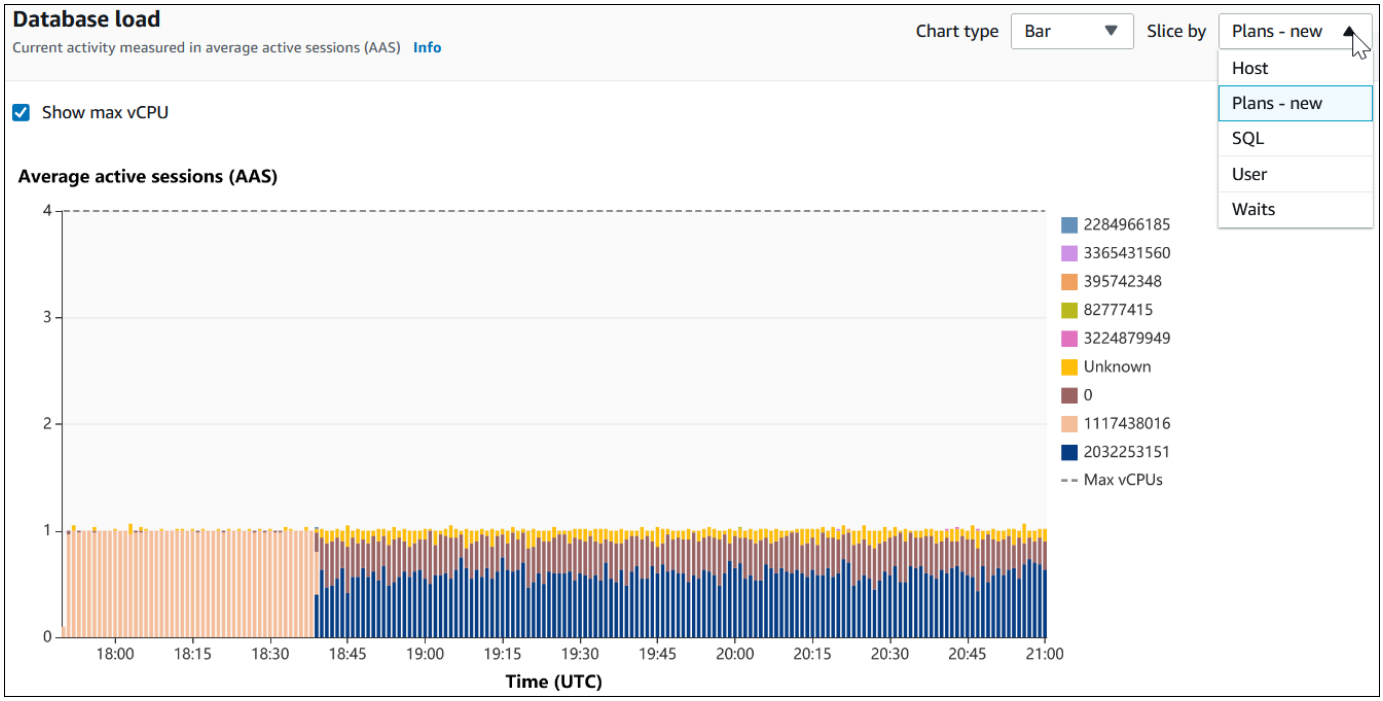
Lorsque vous analysez la charge de base de données sur une base de données Oracle, vous voudrez peut-être savoir quels plans contribuent le plus à la charge de la base de données. Vous pouvez déterminer quels plans contribuent le plus à la charge de base de données à l'aide de la fonctionnalité de capture de plans de Performance Insights.

Pour analyser les plans d'exécution Oracle à l'aide de la console

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.

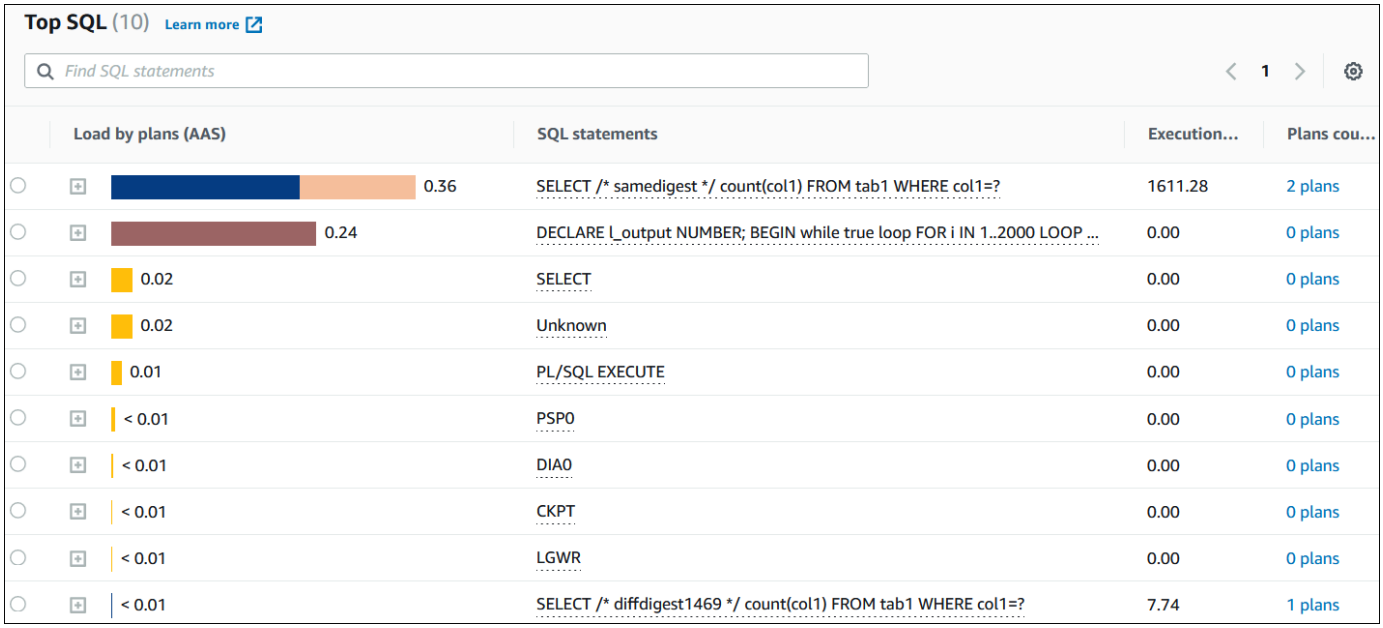
2. Dans le volet de navigation, choisissez Performance Insights.
3. Choisissez une instance de base de données Oracle. Le tableau de bord de Performance Insights s'affiche pour cette instance de base de données.
4. Dans la section Database load (DB load) (Charge de base de données), choisissez Plans à côté de Slice by (Trancher par).

   Le graphique du nombre moyen de sessions actives indique les forfaits utilisés par vos principaux SQL relevés. Les valeurs de hachage du plan apparaissent à droite des carrés à code couleur. Chaque valeur de hachage identifie de manière unique un plan.



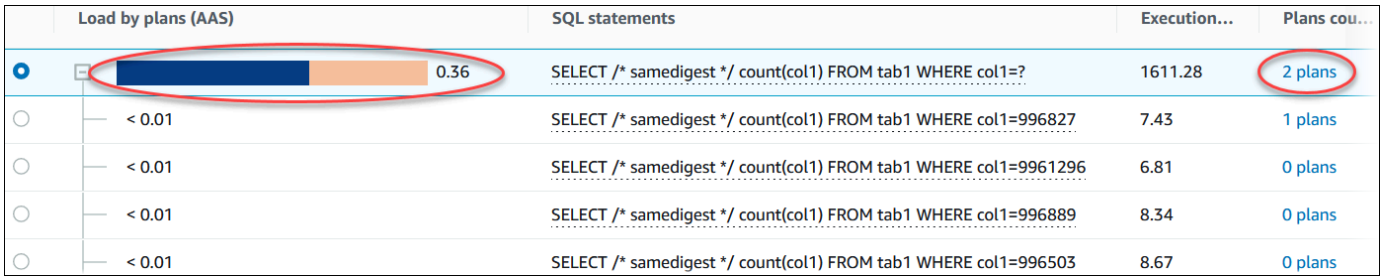
5. Faites défiler la page vers le bas jusqu'à SQL l'onglet Haut.

   Dans l'exemple suivant, le top SQL digest comporte deux plans. Vous pouvez deviner qu'il s'agit d'un récapitulatif grâce au point d'interrogation de l'instruction.



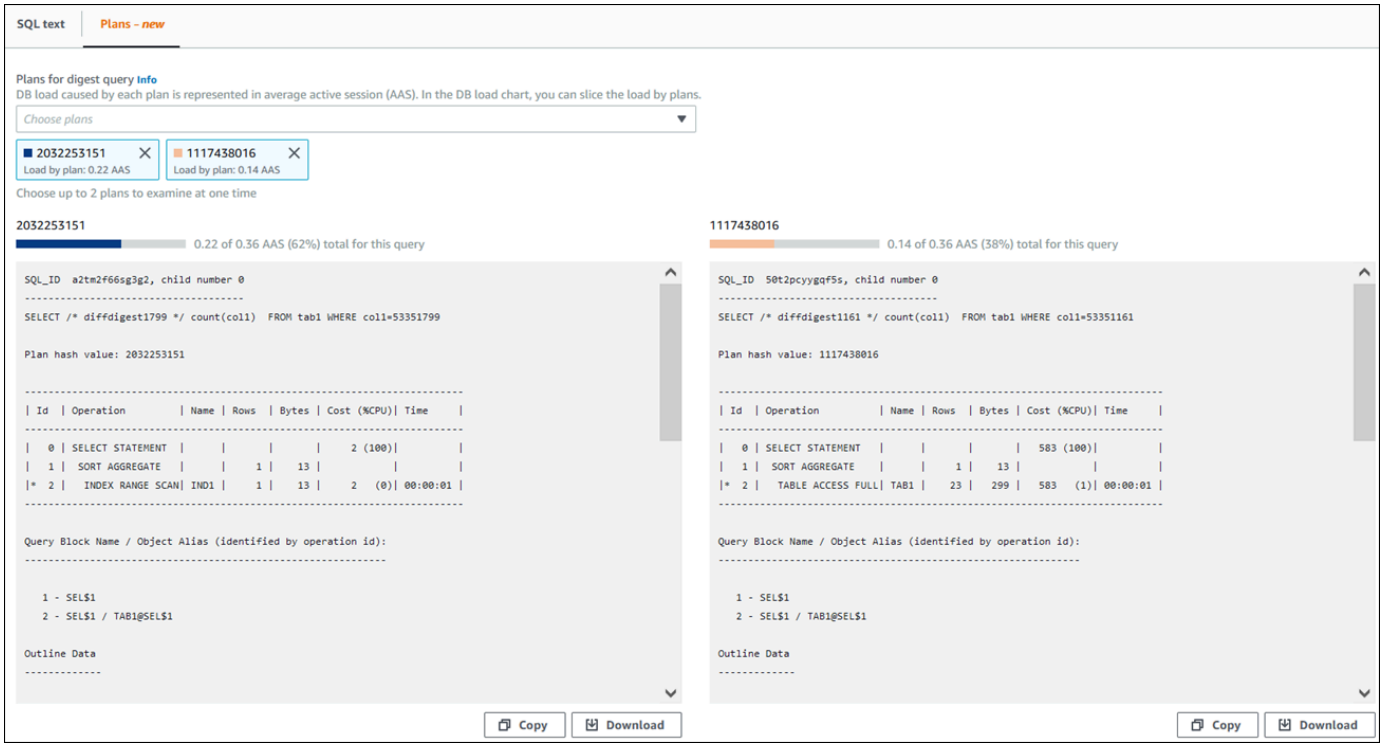
6. Choisissez le récapitulatif pour afficher ses instructions de composants.

   Dans l'exemple suivant, l'instruction SELECT est une requête récapitulative. Les requêtes de composants du récapitulatif utilisent deux plans différents. Les couleurs des plans correspondent au graphique de charge de base de données. Le nombre total de plans dans le récapitulatif est indiqué dans la deuxième colonne.

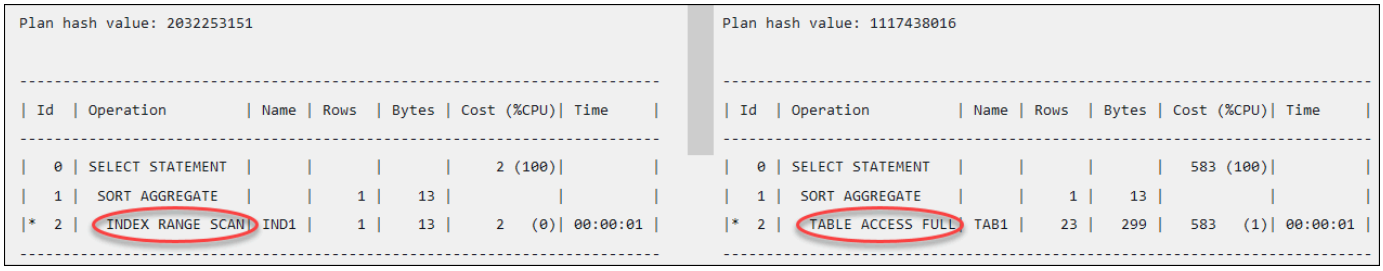


7. Faites défiler et choisissez deux Plans à comparer dans la liste Plans for digest query (Plans pour requête récapitulative).

   Vous pouvez afficher un ou deux plans pour une requête à la fois. La capture d'écran suivante compare les deux plans du récapitulatif, avec le hachage 2032253151 et le hachage 1117438016. Dans l'exemple suivant, 62 % des sessions actives moyennes exécutant cette requête récapitulative utilisent le plan de gauche, tandis que 38 % utilisent celui de droite.



Dans cet exemple, les plans diffèrent d'une manière importante. L'étape 2 du plan 2032253151 utilise une analyse d'index, tandis que le plan 1117438016 utilise une analyse de table complète. Pour une table comportant un grand nombre de lignes, la requête d'une seule ligne est presque toujours plus rapide avec une analyse d'index.



8. (Facultatif) Choisissez Copy (Copier) pour copier le plan dans le presse-papier, ou Download (Téléchargement) pour enregistrer le plan sur votre disque dur.

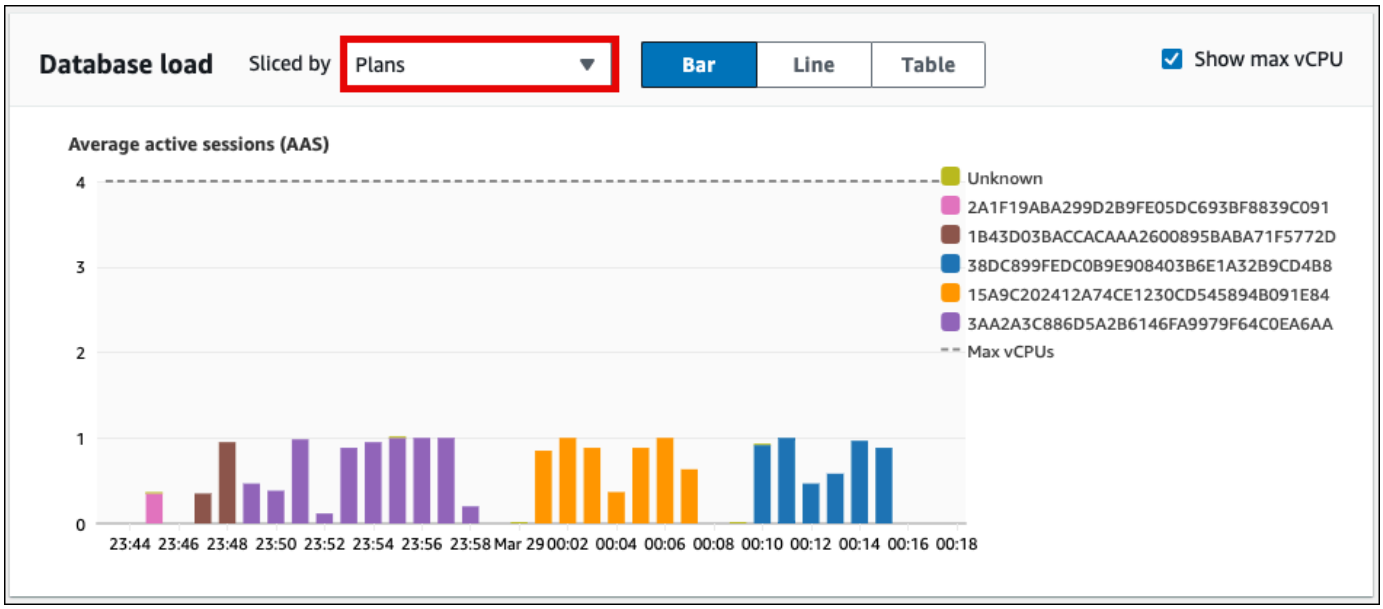
Analyse des plans d'exécution des SQL serveurs à l'aide du tableau de bord Performance Insights pour Amazon RDS

Lorsque vous analysez la charge de base de données sur une base de données SQL serveur, vous souhaiterez peut-être savoir quels plans contribuent le plus à la charge de base de données. Vous pouvez déterminer quels plans contribuent le plus à la charge de base de données à l'aide de la fonctionnalité de capture de plans de Performance Insights.

Pour analyser les plans d'exécution SQL du serveur à l'aide de la console

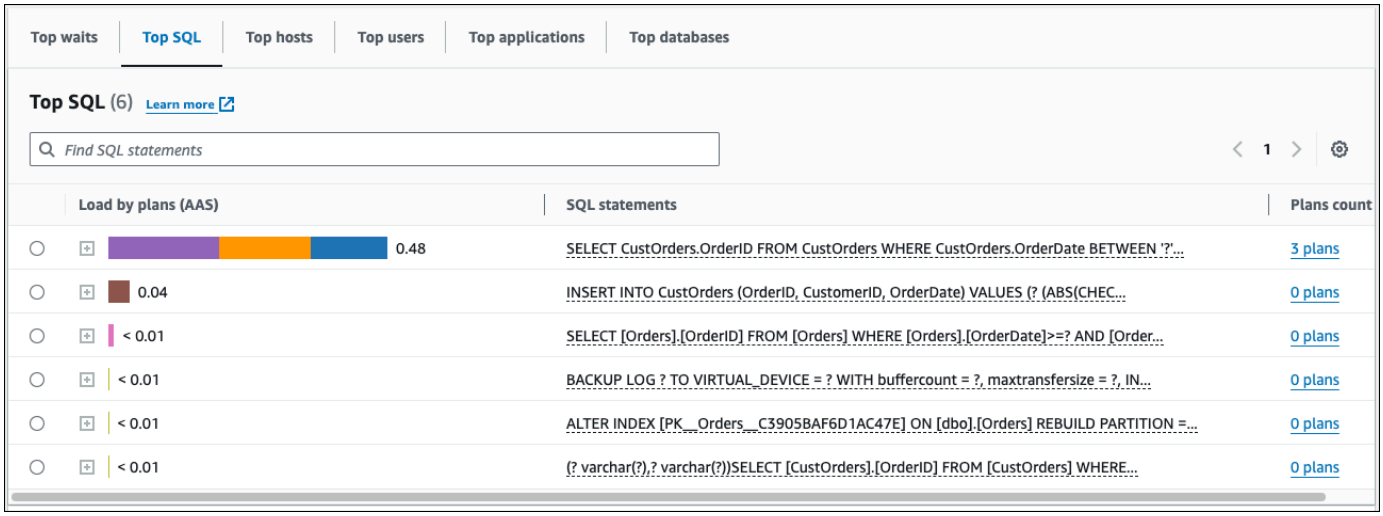
1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Performance Insights.
3. Choisissez une instance de base de données de SQL serveur. Le tableau de bord de Performance Insights s'affiche pour cette instance de base de données.
4. Dans la section Database load (DB load) (Charge de base de données), choisissez Plans à côté de Slice by (Trancher par).

   Le graphique du nombre moyen de sessions actives indique les forfaits utilisés par vos principaux SQL relevés. Les valeurs de hachage du plan apparaissent à droite des carrés à code couleur. Chaque valeur de hachage identifie de manière unique un plan.



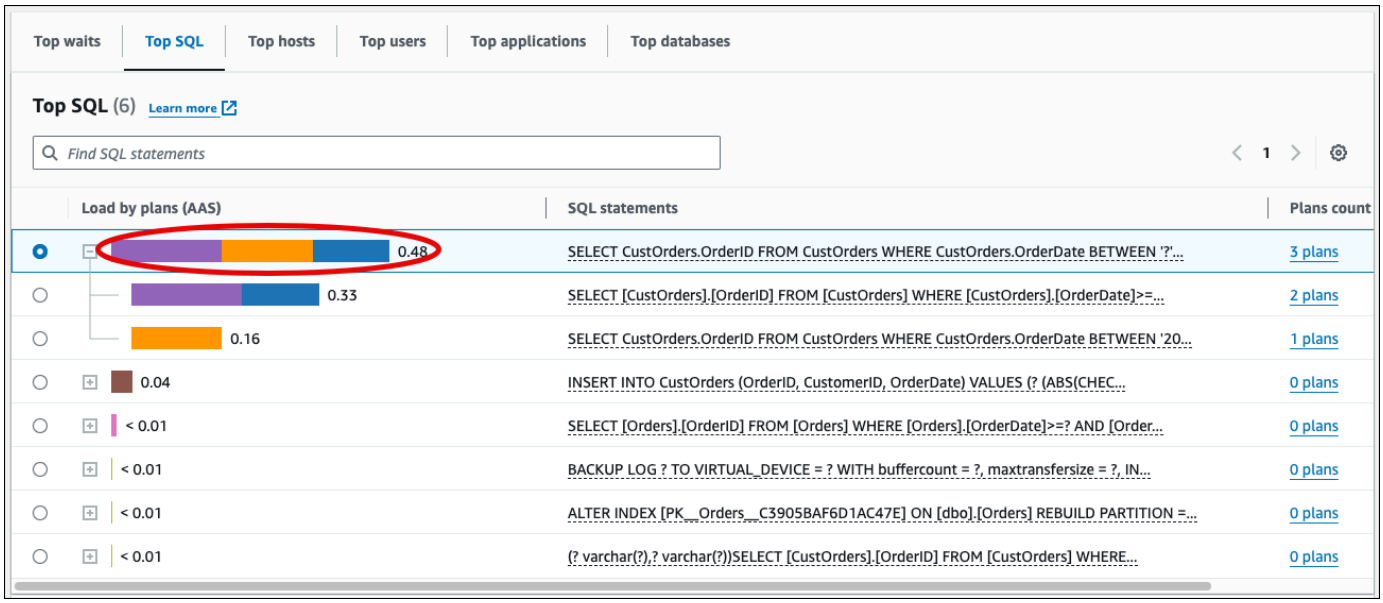
5. Faites défiler la page vers le bas jusqu'à SQL l'onglet Haut.

   Dans l'exemple suivant, le top SQL digest comporte trois plans. La présence d'un point d'interrogation dans l'SQLénoncé indique qu'il s'agit d'un résumé. Pour afficher le SQL relevé complet, choisissez une valeur dans la colonne SQLdes relevés.



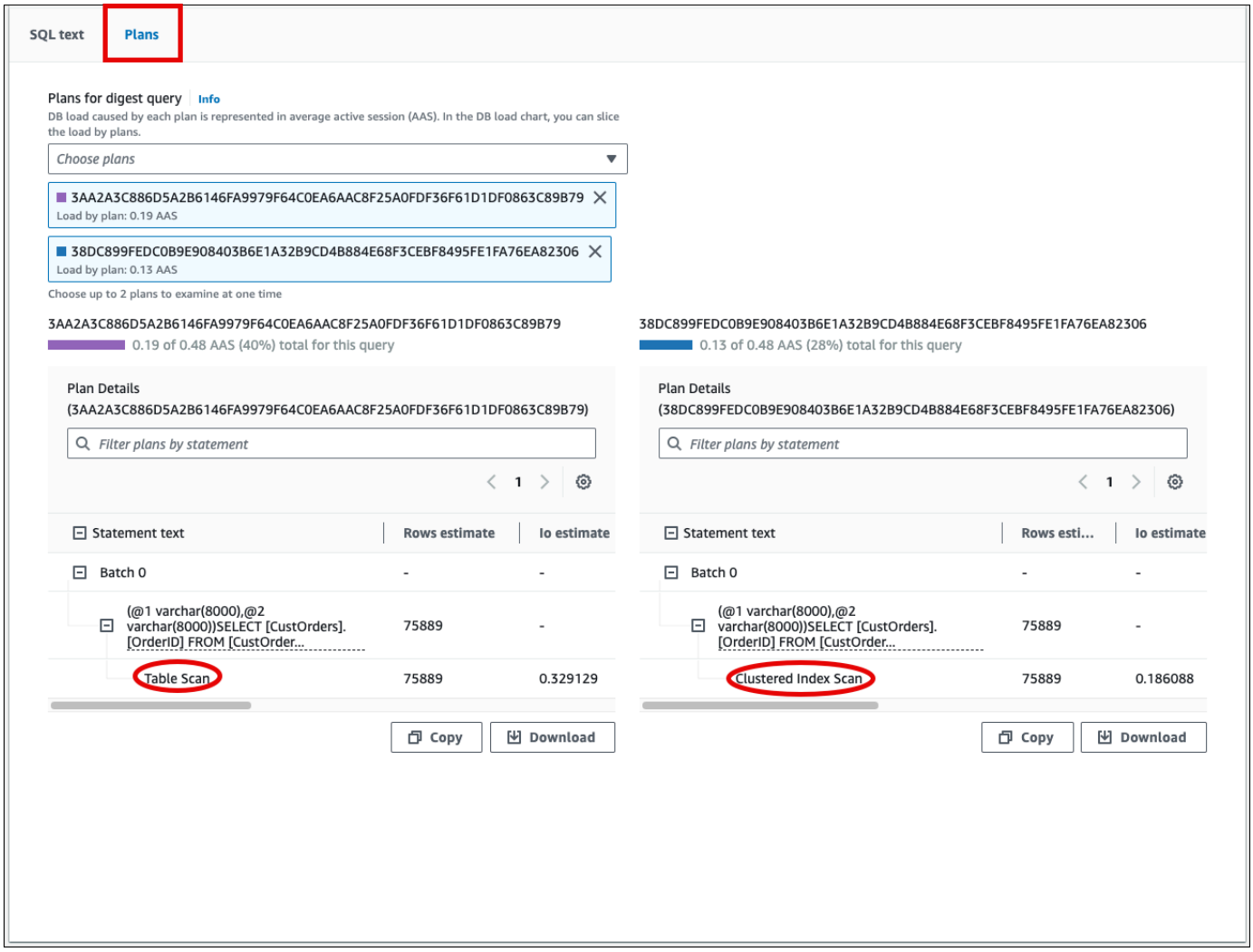
6. Choisissez le récapitulatif pour afficher ses instructions de composants.

   Dans l'exemple suivant, l'instruction `SELECT` est une requête récapitulative. Les requêtes des composants du condensé utilisent trois plans d'exécution différents. Les couleurs attribuées aux plans correspondent au diagramme de charge de la base de données.



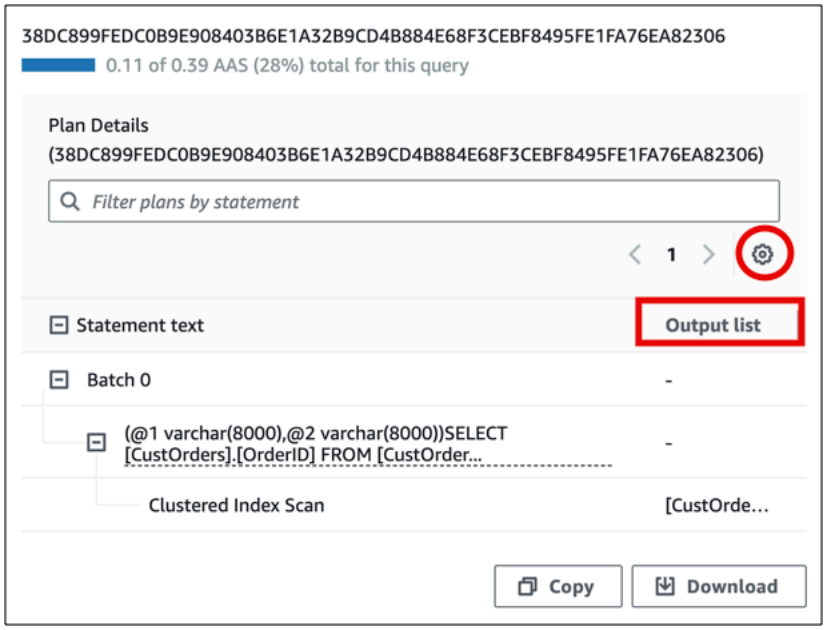
7. Faites défiler et choisissez deux Plans à comparer dans la liste Plans for digest query (Plans pour requête récapitulative).

   Vous pouvez afficher un ou deux plans pour une requête à la fois. La capture d'écran suivante compare deux plans du résumé. Dans l'exemple suivant, 40 % des sessions actives en moyenne exécutant cette requête de résumé utilisent le plan de gauche, tandis que 28 % utilisent le plan de droite.



Dans l'exemple précédent, les plans diffèrent de manière importante. L'étape 2 du plan de gauche utilise un scan de table, tandis que le plan de droite utilise un scan d'index clusterisé. Pour une table comportant un grand nombre de lignes, une requête récupérant une seule ligne est presque toujours plus rapide avec un scan d'index clusterisé.

8. (Facultatif) Cliquez sur l'icône Paramètres dans le tableau Détails du plan pour personnaliser la visibilité et l'ordre des colonnes. La capture d'écran suivante montre le tableau des détails du plan avec la colonne Liste des sorties comme deuxième colonne.



9. (Facultatif) Choisissez Copy (Copier) pour copier le plan dans le presse-papier, ou Download (Téléchargement) pour enregistrer le plan sur votre disque dur.

> **Note**
>
> Performance Insights affiche les plans d'exécution estimés à l'aide d'une arborescence hiérarchique. Le tableau inclut les informations d'exécution partielle pour chaque instruction. Pour plus d'informations sur les colonnes du tableau Détails du plan, consultez SETSHOWPLAN_ ALL dans la documentation SQL du serveur. Pour afficher les informations d'exécution complètes d'un plan d'exécution estimé, choisissez Télécharger pour télécharger le plan, puis téléchargez-le dans SQL Server Management Studio. Pour plus d'informations sur l'affichage d'un plan d'exécution estimé à l'aide de SQL Server Management Studio, voir Afficher un plan d'exécution estimé dans la documentation SQL du serveur.

Consulter les recommandations proactives de Performance Insights

Amazon RDS Performance Insights surveille des indicateurs spécifiques et crée automatiquement des seuils en analysant les niveaux susceptibles de poser problème pour une ressource spécifique. Lorsque les nouvelles valeurs métriques dépassent un seuil prédéfini sur une période donnée, Performance Insights génère une recommandation proactive. Cette recommandation permet d'éviter tout impact futur sur les performances de la base de données. Pour bénéficier de ces

recommandations proactives, vous devez activer Performance Insights avec une période de rétention payante.

Pour plus d'informations sur l'activation de Performance Insights, consultez Activer et désactiver Performance Insights pour Amazon RDS. Pour plus d'informations sur la tarification et la conservation des données pour Performance Insights, consultezTarification et conservation des données pour Performance Insights.

Pour connaître les régions, les moteurs de base de données et les classes d'instance pris en charge pour les recommandations proactives, consultezPrise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

Vous pouvez consulter l'analyse détaillée et les investigations recommandées concernant les recommandations proactives sur la page de détails des recommandations.

Pour plus d'informations sur les recommandations, consultezRecommandations d'Amazon RDS.

Pour consulter l'analyse détaillée d'une recommandation proactive

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, effectuez l'une des opérations suivantes :

   - Choisissez Recommandations.

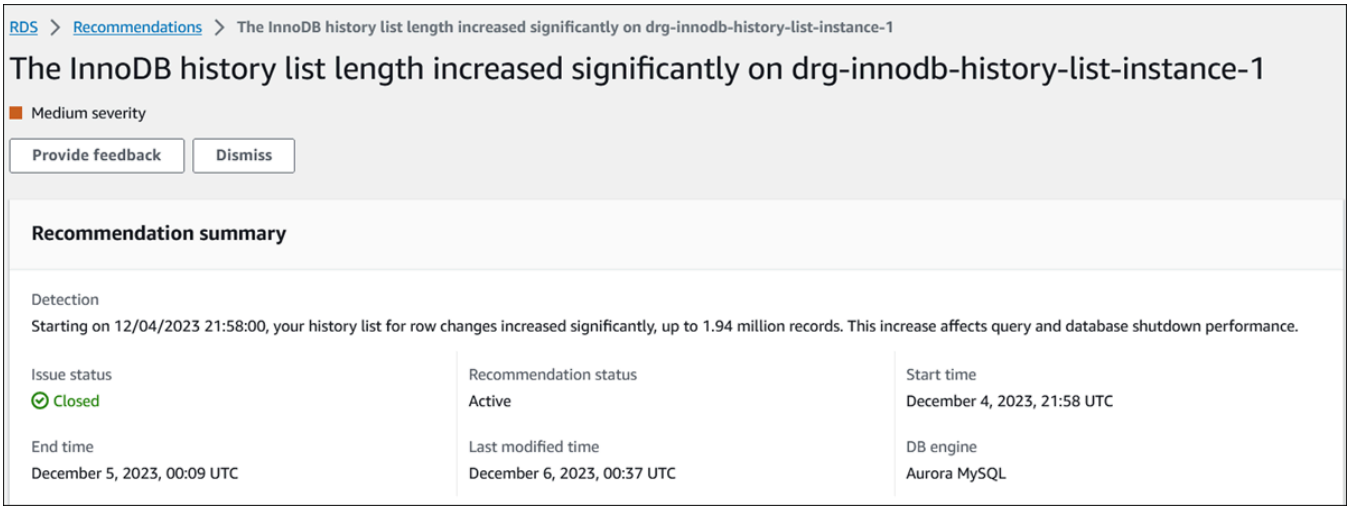
     La page Recommandations affiche une liste de recommandations triées par gravité pour toutes les ressources de votre compte.
   - Choisissez Bases de données, puis sélectionnez Recommandations pour une ressource dans la page des bases de données.

     L'onglet Recommandations affiche les recommandations et leurs détails pour la ressource sélectionnée.
3. Trouvez une recommandation proactive et choisissez Afficher les détails.

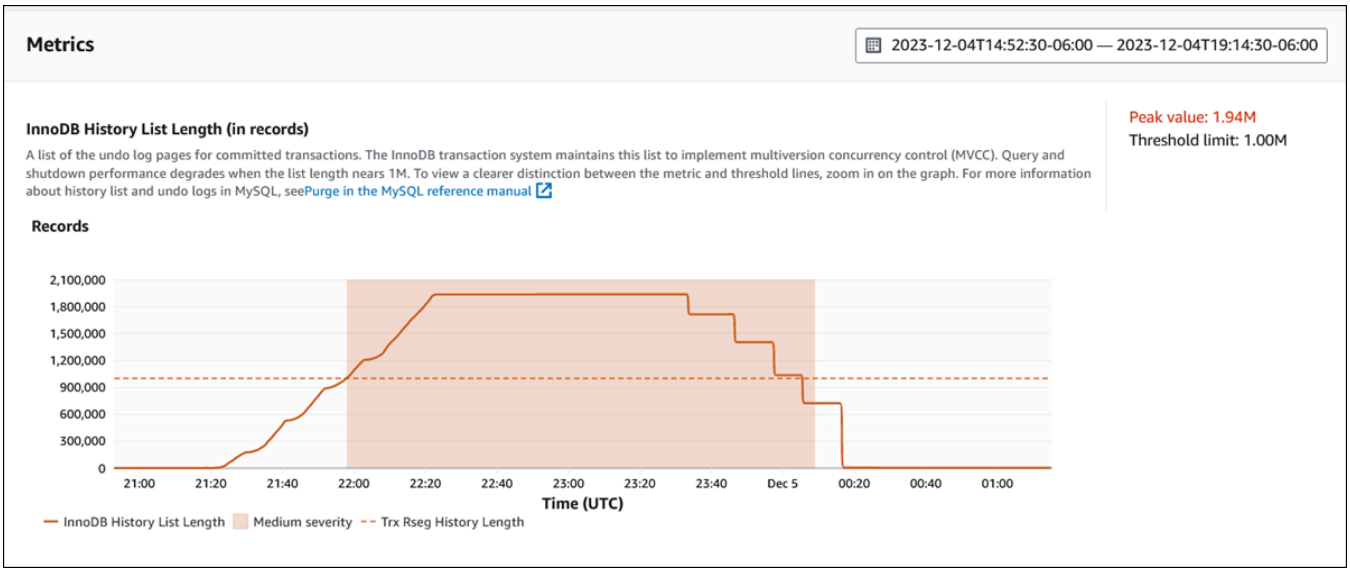
   La page de détails des recommandations s'affiche. Le titre indique le nom de la ressource affectée ainsi que le problème détecté et sa gravité.

   Les composants de la page détaillée des recommandations sont les suivants :

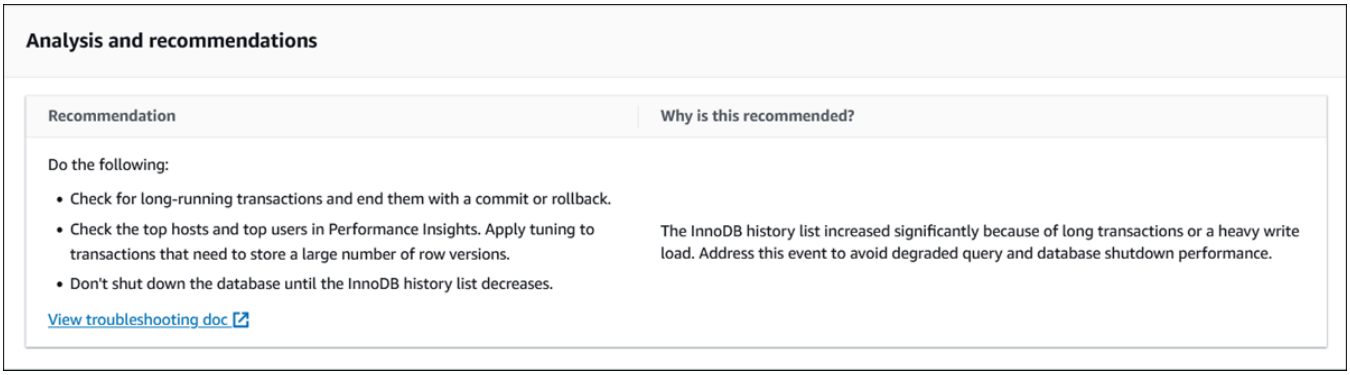
- Résumé des recommandations : problème détecté, état de la recommandation et du problème, heure de début et de fin du problème, heure de modification de la recommandation et type de moteur.



- Métriques — Les graphiques du problème détecté. Chaque graphique affiche un seuil déterminé par le comportement de base de la ressource et les données de la métrique rapportées depuis le début du problème.



- Analyse et recommandations — La recommandation et le motif de la recommandation suggérée.



Vous pouvez examiner la cause du problème, puis exécuter les actions recommandées pour résoudre le problème, ou choisir Ignorer dans le coin supérieur droit pour ignorer la recommandation.

Récupération de métriques avec Performance Insights API for Amazon RDS

Lorsque Performance Insights est activé, il API fournit une visibilité sur les performances de l'instance. Amazon CloudWatch Logs fournit la source officielle pour les mesures de surveillance des ventes pour AWS les services.

Performance Insights fournit une vue spécifique au domaine de la charge de base de données mesurée en nombre moyen de sessions actives ()AAS. Cette métrique apparaît aux API consommateurs sous la forme d'un ensemble de données chronologiques bidimensionnel. La dimension temporelle des données fournit les données de charge de la base de données pour chaque point temporel de la plage de temps interrogée. Chaque point dans te temps décompose la charge globale par rapport aux dimensions demandées, par exemple, `SQL`, `Wait-event`, `User` ou `Host`, mesurée à ce point dans le temps.

Amazon RDS Performance Insights surveille votre instance de base de RDS données Amazon () afin que vous puissiez analyser et résoudre les problèmes de performance de la base de données. Vous pouvez consulter les données de Performance Insights dans AWS Management Console. Performance Insights fournit également un accès public API afin que vous puissiez interroger vos propres données. Vous pouvez utiliser le API pour effectuer les opérations suivantes :

- Déchargement des données dans une base de données
- Ajout de données Performance Insights aux tableaux de bord de surveillance existants

- Création d'outils de surveillance

Pour utiliser Performance InsightsAPI, activez Performance Insights sur l'une de vos RDS instances de base de données Amazon. Pour de plus amples informations sur l'activation de Performance Insights, veuillez consulter Activer et désactiver Performance Insights pour Amazon RDS. Pour plus d'informations sur les Performance InsightsAPI, consultez le Amazon RDS Performance Insights API Reference.

Le Performance Insights API fournit les opérations suivantes.

Action Performance Insights	AWS CLI commande	Description
CreatePerformanceAnalysisReport	aws pi create-performance-analysis-report	Crée un rapport d'analyse des performances pour une période spécifique pour l'instance de base de données. Le résultat est `AnalysisReportId` qui est l'identifiant unique du rapport.
DeletePerformanceAnalysisReport	aws pi delete-performance-analysis-report	Supprime un rapport d'analyse des performances.
DescribeDimensionKeys	aws pi describe-dimension-keys	Récupère les N premières clés de dimension d'une mesure sur une période spécifique.
GetDimensionKeyDetails	aws pi get-dimension-key-details	Récupère les attributs du groupe de dimensions spécifié pour une instance de base de données ou une source de données. Par exemple, si vous spécifiez un SQL identifiant et si les détails de la dimension sont disponibles, le texte intégral de la dimension

Action Performance Insights	AWS CLI commande	Description
		`db.sql.statement` associée à cet identifiant est `GetDimensionKeyDetails` extrait. Cette opération est utile car elle `GetResourceMetrics` `DescribeDimensionKeys` ne prend pas en charge la récupération de texte de SQL déclaration volumineux.
`GetPerformanceAnalysisReport`	`aws pi get-performance-analysis-report`	Récupère le rapport, y compris les informations du rapport. Le résultat inclut l'état du rapport, l'ID du rapport, les détails temporels du rapport, les informations et les recommandations.
`GetResourceMetadata`	`aws pi get-resource-metadata`	Récupérez les métadonnées de différentes fonctions. Par exemple, les métadonnées peuvent indiquer qu'une fonction est activée ou désactivée sur une instance de base de données spécifique.

Action Performance Insights	AWS CLI commande	Description
GetResourceMetrics	aws pi get-resource-metrics	Récupère les métriques Performance Insights d'un ensemble de sources de données, au cours d'une période. Vous pouvez fournir des groupes de dimensions et des dimensions spécifiques, ainsi que des critères d'agrégation et de filtrage, pour chaque groupe.
ListAvailableResourceDimensions	aws pi list-available-resource-dimensions	Récupérez les dimensions pouvant être interrogées pour chaque type de métrique spécifié sur une instance spécifiée.
ListAvailableResourceMetrics	aws pi list-available-resource-metrics	Récupérez toutes les métriques disponibles des types de métriques spécifiés pouvant être interrogés pour une instance de base de données spécifiée.
ListPerformanceAnalysisReports	aws pi list-performance-analysis-reports	Récupère tous les rapports d'analyse disponibles pour l'instance de base de données. Les rapports sont répertoriés en fonction de l'heure de début de chaque rapport.

Action Performance Insights	AWS CLI commande	Description
ListTagsForResource	aws pi list-tags-for-resource	Répertorie toutes les balises de métadonnées ajoutées à la ressource. La liste inclut le nom et la valeur de la balise.
TagResource	aws pi tag-resource	Ajoute des balises de métadonnées à la RDS ressource Amazon. La balise inclut un nom et une valeur.
UntagResource	aws pi untag-resource	Supprime la balise de métadonnées de la ressource.

Pour plus d'informations sur la récupération de statistiques de séries chronologiques et pour AWS CLI des exemples pour Performance Insights, consultez les rubriques suivantes.

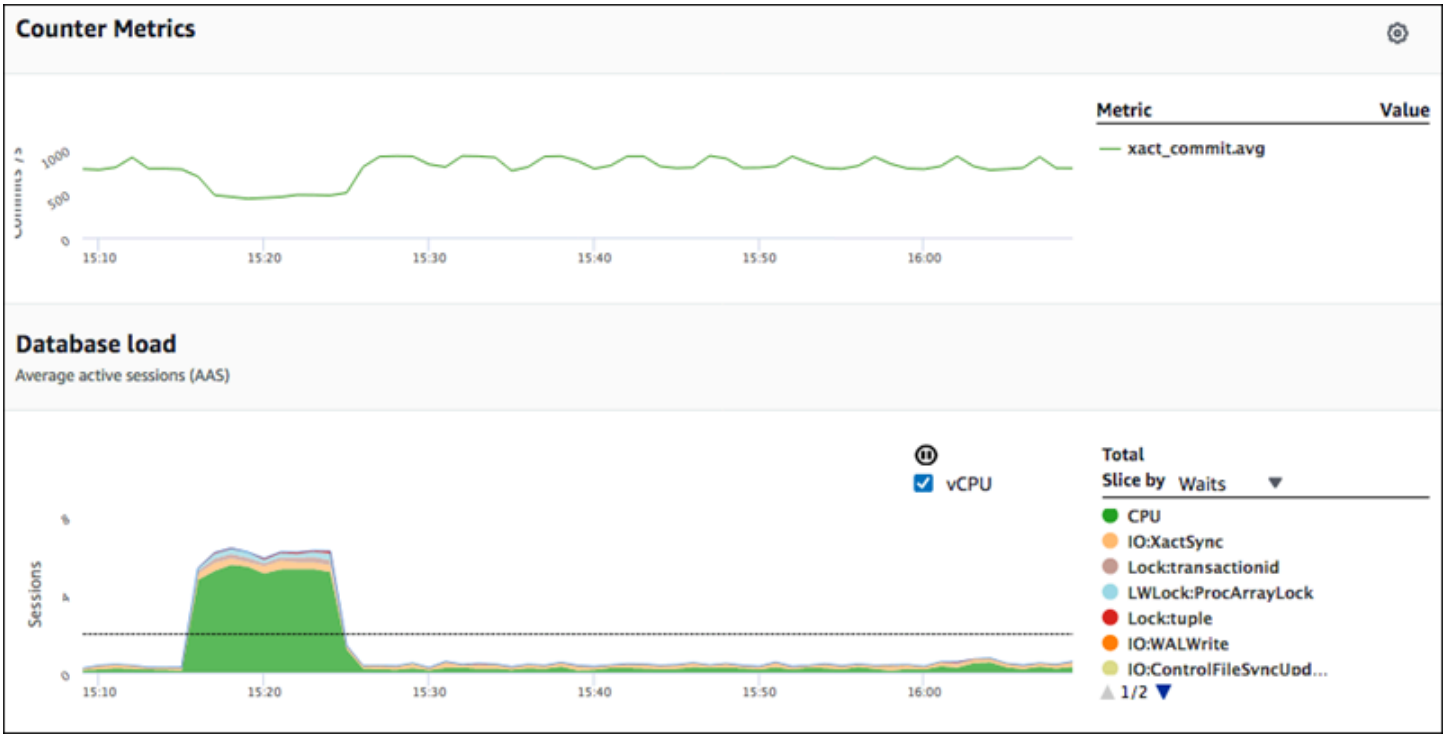
Rubriques

- Récupération de statistiques de séries chronologiques pour Performance Insights
- AWS CLI exemples de Performance Insights

Récupération de statistiques de séries chronologiques pour Performance Insights

L'opération `GetResourceMetrics` récupère une ou plusieurs métriques de série chronologique à partir des données de Performance Insights. `GetResourceMetrics` exige une métrique et une période, et renvoie une réponse contenant la liste des points de données.

Par exemple, les AWS Management Console utilisations `GetResourceMetrics` pour remplir le graphique Counter Metrics et le graphique de charge de la base de données, comme indiqué dans l'image suivante.



Toutes les métriques renvoyées par `GetResourceMetrics` sont des métriques de série chronologique standard, à l'exception de `db.load`. Elle apparaît dans le graphique Database Load (Charge de base de données). La métrique `db.load` est différente des autres métriques de série chronologique, car vous pouvez la décomposer en sous-composants appelés dimensions. Dans l'image précédente, `db.load` est décomposé et regroupé en fonction des états d'attente qui constituent `db.load`.

> **Note**
>
> `GetResourceMetrics` peut également renvoyer la métrique `db.sampleload`, mais la métrique `db.load` est appropriée dans la plupart des cas.

Pour de plus amples informations sur les métriques de compteur renvoyées par `GetResourceMetrics`, veuillez consulter Métrique de compteur de Performance Insights.

Les calculs suivants sont pris en charge pour les métriques :

- Moyenne – Moyenne de la métrique sur une période. Ajoutez `.avg` au nom de la métrique.
- Minimum – Valeur minimale de la métrique sur une période. Ajoutez `.min` au nom de la métrique.
- Maximum – Valeur maximale de la métrique sur une période. Ajoutez `.max` au nom de la métrique.

- Somme – Somme des valeurs de la métrique sur une période. Ajoutez `.sum` au nom de la métrique.
- Nombre échantillon – Nombre de fois où la métrique a été collectée sur une période. Ajoutez `.sample_count` au nom de la métrique.

Par exemple, supposons qu'une métrique soit collectée pendant 300 secondes (5 minutes) et qu'elle soit collectée une fois toutes les minutes. Les valeurs pour chaque minute sont 1, 2, 3, 4 et 5. Dans ce cas, les calculs suivants sont renvoyés :

- Moyenne – 3
- Minimum – 1
- Maximum – 5
- Somme – 15
- Nombre échantillon – 5

Pour plus d'informations sur l'utilisation de la `get-resource-metrics` AWS CLI commande, consultez `get-resource-metrics`.

Pour l'option `--metric-queries`, spécifiez une ou plusieurs requêtes pour lesquelles vous souhaitez obtenir les résultats. Chaque requête se compose d'un paramètre `Metric` obligatoire et des paramètres `GroupBy` et `Filter` facultatifs. Voici un exemple de spécification de l'option `--metric-queries`.

```
{
   "Metric": "string",
   "GroupBy": {
     "Group": "string",
     "Dimensions": ["string", ...],
     "Limit": integer
   },
   "Filter": {"string": "string"
     ...}
```

AWS CLI exemples de Performance Insights

Dans les sections suivantes, découvrez le AWS Command Line Interface (AWS CLI) pour Performance Insights et des AWS CLI exemples d'utilisation.

Rubriques

- Aide intégrée AWS CLI pour Performance Insights
- Récupération de métriques de compteur
- Récupération de la charge de base de données moyenne pour les principaux événements d'attente
- Récupération de la charge moyenne de la base de données pour le haut SQL
- Récupération de la charge moyenne de base de données filtrée par SQL
- Récupération du texte intégral d'une déclaration SQL
- Création d'un rapport d'analyse des performances pour une période donnée
- Récupération d'un rapport d'analyse des performances
- Établissement de la liste de tous les rapports d'analyse des performances pour l'instance de base de données
- Suppression d'un rapport d'analyse des performances
- Ajout d'une balise à un rapport d'analyse des performances
- Établissement de la liste de toutes les balises pour un rapport d'analyse des performances
- Suppression des balises d'un rapport d'analyse des performances

Aide intégrée AWS CLI pour Performance Insights

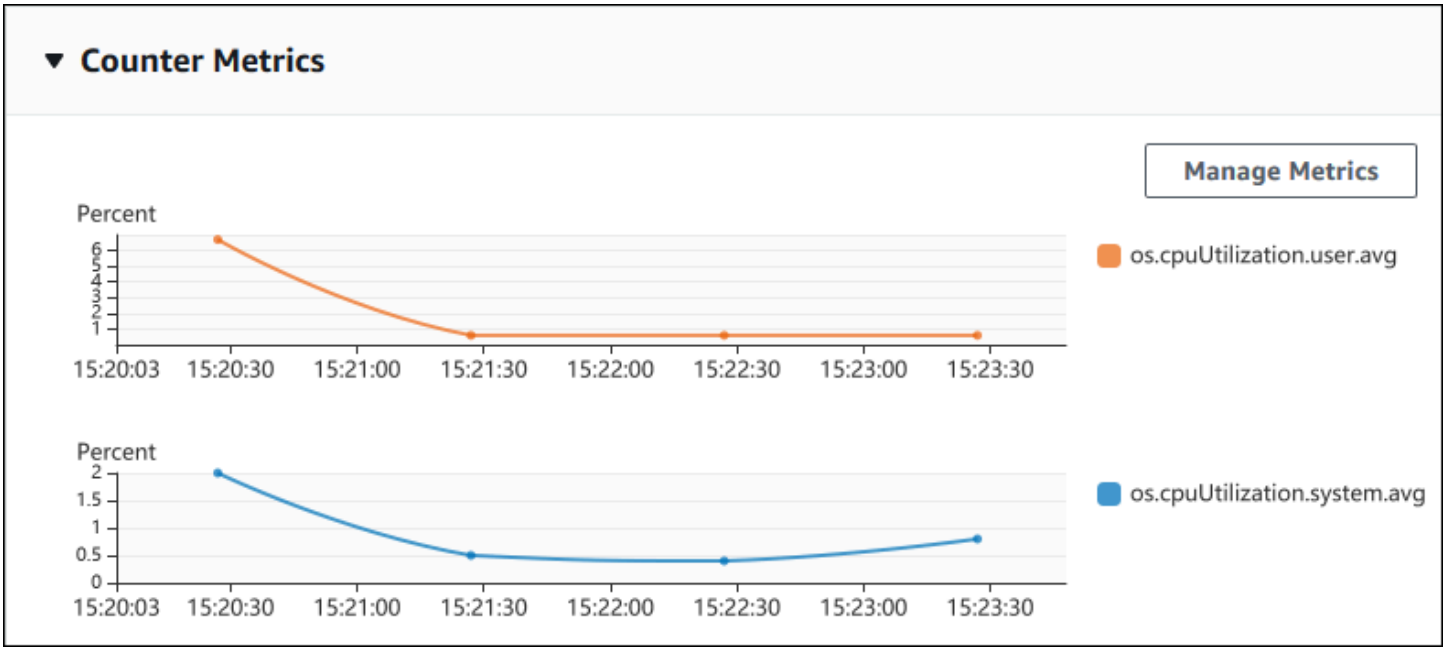
Vous pouvez consulter les données de Performance Insights à l'aide d AWS CLI. Vous pouvez consulter l'aide relative aux AWS CLI commandes de Performance Insights en saisissant ce qui suit sur la ligne de commande.

```
aws pi help
```

Si ce n'est pas le AWS CLI cas, reportez-vous à la section Installation du AWS CLI guide de l'AWS CLI utilisateur pour plus d'informations sur son installation.

Récupération de métriques de compteur

L'image suivante illustre deux graphiques de métriques de compteur dans AWS Management Console.



L'exemple suivant montre comment collecter les mêmes données que celles AWS Management Console utilisées pour générer les deux graphiques contre-métriques.

Dans Linux, macOS, ou Unix:

```
aws pi get-resource-metrics \
   --service-type RDS \
   --identifier db-ID \
   --start-time 2018-10-30T00:00:00Z \
   --end-time   2018-10-30T01:00:00Z \
   --period-in-seconds 60 \
   --metric-queries '[{"Metric": "os.cpuUtilization.user.avg"  },
                      {"Metric": "os.cpuUtilization.idle.avg"}]'
```

Dans Windows:

```
aws pi get-resource-metrics ^
   --service-type RDS ^
   --identifier db-ID ^
   --start-time 2018-10-30T00:00:00Z ^
   --end-time   2018-10-30T01:00:00Z ^
   --period-in-seconds 60 ^
   --metric-queries '[{"Metric": "os.cpuUtilization.user.avg"  },
                      {"Metric": "os.cpuUtilization.idle.avg"}]'
```

Vous pouvez également simplifier la lecture d'une commande en spécifiant un fichier pour l'option `--metrics-query`. L'exemple suivant utilise un fichier nommé query.json pour l'option. Le contenu du fichier est le suivant.

```
[
    {
        "Metric": "os.cpuUtilization.user.avg"
    },
    {
        "Metric": "os.cpuUtilization.idle.avg"
    }
]
```

Exécutez la commande suivante pour utiliser le fichier.

Dans Linux, macOS, ou Unix:

```
aws pi get-resource-metrics \
   --service-type RDS \
   --identifier db-ID \
   --start-time 2018-10-30T00:00:00Z \
   --end-time   2018-10-30T01:00:00Z \
   --period-in-seconds 60 \
   --metric-queries file://query.json
```

Dans Windows:

```
aws pi get-resource-metrics ^
   --service-type RDS ^
   --identifier db-ID ^
   --start-time 2018-10-30T00:00:00Z ^
   --end-time   2018-10-30T01:00:00Z ^
   --period-in-seconds 60 ^
   --metric-queries file://query.json
```

L'exemple précédent spécifie les valeurs suivantes pour les options :

- `--service-type`— RDS pour Amazon RDS
- `--identifier` – ID de ressource de l'instance de base de données
- `--start-time`et `--end-time` — Les ISO 8601 `DateTime` valeurs pour la période à interroger, avec plusieurs formats pris en charge

L'interrogation se déroule pendant un intervalle d'une heure :

- `--period-in-seconds` – `60` pour une requête toutes les minutes
- `--metric-queries` – Tableau de deux requêtes s'appliquant chacune à une métrique.

  Le nom de la métrique utilise des points pour classifier la métrique dans une catégorie utile, l'élément final étant une fonction. Dans l'exemple, la fonction est `avg` pour chaque requête. Comme pour Amazon CloudWatch, les fonctions prises en charge sont `min`,`max`,`total`, et`avg`.

La réponse ressemble à ce qui suit.

```
{
    "Identifier": "db-XXX",
    "AlignedStartTime": 1540857600.0,
    "AlignedEndTime": 1540861200.0,
    "MetricList": [
        { //A list of key/datapoints
            "Key": {
                "Metric": "os.cpuUtilization.user.avg" //Metric1
            },
            "DataPoints": [
                //Each list of datapoints has the same timestamps and same number of
 items
                {
                    "Timestamp": 1540857660.0, //Minute1
                    "Value": 4.0
                },
                {
                    "Timestamp": 1540857720.0, //Minute2
                    "Value": 4.0
                },
                {
                    "Timestamp": 1540857780.0, //Minute 3
                    "Value": 10.0
                }
                //... 60 datapoints for the os.cpuUtilization.user.avg metric
            ]
        },
        {
            "Key": {
                "Metric": "os.cpuUtilization.idle.avg" //Metric2
            },
```

```
            "DataPoints": [
                {
                    "Timestamp": 1540857660.0, //Minute1
                    "Value": 12.0
                },
                {
                    "Timestamp": 1540857720.0, //Minute2
                    "Value": 13.5
                },
                //... 60 datapoints for the os.cpuUtilization.idle.avg metric
            ]
        }
    ] //end of MetricList
} //end of response
```

La réponse contient les éléments `Identifier`, `AlignedStartTime` et `AlignedEndTime`. Étant donné que la valeur de `--period-in-seconds` était définie sur `60`, les heures de début et de fin ont été arrondies à la minute près. Si `--period-in-seconds` était défini sur `3600`, les heures de début et de fin auraient été arrondies à l'heure près.

L'élément `MetricList` dans la réponse comporte un certain nombre d'entrées, chacune associée à une entrée `Key` et `DataPoints`. Chaque élément `DataPoint` comporte une entrée `Timestamp` et `Value`. Chaque liste `Datapoints` répertorie 60 points de données, car les requêtes sont exécutées toutes les minutes pendant une heure, avec `Timestamp1/Minute1`, `Timestamp2/Minute2`, etc. jusqu'à `Timestamp60/Minute60`.

Étant donné que la requête s'applique à deux métriques de compteur différentes, contient deux élément `MetricList`.

Récupération de la charge de base de données moyenne pour les principaux événements d'attente

L'exemple suivant est la même requête que celle AWS Management Console utilisée pour générer un graphique linéaire à aires empilées. Il récupère la valeur de `db.load.avg` sur la dernière heure en divisant la charge conformément aux sept principaux événements d'attente. La commande est identique à la commande de la rubrique Récupération de métriques de compteur. Le contenu du fichier query.json est cependant différent :

```
[
    {
        "Metric": "db.load.avg",
        "GroupBy": { "Group": "db.wait_event", "Limit": 7 }
```

```
    }
]
```

Exécutez la commande suivante.

Dans Linux, macOS, ou Unix:

```
aws pi get-resource-metrics \
   --service-type RDS \
   --identifier db-ID \
   --start-time 2018-10-30T00:00:00Z \
   --end-time   2018-10-30T01:00:00Z \
   --period-in-seconds 60 \
   --metric-queries file://query.json
```

Dans Windows:

```
aws pi get-resource-metrics ^
   --service-type RDS ^
   --identifier db-ID ^
   --start-time 2018-10-30T00:00:00Z ^
   --end-time   2018-10-30T01:00:00Z ^
   --period-in-seconds 60 ^
   --metric-queries file://query.json
```

L'exemple spécifie la métrique de `db.load.avg` et exécute une action `GroupBy` pour les sept principaux événements d'attente. Pour plus de détails sur les valeurs valides pour cet exemple, reportez-vous DimensionGroupà la section Performance Insights API Reference.

La réponse ressemble à ce qui suit.

```
{
    "Identifier": "db-XXX",
    "AlignedStartTime": 1540857600.0,
    "AlignedEndTime": 1540861200.0,
    "MetricList": [
        { //A list of key/datapoints
            "Key": {
                //A Metric with no dimensions. This is the total db.load.avg
                "Metric": "db.load.avg"
            },
            "DataPoints": [
```

```
                //Each list of datapoints has the same timestamps and same number of
 items
                {
                    "Timestamp": 1540857660.0, //Minute1
                    "Value": 0.5166666666666667
                },
                {
                    "Timestamp": 1540857720.0, //Minute2
                    "Value": 0.38333333333333336
                },
                {
                    "Timestamp": 1540857780.0, //Minute 3
                    "Value": 0.26666666666666666
                }
                //... 60 datapoints for the total db.load.avg key
            ]
        },
        {
            "Key": {
                //Another key. This is db.load.avg broken down by CPU
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.name": "CPU",
                    "db.wait_event.type": "CPU"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1540857660.0, //Minute1
                    "Value": 0.35
                },
                {
                    "Timestamp": 1540857720.0, //Minute2
                    "Value": 0.15
                },
                //... 60 datapoints for the CPU key
            ]
        },
        //... In total we have 8 key/datapoints entries, 1) total, 2-8) Top Wait Events
    ] //end of MetricList
} //end of response
```

Dans cette réponse, comporte huit entrée `MetricList`. Une entrée s'applique à la valeur totale de `db.load.avg` et les sept autres entrées s'appliquent à chacune des valeurs de `db.load.avg` divisées conformément à l'un des sept principaux événements d'attente. Contrairement au premier exemple qui comportait une dimension de regroupement, cet exemple doit définir un élément Key pour chaque regroupement de la métrique. Un seul élément Key peut être associé à chaque métrique, comme dans le cas d'utilisation de la métrique de compteur de base.

Récupération de la charge moyenne de la base de données pour le haut SQL

L'exemple suivant regroupe `db.wait_events` les 10 premières SQL déclarations. Il existe deux groupes différents pour les SQL déclarations :

- `db.sql`— La SQL déclaration complète, telle que `select * from customers where customer_id = 123`
- `db.sql_tokenized`— La SQL déclaration tokenisée, telle que `select * from customers where customer_id = ?`

Lors de l'analyse des performances d'une base de données, il peut être utile de considérer les SQL instructions dont les paramètres ne diffèrent que comme un élément logique. Vous pouvez donc utiliser `db.sql_tokenized` lors de l'interrogation. Toutefois, en particulier lorsque vous souhaitez expliquer des plans, il est parfois plus utile d'examiner des SQL instructions complètes avec des paramètres et de les regrouper par requêtes`db.sql`. Il existe une relation parent-enfant entre tokenisé et fullSQL, plusieurs full SQL (enfants) étant regroupés sous le même SQL tokenisé (parent).

La commande illustrée dans cet exemple est identique à la commande de la rubrique Récupération de la charge de base de données moyenne pour les principaux événements d'attente. Le contenu du fichier query.json est cependant différent :

```
[
    {
        "Metric": "db.load.avg",
        "GroupBy": { "Group": "db.sql_tokenized", "Limit": 10 }
    }
]
```

L'exemple suivant utilise `db.sql_tokenized`.

Dans Linux, macOS, ou Unix:

```
aws pi get-resource-metrics \
   --service-type RDS \
   --identifier db-ID \
   --start-time 2018-10-29T00:00:00Z \
   --end-time   2018-10-30T00:00:00Z \
   --period-in-seconds 3600 \
   --metric-queries file://query.json
```

Dans Windows:

```
aws pi get-resource-metrics ^
   --service-type RDS ^
   --identifier db-ID ^
   --start-time 2018-10-29T00:00:00Z ^
   --end-time   2018-10-30T00:00:00Z  ^
   --period-in-seconds 3600 ^
   --metric-queries file://query.json
```

Cet exemple demande plus de 24 heures, avec une heure period-in-seconds.

L'exemple spécifie la métrique de `db.load.avg` et exécute une action `GroupBy` pour les sept principaux événements d'attente. Pour plus de détails sur les valeurs valides pour cet exemple, reportez-vous DimensionGroupà la section Performance Insights API Reference.

La réponse ressemble à ce qui suit.

```
{
    "AlignedStartTime": 1540771200.0,
    "AlignedEndTime": 1540857600.0,
    "Identifier": "db-XXX",

    "MetricList": [ //11 entries in the MetricList
        {
            "Key": { //First key is total
                "Metric": "db.load.avg"
            }
            "DataPoints": [ //Each DataPoints list has 24 per-hour Timestamps and a
 value
                {
                    "Value": 1.6964980544747081,
                    "Timestamp": 1540774800.0
```

```
                },
                //... 24 datapoints
            ]
        },
        {
            "Key": { //Next key is the top tokenized SQL
                "Dimensions": {
                    "db.sql_tokenized.statement": "INSERT INTO authors (id,name,email)
 VALUES\n( nextval(?)  ,?,?)",
                    "db.sql_tokenized.db_id": "pi-2372568224",
                    "db.sql_tokenized.id": "AKIAIOSFODNN7EXAMPLE"
                },
                "Metric": "db.load.avg"
            },
            "DataPoints": [ //... 24 datapoints
            ]
        },
        // In total 11 entries, 10 Keys of top tokenized SQL, 1 total key
    ] //End of MetricList
} //End of response
```

Cette réponse comporte 11 entrées `MetricList` (1 au total, 10 entrées les plus tokenisées SQL), chaque entrée en comptant 24 par heure. `DataPoints`

Pour tokenized SQL, chaque liste de dimensions comporte trois entrées :

- `db.sql_tokenized.statement` — La déclaration tokenisée. SQL
- `db.sql_tokenized.db_id` — Soit l'ID de base de données natif utilisé pour faire référence à la SQL, soit un identifiant synthétique généré par Performance Insights pour vous si l'ID de base de données natif n'est pas disponible. Cet exemple renvoie l'ID synthétique `pi-2372568224`.
- `db.sql_tokenized.id` – ID de la requête dans Performance Insights.

  Dans le AWS Management Console, cet ID est appelé Support ID. Il est nommé ainsi parce que l'ID est une donnée que le AWS Support peut examiner pour vous aider à résoudre un problème lié à votre base de données. AWS prend très au sérieux la sécurité et la confidentialité de vos données, et presque toutes les données sont stockées cryptées avec votre AWS KMS clé. Par conséquent, personne à l'intérieur ne AWS peut consulter ces données. Dans l'exemple précédent, `tokenized.statement` et `tokenized.db_id` sont tous les deux stockés sous forme chiffrée. Si vous rencontrez un problème avec votre base de données, le AWS Support peut vous aider en faisant référence à l'ID de support.

Lors de l'interrogation, il peut s'avérer utile de spécifier une entrée `Group` dans `GroupBy`. Toutefois, pour contrôler les données renvoyées de manière plus précise, spécifier la liste des dimensions. Par exemple, si `db.sql_tokenized.statement` est le seul élément nécessaire, un attribut `Dimensions` peut être ajouté au fichier query.json.

```
[
    {
        "Metric": "db.load.avg",
        "GroupBy": {
            "Group": "db.sql_tokenized",
            "Dimensions":["db.sql_tokenized.statement"],
            "Limit": 10
        }
    }
]
```

Récupération de la charge moyenne de base de données filtrée par SQL



L'image précédente indique qu'une requête particulière est sélectionnée et que le graphique en aires empilées Average active sessions (Sessions actives en moyenne) qui apparaît dans la section supérieure s'y applique. Bien que la requête concerne toujours les sept principaux événements

d'attente globaux, la valeur de la réponse est filtrée. Le filtre permet à la requête de prendre uniquement en compte les sessions qui correspondent à un filtre en particulier.

La API requête correspondante dans cet exemple est similaire à la commande dansRécupération de la charge moyenne de la base de données pour le haut SQL. Le contenu du fichier query.json est cependant différent :

```
[
 {
        "Metric": "db.load.avg",
        "GroupBy": { "Group": "db.wait_event", "Limit": 5  },
        "Filter": { "db.sql_tokenized.id": "AKIAIOSFODNN7EXAMPLE" }
    }
]
```

Dans Linux, macOS, ou Unix:

```
aws pi get-resource-metrics \
   --service-type RDS \
   --identifier db-ID \
   --start-time 2018-10-30T00:00:00Z \
   --end-time   2018-10-30T01:00:00Z \
   --period-in-seconds 60 \
   --metric-queries file://query.json
```

Dans Windows:

```
aws pi get-resource-metrics ^
   --service-type RDS ^
   --identifier db-ID ^
   --start-time 2018-10-30T00:00:00Z ^
   --end-time   2018-10-30T01:00:00Z ^
   --period-in-seconds 60 ^
   --metric-queries file://query.json
```

La réponse ressemble à ce qui suit.

```
{
    "Identifier": "db-XXX",
    "AlignedStartTime": 1556215200.0,
    "MetricList": [
        {
```

```
            "Key": {
                "Metric": "db.load.avg"
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
                    "Value": 1.4878117913832196
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 1.192823803967328
                }
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.type": "io",
                    "db.wait_event.name": "wait/io/aurora_redo_log_flush"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
                    "Value": 1.1360544217687074
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 1.058051341890315
                }
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.type": "io",
                    "db.wait_event.name": "wait/io/table/sql/handler"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
```

```
                    "Value": 0.16241496598639457
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 0.05163360560093349
                }
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.type": "synch",
                    "db.wait_event.name": "wait/synch/mutex/innodb/
aurora_lock_thread_slot_futex"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
                    "Value": 0.11479591836734694
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 0.013127187864644107
                }
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.type": "CPU",
                    "db.wait_event.name": "CPU"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
                    "Value": 0.05215419501133787
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 0.05805134189031505
```

```
                }
            ]
        },
        {
            "Key": {
                "Metric": "db.load.avg",
                "Dimensions": {
                    "db.wait_event.type": "synch",
                    "db.wait_event.name": "wait/synch/mutex/innodb/lock_wait_mutex"
                }
            },
            "DataPoints": [
                {
                    "Timestamp": 1556218800.0,
                    "Value": 0.017573696145124718
                },
                {
                    "Timestamp": 1556222400.0,
                    "Value": 0.002333722287047841
                }
            ]
        }
    ],
    "AlignedEndTime": 1556222400.0
} //end of response
```

Dans cette réponse, toutes les valeurs sont filtrées en fonction de la contribution de tokenized SQL AKIAIOSFODNN7EXAMPLE spécifiée dans le fichier query.json. Les clés peuvent également suivre un ordre différent de celui d'une requête sans filtre, car ce sont les cinq principaux événements d'attente qui ont affecté le filtreSQL.

Récupération du texte intégral d'une déclaration SQL

L'exemple suivant récupère le texte intégral d'une SQL instruction pour une instance `db-10BCD2EFGHIJ3KL4M5NO6PQRS5` de base de données. Le `--group` est `db.sql` et l'`--group-identifier` est `db.sql.id`. Dans cet exemple, *`my-sql-id`* représente un SQL identifiant récupéré en invoquant `pi get-resource-metrics` oupi `describe-dimension-keys`.

Exécutez la commande suivante.

Dans Linux, macOS, ou Unix:

```
aws pi get-dimension-key-details \
```

```
    --service-type RDS \
    --identifier db-10BCD2EFGHIJ3KL4M5NO6PQRS5 \
    --group db.sql \
    --group-identifier my-sql-id \
    --requested-dimensions statement
```

Dans Windows:

```
aws pi get-dimension-key-details ^
    --service-type RDS ^
    --identifier db-10BCD2EFGHIJ3KL4M5NO6PQRS5 ^
    --group db.sql ^
    --group-identifier my-sql-id ^
    --requested-dimensions statement
```

Dans cet exemple, les détails des dimensions sont disponibles. Ainsi, Performance Insights récupère le texte intégral de la SQL déclaration, sans la tronquer.

```
{
    "Dimensions":[
    {
        "Value": "SELECT e.last_name, d.department_name FROM employees e, departments d
 WHERE e.department_id=d.department_id",
        "Dimension": "db.sql.statement",
        "Status": "AVAILABLE"
    },
    ...
    ]
}
```

Création d'un rapport d'analyse des performances pour une période donnée

L'exemple suivant crée un rapport d'analyse des performances avec l'heure de début `1682969503` et l'heure de fin `1682979503` pour la base de données `db-loadtest-0`.

```
aws pi create-performance-analysis-report \
        --service-type RDS \
        --identifier db-loadtest-0 \
        --start-time 1682969503 \
        --end-time 1682979503 \
        --region us-west-2
```

La réponse est l'identifiant unique `report-0234d3ed98e28fb17` du rapport.

```
{
   "AnalysisReportId": "report-0234d3ed98e28fb17"
}
```

Récupération d'un rapport d'analyse des performances

L'exemple suivant extrait les détails du rapport d'analyse pour le rapport `report-0d99cc91c4422ee61`.

```
aws pi get-performance-analysis-report \
--service-type RDS \
--identifier db-loadtest-0 \
--analysis-report-id report-0d99cc91c4422ee61 \
--region us-west-2
```

La réponse fournit le statut du rapport, l'identifiant, les détails temporels et les informations.

```
        {
    "AnalysisReport": {
        "Status": "Succeeded",
        "ServiceType": "RDS",
        "Identifier": "db-loadtest-0",
        "StartTime": 1680583486.584,
        "AnalysisReportId": "report-0d99cc91c4422ee61",
        "EndTime": 1680587086.584,
        "CreateTime": 1680587087.139,
        "Insights": [
           ... (Condensed for space)
        ]
    }
}
```

Établissement de la liste de tous les rapports d'analyse des performances pour l'instance de base de données

L'exemple suivant répertorie tous les rapports d'analyse des performances disponibles pour la base de données `db-loadtest-0`.

```
aws pi list-performance-analysis-reports \
```

```
--service-type RDS \
--identifier db-loadtest-0 \
--region us-west-2
```

La réponse répertorie tous les rapports avec l'ID du rapport, le statut et les détails temporels de la période.


```
{
    "AnalysisReports": [
        {
            "Status": "Succeeded",
            "EndTime": 1680587086.584,
            "CreationTime": 1680587087.139,
            "StartTime": 1680583486.584,
            "AnalysisReportId": "report-0d99cc91c4422ee61"
        },
        {
            "Status": "Succeeded",
            "EndTime": 1681491137.914,
            "CreationTime": 1681491145.973,
            "StartTime": 1681487537.914,
            "AnalysisReportId": "report-002633115cc002233"
        },
        {
            "Status": "Succeeded",
            "EndTime": 1681493499.849,
            "CreationTime": 1681493507.762,
            "StartTime": 1681489899.849,
            "AnalysisReportId": "report-043b1e006b47246f9"
        },
        {
            "Status": "InProgress",
            "EndTime": 1682979503.0,
            "CreationTime": 1682979618.994,
            "StartTime": 1682969503.0,
            "AnalysisReportId": "report-01ad15f9b88bcbd56"
        }
    ]
}
```


Suppression d'un rapport d'analyse des performances

L'exemple suivant supprime le rapport d'analyse pour la base de données `db-loadtest-0`.

```
aws pi delete-performance-analysis-report \
--service-type RDS \
--identifier db-loadtest-0 \
--analysis-report-id report-0d99cc91c4422ee61 \
--region us-west-2
```

Ajout d'une balise à un rapport d'analyse des performances

L'exemple suivant ajoute une balise avec une clé `name` et une valeur `test-tag` au rapport `report-01ad15f9b88bcbd56`.

```
aws pi tag-resource \
--service-type RDS \
--resource-arn arn:aws:pi:us-west-2:356798100956:perf-reports/RDS/db-loadtest-0/
report-01ad15f9b88bcbd56 \
--tags Key=name,Value=test-tag \
--region us-west-2
```

Établissement de la liste de toutes les balises pour un rapport d'analyse des performances

L'exemple suivant répertorie toutes les balises pour le rapport `report-01ad15f9b88bcbd56`.

```
aws pi list-tags-for-resource \
--service-type RDS \
--resource-arn arn:aws:pi:us-west-2:356798100956:perf-reports/RDS/db-loadtest-0/
report-01ad15f9b88bcbd56 \
--region us-west-2
```

La réponse répertorie la valeur et la clé de toutes les balises ajoutées au rapport :

```
{
    "Tags": [
        {
            "Value": "test-tag",
            "Key": "name"
        }
    ]
}
```

Suppression des balises d'un rapport d'analyse des performances

L'exemple suivant montre comment supprimer la balise `name` du rapport `report-01ad15f9b88bcbd56`.

```
aws pi untag-resource \
--service-type RDS \
--resource-arn arn:aws:pi:us-west-2:356798100956:perf-reports/RDS/db-loadtest-0/
report-01ad15f9b88bcbd56 \
--tag-keys name \
--region us-west-2
```

Une fois la balise supprimée, appeler le `list-tags-for-resource` API ne permet pas de répertorier cette balise.

Journalisation des appels Performance Insights à l'aide de AWS CloudTrail

Performance Insights fonctionne avec AWS CloudTrail un service qui fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans Performance Insights. CloudTrailcapture tous les API appels à Performance Insights sous forme d'événements. Cette capture inclut les appels depuis la RDS console Amazon et les appels de code vers les API opérations Performance Insights.

Si vous créez un suivi, vous pouvez activer la diffusion continue d' CloudTrail événements vers un compartiment Amazon S3, y compris des événements pour Performance Insights. Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents dans la CloudTrail console dans Historique des événements. À l'aide des données collectées par CloudTrail, vous pouvez déterminer certaines informations. Ces informations incluent la demande qui a été envoyée à Performance Insights, l'adresse IP à partir de laquelle la demande a été effectuée, l'auteur de la demande et sa date. Elles comprennent également des détails supplémentaires.

Pour en savoir plus CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

Utilisation des informations Performance Insights dans CloudTrail

CloudTrail est activé sur votre AWS compte lorsque vous le créez. Lorsqu'une activité se produit dans Performance Insights, cette activité est enregistrée dans un CloudTrail événement avec d'autres événements de AWS service dans la CloudTrail console dans l'historique des événements. Vous pouvez consulter, rechercher et télécharger les événements récents dans votre AWS compte. Pour

plus d'informations, consultez la section Affichage des événements avec l'historique des CloudTrail événements dans le guide de AWS CloudTrail l'utilisateur.

Pour un enregistrement continu des événements de votre AWS compte, y compris des événements pour Performance Insights, créez un historique. Un suivi permet CloudTrail de fournir des fichiers journaux à un compartiment Amazon S3. Par défaut, lorsque vous créez un parcours dans la console, celui-ci s'applique à toutes les AWS régions. Le journal enregistre les événements de toutes les AWS régions de la AWS partition et transmet les fichiers journaux au compartiment Amazon S3 que vous spécifiez. En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour plus d'informations, consultez les rubriques suivantes dans le Guide de l'utilisateur AWS CloudTrail :

- Vue d'ensemble de la création d'un journal d'activité
- CloudTrail Services et intégrations pris en charge
- Configuration des SNS notifications Amazon pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et réception de fichiers CloudTrail journaux de plusieurs comptes

Toutes les opérations de Performance Insights sont enregistrées CloudTrail et documentées dans la APIréférence Performance Insights. Par exemple, les appels aux `GetResourceMetrics` opérations `DescribeDimensionKeys` et génèrent des entrées dans les fichiers CloudTrail journaux.

Chaque événement ou entrée de journal contient des informations sur la personne ayant initié la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou IAM.
- Si la demande a été effectuée avec les informations d'identification de sécurité temporaires d'un rôle ou d'un utilisateur fédéré.
- Si la demande a été faite par un autre AWS service.

Pour plus d'informations, consultez l'CloudTrail userIdentityélément.

Entrées du fichier journal Performance Insights

Un suivi est une configuration qui permet de transmettre des événements sous forme de fichiers journaux à un compartiment Amazon S3 que vous spécifiez. CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. Un événement représente une demande individuelle

d’une source quelconque. Chaque événement inclut des informations sur l'opération demandée, la date et l'heure de l'opération, les paramètres de la demande, etc. CloudTrailles fichiers journaux ne constituent pas une trace ordonnée des API appels publics, ils n'apparaissent donc pas dans un ordre spécifique.

L'exemple suivant montre une entrée de CloudTrail journal illustrant l'`GetResourceMetrics`opération.


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
         "principalId": "AKIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/johndoe",
        "accountId": "123456789012",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "userName": "johndoe"
    },
    "eventTime": "2019-12-18T19:28:46Z",
    "eventSource": "pi.amazonaws.com",
    "eventName": "GetResourceMetrics",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "72.21.198.67",
    "userAgent": "aws-cli/1.16.240 Python/3.7.4 Darwin/18.7.0 botocore/1.12.230",
    "requestParameters": {
        "identifier": "db-YTDU5J5V66X7CXSCVDFD2V3SZM",
        "metricQueries": [
            {
                "metric": "os.cpuUtilization.user.avg"
            },
            {
                "metric": "os.cpuUtilization.idle.avg"
            }
        ],
        "startTime": "Dec 18, 2019 5:28:46 PM",
        "periodInSeconds": 60,
        "endTime": "Dec 18, 2019 7:28:46 PM",
        "serviceType": "RDS"
    },
    "responseElements": null,
    "requestID": "9ffbe15c-96b5-4fe6-bed9-9fccff1a0525",
    "eventID": "08908de0-2431-4e2e-ba7b-f5424f908433",
    "eventType": "AwsApiCall",
```

```
    "recipientAccountId": "123456789012"
}
```

Performance Insights API et VPC points de terminaison d'interface ()AWS PrivateLink

Vous pouvez l'utiliser AWS PrivateLink pour créer une connexion privée entre vous VPC et Amazon RDS Performance Insights. Vous pouvez accéder à Performance Insights comme s'il se trouvait dans votre ordinateurVPC, sans passer par une passerelle Internet, un NAT appareil, une VPN connexion ou une AWS Direct Connect connexion. Les instances de votre VPC ordinateur n'ont pas besoin d'adresses IP publiques pour accéder à Performance Insights.

Vous établissez cette connexion privée en créant un point de terminaison d'interface optimisé par AWS PrivateLink. Nous créons une interface réseau de point de terminaison dans chaque sous-réseau que vous activez pour le point de terminaison d'interface. Il s'agit d'interfaces réseau gérées par les demandeurs qui servent de point d'entrée au trafic destiné à Performance Insights.

Pour plus d'informations, consultez la section Accès Services AWS par AWS PrivateLink le biais du AWS PrivateLink guide.

Considérations relatives à Performance Insights

Avant de configurer un point de terminaison d'interface pour Performance Insights, consultez les considérations du AWS PrivateLink guide.

Performance Insights permet d'appeler toutes ses API actions via le point de terminaison de l'interface.

Par défaut, l'accès complet à Performance Insights est autorisé via le point de terminaison de l'interface. Pour contrôler le trafic vers Performance Insights via le point de terminaison de l'interface, associez un groupe de sécurité aux interfaces réseau du point de terminaison.

Disponibilité

Performance Insights prend API actuellement en charge VPC les terminaux compatibles Régions AWS avec Performance Insights. Pour plus d'informations sur la disponibilité de Performance Insights, consultez Régions et moteurs de base de données pris en charge pour Performance Insights dans Amazon RDS.

Création d'un point de terminaison d'interface pour Performance Insights

Vous pouvez créer un point de terminaison d'interface pour Performance Insights à l'aide de la VPC console Amazon ou du AWS Command Line Interface (AWS CLI). Pour plus d'informations, consultez Création d'un point de terminaison d'interface dans le Guide AWS PrivateLink .

Créez un point de terminaison d'interface pour Performance Insights en utilisant le nom de service suivant :

Si vous activez le mode privé DNS pour le point de terminaison de l'interface, vous pouvez envoyer des API demandes à Performance Insights en utilisant son DNS nom régional par défaut. Par exemple, `pi.us-east-1.amazonaws.com`.

Création d'une politique relative aux VPC terminaux pour Performance Insights API

Une politique de point de terminaison est une IAM ressource que vous pouvez associer à un point de terminaison d'interface. La politique de point de terminaison par défaut permet un accès complet à Performance Insights via le point de terminaison de l'interface. Pour contrôler l'accès autorisé à Performance Insights depuis votre ordinateurVPC, associez une politique de point de terminaison personnalisée au point de terminaison de l'interface.

Une politique de point de terminaison spécifie les informations suivantes :

- Les principaux qui peuvent effectuer des actions (Comptes AWS IAMutilisateurs et IAM rôles).
- Les actions qui peuvent être effectuées.
- La ressource sur laquelle les actions peuvent être effectuées.

Pour plus d'informations, consultez Contrôle de l'accès aux services à l'aide de politiques de point de terminaison dans le Guide AWS PrivateLink .

Exemple : politique relative aux VPC terminaux pour les actions Performance Insights

Voici un exemple de politique de point de terminaison personnalisée. Lorsque vous associez cette politique au point de terminaison de votre interface, elle donne accès aux actions Performance Insights répertoriées à tous les principaux sur toutes les ressources.

```
{
   "Statement":[
      {
         "Principal":"*",
```

```
         "Effect":"Allow",
         "Action":[
            "rds:CreatePerformanceAnalysisReport",
            "rds:DeletePerformanceAnalysisReport",
            "rds:GetPerformanceAnalysisReport"
         ],
         "Resource":"*"
      }
   ]
}
```

Exemple : politique de VPC point de terminaison qui refuse tout accès à partir d'un AWS compte spécifié

La politique de VPC point de terminaison suivante refuse au AWS compte `123456789012` tout accès aux ressources utilisant le point de terminaison. La politique autorise toutes les actions provenant d'autres comptes.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    },
    {
      "Action": "*",
      "Effect": "Deny",
      "Resource": "*",
      "Principal": { "AWS": [ "123456789012" ] }
     }
   ]
}
```

Adressage IP pour Performance Insights

Les adresses IP permettent VPC à vos ressources de communiquer entre elles et avec des ressources via Internet. Performance Insights prend en charge les deux protocoles IPv4 ainsi que les protocoles d'IPv6adressage. Par défaut, Performance Insights et Amazon VPC utilisent le protocole d'IPv4adressage. Vous ne pouvez pas désactiver ce comportement. Lorsque vous créez unVPC, assurez-vous de spécifier un IPv4 CIDR bloc (une plage d'IPv4adresses privées).

Vous pouvez éventuellement attribuer un IPv6 CIDR bloc à vos sous-réseaux VPC et et attribuer les IPv6 adresses de ce bloc aux RDS ressources de votre sous-réseau. Support du IPv6 protocole augmente le nombre d'adresses IP prises en charge. En utilisant le IPv6 protocole, vous vous assurez de disposer de suffisamment d'adresses disponibles pour la future croissance d'Internet. Les RDS ressources nouvelles et existantes peuvent être utilisées IPv4 et les IPv6 adresses au sein de votreVPC. La configuration, la sécurisation et la traduction du trafic réseau entre les deux protocoles utilisés dans les différentes parties d'une application peuvent entraîner une surcharge opérationnelle. Vous pouvez standardiser le IPv6 protocole pour les RDS ressources Amazon afin de simplifier la configuration de votre réseau. Pour plus d'informations sur les points de terminaison de service et les quotas, consultez la section Points de terminaison et quotas Amazon Relational Database Service.

Pour plus d'informations sur l'adressage IP Amazon RDS , consultez IP Amazon.

Analyse des anomalies de performance avec Amazon DevOps Guru pour Amazon RDS

Amazon DevOps Guru est un service d'exploitation entièrement géré qui aide les développeurs et les opérateurs à améliorer les performances et la disponibilité de leurs applications. DevOpsGuru décharge les tâches associées à l'identification des problèmes opérationnels afin que vous puissiez rapidement mettre en œuvre les recommandations visant à améliorer votre application. Pour plus d'informations, consultez Qu'est-ce qu'Amazon DevOps Guru ? dans le guide de l'utilisateur Amazon DevOps Guru.

DevOpsGuru détecte, analyse et formule des recommandations pour les problèmes opérationnels existants pour tous les moteurs de RDS base de données Amazon. DevOpsGuru for RDS étend cette fonctionnalité en appliquant l'apprentissage automatique aux métriques Performance Insights pour les SQL bases de données for Postgre. Ces fonctionnalités de surveillance permettent à DevOps Guru RDS de détecter et de diagnostiquer les problèmes de performance et de recommander des actions correctives spécifiques. DevOpsGuru for RDS peut également détecter les problèmes dans vos pour Postgre SQL avant qu'ils ne se produisent.

Vous pouvez désormais consulter ces recommandations dans RDS la console. Pour de plus amples informations, veuillez consulter Recommandations d'Amazon RDS.

La vidéo suivante est un aperçu de DevOps Guru forRDS.

Pour en savoir plus sur ce sujet, consultez Amazon DevOps Guru pour RDS Under the Hood.

Rubriques

- Les avantages de DevOps Guru pour RDS
- Comment RDS fonctionne DevOps Guru for
- Configuration de DevOps Guru pour RDS

Les avantages de DevOps Guru pour RDS

Si vous êtes responsable pour Postgre, vous ne savez peut-être pas qu'un événement ou une régression affectant cette base de données est en train de se produire. Lorsque vous prenez connaissance de ce problème, vous ne savez pas toujours pourquoi il se produit ni comment y remédier. Plutôt que de vous adresser à un administrateur de base de données (DBA) pour obtenir

de l'aide ou de vous fier à des outils tiers, vous pouvez suivre les recommandations de DevOps Guru pourRDS.

L'analyse détaillée de DevOps Guru vous permet de bénéficier des avantages suivants pour RDS :

Diagnostic rapide

DevOpsGuru pour surveiller et analyser RDS en permanence la télémétrie des bases de données. Performance Insights, Enhanced Monitoring et Amazon CloudWatch collectent des données de télémétrie pour votre instance de de base de données. DevOpsGuru for RDS utilise des techniques statistiques et d'apprentissage automatique pour exploiter ces données et détecter les anomalies. Pour en savoir plus sur les données de télémétrie, consultez les sections , surveillance de la charge de base de données avec Performance Insights sur Amazon RDS et surveillance des métriques du système d'exploitation avec surveillance améliorée dans le guide de RDSl'utilisateur Amazon.

Résolution rapide

Chaque anomalie identifie le problème de performances et suggère des pistes d'enquête ou des actions correctives. Par exemple, DevOps Guru for RDS peut vous recommander d'étudier des événements d'attente spécifiques. Il peut également vous recommander de régler les paramètres de votre groupe d'applications afin de limiter le nombre de connexions à la base de données. Sur la base de ces recommandations, vous pouvez résoudre les problèmes de performances plus rapidement qu'en effectuant un dépannage manuel.

Insights proactifs

DevOpsGuru for RDS utilise les indicateurs de vos ressources pour détecter les comportements potentiellement problématiques avant qu'ils ne s'aggravent. Par exemple, il peut détecter lorsque votre base de données utilise un nombre croissant de tables temporaires sur disque, ce qui peut avoir un impact sur les performances. DevOpsGuru fournit ensuite des recommandations pour vous aider à résoudre les problèmes avant qu'ils ne s'aggravent.

Connaissance approfondie des ingénieurs Amazon et du machine learning

Pour détecter les problèmes de performance et vous aider à résoudre les goulots d'étranglement, DevOps Guru for RDS s'appuie sur l'apprentissage automatique (ML) et des formules mathématiques avancées. Les ingénieurs de base de données Amazon ont contribué au développement du DevOps Guru for RDS findings, qui résume de nombreuses années de gestion de centaines de milliers de bases de données. En s'appuyant sur ces connaissances collectives, DevOps Guru for RDS peut vous enseigner les meilleures pratiques.

Comment RDS fonctionne DevOps Guru for

DevOpsGuru for RDS collecte des données concernant vos SQL bases de données pour Postgre à partir d'Amazon RDS Performance Insights. La métrique la plus importante est`DBLoad`. DevOpsGuru for RDS utilise les indicateurs Performance Insights, les analyse à l'aide de l'apprentissage automatique et publie des informations sur le tableau de bord.

Un aperçu est un ensemble d'anomalies connexes détectées par DevOps Guru.

Dans DevOps Guru forRDS, une anomalie est un schéma qui s'écarte des performances considérées comme normales pour votre base de données for SQL Postgre.

Insights proactifs

Un insight proactif vous permet de connaître un comportement problématique avant qu'il se produise. Il contient des anomalies et contient des recommandations et des mesures connexes pour vous aider à résoudre les problèmes liés à vos SQL bases de données for Postgre avant qu'ils ne s'aggravent. Ces informations sont publiées dans le tableau de bord DevOps Guru.

Par exemple, DevOps Guru peut détecter que votre SQL base de données RDS for Postgre crée de nombreuses tables temporaires sur disque. Si elle n'est pas corrigée, cette tendance peut entraîner des problèmes de performances. Chaque insight proactif inclut des recommandations de comportement correctif et des liens vers des rubriques pertinentes dans Optimisation RDSpour Postgre SQL Aurora Postgre d'Amazon DevOps Guru. Pour plus d'informations, consultez la section Travailler avec des informations dans DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru.

Insights réactifs

Un insight réactif identifie un comportement anormal lorsqu'il se produit. Si DevOps Guru for RDS détecte des problèmes de performances dans vos instances de SQL base de données pour Postgre, il publie un aperçu réactif dans le tableau de bord DevOps Guru. Pour plus d'informations, consultez la section Travailler avec des informations dans DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru.

Anomalies causales

Une anomalie causale est une anomalie de premier niveau au sein d'un insight réactif. Le chargement de la base de données (charge de la base de données) est l'anomalie causale de DevOps Guru forRDS.

Une anomalie mesure l'impact sur les performances en attribuant un niveau de gravité High (Élevé), Medium (Moyen) ou Low (Faible). Pour en savoir plus, consultez la section Concepts clés de DevOps Guru for RDS dans le guide de l'utilisateur Amazon DevOps Guru.

Si DevOps Guru détecte une anomalie actuelle sur votre instance de base de données, vous êtes alerté sur la page Bases de données de la RDS console. La console vous alerte également des anomalies survenues au cours des dernières 24 heures. Pour accéder à la page des anomalies depuis la RDS console, cliquez sur le lien figurant dans le message d'alerte. La RDS console vous alerte également sur la page de votre pour l'SQLinstance de base de données Postgre.

Anomalies contextuelles

Une anomalie contextuelle est un résultat dans la charge de base de données qui est associé à un insight réactif. Chaque anomalie contextuelle décrit un problème de SQL performance spécifique d' pour Postgre qui nécessite une enquête. Par exemple, DevOps Guru for RDS peut vous recommander d'envisager d'augmenter la CPU capacité ou d'étudier les événements d'attente qui contribuent à la charge de la base de données.

> **Important**
>
> Nous vous recommandons de tester toutes les modifications apportées à une instance test avant de modifier une instance de production. Vous pouvez ainsi mieux appréhender l'impact d'une modification.

Pour en savoir plus, consultez la section Analyse des anomalies dans Amazon RDS dans le guide de l'utilisateur Amazon DevOps Guru.

Configuration de DevOps Guru pour RDS

Pour permettre à DevOps Guru for Amazon RDS de publier des informations pour et Postgre, effectuez les tâches suivantes. RDS

Rubriques

- Configuration des politiques IAM d'accès pour DevOps Guru for RDS
- Activer Performance Insights pour vos instances de SQL base de données pour Postgre
- Activer DevOps Guru et spécifier la couverture des ressources

Configuration des politiques IAM d'accès pour DevOps Guru for RDS

Pour consulter les alertes de DevOps Guru dans la RDS console, votre AWS Identity and Access Management (IAM) l'utilisateur ou le rôle doit respecter l'une des politiques suivantes :

- Le AWS politique gérée `AmazonDevOpsGuruConsoleFullAccess`
- Le AWS stratégie gérée `AmazonDevOpsGuruConsoleReadOnlyAccess` et l'une des politiques suivantes :
  - Le AWS politique gérée `AmazonRDSFullAccess`
  - Politique gérée par le client qui inclut `pi:GetResourceMetrics` et `pi:DescribeDimensionKeys`

Pour de plus amples informations, veuillez consulter Configuration des politiques d'accès pour Performance Insights.

Activer Performance Insights pour vos instances de SQL base de données pour Postgre

DevOpsGuru for RDS s'appuie sur Performance Insights pour ses données. Sans Performance Insights, DevOps Guru publie des anomalies, mais n'inclut pas l'analyse détaillée ni les recommandations.

Lorsque vous créez ou modifiez une SQL instance de base de données RDS pour Postgre, vous pouvez activer Performance Insights. Pour de plus amples informations, veuillez consulter Activer et désactiver Performance Insights pour Amazon RDS.

Activer DevOps Guru et spécifier la couverture des ressources

Vous pouvez activer DevOps Guru pour qu'il surveille vos SQL bases de données for Postgre de l'une des manières suivantes.

Rubriques

- Activer DevOps Guru dans la RDS console
- Ajouter des SQL ressources pour Postgre dans la console DevOps Guru
- Ajouter des SQL ressources pour Postgre à l'aide de AWS CloudFormation

Activer DevOps Guru dans la RDS console

Vous pouvez emprunter plusieurs chemins dans la RDS console Amazon pour activer DevOps Guru.

Rubriques

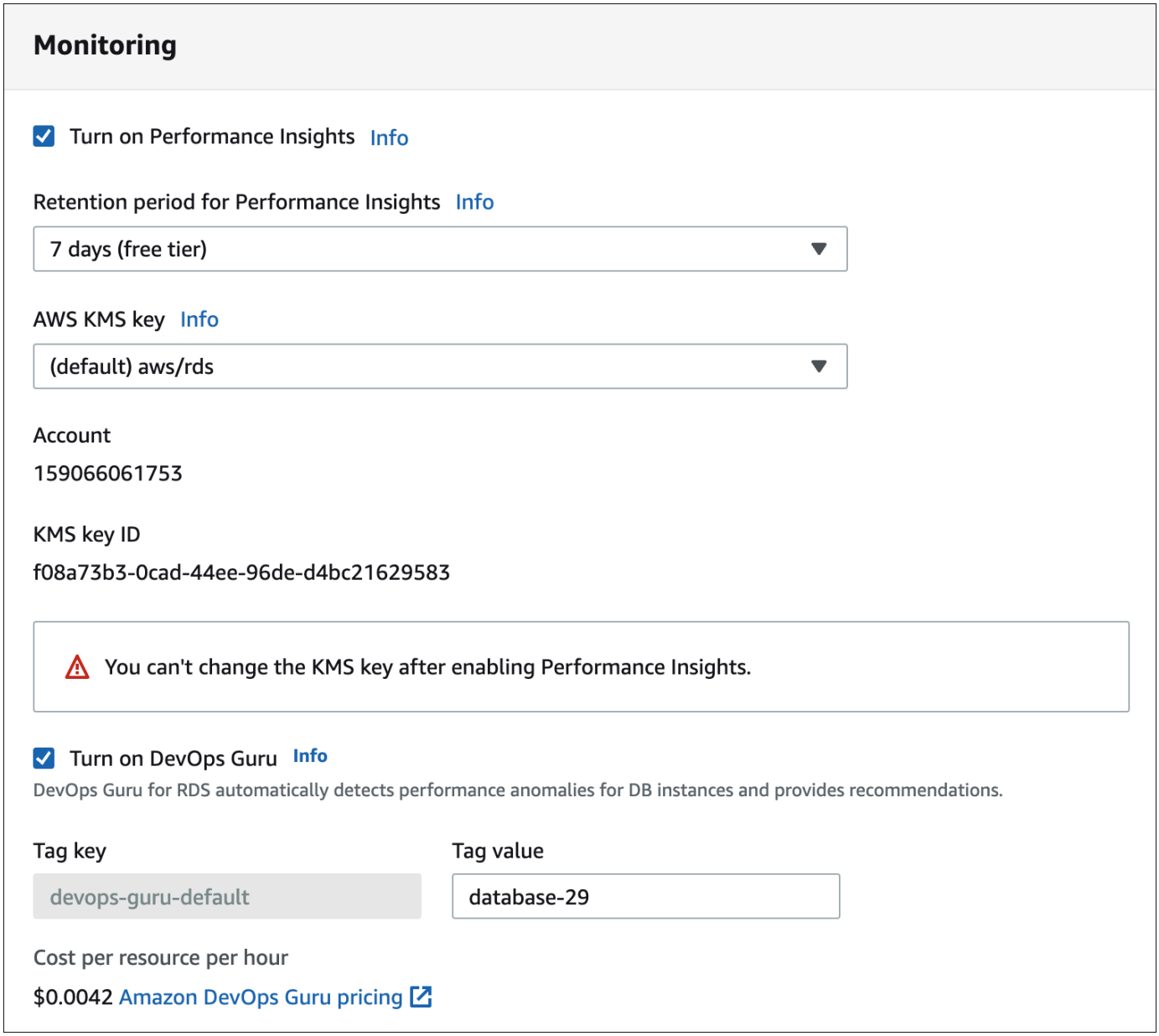
- Activer DevOps Guru lorsque vous créez une base de pour Postgre SQL
- Activer DevOps Guru depuis la bannière de notification
- Répondre à une erreur d'autorisation lorsque vous activez DevOps Guru

Activer DevOps Guru lorsque vous créez une base de pour Postgre SQL

Le flux de création inclut un paramètre qui active la couverture DevOps Guru pour votre base de données. Ce paramètre est activé par défaut lorsque vous choisissez le modèle Production.

Pour activer DevOps Guru lorsque vous créez une base de pour Postgre SQL

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Suivez les étapes de la section Création d'une instance de base de données, jusqu'à l'étape où vous choisissez les paramètres de surveillance, sans la réaliser.
3. Dans Monitoring (Surveillance), choisissez Turn on Performance Insights (Activer Performance Insights). Pour que DevOps Guru RDS puisse fournir une analyse détaillée des anomalies de performance, Performance Insights doit être activé.
4. Choisissez Turn on DevOps Guru.



5. Créez une balise pour votre base de données afin que DevOps Guru puisse la surveiller. Procédez comme suit :

   - Dans le champ de texte Tag key (Clé de balise), saisissez un nom commençant par **`Devops-Guru-`**.
   - Dans le champ de texte Tag value (Valeur de balise), saisissez n'importe quelle valeur. Par exemple, si vous saisissez **`rds-database-1`** le nom de votre SQL base de données pour Postgre, vous pouvez également le saisir **`rds-database-1`** comme valeur de balise.

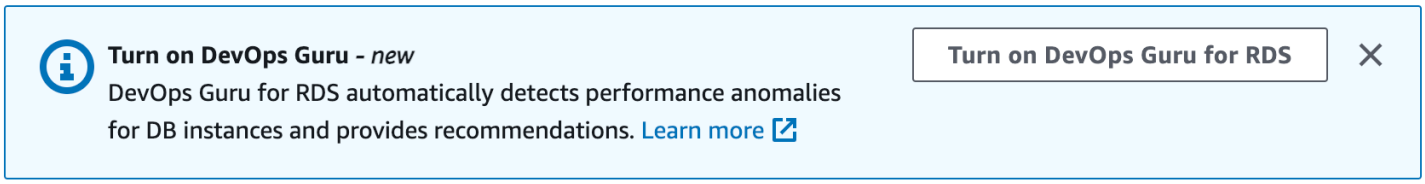
Pour plus d'informations sur les balises, consultez la section « Utiliser des balises pour identifier les ressources dans vos applications DevOps Guru » dans le guide de l'utilisateur Amazon DevOps Guru.

6. Suivez les étapes restantes fournies dans Création d'une instance de base de données.

Activer DevOps Guru depuis la bannière de notification

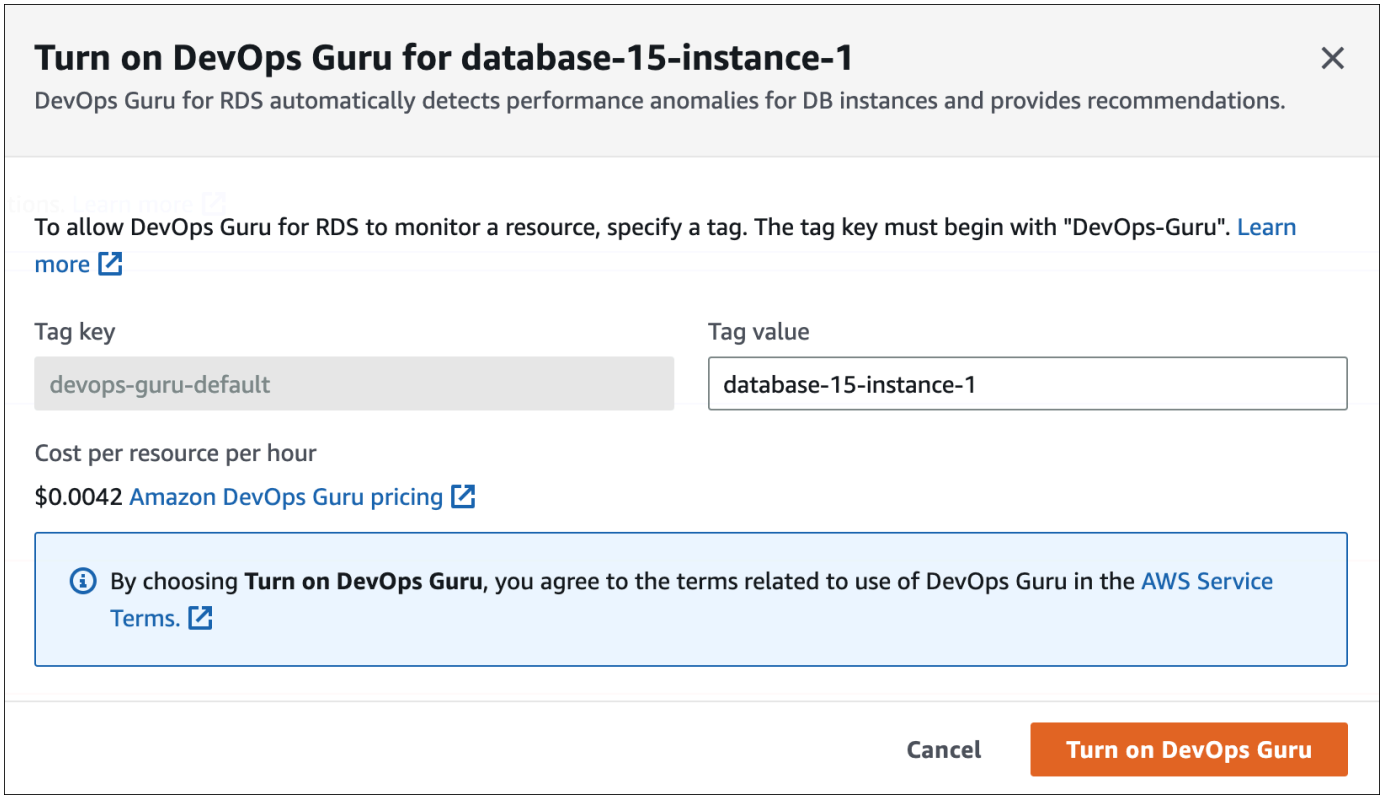
Si vos ressources ne sont pas couvertes par DevOps Guru, Amazon vous en RDS informe par le biais d'une bannière aux emplacements suivants :

- L'onglet Monitoring (Surveillance) d'une instance de cluster de bases de données
- Tableau de bord Performance Insights



Pour activer DevOps Guru pour votre base de pour Postgre SQL

1. Dans la bannière, choisissez Activer DevOps Guru pour RDS.
2. Saisissez un nom de clé et une valeur pour la balise. Pour plus d'informations sur les balises, consultez la section « Utiliser des balises pour identifier les ressources dans vos applications DevOps Guru » dans le guide de l'utilisateur Amazon DevOps Guru.



3. Choisissez Turn on DevOps Guru.

Répondre à une erreur d'autorisation lorsque vous activez DevOps Guru

Si vous activez DevOps Guru depuis la RDS console lorsque vous créez une base de données, la bannière suivante RDS peut s'afficher concernant les autorisations manquantes.

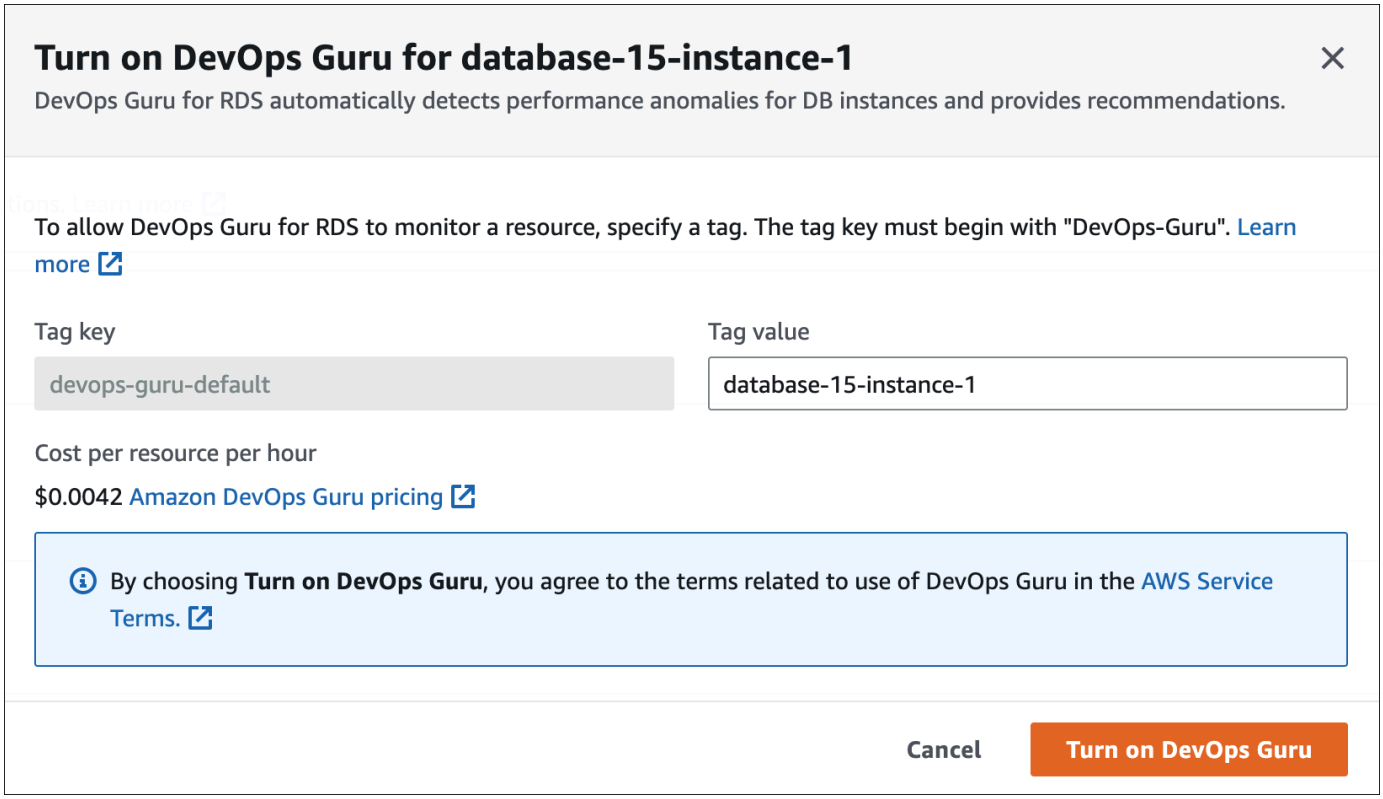


Pour résoudre une erreur d'autorisations

1. Accordez à votre IAM utilisateur ou à votre rôle le rôle géré par l'utilisateur`AmazonDevOpsGuruConsoleFullAccess`. Pour de plus amples informations, veuillez consulter Configuration des politiques IAM d'accès pour DevOps Guru for RDS.
2. Ouvrez la RDS console.
3. Dans le volet de navigation, choisissez Performance Insights.
4. Choisissez une instance de base de données dans le cluster que vous venez de créer.
5. Choisissez l'interrupteur pour lequel vous souhaitez activer DevOpsGuru RDS.

DevOps Guru for RDS

6. Choisissez une valeur de balise. Pour plus d'informations, consultez la section « Utiliser des balises pour identifier les ressources dans vos applications DevOps Guru » dans le guide de l'utilisateur Amazon DevOps Guru.



7. Choisissez Turn on DevOps Guru.

Ajouter des SQL ressources pour Postgre dans la console DevOps Guru

Vous pouvez spécifier la couverture de vos ressources DevOps Guru sur la console DevOps Guru. Suivez l'étape décrite dans Spécifiez la couverture de vos ressources DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru. Lorsque vous modifiez vos ressources analysées, choisissez l'une des options suivantes :

- Choisissez Toutes les ressources du compte pour analyser toutes les ressources prises en charge, y compris les SQL bases de données pour Postgre, dans votre Compte AWS et région.
- Choisissez CloudFormation des piles pour analyser les SQL bases de données for Postgre qui se trouvent dans les piles de votre choix. Pour plus d'informations, voir Utilisation AWS CloudFormation des piles pour identifier les ressources de vos applications DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru.

- Choisissez Tags pour analyser les SQL bases de données for Postgre que vous avez balisées. Pour plus d'informations, consultez la section Utiliser des balises pour identifier les ressources dans vos applications DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru.

Pour plus d'informations, consultez Enable DevOps Guru dans le guide de l'utilisateur Amazon DevOps Guru.

Ajouter des SQL ressources pour Postgre à l'aide de AWS CloudFormation

Vous pouvez utiliser des balises pour ajouter la couverture de vos SQL ressources pour Postgre à vos CloudFormation modèles. La procédure suivante suppose que vous disposez d'un CloudFormation modèle à la fois pour votre instance de SQL base de données pour Postgre et pour votre stack DevOps Guru.

Pour spécifier une instance de SQL base de données pour Postgre à l'aide d'une balise CloudFormation

1. Dans le CloudFormation modèle de votre instance de base de données, définissez une balise à l'aide d'une paire clé/valeur.

   L'exemple suivant attribue la valeur `my-db-instance1` à `Devops-guru-cfn-default` pour une instance de base de données RDS pour PostgreSQL.

   ```
   MyDBInstance1:
     Type: "AWS::RDS::DBInstance"
     Properties:
       DBInstanceIdentifier: my-db-instance1
       Tags:
         - Key: Devops-guru-cfn-default
           Value: devopsguru-my-db-instance1
   ```

2. Dans le CloudFormation modèle de votre pile DevOps Guru, spécifiez le même tag dans votre filtre de collecte de ressources.

   L'exemple suivant configure DevOps Guru pour fournir une couverture à la ressource avec la valeur `my-db-instance1` de balise.

   ```
   DevOpsGuruResourceCollection:
     Type: AWS::DevOpsGuru::ResourceCollection
     Properties:
       ResourceCollectionFilter:
   ```

```
      Tags:
        - AppBoundaryKey: "Devops-guru-cfn-default"
          TagValues:
          - "devopsguru-my-db-instance1"
```

L'exemple suivant fournit une prise en charge pour toutes les ressources à l'intérieur du périmètre de l'application `Devops-guru-cfn-default`.

```
DevOpsGuruResourceCollection:
  Type: AWS::DevOpsGuru::ResourceCollection
  Properties:
    ResourceCollectionFilter:
      Tags:
        - AppBoundaryKey: "Devops-guru-cfn-default"
          TagValues:
          - "*"
```

Pour plus d'informations, voir AWS: DevOpsGuru : : ResourceCollection et AWS: : RDS : : DBInstance dans le AWS CloudFormation Guide de l'utilisateur.

Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée

Surveillez le système d'exploitation de votre instance de base de données en temps réel avec la surveillance améliorée. Lorsque vous voulez voir comment différents processus ou threads utilisent l'UC, les métriques de la surveillance améliorée sont utiles.

Rubriques

- Vue d'ensemble de la surveillance améliorée
- Configuration et activation de la surveillance améliorée
- Afficher les métriques du système d'exploitation dans la RDS console
- Afficher les métriques du système d'exploitation à l'aide CloudWatch de

Vue d'ensemble de la surveillance améliorée

Amazon RDS fournit des métriques en temps réel pour le système d'exploitation sur lequel votre instance de base de données s'exécute. Vous pouvez afficher toutes les métriques système et les informations de processus pour vos instances de base de données RDS sur la console. Vous pouvez gérer les métriques que vous souhaitez surveiller pour chaque instance et personnaliser le tableau de bord en fonction de vos besoins. Pour obtenir une description des métriques de la surveillance améliorée, consultez Métriques du système d'exploitation dans la surveillance améliorée.

RDS fournit les métriques issues de la surveillance améliorée à votre compte Amazon CloudWatch Logs. Vous pouvez créer des filtres CloudWatch de mesures dans CloudWatch Logs et afficher les graphiques sur le CloudWatch tableau de bord. Vous pouvez utiliser la sortie JSON Enhanced Monitoring de CloudWatch Logs dans le système de surveillance de votre choix. Pour plus d'informations, consultez la section Surveillance améliorée dans Amazon RDS. FAQs

Rubriques

- Disponibilité de la surveillance améliorée
- Différences entre les indicateurs de surveillance CloudWatch et les indicateurs améliorés
- Conservation des métriques de surveillance améliorée
- Coût de la surveillance améliorée

Disponibilité de la surveillance améliorée

La surveillance améliorée est disponible pour les moteurs de base de données suivants :

- Db2
- MariaDB
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL

Différences entre les indicateurs de surveillance CloudWatch et les indicateurs améliorés

Un hyperviseur crée et exécute des machines virtuelles (VMs). À l'aide d'un hyperviseur, une instance peut prendre en charge plusieurs invités VMs en partageant virtuellement la mémoire et le processeur. CloudWatch collecte des métriques sur l'utilisation du processeur à partir de l'hyperviseur pour une instance de base de données. En revanche, la surveillance améliorée collecte ses métriques à partir d'un agent sur l'instance de base de données.

Vous constaterez peut-être des différences entre les mesures de surveillance CloudWatch et celles de surveillance améliorée, car la couche hyperviseur effectue une petite quantité de travail. Les différences peuvent être plus importantes si vos instances de base de données utilisent des classes d'instance plus petites. Dans ce scénario, un plus grand nombre de machines virtuelles (VMs) sont probablement gérées par la couche hyperviseur sur une seule instance physique.

Pour obtenir une description des métriques de la surveillance améliorée, consultez Métriques du système d'exploitation dans la surveillance améliorée. Pour plus d'informations sur CloudWatch les métriques, consultez le guide de CloudWatch l'utilisateur Amazon.

Conservation des métriques de surveillance améliorée

Par défaut, les métriques de surveillance améliorée sont stockées pendant 30 jours dans les CloudWatch journaux. Cette période de conservation est différente des CloudWatch indicateurs classiques.

Pour modifier la durée pendant laquelle les métriques sont stockées dans les CloudWatch journaux, modifiez la durée de conservation du groupe de `RDSOSMetrics` journaux dans la CloudWatch

console. Pour plus d'informations, consultez la section Conservation des données du journal des modifications dans CloudWatch les journaux dans le guide de l'utilisateur Amazon CloudWatch Logs.

Coût de la surveillance améliorée

Les métriques de surveillance améliorée sont stockées dans les CloudWatch journaux plutôt que dans CloudWatch les métriques. Le coût de la surveillance améliorée dépend des facteurs suivants :

- La surveillance améliorée ne vous est facturée que si vous dépassez le volume de transfert et de stockage de données fourni par Amazon CloudWatch Logs. Les frais sont basés sur les taux de transfert et de stockage des données des CloudWatch journaux.
- La quantité d'informations transférées pour une instance RDS est directement proportionnelle à la granularité définie pour la fonction de surveillance améliorée. Plus l'intervalle de surveillance est court, plus la fréquence des rapports sur les métriques du système d'exploitation est élevée, ce qui augmente les coûts de surveillance. Pour gérer les coûts, définissez différentes granularités pour différentes instances de vos comptes.
- Les coûts d'utilisation de la surveillance améliorée sont appliqués pour chaque instance de base de données pour laquelle l'option est activée. Surveiller un grand nombre d'instances de bases de données est plus onéreux que de n'en surveiller que quelques unes.
- Les instances de bases de données qui prennent en charge une charge de travail nécessitant des calculs intensifs doivent signaler une activité de traitement du système d'exploitation plus intense et des coûts plus élevés pour la surveillance améliorée.

Pour plus d'informations sur les tarifs, consultez les CloudWatch tarifs Amazon.

Configuration et activation de la surveillance améliorée

Pour utiliser la surveillance améliorée, vous devez créer un rôle IAM, puis activer la surveillance améliorée.

Rubriques

- Création d'un rôle IAM pour la surveillance améliorée
- Activer et désactiver la surveillance améliorée
- Lutter contre le problème de l'adjoint confus

Création d'un rôle IAM pour la surveillance améliorée

La surveillance améliorée nécessite l'autorisation d'agir en votre nom pour envoyer des informations métriques du système d'exploitation à CloudWatch Logs. Vous accordez des autorisations de surveillance améliorée à l'aide d'un rôle AWS Identity and Access Management (IAM). Vous pouvez soit créer ce rôle lorsque vous activez la surveillance améliorée, soit le créer au préalable.

Rubriques

- Création du rôle IAM lorsque vous activez la surveillance améliorée
- Création du rôle IAM avant d'activer la surveillance améliorée

Création du rôle IAM lorsque vous activez la surveillance améliorée

Lorsque vous activez la surveillance améliorée dans la console RDS, Amazon RDS peut créer le rôle IAM requis pour vous. Le rôle est nommé `rds-monitoring-role`. RDS utilise ce rôle pour l'instance de base de données, le réplica en lecture spécifié ou le cluster de base de données Multi-AZ.

Pour créer le rôle IAM lors de l'activation de la surveillance améliorée

1. Suivez les étapes de Activer et désactiver la surveillance améliorée.
2. Définissez Rôle de surveillance sur Par défaut à l'étape où vous choisissez un rôle.

Création du rôle IAM avant d'activer la surveillance améliorée

Vous pouvez créer le rôle requis avant d'activer la surveillance améliorée. Lorsque vous activez la surveillance améliorée, spécifiez le nom de votre nouveau rôle. Vous devez créer ce rôle requis si vous activez la surveillance améliorée à l'aide de l' AWS CLI ou de l'API RDS.

L'utilisateur qui active la surveillance améliorée doit se voir accorder l'autorisation `PassRole`. Pour plus d'informations, consultez l'exemple 2 de la section Octroi à un utilisateur des autorisations pour transmettre un rôle à un AWS service dans le guide de l'utilisateur IAM.

Pour créer un rôle IAM pour la surveillance améliorée Amazon RDS

1. Ouvrez la console IAM à https://console.aws.amazon.coml'adresse.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Sélectionnez Create role (Créer un rôle).

4. Choisissez l'onglet Service AWS , puis choisissez RDS dans la liste de services.
5. Choisissez RDS - Enhanced Monitoring (RDS - Surveillance améliorée), puis Next (Suivant).
6. Assurez-vous que les politiques d'autorisations indiquent Amazon RDSEnhanced MonitoringRole, puis choisissez Next.
7. Dans le champ Role name (Nom de rôle), saisissez un nom pour votre rôle. Par exemple, entrez **`emaccess`**.

   L'entité de confiance correspondant à votre rôle est le AWS service monitoring.rds.amazonaws.com.
8. Choisissez Créer un rôle.

Activer et désactiver la surveillance améliorée

Vous pouvez gérer la surveillance améliorée à l'aide de l'API AWS Management Console AWS CLI, ou RDS. Vous pouvez définir différentes granularités pour la collecte des métriques sur chaque instance de base de données de de base de données.

console

Vous pouvez activer la surveillance améliorée lorsque vous créez une instance, un cluster Multi-AZ de base de données ou un réplica en lecture, ou lorsque vous modifiez une instance ou un cluster de base de données Multi-AZ de base de données. Si vous modifiez une instance de base de données pour activer la surveillance améliorée, vous n'avez pas besoin de redémarrer votre instance de base de données pour que la modification prenne effet.

Vous pouvez activer la surveillance améliorée dans la console RDS lorsque vous effectuez l'une des opérations suivantes sur la page Databases (Bases de données) :

- Création d’une instance ou un cluster Multi-AZ de base de données : choisissez Create database (Créer une base de données).
- Créer un réplica en lecture : choisissez Actions, puis Create read replica (Créer un réplica en lecture).
- Modifier une instance de base de données, un de base de données ou un cluster de base de données multi-AZ : choisissez Modifier.

Pour activer ou désactiver la surveillance améliorée dans la console RDS

1. Descendez jusqu'à Additional configuration (Configuration supplémentaire).
2. Dans Surveillance, choisissez Activer la surveillance améliorée pour votre instance de base de données ou votre réplique de lecture. Désélectionnez l'option pour désactiver la surveillance améliorée .
3. Définissez la propriété Monitoring Role sur le rôle IAM que vous avez créé pour permettre à Amazon RDS de communiquer avec Amazon CloudWatch Logs à votre place, ou choisissez Default pour que RDS crée un rôle nommé pour vous. `rds-monitoring-role`
4. Définissez la propriété Granularity sur l'intervalle, en secondes, entre les points de collecte des métriques pour votre instance de base de données de base de données ou votre réplique de lecture. La propriété Granularité peut être définie sur l'une des valeurs suivantes : `1`, `5`, `10`, `15`, `30` ou `60`.

   La console RDS est actualisée toutes les 5 secondes. Si la granularité est définie sur 1 seconde dans la console RDS, les métriques mises à jour s'affichent toutes les 5 secondes uniquement. Vous pouvez récupérer les mises à jour des métriques en une seconde à l'aide CloudWatch des journaux.

AWS CLI

Pour activer la surveillance améliorée à l'aide des commandes suivantes AWS CLI, définissez l'`--monitoring-interval`option sur une valeur autre que le rôle dans lequel vous l'avez créé `0` et définissez l'`--monitoring-role-arn`option sur le rôle dans lequel vous l'avez crééCréation d'un rôle IAM pour la surveillance améliorée.

- create-db-instance
- create-db-instance-read-réplique
- modify-db-instance
- create-db-cluster(Cluster de bases de données multi-AZ)
- modify-db-cluster(Cluster de bases de données multi-AZ)

L'option `--monitoring-interval` spécifie l'intervalle, en secondes, entre les points lorsque des métriques de surveillance améliorée sont collectées. Les valeurs valides pour l'option sont `0`, `1`, `5`, `10`, `15`, `30` et `60`.

Pour désactiver la surveillance améliorée à l'aide de AWS CLI, définissez l'`--monitoring-interval`option sur `0` dans ces commandes.

Example

L'exemple suivant active la surveillance améliorée pour une instance de base de données :

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --monitoring-interval 30 \
    --monitoring-role-arn arn:aws:iam::123456789012:role/emaccess
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --monitoring-interval 30 ^
    --monitoring-role-arn arn:aws:iam::123456789012:role/emaccess
```

Example

L'exemple suivant active la surveillance améliorée pour une cluster de base de données Multi-AZ :

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-cluster \
    --db-cluster-identifier mydbcluster \
    --monitoring-interval 30 \
    --monitoring-role-arn arn:aws:iam::123456789012:role/emaccess
```

Dans Windows :

```
aws rds modify-db-cluster ^
    --db-cluster-identifier mydbcluster ^
    --monitoring-interval 30 ^
    --monitoring-role-arn arn:aws:iam::123456789012:role/emaccess
```

API RDS

Pour activer la surveillance améliorée à l'aide de l'AP RDS, définissez le paramètre `MonitoringInterval` sur une valeur autre que `0` et définissez le paramètre `MonitoringRoleArn` sur le rôle que vous avez créé dans Création d'un rôle IAM pour la surveillance améliorée. Définissez ces paramètres dans les actions suivantes :

- CréerDBInstance
- CréerDBInstanceReadReplica
- ModifyDBInstance
- Créer DBCluster (cluster de base de données multi-AZ)
- Modifier DBCluster (cluster de bases de données multi-AZ)

Le paramètre `MonitoringInterval` spécifie l'intervalle, en secondes, entre les points lorsque des métriques de surveillance améliorée sont collectées. Les valeurs valides sont `0`, `1`, `5`, `10`, `15`, `30` et `60`.

Pour désactiver la surveillance améliorée à l'aide de l'API RDS, définissez `MonitoringInterval` sur `0`.

Lutter contre le problème de l'adjoint confus

Le problème de député confus est un problème de sécurité dans lequel une entité qui n'est pas autorisée à effectuer une action peut contraindre une entité plus privilégiée à le faire. En AWS, l'usurpation d'identité interservices peut entraîner la confusion des adjoints. L'usurpation d'identité entre services peut se produire lorsqu'un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé et ses autorisations utilisées pour agir sur les ressources d'un autre client auxquelles on ne serait pas autorisé d'accéder autrement. Pour éviter cela, AWS fournit des outils qui vous aident à protéger vos données pour tous les services avec des principaux de service qui ont eu accès aux ressources de votre compte. Pour de plus amples informations, veuillez consulter Le problème de l'adjoint confus.

Afin de limiter les autorisations octroyées par Amazon RDS à un autre service pour la ressource, nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans une politique d'approbation pour votre rôle de surveillance améliorée. Si vous utilisez les deux clés de contexte de condition globale, elles doivent utiliser le même ID de compte.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. Pour Amazon RDS, définissez `aws:SourceArn` sur `arn:aws:rds:`*`Region`*`:`*`my-account-id`*`:db:`*`dbname`*.

L'exemple suivant utilise les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans une politique d'approbation afin d'empêcher le problème d'adjoint confus.

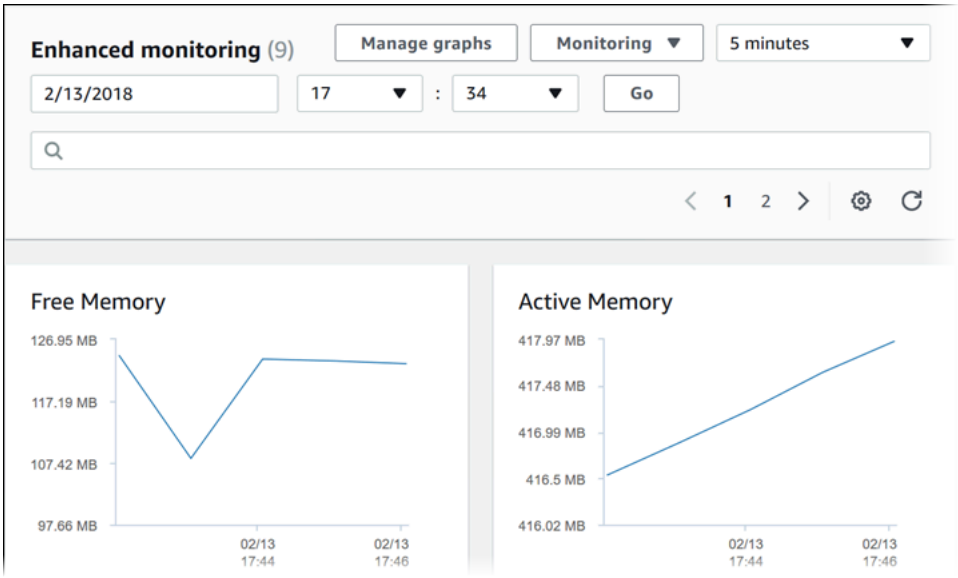
JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "monitoring.rds.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringLike": {
          "aws:SourceArn": "arn:aws:rds:Region:my-account-id:db:dbname"
        },
        "StringEquals": {
          "aws:SourceAccount": "my-account-id"
        }
      }
    }
  ]
}
```

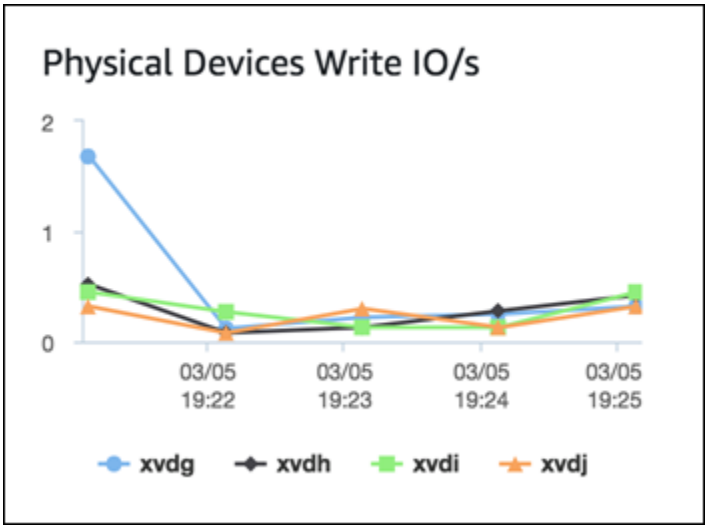
Afficher les métriques du système d'exploitation dans la RDS console

Vous pouvez consulter les métriques du système d'exploitation signalées par Enhanced Monitoring dans la RDS console en choisissant Enhanced monitoring for Monitoring.

L'exemple suivant montre la page de surveillance améliorée. Pour obtenir une description des métriques de la surveillance améliorée, consultez Métriques du système d'exploitation dans la surveillance améliorée.



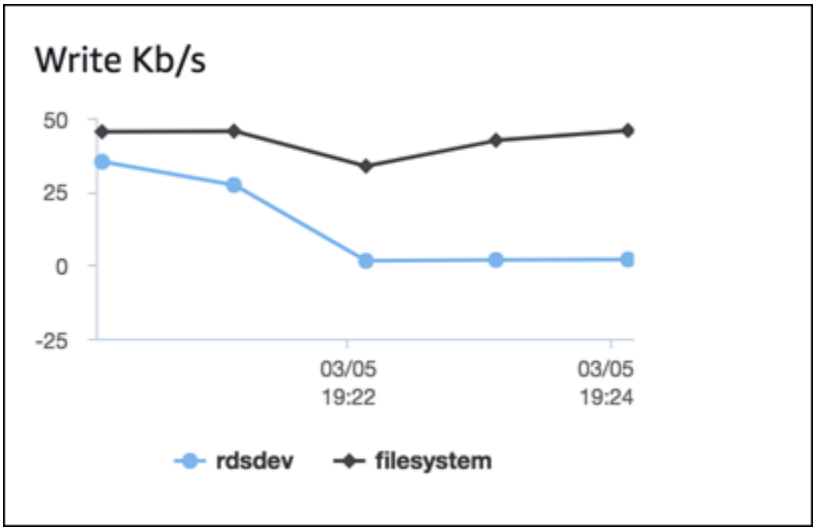
Certaines instances de base de données utilisent plusieurs disques pour le volume de stockage des données de l'instance de base de données. Sur ces instances de base de données, les graphiques Physical Devices (Appareils physiques) montrent les métriques de chacun des disques. Par exemple, le graphique suivant montre les métriques pour quatre disques.



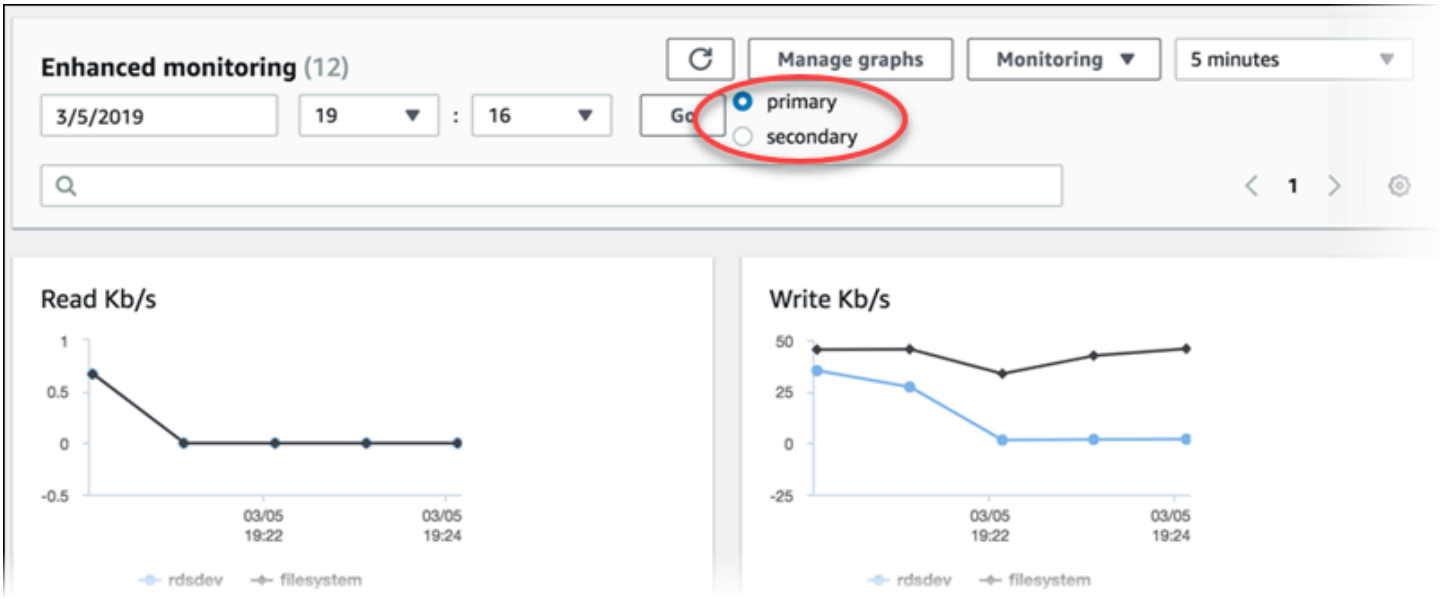
> **Note**
>
> Actuellement, les graphes Physical Devices ne sont pas disponibles pour les instances de base de données Microsoft SQL Server.

Lorsque vous veuillez consulter les graphiques agrégés Disk I/O (I/O de disque) et File system (Système de fichiers), l'appareil rdsdev se rapporte au système de fichiers `/rdsdbdata` dans lequel

tous les journaux et fichiers de base de données sont stockés. L'appareil filesystem se rapporte au système de fichiers / (également appelé système de fichiers racine), dans lequel les fichiers associés au système d'exploitation sont stockés.



Si l'instance de base de données est un déploiement multi-AZ, vous pouvez afficher les métriques de système d'exploitation pour l'instance de base de données principale et son réplica de secours multi-AZ. Dans la vue Enhanced monitoring (Surveillance améliorée), choisissez primary (principal) pour afficher les métriques de système d'exploitation pour l'instance de base de données principale, ou choisissez secondary (secondaire) pour afficher les métriques de système d'exploitation pour le réplica de secours.

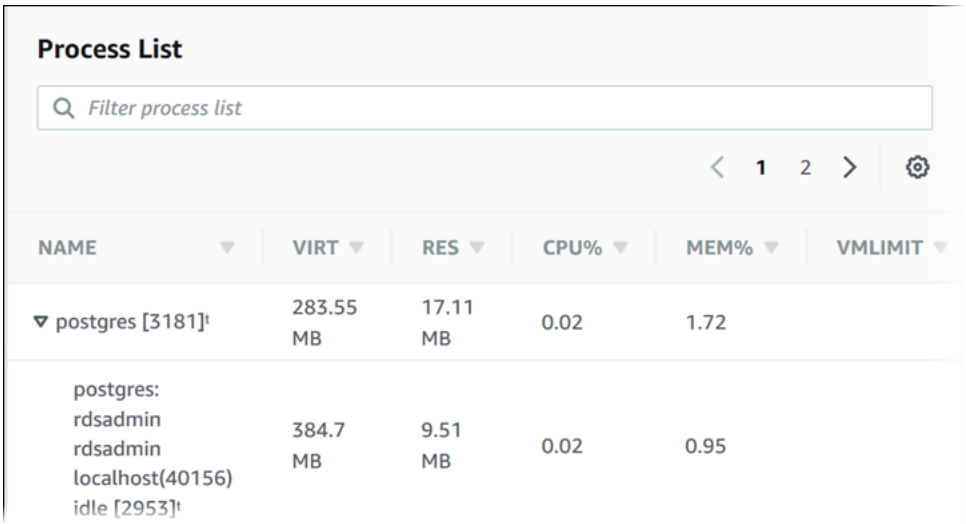


Pour plus d'informations sur les déploiements multi-AZ, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

> **Note**
>
> Actuellement, l'affichage des métriques du système d'exploitation pour un réplica de secours multi-AZ n'est pas pris en charge pour les instances de base de données MariaDB.

Si vous voulez afficher des détails sur les processus qui s'exécutent sur votre instance de base de données, choisissez Liste de processus de système d'exploitation pour Surveillance.

La vue Liste des processus est affichée ci-dessous.



Les métriques de surveillance améliorée affichées dans la vue Liste des processus sont organisées de la manière suivante :

- RDSprocessus enfants — Affiche un résumé des RDS processus qui prennent en charge l'instance de base de données, par exemple pour les instances My SQL DB. Les threads du processus sont imbriqués sous le processus parent. Les threads de processus indiquent uniquement CPU l'utilisation, car les autres métriques sont les mêmes pour tous les threads du processus. La console affiche 100 processus et threads maximum. Il en résulte une combinaison des processus et des threads les plus CPU gourmands en mémoire et en mémoire. S'il existe plus de 50 processus et 50 threads, la console affiche les 50 premiers éléments de chaque catégorie. Cette présentation vous aide à identifier les processus ayant le plus d'impact sur les performances.
- RDSprocessus — Affiche un résumé des ressources utilisées par l'agent de RDS gestion, des processus de surveillance des diagnostics et des autres AWS processus requis pour prendre en charge les RDS instances de base de données.

- OS processes (Processus SE) – Affiche un récapitulatif des processus du noyau et du système, qui ont généralement un faible impact sur les performances.

Les éléments répertoriés pour chaque processus sont les suivants :

- VIRT— Affiche la taille virtuelle du processus.
- RES— Affiche la mémoire physique réellement utilisée par le processus.
- CPU% — Affiche le pourcentage de la CPU bande passante totale utilisée par le processus.
- MEM% — Affiche le pourcentage de la mémoire totale utilisée par le processus.

Les données de surveillance affichées dans la RDS console sont extraites d'Amazon CloudWatch Logs. Vous pouvez également récupérer les métriques d'une instance de base de données sous forme de flux de journal à partir de CloudWatch Logs. Pour de plus amples informations, veuillez consulter Afficher les métriques du système d'exploitation à l'aide CloudWatch de.

Les métriques de surveillance améliorée ne sont pas renvoyées dans les situations suivantes :

- Basculement de l'instance de base de données.
- Modification de la classe d'instance de l'instance de base de données (dimensionnement du calcul).

Les métriques de surveillance améliorée sont renvoyées pendant le redémarrage d'une instance de base de données car seul le moteur de base de données est redémarré. Les métriques concernant le système d'exploitation continuent d'être relevées.

Afficher les métriques du système d'exploitation à l'aide CloudWatch de

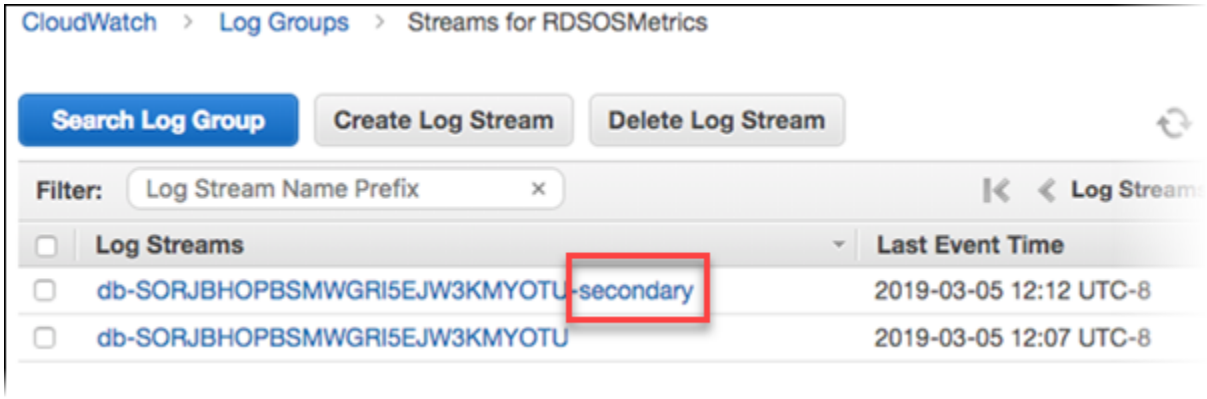
Une fois que vous avez activé la surveillance améliorée pour votre instance de base de données ou votre cluster de de base de données multi-AZ, vous pouvez consulter les métriques correspondantes à l'aide de CloudWatch journaux, chaque flux de journal représentant une instance de base de données ou un cluster de base de données unique surveillé. L'identifiant du flux de journal est l'identifiant de la ressource (`DbiResourceId`) de l'instance de base de données ou du cluster de base de données.

Pour afficher les données du journal de la surveillance améliorée

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.

2. Si nécessaire, choisissez l'instance de base de données ou le Région AWS de base de données Multi-AZ dans lequel se trouve votre instance de base de données. Pour en savoir plus, consultez la section relative aux régions et points de terminaison du document Référence générale Amazon Web Services.
3. Choisissez Logs (Journaux) dans le volet de navigation.
4. RDSOSMetricsChoisissez dans la liste des groupes de journaux.

   Dans un déploiement d'une instance de base de données Multi-AZ, les fichiers journaux avec `-secondary` ajouté au nom correspondent au réplica de secours Multi-AZ.



5. Choisissez le flux de journal à afficher dans la liste des flux de journaux.

Référence des métriques pour Amazon RDS

Dans cette référence, vous trouverez des descriptions des métriques RDSAmazon pour Amazon CloudWatch, Performance Insights et Enhanced Monitoring.

Rubriques

- CloudWatch Métriques Amazon pour Amazon RDS
- CloudWatch Dimensions Amazon pour Amazon RDS
- Statistiques Amazon CloudWatch pour Amazon RDS Performance Insights
- Métrique de compteur de Performance Insights
- Statistiques SQL pour Performance Insights
- Métriques du système d'exploitation dans la surveillance améliorée

CloudWatch Métriques Amazon pour Amazon RDS

CloudWatch Les métriques Amazon fournissent des informations sur les performances et l'état des instances et clusters Amazon RDS, ce qui vous permet de surveiller le comportement du système et de prendre des décisions basées sur les données. Ces indicateurs permettent de suivre l'utilisation des ressources, l'activité des bases de données et l'efficacité opérationnelle, offrant ainsi une visibilité sur les performances de vos instances.

Cette référence décrit les métriques spécifiques disponibles pour Amazon RDS et explique comment les interpréter et les utiliser pour optimiser les performances des bases de données, résoudre les problèmes et garantir une haute disponibilité.

Amazon RDS publie des métriques sur Amazon CloudWatch dans les espaces de `AWS/Usage` noms `AWS/RDS` et.

Rubriques

- Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS
- Mesures CloudWatch d'utilisation d'Amazon pour Amazon RDS (Aurora)

Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS

L'espace de `AWS/RDS` noms d'Amazon CloudWatch inclut les métriques suivantes au niveau de l'instance.

> **Note**
>
> La console Amazon RDS peut afficher des métriques en unités différentes de celles envoyées à Amazon CloudWatch. Par exemple, la console Amazon RDS peut afficher une métrique en mégaoctets (Mo), tandis que la métrique est envoyée à Amazon en octets. CloudWatch

Métrique	Description	S'applique à	Unités
`BinLogDiskUsage`	Quantité d'espace disque occupée par les journaux binaires. Si les sauvegardes automatiques sont activées pour les instances MySQL et MariaDB, y compris les réplicas en lecture, des journaux binaires sont créés.	MariaDB MySQL	Octets
`BurstBalance`	Pourcentage de crédits Burst-bucket I/O SSD à usage général (gp2) disponibles.	Tous	Pourcentage
`CheckpointLag`	Le temps écoulé depuis le dernier point de contrôle.		Secondes
`ConnectionAttempts`	Le nombre de tentatives de connexion à une instance, qu'elles soient réussies ou non.	MySQL	Nombre
`CPUUtilization`	Pourcentage d'utilisation de la CPU.	Tous	Pourcentage
`CPUCreditUsage`	Nombre de crédits UC dépensés par l'instance pour l'utilisation de l'UC. Un crédit de processeur équivaut à un vCPU fonctionnant à 100 % d'utilisation pendant une minute ou une combinaison équivalente de vCPUs, d'utilisation et de temps. Par exemple, vous pouvez		Crédits (minutes vCPU)

Métrique	Description	S'applique à	Unités
	avoir un vCPU fonctionnant à 50 % d'utilisation pendant deux minutes ou deux vCPU CPUs fonctionnant à 25 % d'utilisation pendant deux minutes. Cette métrique s'applique uniquement aux instances `db.t4g`, `db.t2` et `db.t3`. **Note** Nous recommandons d'utiliser uniquement les classes d'instance de base de données T pour les serveurs de développement et de test, ou pour d'autres serveurs non dédiés à la production. Pour plus de détails sur les classes d'instances T, voir Types de classes d'instance de base de données Les métriques de crédits UC sont disponibles uniquement toutes les 5 minutes. Si vous spécifiez une période supérieure à cinq minutes, utilisez la statistique `Sum` au lieu de la statistique `Average`.		

Métrique	Description	S'applique à	Unités
`CPUCredit Balance`	Nombre de crédits UC gagnés qu'une instance a accumulés depuis son lancement ou son démarrage. Pour les instances T2 Standard, le `CPUCredit Balance` inclut également le nombre de crédits de lancement qui ont été accumulés. Les crédits sont accumulés dans le solde de crédits quand ils sont gagnés et supprimés du solde de crédits lorsqu'ils sont dépensés. Le solde de crédits présente une limite maximum qui est déterminée par la taille de l'instance. Une fois que la limite est atteinte, tous les nouveaux crédits gagnés sont rejetés. Pour les instances T2 Standard, les crédits de lancement ne sont pas comptés dans la limite. L'instance peut dépenser les crédits figurant dans le `CPUCreditBalance` pour dépasser le niveau de base de l'utilisation de l'UC. Les crédits figurant dans le `CPUCredit Balance` d'une instance en cours d'exécution n'expirent pas. Lorsque l'instance s'arrête, le `CPUCredit Balance` ne persiste pas, et tous les crédits accumulés sont perdus. Les métriques de crédits UC sont disponibles uniquement toutes les 5 minutes.		Crédits (minutes vCPU)

Métrique	Description	S'applique à	Unités
	Cette métrique s'applique uniquement aux instances `db.t4g` `db.t2` `db.t3`, et. **Note** Nous recommandons d'utiliser uniquement les classes d'instance de base de données T pour les serveurs de développement et de test, ou pour d'autres serveurs non dédiés à la production. Pour plus de détails sur les classes d'instances T, voir Types de classes d'instance de base de données Les crédits de lancement fonctionnent de la même manière sur Amazon RDS et sur Amazon EC2. Pour obtenir plus d'informations, consultez la section Launch credits (Crédits de lancement) dans le Guide de l'utilisateur d'Amazon Elastic Compute Cloud pour les instances Linux.		

Métrique	Description	S'applique à	Unités
`CPUSurplusCreditBalance`	Nombre de crédits excédentaires dépensés par une instance illimitée lorsque la valeur `CPUCreditBalance` est nulle. La valeur de `CPUSurplusCreditBalance` est remboursée progressivement par les crédits UC gagnés. Si le nombre de crédits excédentaires dépasse le nombre maximum de crédits que l'instance peut gagner en 24 heures, les crédits excédentaires dépensés au-dessus du maximum génèrent des frais supplémentaires. Les métriques de crédits CPU sont disponibles toutes les 5 minutes uniquement.	Tous	Crédits (minutes vCPU)

Métrique	Description	S'applique à	Unités
`CPUSurplusCreditsCharged`	Nombre de crédits excédentaires dépensés qui ne sont pas remboursés progressivement par les crédits UC gagnés et qui génèrent donc des frais supplémentaires. Les crédits excédentaires dépensés sont facturés lorsque l'une des situations suivantes se produit : • Les crédits excédentaires dépensés dépassent le nombre maximum de crédits que l'instance peut gagner sur une période de 24 heures. Les crédits excédentaires dépensés au-dessus de ce maximum sont facturés à la fin de l'heure. • L'instance est arrêtée ou résiliée. • L'instance bascule du mode `unlimited` au mode `standard`. Les métriques de crédits CPU sont disponibles toutes les 5 minutes uniquement.	Tous	Crédits (minutes vCPU)

Métrique	Description	S'applique à	Unités
`DatabaseConnections`	Nombre de connexions réseau client à l'instance de base de données. Le nombre de sessions de base de données peut être supérieur à la valeur de la métrique car celle-ci n'inclut pas les éléments suivants : • Sessions qui n'ont plus de connexion réseau mais dont la base de données n'a pas été nettoyée • Sessions créées par le moteur de base de données à ses propres fins • Sessions créées par les capacités d'exécution parallèle du moteur de base de données • Sessions créées par le planificateur de tâches du moteur de base de données • Connexions Amazon RDS	Tous	Nombre
`DiskQueueDepth`	Le nombre de demandes en attente I/Os (lecture/écriture) en attente d'accès au disque.	Tous	Nombre
`DiskQueueDepthLogVolume`	Le nombre de demandes en attente I/Os (lecture/écriture) en attente d'accès au disque du volume du journal.	Instances de base de données avec volume de journal dédié activé	Nombre

Métrique	Description	S'applique à	Unités
`EBSByteBalance%`	Pourcentage de crédits de débit restant dans le compartiment en rafales de votre base de données RDS. Cette métrique est disponible uniquement pour la surveillance basique. La valeur de la métrique est basée sur le débit de tous les volumes, y compris le volume racine, plutôt que sur les seuls volumes contenant des fichiers de base de données. Pour trouver les tailles d'instance compatibles avec cette métrique, consultez les tailles d'instance marquées d'un astérisque (*) dans le tableau EBS optimisé par défaut du guide de EC2 l'utilisateur Amazon. La statistique `Sum` n’est pas applicable pour cette métrique.	Tous	Pourcentage

Métrique	Description	S'applique à	Unités
`EBSIOBalance%`	Pourcentage de I/O crédits restant dans le bucket burst de votre base de données RDS. Cette métrique est disponible uniquement pour la surveillance basique. La valeur de la métrique est basée sur les IOPS de tous les volumes, y compris le volume racine, plutôt que sur les seuls volumes contenant des fichiers de base de données. Pour connaître les tailles d'instance compatibles avec cette métrique, consultez les types d'instances optimisés pour Amazon EBS dans le guide de EC2 l'utilisateur Amazon. La `Sum` statistique ne s'applique pas à cette métrique. Cette métrique est différente de `BurstBalance` . Pour savoir comment utiliser cette métrique, consultez l'article de blog Improving application performance and reducing costs with Amazon EBS-Optimized Instance burst capability.	Tous	Pourcentage
`FailedSQLServerAgentJobsCount`	Nombre de tâches Microsoft SQL Server Agent ayant échoué au cours de la dernière minute.	Microsoft SQL Server	Compte par minute

Métrique	Description	S'applique à	Unités
`FreeableMemory`	Quantité de mémoire vive disponible. Pour les instances de base de données MariaDB, MySQL, Oracle et PostgreSQL, cette métrique rapporte la valeur du champ `MemAvailable` de `/proc/meminfo` .	Tous	Octets
`FreeLocalStorage`	Quantité d'espace de stockage local disponible. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Octets

Métrique	Description	S'applique à	Unités
`FreeLocalStoragePercent`	Pourcentage d'espace de stockage local disponible. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Pourcentage
`FreeStorageSpace`	Quantité d'espace de stockage disponible.	Tous	Octets
`FreeStorageSpaceLogVolume`	Volume d'espace de stockage disponible sur le volume des journaux.	Instances de base de données avec volume de journal dédié activé	Octets
`IamDbAuthConnectionRequests`	Le nombre de demandes de connexion utilisant l'authentification IAM à l'instance de base de données.	Tous	Nombre

Métrique	Description	S'applique à	Unités
`MaximumUsedTransactionIDs`	Le nombre maximum de transactions IDs qui ont été utilisées.	PostgreSQL	Nombre
`NetworkReceiveThroughput`	Trafic de réseau entrant (réception) sur l'instance DB, notamment le trafic de base de données client et le trafic Amazon RDS, utilisé pour la supervision et la réplication.	Tous	Octets par seconde
`NetworkTransmitThroughput`	Trafic de réseau sortant (transmission) sur l'instance DB, comprenant le trafic de base de données client et le trafic Amazon RDS, utilisé pour la supervision et la réplication.	Tous	Octets par seconde
`OldestLogicalReplicationSlotLag`	La taille retardée d'Amazon RDS valide une transaction sur la base de données source et l'heure à laquelle RDS applique la transaction sur la base de données répliquée.	PostgreSQL	Octets
`OldestReplicationSlotLag`	Taille du retard du réplica le plus en retard en termes de données WAL reçues.	PostgreSQL	Octets
`ReadIOPS`	Nombre moyen d' I/O opérations de lecture sur disque par seconde.	Tous	Nombre par seconde

Métrique	Description	S'applique à	Unités
`ReadIOPSLocalStorage`	Nombre moyen d' I/O opérations de lecture du disque vers le stockage local par seconde. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Nombre par seconde
`ReadIOPSLogVolume`	Nombre moyen d' I/O opérations de lecture du disque par seconde pour le volume du journal.	Instances de base de données avec volume de journal dédié activé	Nombre par seconde
`ReadLatency`	Durée moyenne par I/O opération sur le disque.	Tous	Secondes

Métrique	Description	S'applique à	Unités
`ReadLatencyLocalStorage`	Durée moyenne par I/O opération sur disque pour le stockage local. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Secondes
`ReadLatencyLogVolume`	Durée moyenne par I/O opération sur disque pour le volume du journal.	Instances de base de données avec volume de journal dédié activé	Secondes
`ReadThroughput`	Nombre moyen d’octets lus sur le disque par seconde.	Tous	Octets par seconde

Métrique	Description	S'applique à	Unités
`ReadThroughputLocalStorage`	Nombre moyen d'octets lus sur le disque par seconde pour le stockage local. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Octets par seconde
`ReadThroughputLogVolume`	Nombre moyen d'octets lus sur le disque par seconde pour le volume des journaux.	Instances de base de données avec volume de journal dédié activé	Octets par seconde

Métrique	Description	S'applique à	Unités
`ReplicaLag`	Pour les configurations de réplica en lecture, il s'agit du temps pendant lequel l'instance de base de données de réplica en lecture est en retard par rapport à l'instance de base de données source. S'applique aux réplicas en lecture MariaDB, Microsoft SQL Server, MySQL, Oracle et PostgreSQL. Pour les clusters de base de données Multi-AZ, la différence de temps entre la dernière transaction sur l'instance de base de données auteur et la dernière transaction appliquée sur une instance de base de données lecteur.		Secondes
`ReplicationChannelLag`	Pour les configurations de réplication multi-sources, la durée pendant laquelle un canal particulier de la réplique multi-source est en retard par rapport à l'instance de base de données source. Pour de plus amples informations, veuillez consulter the section called “Surveillance des canaux de réplication multi-sources”.	MySQL	Secondes
`ReplicationSlotDiskUsage`	Espace disque utilisé par les fichiers d'emplacement de réplication.	PostgreSQL	Octets

Métrique	Description	S'applique à	Unités
`SwapUsage`	Quantité d'espace d'échange utilisé sur l'instance DB.	MariaDB MySQL Oracle PostgreSQL	Octets
`TempDbAvailableDataSpace`	La quantité d'espace de données disponible sur le tempdb et le volume où se trouve le tempdb. Utilisez cette métrique pour surveiller la disponibilité de l'espace de données tempdb et planifier la capacité en conséquence. Des valeurs faibles peuvent indiquer la nécessité d'augmenter le stockage ou d'optimiser les requêtes qui utilisent fortement tempdb.	SQL Server	Octets
`TempDbAvailableLogSpace`	La quantité d'espace journal disponible sur la base de données tempdb et le volume sur lequel se trouve la base de données tempdb. Utilisez cette métrique pour surveiller la disponibilité de l'espace dans le journal tempdb et empêcher que le journal des transactions ne soit plein. Essentiel pour les charges de travail comportant des transactions importantes ou une simultanéité élevée qui génèrent une activité de journal importante.	SQL Server	Octets

Métrique	Description	S'applique à	Unités
`TempDbDataFileUsage`	Pourcentage de fichiers de données utilisés sur la base de données tempdb. Cette métrique ne tient pas compte de la croissance potentielle des fichiers. Utilisez cette métrique pour surveiller l'utilisation des fichiers de données tempdb et identifier les problèmes de performance potentiels. Des valeurs élevées peuvent indiquer la nécessité d'optimiser les requêtes qui créent de grands objets temporaires ou augmentent la taille de la base de données tempdb.	SQL Server	Pourcentage
`TempDbLogFileUsage`	Pourcentage de fichiers journaux utilisés sur la base de données tempdb. Cette métrique ne tient pas compte de la croissance potentielle des fichiers. Utilisez cette métrique pour surveiller l'utilisation du fichier journal tempdb et éviter les problèmes de performances. Des valeurs élevées peuvent indiquer des transactions de longue durée ou une activité de journalisation excessive susceptible d'avoir un impact sur les performances globales de la base de données.	SQL Server	Pourcentage
`TransactionLogsDiskUsage`	Espace disque utilisé par les journaux de transactions.	PostgreSQL	Octets

Métrique	Description	S'applique à	Unités
`TransactionLogsGeneration`	Taille de journaux de transactions générés par seconde.	PostgreSQL	Octets par seconde
`WriteIOPS`	Nombre moyen d' I/O opérations d'écriture sur disque par seconde.	Tous	Nombre par seconde
`WriteIOPSLocalStorage`	Nombre moyen d' I/O opérations d'écriture sur disque par seconde sur le stockage local. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Nombre par seconde
`WriteIOPSLogVolume`	Nombre moyen d' I/O opérations d'écriture sur disque par seconde pour le volume du journal.	Instances de base de données avec volume de journal dédié activé	Nombre par seconde

Métrique	Description	S'applique à	Unités
`WriteLatency`	Durée moyenne par I/O opération sur le disque.	Tous	Secondes
`WriteLatencyLocalStorage`	Durée moyenne par I/O opération de disque sur le stockage local. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Secondes
`WriteLatencyLogVolume`	Durée moyenne par I/O opération sur disque pour le volume du journal.	Instances de base de données avec volume de journal dédié activé	Secondes
`WriteThroughput`	Nombre moyen d’octets écrits sur le disque par seconde.	Tous	Octets par seconde

Métrique	Description	S'applique à	Unités
`WriteThroughputLogVolume`	Nombre moyen d'octets écrits sur le disque par seconde pour le volume des journaux.	Instances de base de données avec volume de journal dédié activé	Octets par seconde
`WriteThroughputLocalStorage`	Nombre moyen d'octets écrits sur disque par seconde pour le stockage local. Cette métrique s'applique uniquement aux classes d'instances de base de données avec des volumes de stockage d'instance NVMe SSD. Pour plus d'informations sur les EC2 instances Amazon avec des volumes de stockage d'instance NVMe SSD, consultez la section Volumes de stockage d'instance. Les classes d'instance de base de données RDS équivalentes disposent des mêmes volumes de stockage d'instances. Par exemple, les classes d'instance de base de données db.m6gd et db.r6gd possèdent des volumes de stockage d'instance SSD. NVMe		Octets par seconde

Mesures CloudWatch d'utilisation d'Amazon pour Amazon RDS (Aurora)

L'espace de `AWS/Usage` noms d'Amazon CloudWatch inclut les mesures d'utilisation au niveau du compte pour vos quotas de service Amazon RDS. CloudWatch collecte automatiquement les statistiques d'utilisation pour tous Régions AWS.

Pour plus d'informations, consultez les statistiques CloudWatch d'utilisation dans le guide de CloudWatch l'utilisateur Amazon. Pour plus d'informations sur les quotas, consultez Quotas et contraintes pour Amazon RDS et Requesting a quota increase dans le Guide de l'utilisateur de Service Quotas.

Métrique	Description	Unités*
`AllocatedStorage`	Le stockage total pour toutes les instances de base de données. La somme exclut les instances de migration temporaire.	Gigaoctets
`AuthorizationsPerDBSecurityGroup`	Le nombre de règles d'entrée par groupe de sécurité de base de données dans votre Compte AWS. La valeur utilisée est le plus grand nombre de règles d'entrée dans un groupe de sécurité de base de données du compte. Les autres groupes de sécurité de base de données du compte peuvent avoir un nombre inférieur de règles d'entrée.	Nombre
`CustomEndpointsPerDBCluster`	Le nombre de points de terminaison personnalisés par cluster de base de données dans votre Compte AWS. La valeur utilisée est le plus grand nombre de points de terminaison personnalisés dans un cluster de base de données du compte. Les autres clusters de base de données du compte peuvent avoir un nombre inférieur de points de terminaison personnalisés.	Nombre
`CustomEngineVersions`	Le nombre de versions de moteur personnalisées (CEVs) pour Amazon RDS Custom dans votre Compte AWS.	Nombre
`DBClusterParameterGroups`	Nombre de groupes de paramètres de cluster de base de données dans votre Compte AWS. Le compte exclut les groupes de paramètres par défaut.	Nombre
`DBClusterRoles`	Le nombre de rôles AWS Identity and Access Management (IAM) associés par cluster de base de données dans votre Compte AWS. La valeur utilisée est le plus grand nombre de rôles IAM associés pour un cluster de base de données dans le compte. Les autres clusters de base de données	Nombre

Métrique	Description	Unités*
	du compte peuvent avoir un nombre inférieur de rôles IAM associés.	
`DBClusters`	Le nombre de clusters de base de données Amazon Aurora dans votre Compte AWS.	Nombre
`DBInstanceRoles`	Le nombre de rôles AWS Identity and Access Managemen t (IAM) associés par instance de base de données dans votre Compte AWS. La valeur utilisée est le plus grand nombre de rôles IAM associés pour une instance de base de données dans le compte. Les autres instances de base de données du compte peuvent avoir un nombre inférieur de rôles IAM associés.	Nombre
`DBInstances`	Le nombre d'instances de base de données dans votre Compte AWS.	Nombre
`DBParamet erGroups`	Le nombre de groupes de paramètres de base de données dans votre Compte AWS. Le compte exclut les groupes de paramètres de base de données par défaut.	Nombre
`DBSecurityGroups`	Le nombre de groupes de sécurité dans votre Compte AWS. Le compte exclut le groupe de sécurité par défaut et le groupe de sécurité VPC par défaut.	Nombre
`DBSubnetGroups`	Le nombre de groupes de sous-réseaux de base de données dans votre Compte AWS. Le compte exclut le groupe de sous-réseau par défaut.	Nombre
`EventSubs criptions`	Le nombre d'abonnements aux notifications d'événements enregistrés dans votre Compte AWS.	Nombre
`Integrations`	Le nombre d'intégrations zéro ETL avec Amazon Redshift dans votre. Compte AWS	Nombre

Métrique	Description	Unités*
`ManualClusterSnapshots`	Le nombre d'instantanés de cluster de base de données créés manuellement dans votre Compte AWS. Le compte exclut les instantanés non valides.	Nombre
`ManualSnapshots`	Le nombre d'instantanés de la base de données créés manuellement dans votre Compte AWS. Le compte exclut les instantanés non valides.	Nombre
`OptionGroups`	Le nombre de groupes d'options dans votre Compte AWS. Le compte exclut les groupes d'options par défaut.	Nombre
`Proxies`	Le nombre de proxys RDS présents dans votre AWS compte.	Nombre
`ReadReplicasPerMaster`	Le nombre de répliques de lecture par instance de base de données dans votre compte. La valeur utilisée est le plus grand nombre de répliques de lecture pour une instance de base de données dans le compte. Les autres instances de base de données du compte peuvent avoir un nombre inférieur de répliques de lecture.	Nombre
`ReservedDBInstances`	Le nombre d'instances réservées de la base de données dans votre Compte AWS. Le compte exclut les instances retirées ou déclinées.	Nombre
`SubnetsPerDBSubnetGroup`	Le nombre de sous-réseaux par groupe de sous-réseaux de base de données dans votre. Compte AWS Le plus grand nombre de sous-réseaux pour un groupe de sous-réseaux de base de données dans le compte. Les autres groupes de sous-réseaux de base de données du compte peuvent avoir un nombre inférieur de sous-réseaux.	Nombre

> **Note**
>
> Amazon RDS ne publie pas d'unités destinées aux statistiques d' CloudWatchutilisation. Les unités n'apparaissent que dans la documentation.

CloudWatch Dimensions Amazon pour Amazon RDS

Vous pouvez filtrer les données des métriques Amazon RDS en utilisant n'importe quelle dimension dans le tableau suivant.

Dimension	Filtre les données demandées pour . . .
`DBInstanceIdentifier`	Une instance de base de données spécifique.
`DatabaseClass`	Toutes les instances d'une classe de base de données. Par exemple, vous pouvez regrouper des métriques pour toutes les instances qui appartiennent à la classe de base de données `db.r5.large` .
`EngineName`	Le nom du moteur identifié uniquement. Par exemple, vous pouvez regrouper des métriques pour toutes les instances ayant le nom de moteur `postgres`.
`SourceRegion`	La région spécifiée uniquement. Par exemple, vous pouvez regrouper des métriques pour toutes les instances de base de données de la région `us-east-1` .

Statistiques Amazon CloudWatch pour Amazon RDS Performance Insights

Performance Insights publie automatiquement certains indicateurs sur Amazon CloudWatch. Les mêmes données peuvent être consultées à partir de Performance Insights, mais l'ajout des métriques CloudWatch facilite l'ajout CloudWatch d'alarmes. Cela permet également d'ajouter facilement les métriques aux CloudWatch tableaux de bord existants.

Métrique	Description
DBLoad	Nombre de sessions actives pour la base de données. Vous souhaitez généralement obtenir les données relatives au nombre moyen de sessions actives. Dans Performance Insights, ces données sont interrogées sous la forme `db.load.avg` .
DBLoadCPU	Nombre de sessions actives dans lesquelles le type d'événement d'attente est CPU (UC). Dans Performance Insights, ces données sont interrogées sous la forme `db.load.avg` , filtrées par le type d'événement d'attente `CPU`.
DBLoadNon processeur	Nombre de sessions actives dans lesquelles le type d'événement d'attente n'est pas CPU (UC).
DBLoadRelativeToNumVCPUs	Rapport entre la charge de base de données et le nombre de connexions virtuelles CPUs pour la base de données.

> **Note**
>
> Ces métriques ne sont publiées CloudWatch que si l'instance de base de données est chargée.

Vous pouvez examiner ces métriques à l'aide de la CloudWatch console, de ou de l' CloudWatch API. AWS CLI Vous pouvez également examiner d'autres indicateurs de mesure de Performance Insights à l'aide d'une fonction mathématique spéciale. Pour de plus amples informations, veuillez consulter Interroger d'autres indicateurs de compteur Performance Insights dans CloudWatch.

Par exemple, vous pouvez obtenir les statistiques de la `DBLoad` métrique en exécutant la get-metric-statisticscommande.

```
aws cloudwatch get-metric-statistics \
    --region us-west-2 \
    --namespace AWS/RDS \
    --metric-name DBLoad  \
    --period 60 \
    --statistics Average \
    --start-time 1532035185 \
    --end-time 1532036185 \
    --dimensions Name=DBInstanceIdentifier,Value=db-loadtest-0
```

Cet exemple génère une sortie similaire à la suivante.

```
{
  "Datapoints": [
  {
  "Timestamp": "2021-07-19T21:30:00Z",
  "Unit": "None",
  "Average": 2.1
  },
  {
  "Timestamp": "2021-07-19T21:34:00Z",
  "Unit": "None",
  "Average": 1.7
  },
  {
  "Timestamp": "2021-07-19T21:35:00Z",
  "Unit": "None",
  "Average": 2.8
  },
  {
  "Timestamp": "2021-07-19T21:31:00Z",
  "Unit": "None",
  "Average": 1.5
  },
  {
  "Timestamp": "2021-07-19T21:32:00Z",
  "Unit": "None",
  "Average": 1.8
  },
  {
  "Timestamp": "2021-07-19T21:29:00Z",
  "Unit": "None",
  "Average": 3.0
```

```
},
{
"Timestamp": "2021-07-19T21:33:00Z",
"Unit": "None",
"Average": 2.4
}
],
"Label": "DBLoad"
}
```

Pour plus d'informations CloudWatch, consultez Qu'est-ce qu'Amazon CloudWatch ? dans le guide de CloudWatch l'utilisateur Amazon.

Interroger d'autres indicateurs de compteur Performance Insights dans CloudWatch

> **Note**
>
> Si vous activez le mode avancé de Database Insights, Amazon RDS publie les indicateurs de performance Insights sur Amazon CloudWatch. Avec Database Insights, vous n'avez pas besoin d'utiliser la fonction mathématique `DB_PERF_INSIGHTS` métrique. Vous pouvez utiliser le tableau de bord CloudWatch Database Insights pour rechercher, interroger et définir des alarmes pour les compteurs de Performance Insights.

Vous pouvez effectuer des requêtes, créer des alarmes et créer des graphiques sur les métriques RDS Performance Insights à partir de CloudWatch. Vous pouvez accéder aux informations concernant votre de bases de données à l'aide de la fonction mathématique `DB_PERF_INSIGHTS` métrique pour CloudWatch. Cette fonction vous permet d'utiliser les métriques Performance Insights qui ne sont pas directement communiquées CloudWatch pour créer une nouvelle série chronologique.

Vous pouvez utiliser la nouvelle fonction Metric Math en cliquant sur le menu déroulant Ajouter des mathématiques dans l'écran Sélectionner une métrique de la CloudWatch console. Vous pouvez l'utiliser pour créer des alarmes et des graphiques sur les mesures Performance Insights ou sur des combinaisons de CloudWatch mesures Performance Insights, y compris des alarmes haute résolution pour les mesures inférieures à la minute. Vous pouvez également utiliser la fonction par programmation en incluant l'expression Metric Math dans une `get-metric-data`demande. Pour

plus d'informations, consultez Syntaxe et fonctions mathématiques des métriques et Créer une alarme sur les métriques de compteur Performance Insights à partir d'une AWS base de données.

Métrique de compteur de Performance Insights

Les métriques de compteur sont des métriques de performances de base de données et de système d'exploitation dans le tableau de bord Performance Insights. Vous pouvez établir des corrélations entre ces informations et la charge de la base de données pour identifier et analyser les problèmes de performances. Vous devez ajouter une fonction statistique à la métrique pour obtenir les valeurs de la métrique. Par exemple, les fonctions prises en charge pour la métrique `os.memory.active` sont `.avg`, `.min`, `.max`, `.sum` et `.sample_count`.

Les métriques du compteur sont collectées une fois par minute. La collecte des métriques du système d'exploitation dépend de l'activation ou de la désactivation de la surveillance améliorée. Si la surveillance améliorée est désactivée, les métriques du système d'exploitation sont collectées une fois par minute. Si la surveillance améliorée est activée, les métriques du système d'exploitation sont collectées pour la période sélectionnée. Pour plus d'informations sur l'activation et la désactivation de la surveillance améliorée, consultez Activer et désactiver la surveillance améliorée.

Rubriques

- Compteurs de système d'exploitation Performance Insights
- Compteurs Performance Insights pour Amazon RDS for MariaDB et MySQL
- Compteurs Performance Insights pour Amazon RDS for Microsoft SQL Server
- Compteurs Performance Insights pour Amazon RDS for Oracle
- Compteurs Performance Insights pour Amazon RDS for PostgreSQL

Compteurs de système d'exploitation Performance Insights

Les compteurs des systèmes d'exploitation suivants, dont le préfixe est `os`, sont disponibles avec la fonctionnalité Analyse des performances pour tous les moteurs RDS à l'exception de RDS for SQL Server.

Vous pouvez utiliser l'API `ListAvailableResourceMetrics` pour obtenir la liste des métriques de compteur disponibles pour votre instance de base de données. Pour plus d'informations, consultez ListAvailableResourceMetricsle guide de référence des API Amazon RDS Performance Insights.

Compteur	Type	Unité	Métrique	Description
Actif	Mémoire	Kilooctets	os.memory .active	Quantité de mémoire attribuée, en kilo-octets.
Tampons	Mémoire	Kilooctets	os.memory .buffers	Quantité de mémoire utilisée pour la mise en mémoire tampon des I/O demandes avant l'écriture sur le périphérique de stockage, en kilo-octets.
Mis en cache	Mémoire	Kilooctets	os.memory .cached	Quantité de mémoire utilisée pour la mise en cache des E/S basées sur le système de fichiers, en kilo-octets.
Cache de base de données	Mémoire	Octets	os.memory .db.cache	Quantité de mémoire utilisée pour le cache de pages par le processus de base de données, y compris tmpfs (shmem), en octets.

Compteur	Type	Unité	Métrique	Description
Taille de résident défini de base de données	Mémoire	Octets	os.memory.db.residentSetSize	Quantité de mémoire utilisée pour le cache anonyme et d'échange par le processus de base de données, sans inclure tmpfs (shmem), en octets.
Échange de base de données	Mémoire	Octets	os.memory.db.swap	Quantité de mémoire utilisée pour l'échange par le processus de base de données, en octets.
Non intègre	Mémoire	Kilooctets	os.memory.dirty	Quantité de pages mémoire de la RAM ayant été modifiées mais non écrites dans le bloc de données associé dans le stockage, en kilo-octets.
Free	Mémoire	Kilooctets	os.memory.free	Quantité de mémoire non attribuée, en kilo-octets.

Compteur	Type	Unité	Métrique	Description
Grandes pages gratuites	Mémoire	Pages	os.memory.hugePagesFree	Nombre de grandes pages gratuites. Les grandes pages sont une fonction du noyau Linux.
Grandes pages Rsvd	Mémoire	Pages	os.memory.hugePagesRsvd	Nombre de grandes pages dédiées.
Taille des grandes pages	Mémoire	Kilooctets	os.memory.hugePagesSize	Taille de chaque unité de grandes pages, en kilo-octets.
Grandes pages excéd	Mémoire	Pages	os.memory.hugePagesSurp	Nombre de grandes pages excédentaires disponibles par rapport au nombre total.
Total de grandes pages	Mémoire	Pages	os.memory.hugePagesTotal	Nombre total de grandes pages.
Inactif	Mémoire	Kilooctets	os.memory.inactive	Quantité de pages mémoire moins fréquemment utilisées, en kilo-octets.

Compteur	Type	Unité	Métrique	Description
Mappé	Mémoire	Kilooctets	os.memory .mapped	Quantité totale de contenu du système de fichiers mappé en mémoire dans un espace d'adressage de processus, en kilo-octets.
Nombre d'arrêts de mémoire insuffisante	Mémoire	tue	os.memory. outOfMemo ryKillCount	Nombre d'arrêts de mémoire insuffisante survenus au cours du dernier intervalle de collecte.
Tables de pages	Mémoire	Kilooctets	os.memory .pageTables	Quantité de mémoire utilisée par les tables de page, en kilo-octets.
Section	Mémoire	Kilooctets	os.memory.slab	Quantité de structures de données noyau réutilisables, en kilo-octets.
Total	Mémoire	Kilooctets	os.memory.total	Quantité totale de mémoire, en kilo-octets.

Compteur	Type	Unité	Métrique	Description
Écriture différée	Mémoire	Kilooctets	os.memory .writeback	Quantité de pages de modification dans la RAM encore écrites dans le stockage de sauvegarde, en kilo-octets.
Invité	Utilisation de l'UC	Pourcentage	os.cpuUtilization. guest	Pourcentage de l'UC en cours d'utilisation par les programmes invités.
Inactif	Utilisation de l'UC	Pourcentage	os.cpuUtilization. idle	Pourcentage de l'UC inactive.
Irq	Utilisation de l'UC	Pourcentage	os.cpuUtilization. irq	Pourcentage de l'UC en cours d'utilisation par les interruptions logicielles.
Nice	Utilisation de l'UC	Pourcentage	os.cpuUtilization. nice	Pourcentage de l'UC en cours d'utilisation par des programmes s'exécutant avec la priorité la plus faible.

Compteur	Type	Unité	Métrique	Description
Steal	Utilisation de l'UC	Pourcentage	os.cpuUtilization.steal	Pourcentage de l'UC en cours d'utilisation par d'autres machines virtuelles.
Système	Utilisation de l'UC	Pourcentage	os.cpuUtilization.system	Pourcentage de l'UC en cours d'utilisation par le noyau.
Total	Utilisation de l'UC	Pourcentage	os.cpuUtilization.total	Pourcentage total de l'UC en cours d'utilisation. Cette valeur inclut la valeur Nice.
Utilisateur	Utilisation de l'UC	Pourcentage	os.cpuUtilization.user	Pourcentage de l'UC en cours d'utilisation par des programmes utilisateurs.
Attente	Utilisation de l'UC	Pourcentage	os.cpuUtilization.wait	Pourcentage de CPU inutilisé pendant l'attente de I/O l'accès.
Lisez IOs PS	E/S du disque	Demandes par seconde	Système d'exploitation Diskio.<devicename>.lire PS IOs	Nombre d'opérations de lecture par seconde.

Compteur	Type	Unité	Métrique	Description
Écrire IOs PS	E/S du disque	Demandes par seconde	Système d'exploitation Diskio.<devicename>.write PS IOs	Nombre d'opérations d'écriture par seconde.
Longueur file d'attente moyenne	E/S du disque	Requêtes	Système d'exploitation Diskio.<devicename>.avgQueueLen	Le nombre de demandes en attente dans la file d'attente de l'I/O appareil.
Taille demande moyenne	E/S du disque	Requêtes	Système d'exploitation Diskio.<devicename>.avgReqSz	Le nombre de demandes en attente dans la file d'attente de l'I/O appareil.
En attente	E/S du disque	Millisecondes	os.diskIO.<nom_périphérique>.await	Nombre de millisecondes requises pour répondre aux requêtes, y compris le temps d'attente et le temps de service.

Compteur	Type	Unité	Métrique	Description
Lisez IOs PS	E/S du disque	Requêtes	Système d'exploitation Diskio.<devicename>.lire PS IOs	Nombre d'opérations de lecture par seconde.
Ko de lecture	E/S du disque	Kilooctets	os.diskIO.<nom_périphérique>.readKb	Nombre total de kilo-octets lus.
PS de Ko de lecture	E/S du disque	Kilo-octets par seconde	os.diskIO.<nom_périphérique>.readKbPS	Nombre de kilo-octets lus par seconde.
PS Rrqm	E/S du disque	Demandes par seconde	os.diskIO.<nom_périphérique>.rrqmPS	Nombre de requêtes de lecture fusionnées mises en file d'attente par seconde.
TPS	E/S du disque	Transactions par seconde	os.diskIO.<nom_périphérique>.tps	Le nombre de I/O transactions par seconde.
Utilitaire	E/S du disque	Pourcentage	os.diskIO.<nom_périphérique>.util	Pourcentage de temps UC pendant lequel les requêtes ont été émises.

Compteur	Type	Unité	Métrique	Description
Ko d'écriture	E/S du disque	Kilooctets	os.diskIO .<nom_périphérique>.writeKb	Nombre total de kilo-octets écrits.
PS Ko d'écriture	E/S du disque	Kilo-octets par seconde	os.diskIO .<nom_périphérique>.writeKbPS	Nombre de kilo-octets écrits par seconde.
PS Wrqm	E/S du disque	Demandes par seconde	os.diskIO .<nom_périphérique>.wrqmPS	Nombre de requêtes d'écriture fusionnées mises en file d'attente par seconde.
Bloqué	Tâches	Tâches	os.tasks.blocked	Nombre de tâches bloquées.
En cours d'exécution	Tâches	Tâches	os.tasks.running	Nombre de tâches en cours d'exécution.
En veille	Tâches	Tâches	os.tasks.sleeping	Nombre de tâches en veille.
Arrêté(e)	Tâches	Tâches	os.tasks.stopped	Nombre de tâches arrêtées.
Total	Tâches	Tâches	os.tasks.total	Nombre total de tâches.

Compteur	Type	Unité	Métrique	Description
Zombie	Tâches	Tâches	os.tasks.zombie	Nombre de tâches enfant inactives avec une tâche parent active.
Un	Minute moyenne de charge	Processus	os. loadAverageMinute.un	Nombre de processus demandant du temps UC au cours de la dernière minute.
Quinze	Minute moyenne de charge	Processus	os. loadAverageMinute.quinze	Nombre de processus demandant du temps UC au cours des 15 dernières minutes.
Cinq	Minute moyenne de charge	Processus	os. loadAverageMinute.cinq	Nombre de processus demandant du temps UC au cours des 5 dernières minutes.

Compteur	Type	Unité	Métrique	Description
Mis en cache	Swap	Kilooctets	os.swap.cached	Quantité de mémoire d'échange, en kilo-octets, utilisée en tant que mémoire cache.
Free	Swap	Kilooctets	os.swap.free	Quantité de mémoire d'échange libre, en kilo-octets.
Entrée	Swap	Kilooctets	os.swap.in	Quantité de mémoire, en kilo-octets, échangée depuis le disque.
Sortie	Swap	Kilooctets	os.swap.out	Quantité de mémoire, en kilo-octets, échangée vers le disque.
Total	Swap	Kilooctets	os.swap.total	Quantité totale de mémoire d'échange disponible, en kilo-octets.
Nombre maximum de fichiers	Système de fichiers	Dépôt de	os.fileSys.maxFiles	Nombre maximum de fichiers pouvant être créés pour le système de fichiers.

Compteur	Type	Unité	Métrique	Description
Fichiers utilisés	Système de fichiers	Dépôt de	os.fileSys.usedFiles	Nombre de fichiers dans le système de fichiers.
Pourcentage de fichiers utilisés	Système de fichiers	Dépôt de	Système d'exploitation .FileSys.usedFilePercent	Pourcentage de fichiers disponibles en cours d'utilisation.
Pourcentage utilisé	Système de fichiers	Pourcentage	os.fileSys.usedPercent	Pourcentage d'espace de disque du système de fichiers en cours d'utilisation.
Utilisé	Système de fichiers	Kilooctets	os.fileSys.used	Quantité d'espace disque utilisé par des fichiers du système de fichiers, en kilo-octets.
Total	Système de fichiers	Kilooctets	os.fileSys.total	Quantité totale d'espace disque disponible pour le système de fichiers, en kilo-octets.
Rx	Réseau	Octets par seconde	os.network.rx	Nombre d'octets reçus par seconde.

Compteur	Type	Unité	Métrique	Description
Tx	Réseau	Octets par seconde	os.network.tx	Nombre d'octets téléchargés par seconde.
Utilisation d'ACU	Général	Pourcentage	os.general.acuUtilization	Pourcentage de la capacité actuelle par rapport à la capacité maximale configurée.
ACU configurée max.	Général	ACUs	os.general.maxConfiguredAcu	Capacité maximale configurée par l'utilisateur, en unités de capacité Aurora (ACUs).
ACU configurée min.	Général	ACUs	os.general.minConfiguredAcu	La capacité minimale configurée par l'utilisateur, en ACUs.
Num VCPUs	Général	v CPUs	os.general.numVCPUs	Le nombre de virtual CPUs (vCPUs) pour l'instance de base de données.

Compteur	Type	Unité	Métrique	Description
Capacité de base de données sans serveur	Général	ACUs	os.general.serverlessDatabaseCapacity	La capacité actuelle de l'instance, en ACUs.

Compteurs Performance Insights pour Amazon RDS for MariaDB et MySQL

Les compteurs de base de données suivants sont disponibles avec Performance Insights pour Amazon RDS for MariaDB et MySQL.

Rubriques

- Compteurs natifs pour RDS for MariaDB et RDS for MySQL
- Compteurs non natifs pour Amazon RDS for MariaDB et MySQL

Compteurs natifs pour RDS for MariaDB et RDS for MySQL

Les métriques natives sont définies par le moteur de base de données et non par Amazon RDS. Pour les définitions de ces métriques natives, voir Variables d'état du serveur (pour 8.0) et Variables d'état du serveur (pour 8.4) dans la documentation MySQL.

Compteur	Type	Unité	Métrique
Com_analyze	SQL	Requêtes par seconde	db.SQL.Com_analyze
Com_optimize	SQL	Requêtes par seconde	db.SQL.Com_optimize
Com_select	SQL	Requêtes par seconde	db.SQL.Com_select
Connexions	SQL	Nombre de tentatives de connexion par minute	db.Users.Connections

Compteur	Type	Unité	Métrique
		(réussies ou non) au serveur MySQL	
Innodb_rows_deleted	SQL	Lignes par seconde	db.SQL.Innodb_rows_deleted
Innodb_rows_inserted	SQL	Lignes par seconde	db.SQL.Innodb_rows_inserted
Innodb_rows_read	SQL	Lignes par seconde	db.SQL.Innodb_rows_read
Innodb_rows_updated	SQL	Lignes par seconde	db.SQL.Innodb_rows_updated
Select_full_join	SQL	Requêtes par seconde	db.SQL.Select_full_join
Select_full_range_join	SQL	Requêtes par seconde	db.SQL.Select_full_range_join
Select_range	SQL	Requêtes par seconde	db.SQL.Select_range
Select_range_check	SQL	Requêtes par seconde	db.SQL.Select_range_check
Select_scan	SQL	Requêtes par seconde	db.SQL.Select_scan
Slow_queries	SQL	Requêtes par seconde	db.SQL.Slow_queries
Sort_merge_passes	SQL	Requêtes par seconde	db.SQL.Sort_merge_passes

Compteur	Type	Unité	Métrique
Sort_range	SQL	Requêtes par seconde	db.SQL.Sort_range
Sort_rows	SQL	Requêtes par seconde	db.SQL.Sort_rows
Sort_scan	SQL	Requêtes par seconde	db.SQL.Sort_scan
Questions	SQL	Requêtes par seconde	db.SQL.Questions
Innodb_row_lock_time	Locks	Milliseconds (moyenne)	db.Locks.Innodb_row_lock_time
Table_locks_immediate	Locks	Demandes par seconde	db.Locks.Table_locks_immediate
Table_locks_waited	Locks	Demandes par seconde	db.Locks.Table_locks_waited
Aborted_clients	Users	Connexions	db.Users.Aborted_clients
Aborted_connects	Users	Connexions	db.Users.Aborted_connects
max_connections	Users	Connexions	db.User.max_connections
Threads_created	Users	Connexions	db.Users.Threads_created
Threads_running	Users	Connexions	db.Users.Threads_running
Innodb_data_writes	I/O	Opérations par seconde	db.IO.Innodb_data_writes
Innodb_dblwr_writes	I/O	Opérations par seconde	db.IO.Innodb_dblwr_writes

Compteur	Type	Unité	Métrique
Innodb_log_write_requests	I/O	Opérations par seconde	db.IO.Innodb_log_write_requests
Innodb_log_writes	I/O	Opérations par seconde	db.IO.Innodb_log_writes
Innodb_pages_written	I/O	Pages par seconde	db.IO.Innodb_pages_written
Created_tmp_disk_tables	Temp	Tables par seconde	db.Temp.Created_tmp_disk_tables
Created_tmp_tables	Temp	Tables par seconde	db.Temp.Created_tmp_tables
Innodb_buffer_pool_pages_data	Cache	Pages	db.Cache.Innodb_buffer_pool_pages_data
Innodb_buffer_pool_pages_total	Cache	Pages	db.Cache.Innodb_buffer_pool_pages_total
Innodb_buffer_pool_read_requests	Cache	Pages par seconde	db.Cache.Innodb_buffer_pool_read_requests
Innodb_buffer_pool_reads	Cache	Pages par seconde	db.Cache.Innodb_buffer_pool_reads
Opened_tables	Cache	Tables	db.Cache.Opened_tables
Opened_table_definitions	Cache	Tables	db.Cache.Opened_table_definitions
Qcache_hits	Cache	Requêtes	db.Cache.Qcache_hits

Compteurs non natifs pour Amazon RDS for MariaDB et MySQL

Les métriques de compteur non natif sont des compteurs définis par Amazon RDS. Une métrique non native peut être obtenue avec une requête spécifique. Il peut également s'agir d'une métrique

dérivée, pour laquelle deux compteurs natifs ou plus sont utilisés dans les calculs de rapport, de taux d'accès ou de latences.

Compteur	Type	Unité	Métrique	Description	Définition
innodb_buffer_pool_hits	Cache	lit	db.Cache.innoDB_buffer_pool_hits	Nombre de lectures pouvant être réalisées par InnoDB à partir du pool de mémoires tampons.	`innodb_buffer_pool_read_requests - innodb_buffer_pool_reads`
innodb_buffer_pool_hit_rate	Cache	Pourcentage	db.Cache.innoDB_buffer_pool_hit_rate	Pourcentage de lectures pouvant être réalisées par InnoDB à partir du pool de mémoires tampons.	`100 * innodb_buffer_pool_read_requests / (innodb_buffer_pool_read_requests + innodb_buffer_pool_reads)`
innodb_buffer_pool_usage	Cache	Pourcentage	db.Cache.innoDB_buffer_pool_usage	Pourcentage du pool de mémoires tampons Inn	`Innodb_buffer_pool_pages_data / Innodb_buffer_pool`

Compteur	Type	Unité	Métrique	Description	Définition
				contenant des données (pages).	`_pages_total * 100.0`
				Note Cette valeur peut varier lors de l'utilisation de tables compr es. Pour plus d'inform tions, consul les inform ons sur `Innod ffer_ _page ta` et `Innod`	

Compteur	Type	Unité	Métrique	Description	Définition
				ffer_page tal dans les variabl d'état du serveu (pour 8.0) et les variabl d'état du serveu (pour 8.4) dans la docum tion MySQl	
query_cache_hit_rate	Cache	Pourcentage	db.Cache.query_cache_hit_rate	Taux d'accès au cache (de requête) de l'ensemble de résultats My	`Qcache_hits / (QCache_hits + Com_select) * 100`

Compteur	Type	Unité	Métrique	Descripti on	Définition
innodb_datafile_wr ites_to_disk	I/O	écrit	db.IO.inn oDB_dataf ile_writes_to_disk	Nombre d'écritur es de fichier de données Inn sur le disque, sans compter les opération s double write et redo logging write.	`Innodb_da ta_writes - Innodb_lo g_writes - Innodb_db lwr_writes`
innodb_ro ws_changed	SQL	Lignes	db.SQL.in nodb_rows _changed	Nombre total d'opérati ons de ligne InnoDE	`db.SQL.In nodb_rows _inserted + db.SQL.In nodb_rows _deleted + db.SQL. Innodb_ro ws_updated`
active_transactions	Transacti ons	Transacti ons	db.Transa ctions.active_tran sactions	Nombre total de transacti ons actives.	`SELECT COUNT(1) AS active_tr ansactions FROM INFOR MATION_SC HEMA.INNO DB_TRX`

Compteur	Type	Unité	Métrique	Description	Définition
trx_rseg_history_len	Transactions	Aucun	db.Transactions.trx_rseg_history_len	Liste des pages du journal des annulations pour les transactions validées qui est gérée par le système de transactions InnoDB pour implémenter le contrôle de simultanéité multiversion. Pour plus d'informations sur les détails des enregistr	`SELECT COUNT AS trx_rseg_history_len FROM INFORMATION_SCHEMA.INNODB_METRICS WHERE NAME='trx_rseg_history_len'`

Compteur	Type	Unité	Métrique	Description	Définition
				ements du journal d'annulation, consultez InnoDB Multi-Versioning (pour 8.0) et InnoDB Multi-Versioning (pour 8.4) dans la documentation MySQL.	
innodb_deadlocks	Locks	Locks	db.Locks.innodb_deadlocks	Nombre total de blocages.	`SELECT COUNT AS innodb_deadlocks FROM INFORMATION_SCHEMA.INNODB_METRICS WHERE NAME='lock_deadlocks'`

Compteur	Type	Unité	Métrique	Description	Définition
innodb_lock_timeouts	Locks	Locks	db.Locks.innodb_lock_timeouts	Nombre total de verrous qui ont expiré.	`SELECT COUNT AS innodb_lock_timeouts FROM INFORMATION_SCHEMA.INNODB_METRICS WHERE NAME='lock_timeouts'`
innodb_row_lock_waits	Locks	Locks	db.Locks.innodb_row_lock_waits	Nombre total de verrouillages de ligne ayant entraîné une attente.	`SELECT COUNT AS innodb_row_lock_waits FROM INFORMATION_SCHEMA.INNODB_METRICS WHERE NAME='lock_row_lock_waits'`

Compteurs Performance Insights pour Amazon RDS for Microsoft SQL Server

Les compteurs de base de données suivants sont disponibles avec Performance Insights pour RDS for Microsoft SQL Server.

Compteurs natifs pour RDS for Microsoft SQL Server

Les métriques natives sont définies par le moteur de base de données et non par Amazon RDS. Vous trouverez les définitions de ces métriques natives dans Utilisation des objets SQL Server dans la documentation Microsoft SQL Server.

Compteur	Type	Unité	Métrique
Demandes transmises	Méthodes d'accès	Enregistrements par seconde	db.Access Methods.Forwarded Records
Fractionnements de page	Méthodes d'accès	Fractionnements par seconde	db.Access Methods.Page Splits
Buffer cache hit ratio	Gestionnaire des buffers	Ratio	db.Buffer Manager.Buffer cache hit ratio
Espérance de vie des pages	Gestionnaire des buffers	Espérance en secondes	db.Buffer Manager.Page life expectancy
Recherches de page	Gestionnaire des buffers	Recherches par seconde	db.Buffer Manager.Page lookups
Lectures de page	Gestionnaire des buffers	Lectures par seconde	db.Buffer Manager.Page reads
Écritures de page	Gestionnaire des buffers	Écritures par seconde	db.Buffer Manager.Page writes
Transactions actives	Bases de données	Transactions	db.Databases.Active Transactions (_Total)
Octets de journal vidés	Bases de données	Octets vidés par seconde	db.Databases.Log Bytes Flushed (_Total)
Attentes de vidage de journal	Bases de données	Attentes par seconde	db.Databases.Log Flush Waits (_Total)
Vidages de journal	Bases de données	Vidages par seconde	db.Databases.Log Flushes (_Total)
Transactions en écriture	Bases de données	Transactions par seconde	db.Databases.Write Transactions (_Total)

Compteur	Type	Unité	Métrique
Processus bloqués	Statistiques générales	Processus bloqués	db.General Statistics.Processes blocked
Connexions utilisateur	Statistiques générales	Connexions	db.General Statistics.User Connections
Attentes de verrou	Verrous	Attentes par seconde	db.Latches.Latch Waits
Nombre total de verrous	Locks	Verrous par minute	db.Locks.Number of Deadlocks (_Total)
Attributions mémoire en attente	Gestionnaire de la mémoire	Attributions mémoire	db.Memory Manager.Memory Grants Pending
Demandes par lots	Statistiques SQL	Demandes par seconde	db.SQL Statistics.Batch Requests
Compilations SQL	Statistiques SQL	Compilations par seconde	db.SQL Statistics.SQL Compilations
Recompilations SQL	Statistiques SQL	Recompilations par seconde	db.SQL Statistics.SQL Re-Compilations

Compteurs Performance Insights pour Amazon RDS for Oracle

Les compteurs de base de données suivants sont disponibles avec Performance Insights pour RDS Oracle.

Compteurs natifs pour RDS for Oracle

Les métriques natives sont définies par le moteur de base de données et non par Amazon RDS. La section Statistics Descriptions de la documentation Oracle fournit les définitions de ces métriques natives.

> **Note**
>
> Pour la métrique de compteur `CPU used by this session`, l'unité a été transformée des centisecondes natives en sessions actives pour simplifier l'utilisation de la valeur. Par exemple, dans le graphique de la charge de travail de base de données, « CPU send » représente la demande en UC. La métrique de compteur `CPU used by this session` représente la quantité d'UC utilisée par les sessions Oracle. Vous pouvez comparer la valeur de « CPU send » à la métrique de compteur `CPU used by this session`. Lorsque la demande en UC est supérieure à la quantité d'UC utilisée, les sessions sont en attente du temps UC.

Compteur	Type	Unité	Métrique
Quantité d'UC utilisée par cette session	User	Sessions actives	Quantité d'UC utilisée par cette session
Client aller-retour SQL*Net to/from	Utilisateur	Allers-retours par seconde	Client aller-retour DB.User.SQL*Net to/from
Bytes received via SQL*Net from client	User	Octets par seconde	db.User.bytes received via SQL*Net from client
User commits	User	Validations par seconde	db.User.user commits
Logons cumulative	User	Connexions par seconde	db.User.logons cumulative
User calls	User	Appels par seconde	db.User.user calls
Bytes sent via SQL*Net to client	User	Octets par seconde	db.User.bytes sent via SQL*Net to client
User rollbacks	User	Restaurations par seconde	db.User.user rollbacks

Compteur	Type	Unité	Métrique
Redo size	Redo	Octets par seconde	db.Redo.redo size
Parse count (total)	SQL	Analyses par seconde	db.SQL.parse count (total)
Parse count (hard)	SQL	Analyses par seconde	db.SQL.parse count (hard)
Table scan rows gotten	SQL	Lignes par seconde	db.SQL.table scan rows gotten
Sorts (memory)	SQL	Tris par seconde	db.SQL.sorts (memory)
Sorts (disk)	SQL	Tris par seconde	db.SQL.sorts (disk)
Sorts (rows)	SQL	Tris par seconde	db.SQL.sorts (rows)
Physical read bytes	Cache	Octets par seconde	db.Cache.physical read bytes
DB block gets	Cache	Blocs par seconde	db.Cache.db block gets
DBWR checkpoints	Cache	Points de contrôle par minute	db.Cache.DBWR checkpoints
Physical reads	Cache	Lectures par seconde	db.Cache.physical reads
Consistent gets from cache	Cache	Obtentions par seconde	db.Cache.consistent gets from cache
DB block gets from cache	Cache	Obtentions par seconde	db.Cache.db block gets from cache
Consistent gets	Cache	Obtentions par seconde	db.Cache.consistent gets

Compteurs Performance Insights pour Amazon RDS for PostgreSQL

Les compteurs de base de données suivants sont disponibles avec Performance Insights pour Amazon RDS for PostgreSQL.

Rubriques

- Compteurs natifs pour Amazon RDS for PostgreSQL
- Compteurs non natifs pour Amazon RDS for PostgreSQL

Compteurs natifs pour Amazon RDS for PostgreSQL

Les métriques natives sont définies par le moteur de base de données et non par Amazon RDS. La section Viewing Statistics de la documentation PostgreSQL fournit les définitions de ces métriques natives.

Compteur	Type	Unité	Métrique
blks_hit	Cache	Blocs par seconde	db.Cache.blks_hit
buffers_alloc	Cache	Blocs par seconde	db.Cache.buffers_alloc
buffers_checkpoint	Checkpoint	Blocs par seconde	db.Checkpoint.buffers_checkpoint
checkpoint_sync_time	Checkpoint	Millisecondes par point de contrôle	db.Checkpoint.checkpoint_sync_time
checkpoint_write_time	Checkpoint	Millisecondes par point de contrôle	db.Checkpoint.checkpoint_write_time
checkpoints_req	Checkpoint	Points de contrôle par minute	db.Checkpoint.checkpoints_req
checkpoints_timed	Checkpoint	Points de contrôle par minute	db.Checkpoint.checkpoints_timed
maxwritten_clean	Checkpoint	Arrêts de nettoyage Bgwriter par minute	db.Checkpoint.maxwritten_clean

Compteur	Type	Unité	Métrique
deadlocks	Concurrency	Blocages par minute	db.Concurrency.deadlocks
blk_read_time	I/O	Millisecondes	db.IO.blk_read_time
blks_read	I/O	Blocs par seconde	db.IO.blks_read
buffers_backend	I/O	Blocs par seconde	db.IO.buffers_backend
buffers_backend_fsync	I/O	Blocs par seconde	db.IO.buffers_backend_fsync
buffers_clean	I/O	Blocs par seconde	db.IO.buffers_clean
tup_deleted	SQL	Tuples par seconde	db.SQL.tup_deleted
tup_fetched	SQL	Tuples par seconde	db.SQL.tup_fetched
tup_inserted	SQL	Tuples par seconde	db.SQL.tup_inserted
tup_returned	SQL	Tuples par seconde	db.SQL.tup_returned
tup_updated	SQL	Tuples par seconde	db.SQL.tup_updated
temp_bytes	Temp	Octets par seconde	db.Temp.temp_bytes
temp_files	Temp	Fichiers par minute	db.Temp.temp_files
xact_commit	Transactions	Validations par seconde	db.Transactions.xact_commit

Compteur	Type	Unité	Métrique
xact_rollback	Transactions	Restaurations par seconde	db.Transactions.xact_rollback
numbackends	User	Connexions	db.User.numbackends
archived_count	Journal WAL (Write-ahead log)	Fichiers par minute	db.WAL.archived_count

Compteurs non natifs pour Amazon RDS for PostgreSQL

Les métriques de compteur non natif sont des compteurs définis par Amazon RDS. Une métrique non native peut être obtenue avec une requête spécifique. Il peut également s'agir d'une métrique dérivée, pour laquelle deux compteurs natifs ou plus sont utilisés dans les calculs de rapport, de taux d'accès ou de latences.

Compteur	Type	Unité	Métrique	Description	Définition
checkpoint_sync_latency	Checkt	db.Checkpoint.checkpoint_sync_latency		Durée totale consacrée à la partie du traitement de point de contrôle où les fichiers sont synchronisés sur le disque.	`checkpoint_sync_time / (checkpoints_timed + checkpoints_req)`
checkpoint_write_latency	Checkt	db.Checkpoint.checkpoint_write_latency	Durée totale consacrée à la partie du traitement de point de contrôle où les fichiers sont écrits sur le disque.	`checkpoint_write_time / (checkpoints_timed + checkpoints_req)`	

Compteur	Type	Unité	Métrique	Description	Définition
read_latency	I/O	db.IO.r d_later	Durée consacrée à la lecture des blocs de fichier de données par les backends dans cette instance.	`blk_read_time / blks_read`	
idle_in_transaction_aborted_count	État	Séanc	db.state.idle_in_transaction_aborted_count	Le nombre de sessions dans l'`idle in transaction (aborted)` État.	Ne s'applique pas
idle_in_transaction_count	État	Séanc	db.state.idle_in_transaction_count	Le nombre de sessions dans l'`idle in transaction` État.	Ne s'applique pas
idle_in_transaction_max_time	État	Secon	db.state.idle_in_transaction_max_time	Durée de la transaction la plus longue de l'`idle in transaction` État, en secondes.	Ne s'applique pas
active_transactions	Transa ons	Transa ons	db.Transactions.active_transactions	Le nombre de transactions actives.	Ne s'applique pas
blocked_transactions	Transa ons	Transa ons	db.Transactions.blocked_transactions	Le nombre de transactions bloquées.	Ne s'applique pas

Compteur	Type	Unité	Métrique	Description	Définition
oldest_active_logical_replication_slot_xid_age	Transactions	DB.Transactions.oldest_Active_Logical_Replication_Slot_XID_A	L'âge de la transaction la plus ancienne dans un slot de réplication logique actif. Pour de plus amples informations, veuillez consulter Emplacement de réplication logique.	–	
oldest_inactive_logical_replication_slot_xid_age	Transactions	DB.Transactions.OLDEST_Active_Logical_Replication_Slot_XID	L'âge de la transaction la plus ancienne dans un slot de réplication logique inactif. Pour de plus amples informations, veuillez consulter Emplacement de réplication logique.	–	
oldest_prepared_transaction_xid_age	Transactions	DB.Transactions.OLDEST_PREPARED_TRANSACTION_XID_AG	L'âge de la plus ancienne transaction préparée. Pour de plus amples informations, veuillez consulter Transaction préparée.	–	

Compteur	Type	Unité	Métrique	Description	Définition
oldest_running_transaction_xid_age	Transactions	DB.Transactions.DEST_ing_Transaction_D_AGE	L'âge de la plus ancienne transaction en cours. Pour plus d'informations, consultez la section Déclaration active pour la plus ancienne transaction active en cours et État Idle in transaction (Transaction inactive) pour la plus ancienne en cours d'exécution idle-in-transaction.	–	
oldest_hot_standby_feedback_xid_age	Transactions	DB.Transactions.DEST_STANDEEDBID_AG	L'âge de la plus ancienne transaction en cours sur une réplique en lecture lorsque `hot_standby_feedback` cette option est activée. Pour de plus amples informations, veuillez consulter Réplicas en lecture.	–	
max_used_xact_ids	Transactions	Transactions	db.Transactions.max_used_xact_ids	Le nombre de transactions qui n'ont pas été supprimées.	Ne s’applique pas

Compteur	Type	Unité	Métrique	Description	Définition
max_connections	Users	Connes	db.User.max_connections	Le nombre maximum de connexions autorisées pour une instance de base de données tel que configuré dans le `max_connections` paramètre.	Ne s’applique pas
archive_failed_count	WAL	Fichier par minute	db.WAL.archive_failed_count	Nombre de tentatives infructueuses d'archivage de fichiers WAL, en fichiers par minute.	Ne s’applique pas

Statistiques SQL pour Performance Insights

Les statistiques SQL sont des métriques liées aux performances des requêtes SQL qui sont collectées par Performance Insights. Performance Insights collecte des statistiques pour chaque seconde d'exécution d'une requête et pour chaque appel SQL. Les statistiques SQL sont une moyenne pour la plage de temps sélectionnée.

Un récapitulatif SQL est un composite de toutes les requêtes ayant un modèle donné mais n'ayant pas nécessairement les mêmes valeurs littérales. Le récapitulatif remplace les valeurs littérales par un point d'interrogation. Par exemple, `SELECT * FROM emp WHERE lname= ?`. Ce récapitulatif peut inclure les requêtes enfant suivantes :

```
SELECT * FROM emp WHERE lname = 'Sanchez'
SELECT * FROM emp WHERE lname = 'Olagappan'
SELECT * FROM emp WHERE lname = 'Wu'
```

Tous les moteurs prennent en charge les statistiques SQL pour les requêtes récapitulatives.

Pour obtenir des informations de prise en charge de la région, du moteur de base de données et des classes d'instances pour cette fonctionnalité, consultez Prise en charge de la classe d'instances, de la région et du moteur de base de données Amazon RDS pour les fonctionnalités d'analyse des performances.

Rubriques

- Statistiques SQL pour MariaDB et MySQL
- Statistiques SQL pour Amazon RDS for Oracle
- Statistiques SQL pour Amazon RDS for SQL Server
- Statistiques SQL pour RDS PostgreSQL

Statistiques SQL pour MariaDB et MySQL

MariaDB et MySQL collectent des statistiques SQL uniquement au niveau du récapitulatif. Aucune statistique n'est affichée au niveau de l'instruction.

Rubriques

- Statistiques récapitulatives pour MariaDB et MySQL
- Statistiques à la seconde pour MariaDB et MySQL
- Statistiques par l'appel pour MariaDB et MySQL
-

Statistiques récapitulatives pour MariaDB et MySQL

Performance Insights collecte des statistiques de synthèse SQL à partir de la table `events_statements_summary_by_digest`. La table `events_statements_summary_by_digest` est gérée par votre base de données.

La table récapitulative n'a pas de politique d'expulsion. Lorsque le tableau est plein, le message suivant s' AWS Management Console affiche :

```
Performance Insights is unable to collect SQL Digest statistics on new queries because
 the table events_statements_summary_by_digest is full.
Please truncate events_statements_summary_by_digest table to clear the issue. Check the
 User Guide for more details.
```

Dans ce cas, MariaDB et MySQL n'assurent pas le suivi des requêtes SQL. Pour résoudre ce problème, Performance Insights tronque automatiquement la table de synthèse lorsque les deux conditions suivantes sont remplies :

- La table est pleine.
- Performance Insights gère automatiquement le schéma de performance.

  Pour la gestion automatique, le paramètre `performance_schema` doit être défini sur `0` et la Source ne doit pas être définie sur `user`. Si Performance Insights ne gère pas automatiquement le schéma de performance, consultez Présentation du schéma de performance pour Performance Insights sur , Amazon RDS for MariaDB ou MySQL.

Dans le AWS CLI, vérifiez la source d'une valeur de paramètre en exécutant la describe-db-parameterscommande.

Statistiques à la seconde pour MariaDB et MySQL

Les statistiques SQL suivantes sont disponibles pour les instances de bases de données MariaDB et MySQL.

Métrique	Unit
db.sql_tokenized.stats.count_star_per_sec	Appels à la seconde
db.sql_tokenized.stats.sum_timer_wait_per_sec	Latence moyenne par seconde (en ms)
db.sql_tokenized.stats.sum_select_full_join_per_sec	Sélections de jointures complètes par seconde
db.sql_tokenized.stats.sum_select_range_check_per_sec	Sélections de vérifications de plages par seconde
db.sql_tokenized.stats.sum_select_scan_per_sec	Sélections d'analyses par seconde
db.sql_tokenized.stats.sum_sort_merge_passes_per_sec	Tris de transmissions de fusion par seconde
db.sql_tokenized.stats.sum_sort_scan_per_sec	Tris d'analyses par seconde

Métrique	Unit
db.sql_tokenized.stats.sum_sort_rang e_per_sec	Tris de plages par seconde
db.sql_tokenized.stats.sum_sort_rows_per_sec	Tris de lignes par seconde
db.sql_tokenized.stats.sum_rows_affected_per_ sec	Lignes affectées par seconde
db.sql_tokenized.stats.sum_rows_exam ined_per_sec	Lignes examinées par seconde
db.sql_tokenized.stats.sum_rows_sent_per_sec	Lignes envoyées par seconde
db.sql_tokenized.stats.sum_created_tmp_disk_t ables_per_sec	Créations de tables de disques temporaires par seconde
db.sql_tokenized.stats.sum_created_tmp_tables _per_sec	Créations de tables temporaires par seconde
db.sql_tokenized.stats.sum_lock_time_per_sec	Temps de verrouillage par seconde (en millisecondes)

Statistiques par l'appel pour MariaDB et MySQL

Les métriques suivantes fournissent les statistiques par appel pour une instruction SQL.

Métrique	Unité
db.sql_tokenized.stats.sum_timer_wait_per_call	Latence moyenne par appel (en millisecondes)
db.sql_tokenized.stats.sum_select_full_join_p er_call	Sélections de jointures complètes par appel
db.sql_tokenized.stats.sum_select_ra nge_check_per_call	Sélections de vérifications de plages par appel

Métrique	Unité
db.sql_tokenized.stats.sum_select_scan_per_call	Sélections d'analyses par appel
db.sql_tokenized.stats.sum_sort_merge_passes_per_call	Tris de transmissions de fusion par appel
db.sql_tokenized.stats.sum_sort_scan_per_call	Tris d'analyses par appel
db.sql_tokenized.stats.sum_sort_range_per_call	Tris de plages par appel
db.sql_tokenized.stats.sum_sort_rows_per_call	Tris de lignes par appel
db.sql_tokenized.stats.sum_rows_affected_per_call	Lignes affectées par appel
db.sql_tokenized.stats.sum_rows_examined_per_call	Lignes examinées par appel
db.sql_tokenized.stats.sum_rows_sent_per_call	Lignes envoyées par appel
db.sql_tokenized.stats.sum_created_tmp_disk_tables_per_call	Créations de tables de disques temporaires par appel
db.sql_tokenized.stats.sum_created_tmp_tables_per_call	Créations de tables temporaires par appel
db.sql_tokenized.stats.sum_lock_time_per_call	Temps de verrouillage par appel (en ms)

Les statistiques SQL suivantes sont disponibles pour les instances de bases de données MariaDB et MySQL.

Métrique	Unit
db.sql_tokenized.stats.count_star	Calls
db.sql_tokenized.stats.sum_timer_wait	Temps d'attente (en ms)

Métrique	Unit
db.sql_tokenized.stats.sum_select_full_join	Sélectionnez l'adhésion complète
db.sql_tokenized.stats.sum_select_range_check	Sélectionnez les vérifications de plage
db.sql_tokenized.stats.sum_select_scan	Sélectionnez les scans
db.sql_tokenized.stats.sum_sort_merge_passes	Trier les passes de fusion
db.sql_tokenized.stats.sum_sort_scan	Trier les scans
db.sql_tokenized.stats.sum_sort_range	Trier les plages
db.sql_tokenized.stats.sum_sort_rows	Tri des lignes
db.sql_tokenized.stats.sum_rows_affected	Lignes affectées
db.sql_tokenized.stats.sum_rows_examined	Lignes examinées
db.sql_tokenized.stats.sum_rows_sent	Lignes envoyées
db.sql_tokenized.stats.sum_created_tmp_disk_tables	Tables de disques temporaires créées
db.sql_tokenized.stats.sum_created_tmp_tables	Tables temporaires créées
db.sql_tokenized.stats.sum_lock_time	Temps de verrouillage (en ms)

Statistiques SQL pour Amazon RDS for Oracle

Amazon RDS for Oracle collecte des statistiques SQL au niveau de l'instruction et du récapitulatif. Au niveau de l'instruction, la colonne ID représente la valeur de `V$SQL.SQL_ID`. Au niveau du récapitulatif, la colonne ID affiche la valeur de `V$SQL.FORCE_MATCHING_SIGNATURE`.

Si l'ID a la valeur `0` au niveau du récapitulatif, Oracle Database a déterminé qu'il était inopportun de réutiliser cette instruction. Dans ce cas, les instructions SQL enfant peuvent appartenir à différents récapitulatifs. Toutefois, les instructions sont regroupées sous `digest_text` pour la première instruction SQL collectée.

Rubriques

- Statistiques à la seconde pour Oracle
- Statistiques par appel pour Oracle
- Statistiques principales pour Oracle

Statistiques à la seconde pour Oracle

Les métriques suivantes fournissent des statistiques à la seconde pour une requête Oracle SQL.

Mesure	Unité
db.sql.stats.executions_per_sec	Nombre d'exécutions par seconde
db.sql.stats.elapsed_time_per_sec	Exécutions actives moyennes
db.sql.stats.rows_processed_per_sec	Lignes traitées par seconde
db.sql.stats.buffer_gets_per_sec	Buffers obtenus par seconde
db.sql.stats.physical_read_requests_per_sec	Lectures physiques par seconde
db.sql.stats.physical_write_requests_per_sec	Écritures physiques par seconde
db.sql.stats.total_sharable_mem_per_sec	Mémoire totale partageable par seconde (en octets)
db.sql.stats.cpu_time_per_sec	Temps UC par seconde (en millisecondes)

Les mesures suivantes fournissent des statistiques par seconde pour une requête Oracle SQL Digest.

Mesure	Unité
db.sql_tokenized.stats.executions_per_sec	Nombre d'exécutions par seconde
db.sql_tokenized.stats.elapsed_time_per_sec	Exécutions actives moyennes

Mesure	Unité
db.sql_tokenized.stats.rows_processed_per_sec	Lignes traitées par seconde
db.sql_tokenized.stats.buffer_gets_per_sec	Buffers obtenus par seconde
db.sql_tokenized.stats.physical_read_requests_per_sec	Lectures physiques par seconde
db.sql_tokenized.stats.physical_write_requests_per_sec	Écritures physiques par seconde
db.sql_tokenized.stats.total_sharable_mem_per_sec	Mémoire totale partageable par seconde (en octets)
db.sql_tokenized.stats.cpu_time_per_sec	Temps UC par seconde (en millisecondes)

Statistiques par appel pour Oracle

Les métriques suivantes fournissent des statistiques par appel pour une instruction Oracle SQL.

Mesure	Unité
db.sql.stats.elapsed_time_per_exec	Temps écoulé par exécution (en millisecondes)
db.sql.stats.rows_processed_per_exec	Lignes traitées par exécution
db.sql.stats.buffer_gets_per_exec	Buffers obtenus par exécution
db.sql.stats.physical_read_requests_per_exec	Lectures physiques par exécution
db.sql.stats.physical_write_requests_per_exec	Écritures physiques par exécution
db.sql.stats.total_sharable_mem_per_exec	Mémoire totale partageable par exécution (en octets)
db.sql.stats.cpu_time_per_exec	Temps UC par exécution (en millisecondes)

Les métriques suivantes fournissent des statistiques par appel pour une requête récapitulative Oracle SQL.

Mesure	Unité
db.sql_tokenized.stats.elapsed_time_per_exec	Temps écoulé par exécution (en millisecondes)
db.sql_tokenized.stats.rows_processed_per_exec	Lignes traitées par exécution
db.sql_tokenized.stats.buffer_gets_per_exec	Buffers obtenus par exécution
db.sql_tokenized.stats.physical_read_requests_per_exec	Lectures physiques par exécution
db.sql_tokenized.stats.physical_write_requests_per_exec	Écritures physiques par exécution
db.sql_tokenized.stats.total_sharable_mem_per_exec	Mémoire totale partageable par exécution (en octets)
db.sql_tokenized.stats.cpu_time_per_exec	Temps UC par exécution (en millisecondes)

Statistiques principales pour Oracle

Les mesures suivantes fournissent des statistiques principales pour une requête Oracle SQL.

Mesure	Unité
db.sql.stats.exécutions	Nombre d'exécutions
db.sql.stats.elapsed_time	Temps écoulé (en ms)
db.sql.stats.rows_processed	Lignes traitées
db.sql.stats.buffer_gets	Buffer obtient
db.sql.stats.physical_read_requests	Physical reads
db.sql.stats.physical_write_requests	écritures physiques

Mesure	Unité
db.sql.stats.total_sharable_mem	Mémoire partageable totale (en octets)
db.sql.stats.cpu_time	Temps processeur (en ms)

Les mesures suivantes fournissent des statistiques principales pour une requête Oracle SQL Digest.

Mesure	Unité
db.sql_tokenized.stats .exécutions	Nombre d'exécutions
db.sql_tokenized.stats.elapsed_time	Temps écoulé (en ms)
db.sql_tokenized.stats.rows_processed	Lignes traitées
db.sql_tokenized.stats.buffer_gets	Buffer obtient
db.sql_tokenized.stats.physical_read_requests	Physical reads
db.sql_tokenized.stats.physical_write_requests	écritures physiques
db.sql_tokenized.stats.total_sharable_mem	Mémoire partageable totale (en octets)
db.sql_tokenized.stats.cpu_time	Temps processeur (en ms)

Statistiques SQL pour Amazon RDS for SQL Server

Amazon RDS for SQL Server collecte des statistiques SQL au niveau de l'instruction et de la synthèse. Au niveau de l'instruction, la colonne ID représente la valeur de `sql_handle`. Au niveau du récapitulatif, la colonne ID affiche la valeur de `query_hash`.

SQL Server renvoie des valeurs NULL pour `query_hash` pour quelques instructions. Tel est notamment le cas de ALTER INDEX, CHECKPOINT, UPDATE STATISTICS, COMMIT TRANSACTION, FETCH NEXT FROM Cursor et de quelques instructions INSERT, de SELECT @<variable>, des instructions conditionnelles et des procédures stockées exécutables. Dans ce cas, la valeur de `sql_handle` s'affiche en tant qu'ID au niveau de la synthèse de l'instruction.

Rubriques

- Statistiques par seconde pour SQL Server
- Statistiques par appel pour SQL Server
- Statistiques principales pour SQL Server

Statistiques par seconde pour SQL Server

Les métriques suivantes fournissent des statistiques par seconde pour une requête SQL de SQL Server.

Mesure	Unité
db.sql.stats.execution_count_per_sec	Nombre d'exécutions par seconde
db.sql.stats.total_elapsed_time_per_sec	Temps total écoulé par seconde
db.sql.stats.total_rows_per_sec	Nombre total de lignes traitées par seconde
db.sql.stats.total_logical_reads_per_sec	Nombre total de lectures logiques par seconde
db.sql.stats.total_logical_writes_per_sec	Nombre total d'écritures logiques par seconde
db.sql.stats.total_physical_reads_per_sec	Nombre total de lectures physiques par seconde
db.sql.stats.total_worker_time_per_sec	Temps processeur total (en ms)

Les métriques suivantes fournissent des statistiques par seconde pour une requête de synthèse SQL de SQL Server.

Mesure	Unité
db.sql_tokenized.stats.execution_count_per_sec	Nombre d'exécutions par seconde
db.sql_tokenized.stats.total_elapsed_time_per_sec	Temps total écoulé par seconde
db.sql_tokenized.stats.total_rows_per_sec	Nombre total de lignes traitées par seconde

Mesure	Unité
db.sql_tokenized.stats.total_logical_reads_per_sec	Nombre total de lectures logiques par seconde
db.sql_tokenized.stats.total_logical_writes_per_sec	Nombre total d'écritures logiques par seconde
db.sql_tokenized.stats.total_physical_reads_per_sec	Nombre total de lectures physiques par seconde
db.sql_tokenized.stats.total_worker_time_per_sec	Temps processeur total (en ms)

Statistiques par appel pour SQL Server

Les métriques suivantes fournissent les statistiques par appel pour une instruction SQL de SQL Server.

Mesure	Unité
db.sql.stats.total_elapsed_time_per_call	Temps total écoulé par exécution (en ms)
db.sql.stats.total_rows_per_call	Nombre total de lignes traitées par exécution
db.sql.stats.total_logical_reads_per_call	Nombre total de lectures logiques par exécution
db.sql.stats.total_logical_writes_per_call	Nombre total d'écritures logiques par exécution
db.sql.stats.total_physical_reads_per_call	Nombre total de lectures physiques par exécution
db.sql.stats.total_worker_time_per_call	Temps processeur total par exécution (en ms)

Les métriques suivantes fournissent des statistiques par appel pour une requête de synthèse SQL de SQL Server.

Mesure	Unité
db.sql_tokenized.stats.total_elapsed_time_per_call	Temps total écoulé par exécution
db.sql_tokenized.stats.total_rows_per_call	Nombre total de lignes traitées par exécution
db.sql_tokenized.stats.total_logical_reads_per_call	Nombre total de lectures logiques par exécution
db.sql_tokenized.stats.total_logical_writes_per_call	Nombre total d'écritures logiques par exécution
db.sql_tokenized.stats.total_physical_reads_per_call	Nombre total de lectures physiques par exécution
db.sql_tokenized.stats.total_worker_time_per_call	Temps processeur total par exécution (en ms)

Statistiques principales pour SQL Server

Les mesures suivantes fournissent des statistiques principales pour une requête SQL Server.

Mesure	Unité
db.sql.stats.execution_count	Nombre d'exécutions
db.sql.stats.total_elapsed_time	Temps total écoulé (en ms)
db.sql.stats.total_rows	Nombre total de lignes traitées
db.sql.stats.total_logical_reads	Nombre total de lectures logiques
db.sql.stats.total_logical_writes	Nombre total d'écritures logiques
db.sql.stats.total_physical_reads	Nombre total de lectures physiques
db.sql.stats.total_worker_time	Temps processeur total (en ms)

Les mesures suivantes fournissent des statistiques principales pour une requête SQL Server SQL Digest.

Mesure	Unité
db.sql_tokenized.stats.execution_count	Numéro d'exécution
db.sql_tokenized.stats.total_elapsed_time	Temps total écoulé (en ms)
db.sql_tokenized.stats.total_rows	Nombre total de lignes traitées
db.sql_tokenized.stats.total_logical_reads	Nombre total de lectures logiques
db.sql_tokenized.stats.total_logical_writes	Nombre total d'écritures logiques
db.sql_tokenized.stats.total_physical_reads	Nombre total de lectures physiques
db.sql_tokenized.stats.total_worker_time	Temps processeur total (en ms)

Statistiques SQL pour RDS PostgreSQL

Pour chaque appel SQL et pour chaque seconde d'exécution d'une requête, Performance Insights collecte des statistiques SQL. RDS for PostgreSQL collecte des statistiques SQL uniquement au niveau des récapitulatifs. Aucune statistique n'est affichée au niveau des déclarations.

Vous trouverez ci-dessous des informations sur les statistiques de niveau récapitulatif pour RDS for PostgreSQL.

Rubriques

- Statistiques récapitulatives pour RDS PostgreSQL
- Statistiques récapitulatives à la seconde pour RDS PostgreSQL
- Statistiques récapitulatives par appel pour RDS PostgreSQL
- Statistiques principales pour RDS PostgreSQL

Statistiques récapitulatives pour RDS PostgreSQL

Pour afficher les statistiques récapitulatives SQL, RDS PostgreSQL doit charger la bibliothèque `pg_stat_statements`. Pour les instances de base de données PostgreSQL compatibles

avec PostgreSQL 11 ou version ultérieure, la base de données charge cette bibliothèque par défaut. Pour les instances de base de données PostgreSQL compatibles avec PostgreSQL 10 ou version antérieure, activez cette bibliothèque manuellement. Pour l'activer manuellement, ajoutez `pg_stat_statements` à `shared_preload_libraries` dans le groupe de paramètres de base de données associé à l'instance de base de données. Puis, redémarrez votre instance de base de données. Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).

> **Note**
>
> Performance Insights peut uniquement collecter des statistiques pour les requêtes non tronquées dans `pg_stat_activity`. Par défaut, les bases de données PostgreSQL tronquent les requêtes de plus de 1 024 octets. Pour augmenter la taille de la requête, modifiez le paramètre `track_activity_query_size` dans le groupe de paramètres de base de données associé à votre instance de base de données. Lorsque vous modifiez ce paramètre, un redémarrage d'instance de base de données est obligatoire.

Statistiques récapitulatives à la seconde pour RDS PostgreSQL

Les statistiques récapitulatives SQL suivantes sont disponibles pour les instances de base de données PostgreSQL.

Métrique	Unité
db.sql_tokenized.stats.calls_per_sec	Appels par seconde
db.sql_tokenized.stats.rows_per_sec	Lignes par seconde
db.sql_tokenized.stats.total_time_per_sec	Exécutions actives moyennes par seconde
db.sql_tokenized.stats.shared_blks_hit_per_sec	Accès en masse par seconde
db.sql_tokenized.stats.shared_blks_read_per_sec	Lectures en masse par seconde
db.sql_tokenized.stats.shared_blks_dirtied_per_sec	Blocs salis par seconde

Métrique	Unité
db.sql_tokenized.stats.shared_blks_written_per_sec	Écritures en masse par seconde
db.sql_tokenized.stats.local_blks_hit_per_sec	Nombre de blocs locaux par seconde
db.sql_tokenized.stats.local_blks_read_per_sec	Lectures par bloc local par seconde
db.sql_tokenized.stats.local_blks_dirtied_per_sec	Bloc local souillé par seconde
db.sql_tokenized.stats.local_blks_written_per_sec	Écritures par bloc local par seconde
db.sql_tokenized.stats.temp_blks_written_per_sec	Écritures temporaires par seconde
db.sql_tokenized.stats.temp_blks_read_per_sec	Lectures temporaires par seconde
db.sql_tokenized.stats.blk_read_time_per_sec	Lectures simultanées moyennes par seconde
db.sql_tokenized.stats.blk_write_time_per_sec	Écritures simultanées moyennes par seconde

Statistiques récapitulatives par appel pour RDS PostgreSQL

Les métriques suivantes fournissent les statistiques par appel pour une instruction SQL.

Métrique	Unité
db.sql_tokenized.stats.rows_per_call	Lignes par appel
db.sql_tokenized.stats.avg_latency_per_call	Latence moyenne par appel (en millisecondes)
db.sql_tokenized.stats.shared_blks_hit_per_call	Accès en masse par appel
db.sql_tokenized.stats.shared_blks_read_per_call	Lectures en masse par appel

Métrique	Unité
db.sql_tokenized.stats.shared_blks_written_per_call	Écritures en masse par appel
db.sql_tokenized.stats.shared_blks_dirtied_per_call	Blocs salis par appel
db.sql_tokenized.stats.local_blks_hit_per_call	Nombre d'accès par bloc local par appel
db.sql_tokenized.stats.local_blks_read_per_call	Lectures par bloc local par appel
db.sql_tokenized.stats.local_blks_dirtied_per_call	Bloc local sale par appel
db.sql_tokenized.stats.local_blks_written_per_call	Écritures de blocs locaux par appel
db.sql_tokenized.stats.temp_blks_written_per_call	Écritures de blocs temporaires par appel
db.sql_tokenized.stats.temp_blks_read_per_call	Lectures de blocs temporaires par appel
db.sql_tokenized.stats.blk_read_time_per_call	Temps de lecture par appel (en ms)
db.sql_tokenized.stats.blk_write_time_per_call	Temps d'écriture par appel (en ms)

Statistiques principales pour RDS PostgreSQL

Les statistiques SQL suivantes sont disponibles pour les instances de base de données PostgreSQL.

Mesure	Unité
db.sql_tokenized.stats.calls	Calls
db.sql_tokenized.stats.rows	Lignes
db.sql_tokenized.stats.total_time	Temps total (en ms)
db.sql_tokenized.stats.shared_blks_hit	Bloquer les hits

Mesure	Unité
db.sql_tokenized.stats.shared_blks_read	Bloquer les lectures
db.sql_tokenized.stats.shared_blks_dirtied	Blocs sales
db.sql_tokenized.stats.shared_blks_written	Écriture par blocs
db.sql_tokenized.stats.local_blks_hit	Nombre de blocages locaux
db.sql_tokenized.stats.local_blks_read	Lectures de blocs locales
db.sql_tokenized.stats.local_blks_dirtied	Blocs locaux sales
db.sql_tokenized.stats.local_blks_written	écritures par blocs locales
db.sql_tokenized.stats.temp_blks_written	écritures temporaires
db.sql_tokenized.stats.temp_blks_read	Lectures temporaires
db.sql_tokenized.stats.blk_read_time	Nombre moyen de lectures simultanées (en ms)
db.sql_tokenized.stats.blk_write_time	Nombre moyen d'écritures simultanées (en ms)

Pour de plus amples informations sur ces métriques, veuillez consulter pg_stat_statements dans la documentation PostgreSQL.

Métriques du système d'exploitation dans la surveillance améliorée

Amazon RDS fournit des métriques en temps réel pour le système d'exploitation (OS) sur lequel s'exécute votre d'instances de base de données. RDS fournit les métriques issues de la surveillance améliorée à votre compte Amazon CloudWatch Logs. Les tableaux suivants répertorient les métriques du système d'exploitation disponibles avec Amazon CloudWatch Logs.

Rubriques

- Métriques du système d'exploitation pour DB2, MariaDB, SQL My, Oracle et Postgre SQL
- Métriques du système d'exploitation pour Microsoft SQL Server

Métriques du système d'exploitation pour DB2, MariaDB, SQL My, Oracle et Postgre SQL

Groupe	Métrique	Nom de la console	Description
General	engine	Ne s'applique pas	Moteur de base de données de l'instance de base de données.
	instanceID	Ne s'applique pas	Identifiant de l'instance de base de données.
	instanceResourceID	Ne s’applique pas	Identifiant immuable pour l'instance de base de données unique à une AWS région, également utilisé comme identifiant du flux de journal.
	numVCPUs	Ne s’applique pas	Le nombre de virtuels CPUs pour l'instance de base de données.
	timestamp	Ne s’applique pas	Heure à laquelle la métrique a été évaluée.
	uptime	Ne s'applique pas	Temps d'activité de l'instance de base de données.
	version	Ne s’applique pas	Version du format de flux JSON des métriques du système d'exploitation.
cpuUtilization	guest	CPUInvité	Pourcentage d'CPUutilisation par les programmes invités.
	idle	CPUInactif	Le pourcentage de CPU ce montant est inactif.

Groupe	Métrique	Nom de la console	Description
	`irq`	CPU IRQ	Pourcentage d'CPUinterruptions du logiciel en cours d'utilisation.
	`nice`	CPUSympa	Pourcentage d'actifs CPU utilisés par les programmes exécutés avec la priorité la plus basse.
	`steal`	CPUVoler	Pourcentage de CPU machines virtuelles utilisées par d'autres machines virtuelles.
	`system`	CPUSystème	Pourcentage d'CPUénergie utilisée par le noyau.
	`total`	CPUTotal	Pourcentage total du produit CPU en cours d'utilisation. Cette valeur inclut la valeur `nice`.
	`user`	CPUUtilisateur	Pourcentage d'CPUapplications utilisées par les programmes utilisateur.
	`wait`	CPUAttendre	Pourcentage de données CPU non utilisées pendant l'attente d'un accès aux E/S.
`diskIO`	`avgQueueLen`	Taille moyenne de la file d'attente	Nombre de requêtes en attente dans la file d'attente du périphérique d'I/O.
	`avgReqSz`	Taille moyenne de la demande	Taille moyenne de requête, en kilo-octets.
	`await`	E/S disque en attente	Nombre de millisecondes requises pour répondre aux requêtes, y compris le temps d'attente et le temps de service.

Groupe	Métrique	Nom de la console	Description
	`device`	Ne s'applique pas	Identifiant du périphérique de disque en cours d'utilisation.
	`readIOsPS`	E/S lecture	Nombre d'opérations de lecture par seconde.
	`readKb`	Total lecture	Nombre total de kilo-octets lus.
	`readKbPS`	Ko/s lecture	Nombre de kilo-octets lus par seconde.
	`readLatency`	Latence de lecture	Temps écoulé entre l'envoi d'une requête d'I/O de lecture et sa fin, en millisecondes. Cette métrique est uniquement disponible pour Amazon Aurora.
	`readThroughput`	Débit de lecture	Quantité de débit réseau utilisée par les demandes adressées au cluster DB, en octets par seconde. Cette métrique est uniquement disponible pour Amazon Aurora.
	`rrqmPS`	Rrqms	Nombre de requêtes de lecture fusionnées mises en file d'attente par seconde.
	`tps`	TPS	Nombre de transactions d'I/O par seconde.
	`util`	Util E/S disque	Pourcentage de CPU temps pendant lequel les demandes ont été émises.
	`writeIOsPS`	E/S écriture	Nombre d'opérations d'écriture par seconde.

Groupe	Métrique	Nom de la console	Description
	writeKb	Total écriture	Nombre total de kilo-octets écrits.
	writeKbPS	Ko/s écriture	Nombre de kilo-octets écrits par seconde.
	writeLatency	Latence en écriture	Temps moyen écoulé entre l'envoi d'une requête d'I/O d'écriture et sa fin, en millisecondes. Cette métrique est uniquement disponible pour Amazon Aurora.
	writeThroughput	Débit d'écriture	Quantité de débit réseau utilisée par les réponses du cluster DB, en octets par seconde. Cette métrique est uniquement disponible pour Amazon Aurora.
	wrqmPS	Wrqms	Nombre de requêtes d'écriture fusionnées mises en file d'attente par seconde.
physicalDeviceIO	avgQueueLen	Taille moyenne de la file d'attente pour les périphériques physiques	Nombre de requêtes en attente dans la file d'attente du périphérique d'I/O.

Groupe	Métrique	Nom de la console	Description
	`avgReqSz`	Taille moyenne de la demande pour les périphériques physiques	Taille moyenne de requête, en kilo-octets.
	`await`	E/S disque en attente pour les périphériques physiques	Nombre de millisecondes requises pour répondre aux requêtes, y compris le temps d'attente et le temps de service.
	`device`	Ne s'applique pas	Identifiant du périphérique de disque en cours d'utilisation.
	`readIOsPS`	E/S lecture pour périphériques physiques	Nombre d'opérations de lecture par seconde.
	`readKb`	Nombre total de lecture pour les périphériques physiques	Nombre total de kilo-octets lus.

Groupe	Métrique	Nom de la console	Description
	`readKbPS`	Lecture ko/s pour les périphériques physiques	Nombre de kilo-octets lus par seconde.
	`rrqmPS`	Rrqms pour les périphériques physiques	Nombre de requêtes de lecture fusionnées mises en file d'attente par seconde.
	`tps`	Appareils physiques TPS	Nombre de transactions d'I/O par seconde.
	`util`	Util. d'E/S de disque pour les périphériques physiques	Pourcentage de CPU temps pendant lequel les demandes ont été émises.
	`writeIOsPS`	Périphériques physiques en écriture IO/s	Nombre d'opérations d'écriture par seconde.

Groupe	Métrique	Nom de la console	Description
	`writeKb`	Nombre total d'écritures pour les périphériques physiques	Nombre total de kilo-octets écrits.
	`writeKbPS`	Écriture ko/s pour les périphériques physiques	Nombre de kilo-octets écrits par seconde.
	`wrqmPS`	Wrqms pour les périphériques physiques	Nombre de requêtes d'écriture fusionnées mises en file d'attente par seconde.
`fileSys`	`maxFiles`	Nombre maximum d'inodes	Nombre maximum de fichiers pouvant être créés pour le système de fichiers.
	`mountPoint`	Ne s'applique pas	Chemin vers le système de fichiers.
	`name`	Ne s'applique pas	Nom du système de fichiers.
	`total`	Total système de fichiers	Quantité totale d'espace disque disponible pour le système de fichiers, en kilo-octets.

Groupe	Métrique	Nom de la console	Description
	`used`	Système de fichiers utilisé	Quantité d'espace disque utilisé par des fichiers du système de fichiers, en kilo-octets.
	`usedFilePercent`	Inodes utilisés	Pourcentage de fichiers disponibles en cours d'utilisation.
	`usedFiles`	% utilisé	Nombre de fichiers dans le système de fichiers.
	`usedPercent`	Système de fichiers utilisé	Pourcentage d'espace de disque du système de fichiers en cours d'utilisation.
`loadAverageMinute`	`fifteen`	Charge moyenne 15 min	Nombre de processus ayant demandé du CPU temps au cours des 15 dernières minutes.
	`five`	Charge moyenne 5 min	Nombre de processus ayant demandé du CPU temps au cours des 5 dernières minutes.
	`one`	Charge moyenne 1 min	Nombre de processus ayant demandé du CPU temps au cours de la dernière minute.
`memory`	`active`	Mémoire active	Quantité de mémoire attribuée, en kilo-octets.
	`buffers`	Mémoire mise en tampon	Quantité de mémoire utilisée pour la mise en mémoire tampon des demandes I/O avant écriture dans le périphérique de stockage, en kilo-octets.
	`cached`	Mémoire mise en cache	Quantité de mémoire utilisée pour la mise en cache des I/O basées sur le système de fichiers.

Groupe	Métrique	Nom de la console	Description
	`dirty`	Mémoire corrompue	Le nombre de pages de mémoire RAM qui ont été modifiées mais qui n'ont pas été écrites dans leur bloc de données associé en stockage, en kilo-octets.
	`free`	Mémoire libre	Quantité de mémoire non attribuée, en kilo-octets.
	`hugePagesFree`	Grandes pages gratuites	Nombre de grandes pages gratuites. Les grandes pages sont une fonction du noyau Linux.
	`hugePagesRsvd`	Grandes pages Rsvd	Nombre de grandes pages dédiées.
	`hugePagesSize`	Taille des grandes pages	Taille de chaque unité de grandes pages, en kilo-octets.
	`hugePagesSurp`	Grandes pages excéd	Nombre de grandes pages excédentaires disponibles par rapport au nombre total.
	`hugePagesTotal`	Total de grandes pages	Le nombre total de grandes pages.
	`inactive`	Mémoire inactive	Quantité de pages mémoire moins fréquemment utilisées, en kilo-octets.
	`mapped`	Mémoire mappée	Quantité totale de contenu du système de fichiers mappé en mémoire dans un espace d'adressage de processus, en kilo-octets.
	`pageTables`	Tables de pages	Quantité de mémoire utilisée par les tables de page, en kilo-octets.

Groupe	Métrique	Nom de la console	Description
	`slab`	Mémoire de section	Quantité de structures de données noyau réutilisables, en kilo-octets.
	`total`	Mémoire totale	Quantité totale de mémoire, en kilo-octets.
	`writeback`	Mémoire en écriture différée	Le nombre de pages sales RAM qui s'y trouvent encore en cours d'écriture sur le stockage de sauvegarde, en kilo-octets.
`network`	`interface`	Ne s’applique pas	Identifiant pour l'interface réseau utilisée pour l'instance de base de données.
	`rx`	RX	Nombre d'octets reçus par seconde.
	`tx`	TX	Nombre d'octets téléchargés par seconde.
`processList`	`cpuUsedPc`	CPU %	Pourcentage CPU utilisé par le processus.
	`id`	Ne s’applique pas	Identifiant du processus.
	`memoryUsedPc`	MEM%	Pourcentage de mémoire utilisé par le processus.
	`name`	Ne s'applique pas	Nom du processus.
	`parentID`	Ne s'applique pas	Identifiant de processus pour le processus parent du processus.

Groupe	Métrique	Nom de la console	Description
	`rss`	RES	Quantité RAM allouée au processus, en kilo-octets.
	`tgid`	Ne s’applique pas	Identifiant du groupe de threads. Numéro représentant l'ID du processus auquel appartient le thread. Cet identifiant permet de regrouper les threads d'un même processus.
	`vss`	VIRT	Quantité de mémoire virtuelle allouée au processus, en kilo-octets.
`swap`	`total`	Swap	Quantité de mémoire d'échange disponible, en kilo-octets.
	`in`	Swaps dans	Quantité de mémoire, en kilo-octets, échangée depuis le disque.
	`out`	Swaps vers	Quantité de mémoire, en kilo-octets, échangée vers le disque.
	`free`	Swap libre	Quantité de mémoire d'échange libre, en kilo-octets.
	`cached`	Swap validé	Quantité de mémoire d'échange, en kilo-octets, utilisée en tant que mémoire cache.
`tasks`	`blocked`	Tâches bloquées	Nombre de tâches bloquées.
	`running`	Tâches en cours d'exécution	Nombre de tâches en cours d'exécution.
	`sleeping`	Tâches en veille	Nombre de tâches en veille.
	`stopped`	Tâches arrêtées	Nombre de tâches arrêtées.

Groupe	Métrique	Nom de la console	Description
	`total`	Total de tâches	Nombre total de tâches.
	`zombie`	Tâches zombies	Nombre de tâches enfant inactives avec une tâche parent active.

Métriques du système d'exploitation pour Microsoft SQL Server

Groupe	Métrique	Nom de la console	Description
`General`	`engine`	Ne s'applique pas	Moteur de base de données de l'instance de base de données.
	`instanceID`	Ne s'applique pas	Identifiant de l'instance de base de données.
	`instanceResourceID`	Ne s’applique pas	Identifiant immuable pour l'instance de base de données unique à une AWS région, également utilisé comme identifiant du flux de journal.
	`numVCPUs`	Ne s’applique pas	Le nombre de virtuels CPUs pour l'instance de base de données.
	`timestamp`	Ne s’applique pas	Heure à laquelle la métrique a été évaluée.
	`uptime`	Ne s'applique pas	Temps d'activité de l'instance de base de données.
	`version`	Ne s’applique pas	Version du format de flux JSON des métriques du système d'exploitation.
`cpuUtilization`	`idle`	CPUInactif	Le pourcentage de CPU ce montant est inactif.

Groupe	Métrique	Nom de la console	Description
	`kern`	CPUNoyau	Pourcentage d'CPUénergie utilisée par le noyau.
	`user`	CPUUtilisateur	Pourcentage d'CPUapplications utilisées par les programmes utilisateur.
`disks`	`name`	Ne s’applique pas	Identifiant du disque.
	`totalKb`	Espace disque total	Espace total du disque, en kilo-octets.
	`usedKb`	Espace disque utilisé	Quantité d'espace utilisé sur le disque, en kilo-octets.
	`usedPc`	% d'espace disque utilisé	Pourcentage d'espace utilisé sur le disque.
	`availKb`	Espace disque disponible	Espace disponible sur le disque, en kilo-octets.
	`availPc`	% d'espace disque disponible	Pourcentage d'espace disponible sur le disque.
	`rdCountPS`	Lectures/s	Nombre d'opérations de lecture par seconde.
	`rdBytesPS`	Ko/s lecture	Nombre d'octets lus par seconde.
	`wrCountPS`	E/S écriture	Nombre d'opérations d'écriture par seconde.
	`wrBytesPS`	Ko/s écriture	Nombre d'octets écrits par seconde.

Groupe	Métrique	Nom de la console	Description
memory	commitTotKb	Total de validation	Quantité d'espace d'adressage virtuel en cours d'utilisation basé sur le fichier d'échange, c'est-à-dire la charge d'écriture actuelle. Cette valeur est composée de la mémoire principale (RAM) et du disque (pagefiles).
	commitLimitKb	Validation maximale	Valeur maximum acceptée pour la métrique commitTotKb . Cette valeur est la somme de la taille actuelle du fichier de page et de la mémoire physique disponible pour le contenu paginable, à l'exception de RAM celle affectée aux zones non paginables.
	commitPeakKb	Pic de validation	Plus grande valeur de la métrique commitTotKb depuis le dernier démarrage du système d'exploitation.
	kernTotKb	Mémoire totale du noyau	Quantité de mémoire dans les groupes de noyaux paginés et non paginés, en kilo-octets.
	kernPagedKb	Mémoire du noyau paginée	Quantité de mémoire dans le groupe de noyaux paginés, en kilo-octets.
	kernNonpagedKb	Mémoire du noyau non paginée	Quantité de mémoire dans le groupe de noyaux non paginés, en kilo-octets.
	pageSize	Taille de page	Taille d'une page, en octets.
	physTotKb	Mémoire totale	Quantité de mémoire physique, en kilo-octets.

Groupe	Métrique	Nom de la console	Description
	`physAvailKb`	Mémoire disponible	Quantité de mémoire physique disponible, en kilo-octets.
	`sqlServerTotKb`	SQLMémoire totale du serveur	Quantité de mémoire dédiée au SQL serveur, en kilo-octets.
	`sysCacheKb`	Cache système	Quantité de mémoire cache du système, en kilo-octets.
`network`	`interface`	Ne s'applique pas	Identifiant pour l'interface réseau utilisée pour l'instance de base de données.
	`rdBytesPS`	Lecture réseau Ko/s	Nombre d'octets reçus par seconde.
	`wrBytesPS`	Écriture réseau Ko/s	Nombre d'octets envoyés par seconde.
`processList`	`cpuUsedPc`	% utilisé	Pourcentage CPU utilisé par le processus.
	`memUsedPc`	MEM%	Pourcentage de mémoire totale utilisé par le processus.
	`name`	Ne s'applique pas	Nom du processus.
	`pid`	Ne s'applique pas	Identifiant du processus. Cette valeur n'est pas présente pour les processus détenus par AmazonRDS.
	`ppid`	Ne s’applique pas	Identifiant de processus pour le parent de ce processus. Cette valeur est uniquement présente pour les processus enfant.

Groupe	Métrique	Nom de la console	Description
	tid	Ne s'applique pas	Identifiant du thread. Cette valeur est uniquement présente pour les threads. Le processus propriétaire peut être identifié à l'aide de la valeur pid.
	workingSetKb	Ne s'applique pas	Quantité de mémoire dans l'ensemble de travail privé, plus la quantité de mémoire en cours d'utilisation par le processus et pouvant être partagée avec d'autres processus, en kilo-octets.
	workingSetPrivKb	Ne s'applique pas	Quantité de mémoire en cours d'utilisation par un processus, mais ne pouvant pas être partagée avec d'autres processus, en kilo-octets.
	workingSetShareableKb	Ne s'applique pas	Quantité de mémoire en cours d'utilisation par un processus et pouvant être partagée avec d'autres processus, en kilo-octets.
	virtKb	Ne s'applique pas	Quantité d'espace d'adressage virtuel utilisé par le processus, in kilobytes. L'utilisation d'un espace d'adressage virtuel n'implique pas forcément l'utilisation correspondante de disque ou de pages de mémoire principale.
system	handles	Poignées	Nombre de handles utilisés par le système.
	processes	Processus	Nombre de processus s'exécutant sur le système.
	threads	Threads	Nombre de threads s'exécutant sur le système.

Surveillance des événements, des journaux et des flux dans un d'instances de base de données RDSAmazon

Lorsque vous surveillez vos bases de données RDSAmazon et vos autres AWS solutions, votre objectif est de maintenir les points suivants :

- Fiabilité
- Disponibilité
- Performance
- Sécurité

Surveillance des métriques dans une RDS instance Amazon explique la surveillance de votre instance à l'aide de métriques. Une solution complète doit également surveiller les événements de base de données, les fichiers journaux et les flux d'activité. AWS met à votre disposition les outils de surveillance suivants :

- Amazon EventBridge est un service de bus d'événements sans serveur qui permet de connecter facilement vos applications à des données provenant de diverses sources. EventBridge fournit un flux de données en temps réel à partir de vos propres applications, applications oftware-as-a S-Service (SAAS) et AWS services. EventBridge achemine ces données vers des cibles telles que AWS Lambda. Cela vous permet de surveiller les événements qui se produisent dans les services et de créer des architectures basées sur les événements. Pour plus d'informations, consultez le guide de EventBridge l'utilisateur Amazon.
- Amazon CloudWatch Logs fournit un moyen de surveiller, de stocker et d'accéder à vos fichiers journaux à partir d'instances RDSAmazon et d'autres sources. AWS CloudTrail Amazon CloudWatch Logs peut surveiller les informations contenues dans les fichiers journaux et vous avertir lorsque certains seuils sont atteints. Vous pouvez également archiver vos données de journaux dans une solution de stockage hautement durable. Pour plus d'informations, consultez le guide de l'utilisateur d'Amazon CloudWatch Logs.
- AWS CloudTrailenregistre API les appels et les événements connexes effectués par ou en votre nom Compte AWS. CloudTrail fournit les fichiers journaux dans un compartiment Amazon S3 que vous spécifiez. Vous pouvez identifier les utilisateurs et les comptes appelés AWS, l'adresse IP source à partir de laquelle les appels ont été effectués et la date des appels. Pour plus d'informations, consultez le AWS CloudTrail Guide de l'utilisateur .

- Database Activity Streams est une fonctionnalité d'RDSAmazon qui fournit un flux en temps quasi réel de l'activité de votre instance de de base de données. Amazon RDS redirige les activités vers un flux de données Amazon Kinesis. Le flux Kinesis est créé automatiquement. Kinesis vous permet de configurer des AWS services tels qu'Amazon Data Firehose, de consommer le flux et AWS Lambda de stocker les données.

Rubriques

- Afficher les journaux, les événements et les flux dans la RDS console Amazon
- Surveillance des RDS événements Amazon
- Surveillance des fichiers journaux RDSAmazon
- Surveillance (Amazon RDS API appelle) AWS CloudTrail
- Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données
- Surveillance des menaces avec Amazon GuardDuty RDS Protection

Afficher les journaux, les événements et les flux dans la RDS console Amazon

Amazon RDS s'intègre Services AWS à pour afficher des informations sur les journaux, les événements et les flux d'activité des bases de données dans la RDS console.

L'onglet Logs & events de votre d'instances de base de RDSdonnées affiche les informations suivantes :

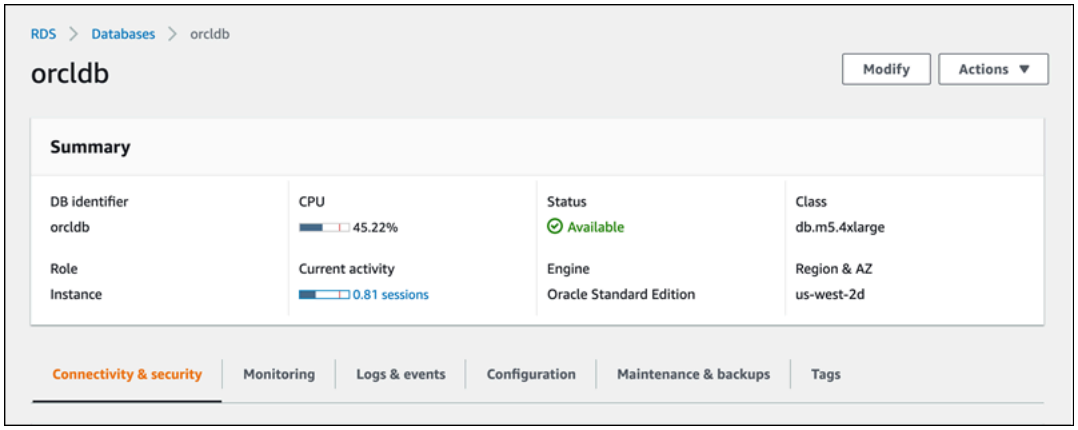
- CloudWatch Alarmes Amazon : affiche toutes les alarmes métriques que vous avez configurées pour l'instance de base . Si vous n'avez pas configuré d'alarmes, vous pouvez les créer dans la RDS console. Pour de plus amples informations, veuillez consulter Surveillance des métriques Amazon RDS () avec Amazon CloudWatch.
- Événements récents : affiche un résumé des événements (modifications de l'environnement) pour votre instance de base de RDSdonnées . Pour de plus amples informations, veuillez consulter Consulter les RDS événements Amazon.
- Journaux : affiche les fichiers journaux de base de données générés par une instance de base de données . Pour de plus amples informations, veuillez consulter Surveillance des fichiers journaux RDSAmazon.

L'onglet Configuration (Configuration) affiche des informations sur les flux d'activité de base de données.

Pour afficher les journaux, les événements et les flux de votre d'instances de base de données dans la RDS console

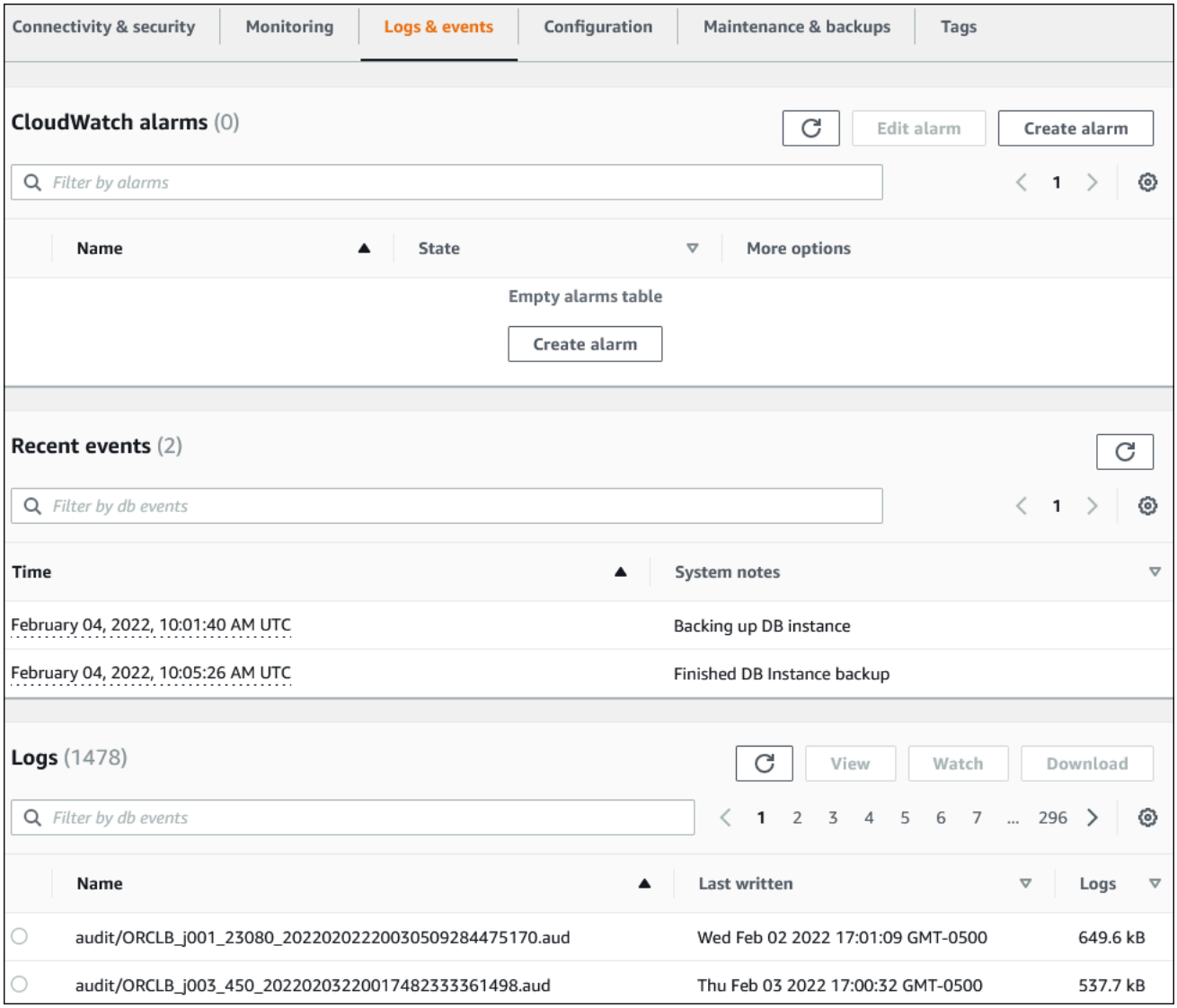
1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez le nom du d'instances de base de données que vous souhaitez surveiller.

   La page Databases (Bases de données) s'affiche. L'exemple suivant montre une base de données Oracle nommée `orclb` .



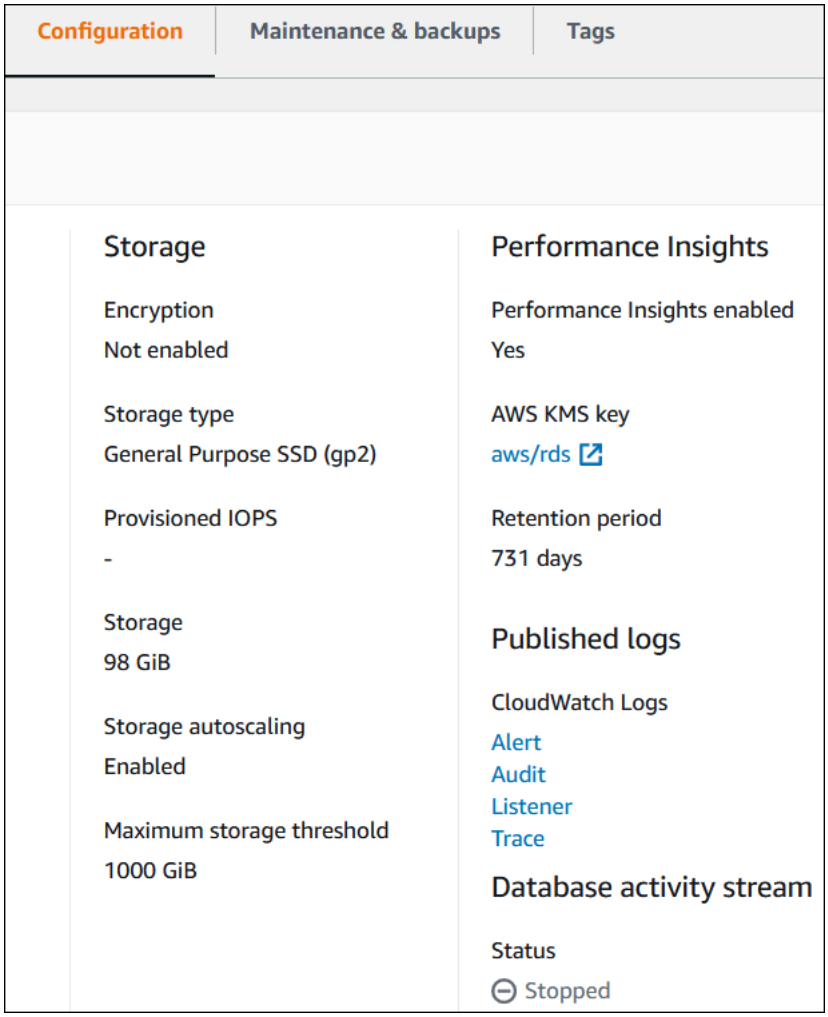
4. Choisissez Logs & events (Journaux et événements).

   La section Logs & events (Journaux et événements) et événements s'affiche.



5. Choisissez Configuration.

   L'exemple suivant montre l'état des flux d'activité de base de données pour votre instance de base de données.


Configuration
Maintenance & backups
Tags
Storage
Encryption
Not enabled
Storage type
General Purpose SSD (gp2)
Provisioned IOPS
-
Storage
98 GiB
Storage autoscaling
Enabled
Maximum storage threshold
1000 GiB
Performance Insights
Performance Insights enabled
Yes
AWS KMS key
aws/rds
Retention period
731 days
Published logs
CloudWatch Logs
Alert
Audit
Listener
Trace
Database activity stream
Status
Stopped

Surveillance des RDS événements Amazon

Un évènement indique un changement dans un environnement. Il peut s'agir d'un AWS environnement, d'un service ou d'une application partenaire SaaS, ou d'une application ou d'un service personnalisé. Pour une description des RDSévénements , voirCatégories d'événements Amazon RDS et messages d'événements .

Rubriques

- Vue d'ensemble des événements pour Amazon RDS
- Consulter les RDS événements Amazon
- Utilisation des notifications d'RDSévénements Amazon
- Création d'une règle qui se déclenche sur un événement Amazon RDS
- Catégories d'événements Amazon RDS et messages d'événements

Vue d'ensemble des événements pour Amazon RDS

Un RDSévénement indique un changement dans l'RDSenvironnement Amazon . Par exemple, Amazon RDS Amazon génère un événement lorsqu'un , l'état d'une instance de base de données passe de « en attente » à « en cours d'exécution ». Amazon RDS diffuse des événements EventBridge en temps quasi réel.

> **Note**
>
> Amazon RDS diffuse des événements dans la mesure du possible. Nous vous recommandons d'éviter d'écrire des programmes en fonction de la présence d'évènements de notification ou de leur ordre, car il peut ne pas y en avoir ou ils peuvent ne pas être dans l'ordre défini.

Amazon RDS enregistre les événements liés aux ressources suivantes :

- Instances de base de données

  Pour obtenir une liste des événements d'instance de base de données, consultez Évènements d'instance de base de données.
- Groupes de paramètres DB

Pour obtenir une liste des événements de groupe de paramètres de base de données, consultez Évènements de groupe de paramètres de base de données.

- Groupes de sécurité DB

  Pour obtenir la liste des événements relatifs aux groupes de sécurité de la base de données, consultez Évènements de groupe de sécurité de base de données.

- Instantanés de la base de données

  Pour obtenir la liste des événements liés aux instantanés de la base de données, consultez Évènements d'instantané de bases de données.

- RDSÉvénements liés au proxy

  Pour obtenir la liste des événements RDS liés au proxy, consultezÉvénements RDS Proxy.

- Événements de déploiement bleu/vert

  Pour obtenir la liste des événements de déploiement bleu/vert, consultez Événements de déploiement bleu/vert.

Les informations collectées sont les suivantes :

- Date et heure de l'évènement.
- Nom de la source et type de source de l'évènement
- Message associé à l'événement
- Les notifications d'événements incluent des balises datant du moment où le message a été envoyé et peuvent ne pas refléter les balises au moment où l'événement s'est produit.

Consulter les RDS événements Amazon

Vous pouvez récupérer les informations d'événement suivantes pour vos ressources RDSAmazon :

- Nom de la ressource
- Type de ressource
- Heure de l'événement
- Résumé du message de l'événement

Vous pouvez accéder aux événements dans les sections suivantes du AWS Management Console :

- L'onglet Événements, qui affiche les événements des dernières 24 heures.
- Le tableau des événements récents de la section Journaux et événements de l'onglet Bases de données, qui peut afficher les événements survenus au cours des deux dernières semaines.

Vous pouvez également récupérer des événements à l'aide de la AWS CLI commande describe-events ou de l'DescribeEventsRDSAPIopération. Si vous utilisez le AWS CLI ou RDS API pour afficher les événements, vous pouvez récupérer les événements des 14 derniers jours au maximum.

> **Note**
>
> Si vous devez stocker des événements pendant de longues périodes, vous pouvez envoyer RDS des événements Amazon à EventBridge. Pour plus d'informations, consultez Création d'une règle qui se déclenche sur un événement Amazon RDS.

Pour une description des événements RDSAmazon , consultezCatégories d'événements Amazon RDS et messages d'événements .

Pour accéder à des informations détaillées sur les événements utilisant AWS CloudTrail, notamment les paramètres de demande, voirCloudTrail événements.

Console

Pour consulter tous les RDS événements Amazon des dernières 24 heures

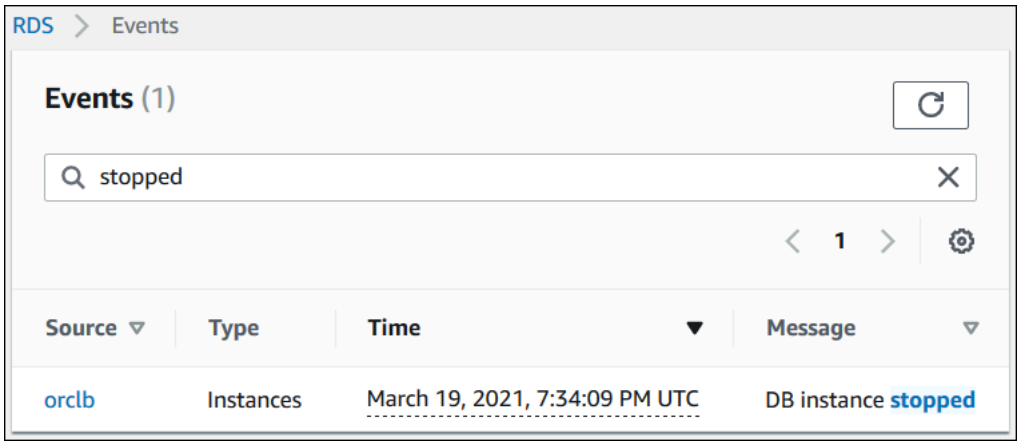
1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Dans le volet de navigation, sélectionnez Events.

   Les évènements disponibles s'affichent sous forme de liste.

3. (Facultatif) Entrez un terme de recherche pour filtrer vos résultats.

   L'exemple suivant montre une liste d'événements filtrés par les caractères **`stopped`**.



AWS CLI

Pour afficher tous les événements générés au cours de la dernière heure, appelez la commande describe-events sans paramètres.

```
aws rds describe-events
```

L'exemple de sortie suivant montre qu'une instance de base de données a été arrêtée.

```
{
    "Events": [
        {
            "EventCategories": [
                "notification"
            ],
            "SourceType": "db-instance",
            "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:testinst",
            "Date": "2022-04-22T21:31:00.681Z",
            "Message": "DB instance stopped",
            "SourceIdentifier": "testinst"
        }
    ]
}
```

Pour afficher tous les RDS événements Amazon des 10080 dernières minutes (7 jours), appelez la AWS CLI commande describe-events et définissez le `--duration` paramètre sur. `10080`

```
aws rds describe-events --duration 10080
```

L'exemple suivant montre les événements dans la plage de temps spécifiée pour l'instance de base de données *`test-instance`*.

```
aws rds describe-events \
    --source-identifier test-instance \
    --source-type db-instance \
    --start-time 2022-03-13T22:00Z \
    --end-time 2022-03-13T23:59Z
```

L'exemple de sortie suivant montre l'état d'une sauvegarde.

```
{
    "Events": [
        {
            "SourceType": "db-instance",
            "SourceIdentifier": "test-instance",
            "EventCategories": [
                "backup"
            ],
            "Message": "Backing up DB instance",
            "Date": "2022-03-13T23:09:23.983Z",
            "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
        },
        {
            "SourceType": "db-instance",
            "SourceIdentifier": "test-instance",
            "EventCategories": [
                "backup"
            ],
            "Message": "Finished DB Instance backup",
            "Date": "2022-03-13T23:15:13.049Z",
            "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance"
        }
    ]
}
```

API

Vous pouvez consulter tous les événements des RDS instances Amazon des 14 derniers jours en appelant l'DescribeEventsRDSAPIopération et en définissant le `Duration` paramètre sur`20160`.

Utilisation des notifications d'RDSévénements Amazon

Amazon RDS utilise le service Amazon Simple Notification (AmazonSNS) pour envoyer une notification lorsqu'un RDS événement Amazon se produit. Ces notifications peuvent se présenter sous n'importe quel formulaire de notification pris en charge par Amazon SNS pour une AWS région, tel qu'un e-mail, un SMS ou un appel vers un HTTP terminal.

Rubriques

- Présentation de la notification RDS d'événements Amazon
- Octroi d'autorisations de publication de notifications dans une rubrique Amazon SNS
- S'abonner aux notifications d'RDSévénements Amazon
- Balises et attributs de notification d'RDSévénements Amazon
- Répertorier les abonnements aux notifications d'RDSévénements Amazon
- Modifier un abonnement aux notifications RDS d'événements Amazon
- Ajouter un identifiant de source à un abonnement aux notifications d'RDSévénements Amazon
- Supprimer un identifiant source d'un abonnement aux notifications d'RDSévénements Amazon
- Liste des catégories de notifications RDS d'événements Amazon
- Supprimer un abonnement aux notifications RDS d'événements Amazon

Présentation de la notification RDS d'événements Amazon

Amazon RDS regroupe les événements dans des catégories auxquelles vous pouvez vous abonner afin que vous puissiez être averti lorsqu'un événement de cette catégorie se produit.

Rubriques

- RDSressources éligibles à l'abonnement à un événement
- Procédure de base pour s'abonner aux notifications d'RDSévénements Amazon
- Envoi de notifications RDS d'événements
- Facturation des notifications RDS d'événements Amazon
- Exemples d'RDSévénements Amazon utilisant Amazon EventBridge

RDSressources éligibles à l'abonnement à un événement

Vous pouvez vous abonner à une catégorie d'évènement pour les ressources suivantes :

- instance de base de données
- Snapshot DB
- Groupe de paramètres de base de données
- Groupe de sécurité de base de données
- RDSProxy
- Versions de moteur personnalisées

Par exemple, si vous vous abonnez à la catégorie de sauvegarde d'une instance de base de données donnée, vous recevez une notification chaque fois que survient un évènement lié à la sauvegarde et qui affecte l'instance de base de données. Si vous vous abonnez à la catégorie de modification de configuration pour une instance de base de données, vous recevez une notification en cas de modification de l'instance de base de données. Vous recevez également une notification en cas de modification d'un abonnement à une notification d'évènements.

Vous pouvez créer plusieurs abonnements différents. Par exemple, vous pouvez vouloir créer un abonnement qui reçoit toutes les notifications d'évènements pour l'ensemble des instances de base de données, et un autre incluant uniquement les évènements critiques pour un sous-ensemble des instances de base de données. Pour le deuxième abonnement, spécifiez une ou plusieurs instances de base de données dans le filtre.

Procédure de base pour s'abonner aux notifications d'RDSévénements Amazon

Le processus de souscription aux notifications d'RDSévénements Amazon est le suivant :

1. Vous créez un abonnement aux notifications d'RDSévénements Amazon à l'aide de la RDS console Amazon AWS CLI, ouAPI.

   Amazon RDS utilise le ARN nom d'un SNS sujet Amazon pour identifier chaque abonnement. La RDS console Amazon crée le ARN pour vous lorsque vous créez l'abonnement. Créez le à ARN l'aide de la SNS console Amazon AWS CLI, du ou de l'Amazon SNSAPI.
2. Amazon RDS envoie un e-mail ou un SMS message d'approbation aux adresses que vous avez indiquées avec votre abonnement.
3. Pour confirmer votre abonnement, cliquez sur le lien dans la notification que vous avez reçue.
4. La RDS console Amazon met à jour la section My Event Subscriptions avec le statut de votre abonnement.

5. Amazon RDS commence à envoyer les notifications aux adresses que vous avez fournies lors de la création de l'abonnement.

Pour en savoir plus sur la gestion des identités et des accès lors de l'utilisation d'AmazonSNS, consultez la section Gestion des identités et des accès dans Amazon SNS dans le manuel Amazon Simple Notification Service Developer Guide.

Vous pouvez l'utiliser AWS Lambda pour traiter les notifications d'événements provenant d'une instance de base de données. Pour plus d'informations, consultez la section Utilisation AWS Lambda avec Amazon RDS dans le manuel du AWS Lambda développeur.

Envoi de notifications RDS d'événements

Amazon RDS envoie des notifications aux adresses que vous fournissez lors de la création de l'abonnement. La notification peut inclure des attributs de message fournissant des métadonnées structurées relatives au message. Pour plus d'informations sur les attributs de message, consultez Catégories d'événements Amazon RDS et messages d'événements .

Les notifications d'évènement peuvent prendre jusqu'à cinq minutes pour être livrées.

> **Important**
>
> Amazon RDS ne garantit pas l'ordre des événements envoyés dans un flux d'événements. L'ordre des évènements est susceptible de changer.

Lorsqu'Amazon SNS envoie une notification à un abonné HTTP ou à un point de HTTPS terminaison, le corps du POST message envoyé au point de terminaison contient un JSON document. Pour plus d'informations, consultez le SNSmessage et les JSON formats Amazon dans le guide du développeur Amazon Simple Notification Service.

Vous pouvez configurer SNS pour vous avertir par SMS. Pour plus d'informations, consultez la section Messagerie texte mobile (SMS) dans le guide du développeur Amazon Simple Notification Service.

Pour désactiver les notifications sans supprimer un abonnement, choisissez Non pour Enabled dans la RDS console Amazon. Vous pouvez également définir le `Enabled` paramètre à `false` l'aide de AWS CLI ou Amazon RDSAPI.

Facturation des notifications RDS d'événements Amazon

La facturation des notifications RDS d'événements Amazon s'effectue via AmazonSNS. SNSDes frais Amazon s'appliquent lors de l'utilisation des notifications d'événements. Pour plus d'informations sur la SNS facturation Amazon, consultez la tarification d'Amazon Simple Notification Service.

Exemples d'RDSévénements Amazon utilisant Amazon EventBridge

Les exemples suivants illustrent différents types d'RDSévénements Amazon sous forme de JSON format. Pour un didacticiel qui explique comment capturer et afficher des événements au JSON format, voirTutoriel : Consigner les modifications de l'état d'une instance de base de données à l'aide EventBridge.

Rubriques

- Exemple d'évènement d'instance de base de données
- Exemple d'évènement de groupe de paramètres de base de données
- Exemple d'évènement d'instantané de base de données

Exemple d'évènement d'instance de base de données

Voici un exemple d'événement d'instance de base de données au JSON format. L'événement indique qu'un basculement multi-AZ a été RDS effectué pour l'instance nommée. `my-db-instance` L'ID de l'événement est RDS EVENT -0049.

```
{
  "version": "0",
  "id": "68f6e973-1a0c-d37b-f2f2-94a7f62ffd4e",
  "detail-type": "RDS DB Instance Event",
  "source": "aws.rds",
  "account": "123456789012",
  "time": "2018-09-27T22:36:43Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:rds:us-east-1:123456789012:db:my-db-instance"
  ],
  "detail": {
    "EventCategories": [
      "failover"
    ],
    "SourceType": "DB_INSTANCE",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:db:my-db-instance",
```

```
    "Date": "2018-09-27T22:36:43.292Z",
    "Message": "A Multi-AZ failover has completed.",
    "SourceIdentifier": "my-db-instance",
    "EventID": "RDS-EVENT-0049"
  }
}
```

Exemple d'évènement de groupe de paramètres de base de données

Voici un exemple d'événement de groupe de paramètres de base de données au JSON format. L'évènement indique que le paramètre `time_zone` a été mis à jour dans le groupe de paramètres `my-db-param-group`. L'ID de l'événement est RDS EVENT -0037.

```
{
  "version": "0",
  "id": "844e2571-85d4-695f-b930-0153b71dcb42",
  "detail-type": "RDS DB Parameter Group Event",
  "source": "aws.rds",
  "account": "123456789012",
  "time": "2018-10-06T12:26:13Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:rds:us-east-1:123456789012:pg:my-db-param-group"
  ],
  "detail": {
    "EventCategories": [
      "configuration change"
    ],
    "SourceType": "DB_PARAM",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:pg:my-db-param-group",
    "Date": "2018-10-06T12:26:13.882Z",
    "Message": "Updated parameter time_zone to UTC with apply method immediate",
    "SourceIdentifier": "my-db-param-group",
    "EventID": "RDS-EVENT-0037"
  }
}
```

Exemple d'évènement d'instantané de base de données

Voici un exemple d'événement de capture d'écran de base de données au JSON format. L'évènement montre la suppression de l'instantané nommé `my-db-snapshot`. L'ID de l'événement est RDS EVENT -0041.

```
{
  "version": "0",
  "id": "844e2571-85d4-695f-b930-0153b71dcb42",
  "detail-type": "RDS DB Snapshot Event",
  "source": "aws.rds",
  "account": "123456789012",
  "time": "2018-10-06T12:26:13Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:rds:us-east-1:123456789012:snapshot:rds:my-db-snapshot"
  ],
  "detail": {
    "EventCategories": [
      "deletion"
    ],
    "SourceType": "SNAPSHOT",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:snapshot:rds:my-db-snapshot",
    "Date": "2018-10-06T12:26:13.882Z",
    "Message": "Deleted manual snapshot",
    "SourceIdentifier": "my-db-snapshot",
    "EventID": "RDS-EVENT-0041"
  }
}
```

Octroi d'autorisations de publication de notifications dans une rubrique Amazon SNS

Pour autoriser Amazon RDS à publier des notifications sur une rubrique Amazon Simple Notification Service (Amazon SNS), associez une politique AWS Identity and Access Management (IAM) à la rubrique de destination. Pour plus d'informations sur les autorisations, consultez Exemples de cas pour le contrôle d'accès Amazon Simple Notification Service dans le Guide du développeur Amazon Simple Notification Service.

Par défaut, une rubrique Amazon SNS dispose d'une politique permettant à toutes les ressources Amazon RDS d'un même compte d'y publier des notifications. Vous pouvez attacher une politique personnalisée pour autoriser les notifications entre comptes ou pour restreindre l'accès à certaines ressources.

Voici un exemple de politique IAM que vous attachez à la rubrique Amazon SNS de destination. Il limite la rubrique aux instances de base de données dont les noms correspondent au préfixe spécifié. Pour utiliser cette politique, spécifiez les valeurs suivantes :

- `Resource` : Amazon Resource Name (ARN) de votre rubrique Amazon SNS
- `SourceARN` : ARN de votre ressource RDS
- `SourceAccount`— Votre Compte AWS identifiant

Pour consulter la liste des types de ressources et leurs caractéristiques ARNs, consultez la section Ressources définies par Amazon RDS dans le Service Authorization Reference.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "events.rds.amazonaws.com"
      },
      "Action": [
        "sns:Publish"
      ],
      "Resource": "arn:aws:sns:us-east-1:123456789012:topic_name",
      "Condition": {
```

```
        "ArnLike": {
          "aws:SourceArn": "arn:aws:rds:us-east-1:123456789012:db:prefix-*"
        },
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

S'abonner aux notifications d'RDSévénements Amazon

Le moyen le plus simple de créer un abonnement est d'utiliser la RDS console. Si vous choisissez de créer des abonnements aux notifications d'événements à l'aide du CLI ouAPI, vous devez créer une rubrique Amazon Simple Notification Service et vous abonner à cette rubrique via la SNS console Amazon ou Amazon SNSAPI. Vous devrez également conserver le nom de ressource Amazon (ARN) du sujet, car il est utilisé lors de la soumission de CLI commandes ou d'APIopérations. Pour plus d'informations sur la création d'un SNS sujet et l'abonnement à celui-ci, consultez Getting started with Amazon SNS dans le manuel Amazon Simple Notification Service Developer Guide.

Vous pouvez spécifier le type de source dont vous souhaitez être informé et la RDS source Amazon qui déclenche l'événement :

Source type (Type de source)

Type de source. Par exemple, Source Type (Type de source) pourrait être Instances. Vous devez choisir un type de source.

Resources pour inclure

Les RDS ressources Amazon qui génèrent les événements. Par exemple, vous pouvez choisir Sélectionner des instances spécifiques, puis myDBInstance1.

Le tableau suivant explique le résultat lorsque vous spécifiez ou ne spécifiez pas ***Resources*** à inclure.

Ressources à inclure	Description	Exemple
Spécifié	RDSvous informe de tous les événements relatifs à la ressource spécifiée uniquement.	Si votre type de source est Instances et que votre ressource est myDBInstance1, vous RDS informe de tous les événements pour myDBInstance 1 uniquement.
Non spécifiée	RDSvous informe des événements relatifs au type de source spécifié pour toutes vos RDS ressources Amazon.	Si votre type de source est Instances, vous RDS informe

Ressources à inclure	Description	Exemple
		de tous les événements liés aux instances dans votre compte.

Un abonné à un SNS sujet Amazon reçoit par défaut tous les messages publiés sur le sujet. Pour recevoir uniquement un sous-ensemble des messages, l'abonné doit attribuer une politique de filtre à l'abonnement à la rubrique. Pour plus d'informations sur le filtrage des SNS messages, consultez le filtrage des SNS messages Amazon dans le manuel Amazon Simple Notification Service Developer Guide

Console

Pour vous abonner à la notification RDS d'événements

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Abonnements aux évènements.
3. Dans le volet Abonnements aux événements, choisissez Créer un abonnement aux événements.
4. Entrez les détails de votre abonnement comme suit :

    a. Dans le champ Nom, entrez un nom pour l'abonnement à la notification d'événements.

    b. Pour Send notification to: (Envoyer les notifications à), effectuez l'une des opérations suivantes :

    - Choisissez New email topic (Nouvelle rubrique d'e-mail). Saisissez un nom pour votre rubrique d'e-mail et une liste de bénéficiaires. Nous vous recommandons de configurer les abonnements aux événements sur la même adresse e-mail que celle du contact principal de votre compte. Les recommandations, les événements de service et les messages de santé personnels sont envoyés via différents canaux. Les abonnements sur la même adresse e-mail garantissent que tous les messages sont regroupés en un seul endroit.
    - Choisissez Amazon Resource Name (ARN). Choisissez ensuite Amazon existant SNS ARN pour un SNS sujet Amazon.

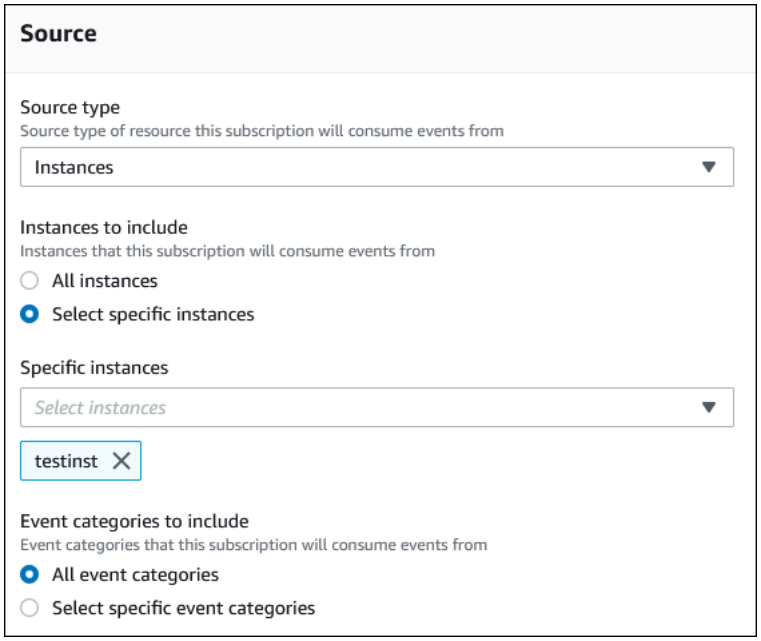
      Si vous souhaitez utiliser une rubrique activée pour le chiffrement côté serveur (SSE), accordez à Amazon RDS les autorisations nécessaires pour accéder au. AWS KMS

key Pour plus d'informations, consultez Activer la compatibilité entre les sources d'événements provenant de AWS services et de sujets chiffrés dans le manuel Amazon Simple Notification Service Developer Guide.

c. Pour Type de source, choisissez un type de source. Par exemple, choisissez Instances ou Parameter groups (Groupes de paramètres) (Instantanés de clusters).

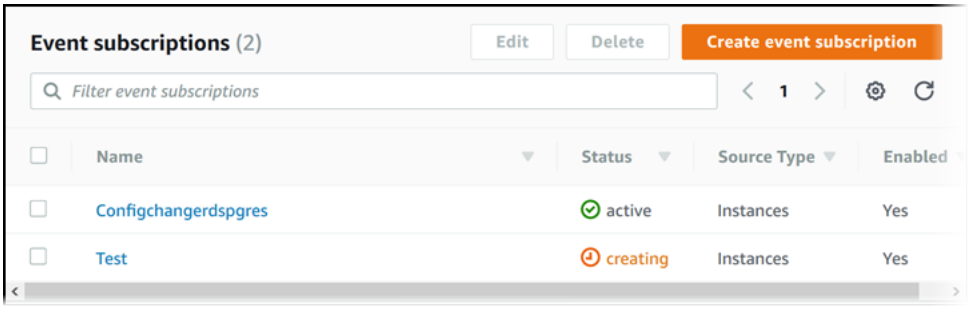
d. Choisissez les catégories d'événements et les ressources pour lesquelles vous souhaitez recevoir des notifications d'événements.

   L'exemple suivant configure les notifications d'événements pour l'instance de base de données nommée `testinst`.



e. Sélectionnez Create (Créer).

La RDS console Amazon indique que l'abonnement est en cours de création.



AWS CLI

Pour vous abonner à la notification d'RDSévénement, utilisez la AWS CLI `create-event-subscription`commande. Incluez les paramètres requis suivants :

- `--subscription-name`
- `--sns-topic-arn`

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-event-subscription \
    --subscription-name myeventsubscription \
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:myawsuser-RDS \
    --enabled
```

Dans Windows :

```
aws rds create-event-subscription ^
    --subscription-name myeventsubscription ^
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:myawsuser-RDS ^
    --enabled
```

API

Pour vous abonner aux notifications RDS d'événements Amazon, appelez la RDS API fonction Amazon `CreateEventSubscription`. Incluez les paramètres requis suivants :

- `SubscriptionName`
- `SnsTopicArn`

Balises et attributs de notification d'RDSévénements Amazon

Lorsqu'Amazon RDS envoie une notification d'événement à Amazon Simple Notification Service (SNS) ou à Amazon EventBridge, la notification contient des attributs de message et des balises d'événement. RDSenvoie les attributs du message séparément avec le message, tandis que les balises d'événement se trouvent dans le corps du message. Utilisez les attributs du message et les RDS balises Amazon pour ajouter des métadonnées à vos ressources. Vous pouvez modifier ces balises avec vos propres notations sur les instances de base de données Pour plus d'informations sur le balisage des RDS ressources Amazon, consultezMarquage d' Amazon RDS.

Par défaut, Amazon SNS et Amazon EventBridge reçoivent tous les messages qui leur sont envoyés. SNSet EventBridge peut filtrer le message et envoyer les notifications vers le mode de communication préféré, tel qu'un e-mail, un SMS ou un appel vers un HTTP terminal.

> **Note**
>
> La notification envoyée par e-mail ou SMS ne comportera pas de balises d'événement.

Le tableau suivant indique les attributs des messages pour les RDS événements envoyés à l'abonné à la rubrique.

Attribut RDS d'événement Amazon	Description
EventID	Identifiant du message RDS d'événement, par exemple, RDS EVENT -0006.
Ressource	L'ARNidentifiant de la ressource émettant l'événement, par exemple,`arn:aws:rds:ap-southeast-2:123456789012:db:database-1` .

Les RDS balises fournissent des données sur la ressource affectée par l'événement de service. RDSajoute l'état actuel des balises dans le corps du message lorsque la notification est envoyée à SNS ou EventBridge.

Pour plus d'informations sur le filtrage des attributs des messages pourSNS, consultez le filtrage des SNS messages Amazon dans le manuel Amazon Simple Notification Service Developer Guide.

Pour plus d'informations sur le filtrage des balises d'événements pour EventBridge, consultez la section Opérateurs de comparaison à utiliser dans les modèles d'événements sur Amazon EventBridge dans le guide de EventBridge l'utilisateur Amazon.

Pour plus d'informations sur le filtrage des balises basées sur la charge utile pourSNS, consultez Présentation du filtrage des messages basé sur la charge utile pour Amazon SNS

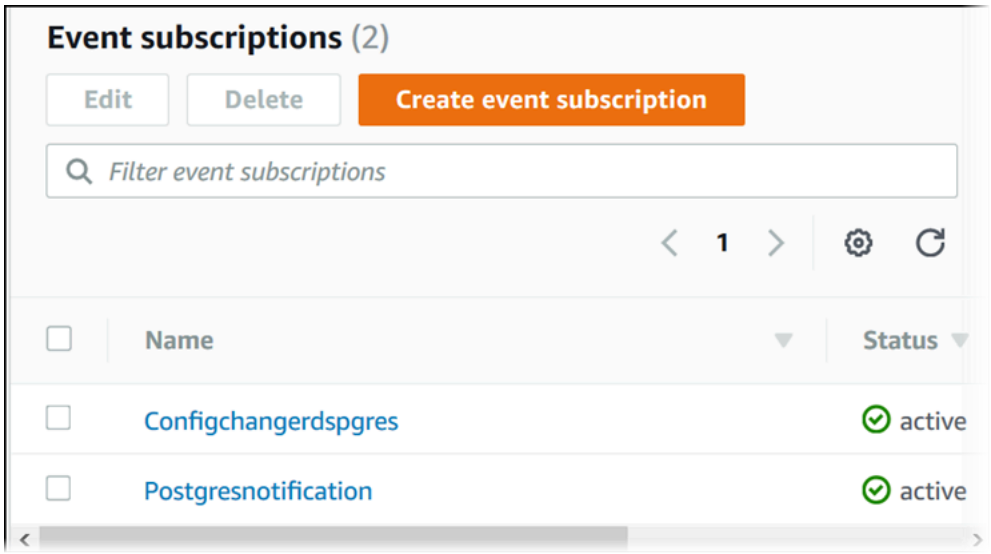
Répertorier les abonnements aux notifications d'RDSÉvénements Amazon

Vous pouvez répertorier vos abonnements actuels aux notifications RDS d'événements Amazon.

Console

Pour répertorier vos abonnements actuels aux notifications RDS d'événements Amazon

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Abonnements aux événements. Le volet Abonnements aux événements affiche tous les abonnements aux notifications d'événements.



AWS CLI

Pour répertorier vos abonnements actuels aux notifications d'RDSÉvénements Amazon, utilisez la AWS CLI `describe-event-subscriptions`commande.

Example

L'exemple suivant décrit tous les abonnements aux évènements.

```
aws rds describe-event-subscriptions
```

L'exemple suivant décrit l'abonnement `myfirsteventsubscription`.

```
aws rds describe-event-subscriptions --subscription-name myfirsteventsubscription
```

API

Pour répertorier vos abonnements actuels aux notifications d'RDSévénements Amazon, lancez l'RDSAPI`DescribeEventSubscriptions`action Amazon.

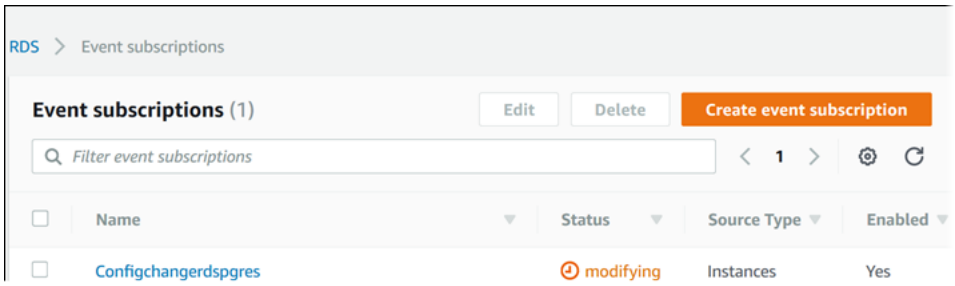
Modifier un abonnement aux notifications RDS d'événements Amazon

Après avoir créé un abonnement, vous pouvez modifier le nom de l'abonnement, l'identifiant de la source, les catégories ou le sujetARN.

Console

Pour modifier un abonnement aux notifications RDS d'événements Amazon

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Abonnements aux événements.
3. Dans le volet Abonnements aux événements, choisissez l'abonnement que vous voulez modifier, puis choisissez Modifier.
4. Apportez les modifications requises à l'abonnement dans la section Cible ou Source.
5. Choisissez Edit. La RDS console Amazon indique que l'abonnement est en cours de modification.



AWS CLI

Pour modifier un abonnement aux notifications d'RDSÉvénements Amazon, utilisez la AWS CLI `modify-event-subscription`commande. Incluez le paramètre requis suivant :

- `--subscription-name`

Example

Le code suivant active `myeventsubscription`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-event-subscription \
    --subscription-name myeventsubscription \
    --enabled
```

Dans Windows :

```
aws rds modify-event-subscription ^
    --subscription-name myeventsubscription ^
    --enabled
```

API

Pour modifier un RDS événement Amazon, appelez l'RDSAPIopération Amazon `ModifyEventSubscription`. Incluez le paramètre requis suivant :

- `SubscriptionName`

Ajouter un identifiant de source à un abonnement aux notifications d'RDSévénements Amazon

Vous pouvez ajouter un identifiant de source (la RDS source Amazon générant l'événement) à un abonnement existant.

Console

Vous pouvez facilement ajouter ou supprimer des identifiants de source à l'aide de la RDS console Amazon en les sélectionnant ou en les désélectionnant lors de la modification d'un abonnement. Pour de plus amples informations, veuillez consulter Modifier un abonnement aux notifications RDS d'événements Amazon.

AWS CLI

Pour ajouter un identifiant de source à un abonnement aux notifications d'RDSévénements Amazon, utilisez la AWS CLI `add-source-identifier-to-subscription`commande. Incluez les paramètres requis suivants :

- `--subscription-name`
- `--source-identifier`

Example

L'exemple suivant ajoute l'identifiant source `mysqldb` à l'abonnement `myrdseventsubscription`

Pour LinuxmacOS, ou Unix :

```
aws rds add-source-identifier-to-subscription \
    --subscription-name myrdseventsubscription \
    --source-identifier mysqldb
```

Dans Windows :

```
aws rds add-source-identifier-to-subscription ^
    --subscription-name myrdseventsubscription ^
    --source-identifier mysqldb
```

API

Pour ajouter un identifiant de source à un abonnement aux notifications d'RDSévénements Amazon, appelez Amazon RDS API `AddSourceIdentifierToSubscription`. Incluez les paramètres requis suivants :

- `SubscriptionName`
- `SourceIdentifier`

Supprimer un identifiant source d'un abonnement aux notifications d'RDSévénements Amazon

Vous pouvez supprimer un identifiant de source (la RDS source Amazon générant l'événement) d'un abonnement si vous ne souhaitez plus être informé des événements liés à cette source.

Console

Vous pouvez facilement ajouter ou supprimer des identifiants de source à l'aide de la RDS console Amazon en les sélectionnant ou en les désélectionnant lors de la modification d'un abonnement. Pour de plus amples informations, veuillez consulter Modifier un abonnement aux notifications RDS d'événements Amazon.

AWS CLI

Pour supprimer un identifiant de source d'un abonnement aux notifications d'RDSévénements Amazon, utilisez la AWS CLI `remove-source-identifier-from-subscription`commande. Incluez les paramètres requis suivants :

- `--subscription-name`
- `--source-identifier`

Example

L'exemple suivant supprime l'identifiant source `mysqldb` de l'abonnement `myrdseventsubscription`.

Pour LinuxmacOS, ou Unix :

```
aws rds remove-source-identifier-from-subscription \
    --subscription-name myrdseventsubscription \
    --source-identifier mysqldb
```

Dans Windows :

```
aws rds remove-source-identifier-from-subscription ^
    --subscription-name myrdseventsubscription ^
    --source-identifier mysqldb
```

API

Pour supprimer un identifiant de source d'un abonnement aux notifications d'RDSévénements Amazon, utilisez la RDS API `RemoveSourceIdentifierFromSubscription`commande Amazon. Incluez les paramètres requis suivants :

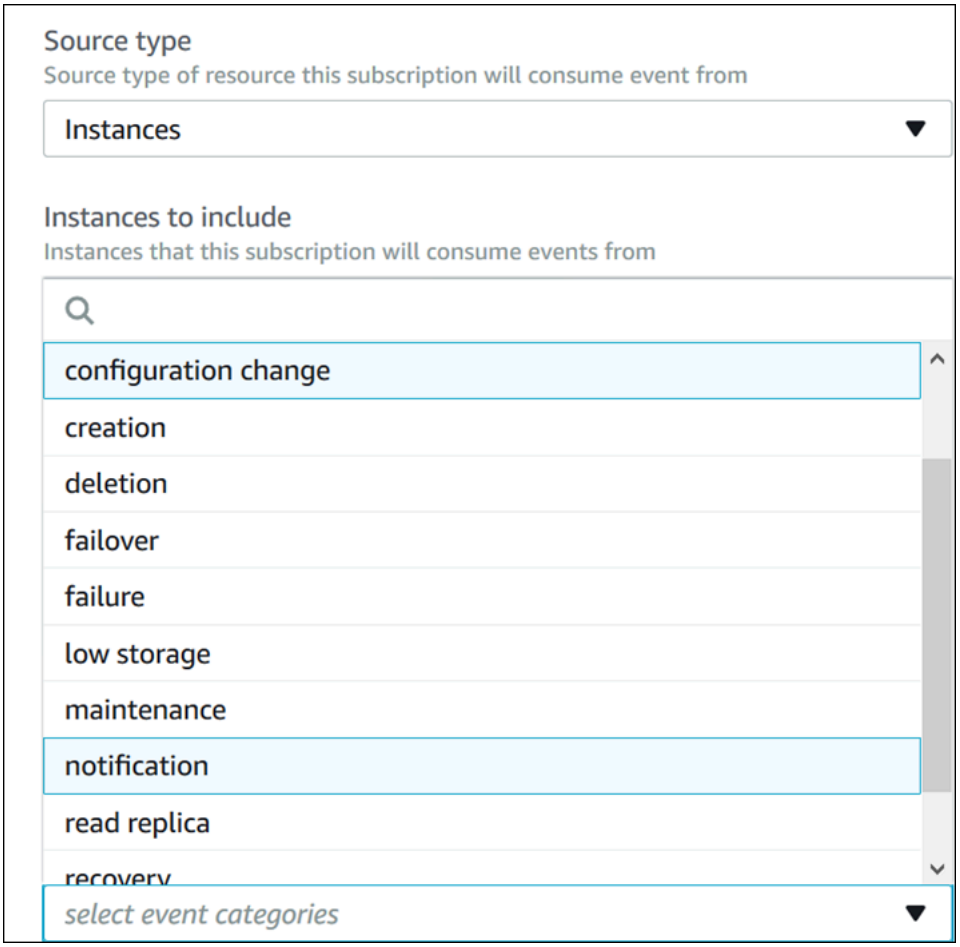
- `SubscriptionName`
- `SourceIdentifier`

Liste des catégories de notifications RDS d'événements Amazon

Tous les évènements d'un type de ressource sont regroupés en catégories. Pour afficher la liste des catégories disponibles, utilisez les procédures suivantes.

Console

Lorsque vous créez ou modifiez un abonnement aux notifications d'événements, les catégories d'événements sont affichées dans la RDS console Amazon. Pour de plus amples informations, veuillez consulter Modifier un abonnement aux notifications RDS d'événements Amazon.



AWS CLI

Pour répertorier les catégories de notifications d'RDSévénements Amazon, utilisez la AWS CLI `describe-event-categories`commande. Cette commande n'a aucun paramètre requis.

Example

```
aws rds describe-event-categories
```

API

Pour répertorier les catégories de notifications d'RDSévénements Amazon, utilisez la RDS API `DescribeEventCategories`commande Amazon. Cette commande n'a aucun paramètre requis.

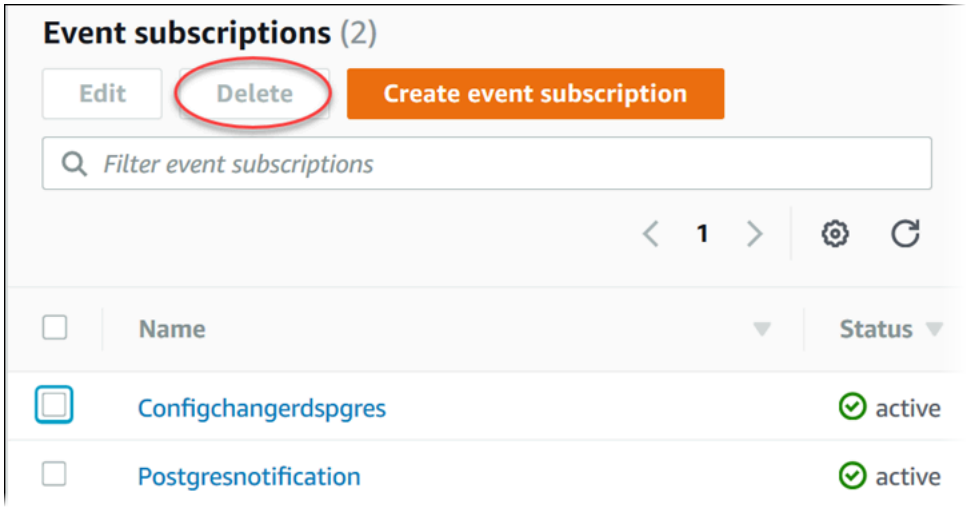
Supprimer un abonnement aux notifications RDS d'événements Amazon

Vous pouvez supprimer un abonnement lorsque vous n'en avez plus besoin. Tous les abonnés à la rubrique ne reçoivent plus les notifications d'évènements spécifiées par l'abonnement.

Console

Pour supprimer un abonnement aux notifications RDS d'événements Amazon

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez DB Event Subscriptions (Abonnements aux évènements de base de données).
3. Dans le volet My DB Event Subscriptions (Mes abonnements aux évènements de base de données), cliquez sur l'abonnement que vous souhaitez supprimer.
4. Sélectionnez Delete (Supprimer).
5. La RDS console Amazon indique que l'abonnement est en cours de suppression.



AWS CLI

Pour supprimer un abonnement aux notifications d'RDSévénements Amazon, utilisez la AWS CLI `delete-event-subscription`commande. Incluez le paramètre requis suivant :

- `--subscription-name`

Example

L'exemple suivant supprime l'abonnement `myrdssubscription`.

```
aws rds delete-event-subscription --subscription-name myrdssubscription
```

API

Pour supprimer un abonnement aux notifications d'RDSévénements Amazon, utilisez la RDS API `DeleteEventSubscription`commande. Incluez le paramètre requis suivant :

- `SubscriptionName`

Création d'une règle qui se déclenche sur un événement Amazon RDS

Amazon vous permet EventBridge d'automatiser les AWS services et de répondre aux événements du système tels que les problèmes de disponibilité des applications ou les modifications des ressources.

Rubriques

- Création de règles pour envoyer des événements Amazon RDS à Amazon EventBridge
- Tutoriel : Consigner les modifications de l'état d'une instance de base de données à l'aide EventBridge

Création de règles pour envoyer des événements Amazon RDS à Amazon EventBridge

Vous pouvez écrire des règles simples pour préciser les événements Amazon RDS qui vous intéressent et les actions automatisées à effectuer quand un événement correspond à une règle. Vous pouvez définir diverses cibles, telles qu'une AWS Lambda fonction ou une rubrique Amazon SNS, qui reçoivent des événements au format JSON. Par exemple, vous pouvez configurer Amazon RDS Amazon pour envoyer des événements à Amazon EventBridge chaque fois qu'une instance de base de données est créée ou supprimée. Pour plus d'informations, consultez le guide de l'utilisateur Amazon CloudWatch Events et le guide de EventBridge l'utilisateur Amazon.

Pour créer une règle qui se déclenche sur un évènement RDS :

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Sous Évènements dans le panneau de navigation, choisissez Règles.
3. Choisissez Create rule.
4. Dans Event source, effectuez l'une des opérations suivantes :
    a. Choisissez Modèle d'événement.
    b. Pour Service Name (Nom du service), choisissez Relational Database Service (RDS) (Service de base de données relationnelle).
    c. Pour Event Type (Type d'évènement), choisissez le type de ressource Amazon RDS qui déclenche l'évènement. Par exemple, si une instance de base de données déclenche l'événement, choisissez RDS DB Instance Event (Événement relatif à l'instance de base de données RDS).

5. Pour les cibles, choisissez Ajouter une cible et choisissez le AWS service qui doit agir lorsqu'un événement du type sélectionné est détecté.
6. Dans les autres champs de cette section, entrez des informations spécifiques à ce type de cible, le cas échéant.
7. Pour de nombreux types de cibles, EventBridge nécessite des autorisations pour envoyer des événements à la cible. Dans ces cas, EventBridge vous pouvez créer le rôle IAM nécessaire au déroulement de votre événement :

   - Pour créer un rôle IAM automatiquement, choisissez Create a new role for this specific resource.
   - Pour utiliser un rôle IAM que vous avez créé auparavant, choisissez Utiliser le rôle existant.
8. Le cas échéant, répétez les étapes 5 à 7 afin d'ajouter une autre cible pour cette règle.
9. Sélectionnez Configure details. Dans Rule definition, saisissez un nom et une description pour la règle.

   Le nom de la règle doit être unique au sein de cette région.
10. Choisissez Create rule.

Pour plus d'informations, consultez la section Création d'une EventBridge règle déclenchant un événement dans le guide de CloudWatch l'utilisateur Amazon.

Tutoriel : Consigner les modifications de l'état d'une instance de base de données à l'aide EventBridge

Dans ce didacticiel, vous allez créer une AWS Lambda fonction qui enregistre les changements d'état d'une instance Amazon RDS. Vous créez ensuite une règle qui exécute la fonction chaque fois qu'il y a un changement d'état d'une instance de base de données RDS existante. Le didacticiel suppose que vous avez d'une petite instance de test en cours d'exécution, que vous pouvez arrêter momentanément.

> **Important**
>
> N'appliquez pas ce tutoriel à une instance de base de données de production en cours d'exécution.

Rubriques

- Étape 1 : Création d'une AWS Lambda fonction
- Étape 2 : création d'une règle
- Étape 3 : test de la règle

Étape 1 : Création d'une AWS Lambda fonction

Créez une fonction Lambda pour enregistrer les évènements de changement d'état. Vous spécifiez cette fonction lors de la création de votre règle.

Pour créer une fonction Lambda

1. Ouvrez la AWS Lambda console à l'adresse https://console.aws.amazon.com/lambda/.
2. Si vous utilisez Lambda pour la première fois, une page de bienvenue s'affiche. Sélectionnez Pour commencer. Sinon, choisissez Créer la fonction.
3. Choisissez Créer à partir de scratch.
4. Sur la page Create function (Créer une fonction), procédez de la façon suivante :
    a. Saisissez un nom et une description pour la fonction Lambda. Par exemple, nommez la fonction **`RDSInstanceStateChange`**.
    b. Dans Runtime, sélectionnez Node.js 16x.
    c. Pour Architecture, choisissez x86_64.
    d. Pour Execution role (Rôle d'exécution), effectuez l'une des opérations suivantes :
        - Choisissez Create a new role with basic Lambda permissions (Créer un rôle avec les autorisations Lambda standard).
        - Pour Existing role (Rôle existant), sélectionnez Use an existing role (Utiliser un rôle existant). Choisissez le rôle que vous voulez utiliser.
    e. Choisissez Créer une fonction.
5. Sur la InstanceStateChange page RDS, procédez comme suit :
    a. Dans Code source (Source de code), sélectionnez index.js.
    b. Dans le panneau index.js, supprimez le code existant.
    c. Saisissez le code suivant :

```
console.log('Loading function');
```

```
exports.handler = async (event, context) => {
    console.log('Received event:', JSON.stringify(event));
};
```

d. Choisissez Deploy (Déployer).

Étape 2 : création d'une règle

Créez une règle pour exécuter votre fonction Lambda chaque fois que vous lancez une instance Amazon RDS.

Pour créer la EventBridge règle

1. Ouvrez la EventBridge console Amazon à l'adresse https://console.aws.amazon.com/events/.
2. Dans le volet de navigation, choisissez Règles.
3. Choisissez Créer une règle.
4. Saisissez un nom et une description pour la règle. Par exemple, entrez **`RDSInstanceStateChangeRule`**.
5. Sélectionnez Rule with an event pattern (Règle avec un modèle d'événement), puis sélectionnez Next (Suivant).
6. Dans Source de l'événement, choisissez AWS des événements ou des événements EventBridge partenaires.
7. Faites défiler la page vers le bas jusqu'à la section Event pattern (Modèle d'événement).
8. Pour Event source (Source d’événement), choisissez Services AWS.
9. Pour Service AWS , choisissez Relational Database Service (RDS).
10. Pour Event type (Type d'évènement), sélectionnez RDS DB Instance Event (évènement d'instance de base de données RDS).
11. Laissez le modèle d'événement par défaut. Ensuite, sélectionnez Suivant.
12. Pour Types de cibles, choisissez service AWS .
13. Pour Select a target (Sélectionner une cible), choisissez Lambda Function (Fonction Lambda).
14. Dans Function (Fonction), choisissez la fonction Lambda que vous avez créée. Ensuite, sélectionnez Suivant.
15. Dans la rubrique Configure tags (Configurer les balises), sélectionnez Next (Suivant).
16. Passez en revue les étapes de votre règle. Puis, choisissez Create rule (Créer une règle).

Étape 3 : test de la règle

Pour tester votre règle, arrêtez une instance de base de données RDS. Après avoir attendu quelques minutes que l'instance s'arrête, vous pouvez vérifier que votre fonction Lambda a été appelée.

Pour tester votre règle en arrêtant une instance de base de données

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Arrêter une instance de base de données RDS.
3. Ouvrez la EventBridge console Amazon à l'adresse https://console.aws.amazon.com/events/.
4. Dans le volet de navigation, cliquez sur Rules (Règles), puis sur le nom de la règle que vous avez créée.
5. Dans Détails des règles, choisissez Surveillance.

   Vous êtes redirigé vers la CloudWatch console Amazon. Si vous n'êtes pas redirigé, cliquez sur Afficher les statistiques dans CloudWatch.
6. Dans All metrics (Toutes les métriques), cliquez sur le nom de la règle que vous avez créée.

   Le graphique doit indiquer que la règle a été appelée.
7. Dans le panneau de navigation, sélectionnez Groupes de journaux.
8. Cliquez sur le nom du groupe de journaux pour votre fonction Lambda (/aws/lambda/***function-name***).
9. Choisissez le nom du flux de journaux pour afficher les données fournies par la fonction concernant l'instance que vous avez lancée. Vous devez voir un évènement reçu semblable à ce qui suit :

```
{
    "version": "0",
    "id": "12a345b6-78c9-01d2-34e5-123f4ghi5j6k",
    "detail-type": "RDS DB Instance Event",
    "source": "aws.rds",
    "account": "111111111111",
    "time": "2021-03-19T19:34:09Z",
    "region": "us-east-1",
    "resources": [
        "arn:aws:rds:us-east-1:111111111111:db:testdb"
    ],
    "detail": {
        "EventCategories": [
```

```
            "notification"
        ],
        "SourceType": "DB_INSTANCE",
        "SourceArn": "arn:aws:rds:us-east-1:111111111111:db:testdb",
        "Date": "2021-03-19T19:34:09.293Z",
        "Message": "DB instance stopped",
        "SourceIdentifier": "testdb",
        "EventID": "RDS-EVENT-0087"
    }
}
```

Pour voir plus d'exemples d'événements RDS au format JSON, consultez Vue d'ensemble des événements pour Amazon RDS.

10. (Facultatif) Lorsque vous avez terminé, vous pouvez ouvrir la console Amazon RDS et lancer l'instance que vous avez arrêtée.

Catégories d'événements Amazon RDS et messages d'événements

Amazon RDS génère un nombre important d'événements dans des catégories auxquelles vous pouvez vous abonner à l'aide de la console Amazon RDS ou de l'API. AWS CLI

Rubriques

- Évènements de cluster de base de données
- Évènements d'instantané de cluster de base de données
- Évènements d'instance de base de données
- Évènements de groupe de paramètres de base de données
- Évènements de groupe de sécurité de base de données
- Évènements d'instantané de bases de données
- Événements RDS Proxy
- Événements de déploiement bleu/vert
- Événements de version du moteur personnalisés

Évènements de cluster de base de données

Le tableau suivant affiche la catégorie d'événement et la liste des événements lorsqu'un cluster de base de données est le type source.

Pour plus d'informations sur les déploiements de clusters de bases de données multi-AZ, consultez. Déploiements de clusters de bases de données multi-AZ pour Amazon RDS

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0016	Réinitialisation des informations d'identification principales.	Aucun
création	RDS-EVENT-0170	cluster de base de données créé.	Aucun
basculement	RDS-EVENT-0069	Le basculement du cluster a échoué. Vérifiez l'état de	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
		santé de vos instances de cluster et réessayez.	
basculement	RDS-EVENT-0070	Promouvoir à nouveau la primaire précédente :*name*.	Aucun
basculement	RDS-EVENT-0071	Basculement terminé vers l'instance de base de données :*name*.	Aucun
basculement	RDS-EVENT-0072	Démarrage du même basculement AZ vers l'instance de base de données :*name*.	Aucun
basculement	RDS-EVENT-0073	Basculement cross-AZ lancé vers une instance de base de données :*name*.	Aucun
échec	RDS-EVENT-0354	Vous ne pouvez pas créer le cluster de base de données en raison de ressources incompatibles. *message*.	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0355	Le cluster de base de données ne peut pas être créé en raison de limites de ressources insuffisantes. *message*.	*message*Cela inclut des détails sur la panne.
maintenance	RDS-EVENT-0156	Une mise à niveau de version mineure du moteur de base de données est disponible pour le cluster de base de données.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance	RDS-EVENT-0173	La version du moteur de cluster de base de données a été mise à niveau.	Les correctifs ont été appliqués au cluster de base de données.
maintenance	RDS-EVENT-0174	Le cluster de base de données est dans un état qui ne peut pas être mis à niveau.	Aucun
maintenance	RDS-EVENT-0176	La version majeure du moteur de cluster de base de données a été mise à niveau.	Aucun
maintenance	RDS-EVENT-0177	La mise à niveau du cluster de base de données est en cours.	Aucun
maintenance	RDS-EVENT-0286	La mise à niveau de *version_number* la version du moteur du cluster de base de données Le cluster reste en ligne.	Aucun
maintenance	RDS-EVENT-0287	Une mise à niveau nécessaire du système d'exploitation a été détectée.	Aucun
maintenance	RDS-EVENT-0288	La mise à niveau du système d'exploitation du cluster est en cours de démarrage.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance	RDS-EVENT-0289	La mise à niveau du système d'exploitation du cluster est terminée.	Aucun
maintenance	RDS-EVENT-0290	Le cluster de base de données a été corrigé : version source *version_number* =>*new_version_number* .	Aucun
maintenance	RDS-EVENT-0410	La pré-vérification a commencé pour la mise à niveau de la version du moteur de cluster de base de données.	Aucun
maintenance	RDS-EVENT-0412	La vérification préalable de la mise à niveau de la version du moteur de cluster de base de données a échoué ou a expiré.	Aucun
maintenance	RDS-EVENT-0413	Les tâches préalables à la mise à niveau du cluster de bases de données sont en cours.	Aucun
maintenance	RDS-EVENT-0414	Les tâches post-mise à niveau du cluster de bases de données sont en cours.	Aucun
maintenance	RDS-EVENT-0417	La mise à niveau de la version du moteur de cluster de base de données a démarré.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0172	Cluster renommé de *name* à*name*.	Aucun
réplica en lecture	RDS-EVENT-0411	La pré-vérification est terminée pour la mise à niveau de la version du moteur de cluster de base de données.	Aucun

Évènements d'instantané de cluster de base de données

Le tableau suivant affiche la catégorie d'événement et la liste des événements lorsqu'un instantané de cluster de base de données est le type source.

Catégorie	ID d'évènement RDS	Message	Remarques
sauvegarde	RDS-EVENT-0074	Création d'un instantané de cluster manuel.	Aucun
sauvegarde	RDS-EVENT-0075	Instantané de cluster manuel créé.	Aucun
notification	RDS-EVENT-0162	La tâche d'exportation de l'instantané de cluster a échoué.	Aucun
notification	RDS-EVENT-0163	La tâche d'exportation de l'instantané de cluster a été annulée.	Aucun
notification	RDS-EVENT-0164	La tâche d'exportation de l'instantané de cluster est terminée.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
sauvegarde	RDS-EVENT-0168	Création d'instantané de cluster automatisé.	Aucun
sauvegarde	RDS-EVENT-0169	Instantané de cluster automatisé créé.	Aucun

Évènements d'instance de base de données

Le tableau suivant recense la catégorie d'évènement et la liste des évènements lorsqu'une instance de base de données est le type source.

Catégorie	ID d'évènement RDS	Message	Remarques
disponibilité	RDS-EVENT-0004	L'instance de base de données s'est arrêtée.	Aucun
disponibilité	RDS-EVENT-0006	Instance de base de données redémarrée.	Aucun
disponibilité	RDS-EVENT-0022	Erreur lors du redémarrage de MySQL :*message*.	Une erreur s'est produite lors du redémarrage MySQL.
disponibilité	RDS-EVENT-0221	L'instance de base de données a atteint le seuil de stockage complet et la base de données a été arrêtée. Vous pouvez augmenter le stockage alloué pour résoudre ce problème.	Aucun
disponibilité	RDS-EVENT-0222	La capacité de stockage disponible pour l'instanc	S'applique uniquement à RDS pour MySQL lorsqu'un

Catégorie	ID d'évènement RDS	Message	Remarques
		e de base *percentage* de données *name* est faible par rapport au stockage alloué [Stockage alloué :*amount*, Stockage gratuit :*amount*]. La base de données sera fermée pour éviter toute corruption si le stockage disponible est inférieur à*amount*. Vous pouvez augmenter le stockage alloué pour résoudre ce problème.	e instance de base de données consomme plus de 90 % du stockage alloué. Surveillez l'espace de stockage d'une instance de base de données à l'aide de la métrique Free Storage Space. Pour de plus amples informations, veuillez consulter Stockage d'instance de base de données Amazon RDS.
disponibilité	RDS-EVENT-0330	La capacité de stockage disponible du volume de journal des transactions dédié est trop faible pour une instance de base de données*name*. Le volume de stockage gratuit du journal correspond à *percentage* l'espace de stockage alloué. [Stockage alloué :*amount*, Stockage libre :*amount*] La base de données sera fermée pour éviter toute corruption si le stockage disponible est inférieur à*amount*. Vous pouvez désactiver le volume dédié au journal des transactions pour résoudre ce problème.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).

Catégorie	ID d'évènement RDS	Message	Remarques
disponibilité	RDS-EVENT-0331	La capacité de stockage disponible du volume de journal des transactions dédié est trop faible pour une instance de base de données*name*. Le stockage gratuit du volume de log *percentage* correspond au stockage provisionné. [Stockage provisionné :*amount*, Stockage gratuit :*amount*] Vous pouvez désactiver le volume dédié au journal des transactions pour résoudre ce problème.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).
disponibilité	RDS-EVENT-0396	Amazon RDS a prévu un redémarrage de cette réplique de lecture lors de la prochaine fenêtre de maintenance de cette instance, après la rotation du mot de passe utilisateur interne.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
disponibilité	RDS-EVENT-0419	Amazon RDS n'a pas pu accéder à la clé de KMS chiffrement de l'instance *name* de base de données. Cette base de données sera placée dans un état inaccessible. Reportez-vous à la section sur la résolution des problèmes dans la documentation Amazon RDS pour plus de détails.	Aucun
sauvegarde	RDS-EVENT-0001	Sauvegarde de l'instance de base de données.	Aucun
sauvegarde	RDS-EVENT-0002	Sauvegarde de l'instanc e de base de données terminée.	Aucun
sauvegarde	RDS-EVENT-0086	Nous ne sommes pas en mesure d'associer le groupe d'options *name* à l'instance de base de données*name*. Vérifiez que le groupe d'options *name* est pris en charge dans votre classe d'instanc e de base de données et dans votre configuration. Si c'est le cas, vérifiez tous les paramètres du groupe d'options et réessayez.	Pour de plus amples informations, veuillez consulter Utilisation de groupes d'options.

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0011	Mis à jour pour utiliser DBParameter le groupe*name*.	Aucun
modification de configuration	RDS-EVENT-0012	Application de la modification à la classe d'instance de base de données.	Aucun
modification de configuration	RDS-EVENT-0014	Fin de l'application de la modification à la classe d'instance de base de données.	Aucun
modification de configuration	RDS-EVENT-0016	Réinitialisation des informations d'identification principales.	Aucun
modification de configuration	RDS-EVENT-0017	Fin de l'application de la modification au stockage alloué.	Aucun
modification de configuration	RDS-EVENT-0018	Application de la modification au stockage alloué.	Aucun
modification de configuration	RDS-EVENT-0024	Application de la modification pour effectuer une conversion vers une instance de base de données multi-AZ.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0025	Fin de l'application de la modification pour effectuer une conversion vers une instance de base de données multi-AZ.	Aucun
modification de configuration	RDS-EVENT-0028	Désactivation des sauvegardes automatiques.	Aucun
modification de configuration	RDS-EVENT-0029	Fin de l'application de la modification pour effectuer une conversion vers une instance de base de données standard (mono-AZ).	Aucun
modification de configuration	RDS-EVENT-0030	Application de la modification pour effectuer une conversion vers une instance de base de données standard (mono-AZ).	Aucun
modification de configuration	RDS-EVENT-0032	Sauvegardes automatiques activées.	Aucun
modification de configuration	RDS-EVENT-0033	Certains *`number`* utilisateurs correspondent au nom d'utilisateur principal ; seuls ceux qui ne sont pas liés à un hôte spécifique sont réinitialisés.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0067	Réinitialisation de votre mot de passe impossible. Informations sur les erreurs :*message*.	Aucun
modification de configuration	RDS-EVENT-0078	L'intervalle de surveillance a été modifié en*number*.	La configuration Supervision améliorée a été modifiée.
modification de configuration	RDS-EVENT-0092	Fin de la mise à jour du groupe de paramètres de la base de données.	Aucun
modification de configuration	RDS-EVENT-0217	Application de la modification initiée par scalabilité automatique au stockage alloué.	Aucun
modification de configuration	RDS-EVENT-0218	Vous avez fini d'appliquer la modification initiée par scalabilité automatique au stockage alloué.	Aucun
modification de configuration	RDS-EVENT-0295	La mise à niveau de la configuration du stockage a commencé.	Aucun
modification de configuration	RDS-EVENT-0296	La mise à niveau de la configuration du stockage est terminée.	Aucun
modification de configuration	RDS-EVENT-0332	Le volume de journal dédié est désactivé.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0333	La désactivation du volume de journal dédié a commencé.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).
modification de configuration	RDS-EVENT-0334	L'activation du volume de journal dédié a commencé.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).
modification de configuration	RDS-EVENT-0335	Le volume de journal dédié est activé.	Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).
modification de configuration	RDS-EVENT-0383	*engine version*ne supporte pas le plugin memcached. RDS continuera à mettre à niveau votre instance de base de données et supprimera ce plugin.	À partir de MySQL 8.3.0, le plugin memcached n'est plus pris en charge. Pour plus d'informations, consultez les modifications apportées à MySQL 8.3.0 (16/01/2022, Innovation Release).
création	RDS-EVENT-0005	Instance de base de données créée.	Aucun
suppression	RDS-EVENT-0003	Instance de base de données supprimée.	Aucun
basculement	RDS-EVENT-0013	Basculement d'instance multi-AZ démarré.	Un basculement multi-AZ ayant entraîné la promotion d'une instance de base de données de secours a démarré.

Catégorie	ID d'évènement RDS	Message	Remarques
basculement	RDS-EVENT-0015	Basculement multi-AZ vers le mode veille terminé : la propagation DNS peut prendre quelques minutes.	Un basculement multi-AZ ayant entraîné la promotion d'une instance de base de données de secours est terminé. Le transfert du DNS vers la nouvelle instance de base de données principale peut prendre plusieurs minutes.
basculement	RDS-EVENT-0034	Abandon du basculement demandé par l'utilisateur, car un basculement s'est récemment produit sur l'instance de base de données.	Amazon RDS ne tente pas d'effectuer le basculement demandé, car un basculement s'est récemment produit sur l'instance de base de données.
basculement	RDS-EVENT-0049	Basculement de l'instance multi-AZ terminé.	Aucun
basculement	RDS-EVENT-0050	Démarrage de l'activation de l'instance multi-AZ.	Une activation Multi-AZ a démarré après une restauration réussie de l'instance de base de données. Cet événement se produit si Amazon RDS promeut l'instance de base de données principale vers le même AZ que l'instance de base de données principale précédente.

Catégorie	ID d'évènement RDS	Message	Remarques
basculement	RDS-EVENT-0051	Activation de l'instance multi-AZ terminée.	Une activation Multi-AZ est terminée. Votre base de données doit être accessibl e maintenant.
basculement	RDS-EVENT-0065	Récupéré après un basculement partiel.	Aucun
échec	RDS-EVENT-0031	Instance de base de données mise en *name* état. RDS vous recommand e de lancer un point-in-time-restore.	L'instance de base de données a échoué en raison d'une configura tion incompatible ou d'un problème de stockage sous-jacent. Commencez un point-in-time-restore pour l'instance de base de données.
échec	RDS-EVENT-0035	Instance de base de données insérée dans*state*. *message*.	L'instance de base de données a des paramètres non valides. Par exemple, si l'instance de base de données n'a pas pu démarrer, un paramètre lié à la mémoire étant défini avec une valeur trop élevée pour cette classe d'instanc e, votre action consiste à modifier le paramètre et à redémarrer l'instance de base de données.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0036	Instance de base de données dans*state*. *message*.	L'instance de base de données se trouve sur un réseau non compatible. Certains sous-réseaux spécifiés ne IDs sont pas valides ou n'existent pas.
échec	RDS-EVENT-0058	L'installation de Statspack a échoué. *message*.	Erreur lors de la création du compte d'utilisateur Oracle Statspack `PERFSTAT`. Supprimez le compte avant d'ajouter l'option `STATSPACK` .
échec	RDS-EVENT-0079	Amazon RDS n'a pas été en mesure de créer des informations d'identification pour une surveillance améliorée. Cette fonction a été désactivée. Cela est probablement dû au fait que votre compte rds-monitoring-role n'est pas présent et qu'il n'est pas configuré correctement. Reportez-vous à la section sur la résolution des problèmes dans la documentation Amazon RDS pour plus de détails.	La supervision améliorée ne peut pas être activée sans le rôle IAM de surveillance améliorée. Pour obtenir des informations sur la création du rôle IAM, consultez Pour créer un rôle IAM pour la surveillance améliorée Amazon RDS.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0080	Amazon RDS n'a pas pu configurer la surveillance améliorée sur votre instance : *name* cette fonctionnalité a été désactivée. Cela est probablement dû au fait que votre compte rds-monitoring-role n'est pas présent et qu'il n'est pas configuré correctement. Reportez-vous à la section sur la résolution des problèmes dans la documentation Amazon RDS pour plus de détails.	La surveillance améliorée a été désactivée en raison d'une erreur lors de la modification de la configuration. Il est probable que le rôle IAM de surveillance améliorée ne soit pas configuré correctement. Pour obtenir des informations sur la création du rôle IAM de surveillance améliorée, consultez Pour créer un rôle IAM pour la surveillance améliorée Amazon RDS.
échec	RDS-EVENT-0081	Amazon RDS n'a pas pu créer d'informations d'identification pour l'*name*option. Cela est dû au fait que le rôle *name* IAM n'est pas correctement configuré dans votre compte. Reportez-vous à la section sur la résolution des problèmes dans la documentation Amazon RDS pour plus de détails.	Le rôle IAM que vous utilisez pour accéder à votre compartiment Amazon S3 pour la sauvegarde et la restauration natives SQL Server est mal configuré. Pour plus d'informations, consultez Configuration pour les sauvegarde et restauration natives.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0165	L'instance de base de données RDS Custom se trouve en dehors du périmètre de prise en charge.	Il est de votre responsabilité de corriger les problèmes de configuration qui font passer votre instance de base de données RDS Custom à l'état `unsupported-configuration` . Si le problème concerne l' AWS infrastructure, vous pouvez utiliser la console ou le AWS CLI pour le résoudre. Si le problème concerne le système d'exploitation ou la configuration de la base de données, vous pouvez vous connecter à l'hôte pour le résoudre. Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0188	L'état de l'instance de base de données ne peut pas être mis à niveau. *message*	Amazon RDS n'a pas pu mettre à niveau une instance de base de données MySQL en raison d'incompatibilités liées au dictionnaire de données. L'instance de base de données a été rétablie vers la version 5.7 en raison de l'échec d'une tentative de mise à niveau vers la version 8.0, ou rétablie vers la version 8.0 de MySQL en raison de l'échec d'une tentative de mise à niveau vers la version 8.4. Pour de plus amples informations, veuillez consulter Annulation après échec de la mise à niveau.
échec	RDS-EVENT-0219	L'instance de base de données est dans un état non valide. Aucune action n'est nécessaire. La scalabilité automatique retentera plus tard.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0220	L'instance de base de données est dans la période de refroidissement après une opération précédente de mise à l'échelle du stockage. Nous optimisons votre instance de base de données. Cette opération prend au moins six heures. Aucune action n'est nécessaire. La scalabilité automatique retentera après la période de refroidissement.	Aucun
échec	RDS-EVENT-0223	Le dimensionnement automatique du stockage n'est pas en mesure de redimensionner le stockage pour la raison :*reason*.	Aucun
échec	RDS-EVENT-0224	La mise à l'échelle automatique du stockage a déclenché une tâche de mise à l'échelle du stockage en attente qui atteindrait ou dépasserait le seuil de stockage maximal. Augmentez le seuil de stockage maximal.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0237	L'instance de base de données comporte un type de stockage qui est actuellement indisponible dans la zone de disponibilité. La scalabilité automatique retentera plus tard.	Aucun
échec	RDS-EVENT-0254	Le quota de stockage sous-jacent pour ce compte client a dépassé la limite. Augmentez le quota de stockage autorisé pour permettre la mise à l'échelle sur l'instance.	Aucun
échec	RDS-EVENT-0278	La création de l'instance de base de données a échoué. *message*	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0279	La promotion de la réplique de lecture personnalisée RDS a échoué. *message*	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0280	RDS Custom n'a pas pu mettre à niveau l'instance de base de données car la pré-vérification a échoué. *message*	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0281	RDS Custom n'a pas pu modifier l'instance de base de données car la pré-vérification a échoué. *message*	*message*Cela inclut des détails sur la panne.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0282	RDS Custom n'a pas pu modifier l'instance de base de données, car les autorisations d'adresses IP Elastic ne sont pas correctes. Veuillez confirmer que l'adresse IP Elastic est marquée avec `AWSRDSCustom` .	Aucun
échec	RDS-EVENT-0283	RDS Custom n'a pas pu modifier l'instance de base de données, car la limite d'adresses IP Elastic a été atteinte dans votre compte. Libérez Elastic non utilisé IPs ou demandez une augmentation du quota de votre limite d'adresses IP Elastic.	Aucun
échec	RDS-EVENT-0284	RDS Custom n'a pas pu convertir l'instance en haute disponibilité car le pré-contrôle a échoué. *`message`*	*`message`*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0285	RDS Custom n'a pas pu créer un instantané final pour l'instance de base de données car*`message`*.	*`message`*Cela inclut des détails sur la panne.

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0421	RDS Custom n'a pas pu convertir l'instance de base de données en un déploiement multi-AZ :. *message* L'instance restera un déploiement mono-AZ. Consultez le guide de l'utilisateur RDS pour plus d'informations sur les déploiements multi-AZ pour RDS Custom for Oracle.	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0306	La mise à niveau de la configuration du stockage a échoué. Veuillez réessayer.	Aucun
échec	RDS-EVENT-0315	Impossible de déplacer la base de données réseau incompatible vers *name* l'état disponible : *message*	La configuration réseau de la base de données n'est pas valide. La base de données n'a pas pu être déplacée du réseau incompatible vers un réseau disponible.
échec	RDS-EVENT-0328	Impossible de joindre un hôte à un domaine. Le statut d'adhésion au domaine, par exemple, *instancename* a été défini sur Échec.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0329	Impossible de joindre un hôte à votre domaine. Au cours du processus de jonction de domaine, Microsoft Windows a renvoyé le code d'erreur*message*. Vérifiez les configurations de votre réseau et de vos autorisations et émettez une `modify-db-instance` demande pour tenter à nouveau de rejoindre le domaine.	Lorsque vous utilisez un Active Directory autogéré, consultezRésolution des problèmes liés à Active Directory autogéré.
échec	RDS-EVENT-0353	L'instance de base de données ne peut pas être créée en raison de limites de ressources insuffisantes. *message*.	*message*Cela inclut des détails sur la panne.
échec	RDS-EVENT-0356	RDS n'a pas pu configurer le point de terminaison Kerberos dans votre domaine. Cela peut empêcher l'authentification Kerberos pour votre instance de base de données. Vérifiez la configuration réseau entre votre instance de base de données et les contrôleurs de domaine.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
échec	RDS-EVENT-0418	Amazon RDS ne parvient pas à accéder à la clé de KMS chiffrement de l'instance *name* de base de données. Cela est probablement dû à la désactivation de la clé ou à l'impossibilité pour Amazon RDS d'y accéder. Si cela continue, la base de données sera placée dans un état inaccessible. Reportez-vous à la section sur la résolution des problèmes dans la documentation Amazon RDS pour plus de détails.	Aucun
échec	RDS-EVENT-0420	Amazon RDS peut désormais accéder avec succès à la clé de KMS chiffrement de l'instance *name* de base de données.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
stockage faible	RDS-EVENT-0007	L'espace de stockage alloué est épuisé. Allouez du stockage supplémen taire pour résoudre le problème.	L'espace de stockage alloué pour l'instance de base de données a été consommé. Pour résoudre ce problème, allouez de l'espace de stockage supplémen taire à l'instance de base de données. Pour plus d'informations, consultez la FAQ RDS. Vous pouvez surveiller l'espace de stockage pour une instance de base de données à l'aide de la métrique Free Storage Space (Espace de stockage libre).
stockage faible	RDS-EVENT-0089	La capacité de stockage gratuite pour l'instance de base *percentage* de données : *name* est faible par rapport au stockage provisionné [Stockage provisionné :*size*, Stockage gratuit :*size*]. Vous pouvez augmenter le stockage provisionné pour résoudre ce problème.	L'instance de base de données a consommé plus de 90 % de son stockage alloué. Vous pouvez surveiller l'espace de stockage pour une instance de base de données à l'aide de la métrique Free Storage Space (Espace de stockage libre).

Catégorie	ID d'évènement RDS	Message	Remarques
stockage faible	RDS-EVENT-0227	Le stockage de votre cluster Aurora est dangereusement bas, il ne reste que des *amount* téraoctets. Veuillez prendre des mesures pour réduire la charge de stockage de votre cluster.	Le sous-système de stockage Aurora manque d'espace.
maintenance	RDS-EVENT-0026	Application de correctifs hors ligne à l'instance de base de données.	La maintenance hors connexion de l'instance de base de données est en cours. L'instance de base de données n'est pas disponible actuellement.
maintenance	RDS-EVENT-0027	Application de correctifs hors ligne à l'instance de base de données terminée.	La maintenance hors connexion de l'instanc e de base de données est terminée. L'instance de base de données est désormais disponible.
maintenance	RDS-EVENT-0047	Instance de base de données corrigée.	Aucun
maintenance	RDS-EVENT-0155	Une mise à niveau de version mineure du moteur de base de données est disponible pour l'instance de base de données.	Aucun
maintenance	RDS-EVENT-0178	La mise à niveau de l'instance de base de données est en cours.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance	RDS-EVENT-0264	La vérification préalable a commencé pour la mise à niveau de la version du moteur de base de données.	Aucun
maintenance	RDS-EVENT-0265	La vérification préalable est terminée pour la mise à niveau de la version du moteur de base de données.	Aucun
maintenance	RDS-EVENT-0266	Le temps d'arrêt a commencé pour l'instance de base de données.	Aucun
maintenance	RDS-EVENT-0267	La mise à niveau de la version du moteur a commencé.	Aucun
maintenance	RDS-EVENT-0268	La mise à niveau de la version du moteur est terminée.	Aucun
maintenance	RDS-EVENT-0269	Les tâches postérieures à la mise à niveau sont en cours.	Aucun
maintenance	RDS-EVENT-0270	La mise à niveau de version du moteur de base de données a échoué. La restauration de la mise à niveau de la version du moteur a réussi.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance	RDS-EVENT-0398	En attente de la fin de la mise à niveau de la version du moteur de base de données sur l'instance de base de données principal e.	Émis sur une réplique en lecture lors d'une mise à niveau majeure d'une version du moteur.
maintenance	RDS-EVENT-0399	En attente de la fin de la mise à niveau de la version du moteur de base de données sur les répliques lues.	Émis sur le moteur de base de données source lors d'une mise à niveau majeure d'une version du moteur.
maintenance	RDS-EVENT-0422	RDS remplacera l'hôte de l'instance de base de données *name* en raison d'une action de maintenan ce en attente.	Aucun
maintenance, échec	RDS-EVENT-0195	*message*	La mise à jour du fichier sur le fuseau horaire Oracle a échoué. Pour de plus amples informations, veuillez consulter Mise à niveau automatique du fichier sur le fuseau horaire Oracle.

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance, notification	RDS-EVENT-0191	La mise à jour d'une nouvelle version du fichier sur le fuseau horaire est disponible.	Si vous mettez à jour votre moteur de base de données RDS for Oracle, Amazon RDS génère cet événement si vous n'avez pas choisi de mise à niveau du fichier sur le fuseau horaire et que la base de données n'utilise pas le dernier fichier sur le fuseau horaire de l'heure d'été disponible sur l'instanc e. Pour de plus amples informations, veuillez consulter Mise à niveau automatique du fichier sur le fuseau horaire Oracle.
maintenance, notification	RDS-EVENT-0192	La mise à jour de votre fichier sur le fuseau horaire a démarré.	La mise à niveau de votre fichier sur le fuseau horaire Oracle a commencé. Pour plus d'informations, consultez Mise à niveau automatique du fichier sur le fuseau horaire Oracle.

Catégorie	ID d'évènement RDS	Message	Remarques
maintenance, notification	RDS-EVENT-0193	Aucune mise à jour n'est disponible pour la version actuelle du fichier sur le fuseau horaire.	Votre instance de base de données Oracle utilise la dernière version du fichier sur le fuseau horaire, et l'une des conditions suivantes est vraie : • Vous avez récemment ajouté l'option `TIMEZONE_FILE_AUTOUPGRADE` . • Votre moteur de base de données Oracle est en cours de mise à niveau. Pour plus d'informations, consultez Mise à niveau automatique du fichier sur le fuseau horaire Oracle.
maintenance, notification	RDS-EVENT-0194	La mise à jour de votre fichier sur le fuseau horaire est terminée.	La mise à jour de votre fichier sur le fuseau horaire Oracle est terminée. Pour de plus amples informations, veuillez consulter Mise à niveau automatique du fichier sur le fuseau horaire Oracle.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0044	*message*	Il s'agit d'une notification émise par l'opérateur. Pour plus d'informations, consultez le message de l'évènement.
notification	RDS-EVENT-0048	La mise à niveau du moteur de base de données est retardée, car cette instance comporte des réplicas en lecture qui doivent d'abord être mis à niveau.	L'application des correctif s de l'instance de base de données a été retardée.
notification	RDS-EVENT-0054	*message*	Le moteur de stockage MySQL que vous utilisez n'est pas InnoDB, lequel est le moteur de stockage MySQL recommandé pour Amazon RDS. Pour obtenir des informations sur les moteurs de stockage MySQL, consultez Moteurs de stockage pris en charge pour RDS for MySQL.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0055	*message*	Le nombre de tables de votre instance de base de données dépasse les bonnes pratiques recommandées pour Amazon RDS. Réduisez le nombre de tables de votre instance de base de données. Pour plus d'informations sur les bonnes pratiques recommandées, consultez Directives opérationnelles de base Amazon RDS.
notification	RDS-EVENT-0056	*message*	Le nombre de bases de données de votre instance de base de données dépasse les bonnes pratiques recommand ées pour Amazon RDS. Réduisez le nombre de bases de données de votre instance de base de données. Pour plus d'informations sur les bonnes pratiques recommandées, consultez Directives opérationnelles de base Amazon RDS.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0064	La rotation de la clé de chiffrement TDE a réussi.	Pour plus d'informations sur les bonnes pratiques recommandées, consultez Directives opérationnelles de base Amazon RDS.
notification	RDS-EVENT-0084	Impossible de convertir l'instance de base de données en instance multi-AZ : *message*.	Vous avez essayé de convertir une instance de base de données en environnement Multi-AZ, mais elle contient des groupes de fichiers en mémoire qui ne sont pas pris en charge pour plusieurs environnements Multi-AZ. Pour plus d'informations, consultez Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.
notification	RDS-EVENT-0087	Instance de base de données arrêtée.	Aucun
notification	RDS-EVENT-0088	Instance de base de données démarrée.	Aucun
notification	RDS-EVENT-0154	L'instance de base de données est en cours de démarrage dans la mesure où elle a dépassé le temps maximum autorisé pour son arrêt.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0157	Impossible de modifier la classe d'instance de base de données. *message*.	RDS ne peut pas modifier la classe d'instance de base de données car la classe d'instance cible ne peut pas prendre en charge le nombre de bases de données figurant dans l'instance de base de données source. Le message d'erreur suivant apparaît : "The instance has N databases, but after conversion it would only support N" (L'instance comporte N bases de données, mais après la conversion, elle n'en prendrait en charge que N). Pour plus d'informations, consultez Limites propres aux instances de bases de données Microsoft SQL Server.
notification	RDS-EVENT-0158	L'instance de base de données est dans un état qui ne peut pas être mise à niveau :*message*.	Aucun
notification	RDS-EVENT-0167	*message*	La configuration du périmètre de prise en charge de RDS Custom a été modifiée.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0189	Les crédits de solde de la rafale gp2 pour l'instance de base de données RDS sont faibles. Pour résoudre ce problème, réduisez l'utilisation IOPS ou modifiez vos paramètres de stockage pour augmenter les performances.	Les crédits de solde de la rafale gp2 pour l'instance de base de données RDS sont faibles. Pour résoudre ce problème, réduisez l'utilisation IOPS ou modifiez vos paramètres de stockage pour augmenter les performances. Pour de plus amples informations, veuillez consulter Crédits I/O et performances en rafale dans le Guide de l'utilisateur Amazon Elastic Compute Cloud.
notification	RDS-EVENT-0225	La taille de stockage allouée en *amount* Go approche le seuil de stockage maximal en *amount* Go. Augmentez le seuil de stockage maximal.	Cet événement est invoqué lorsque le stockage alloué atteint 80 % du seuil de stockage maximal. Pour éviter cet événement, augmentez le seuil de stockage maximum.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0231	La modification du stockage de votre instance de base de données a rencontré une erreur interne. La requête de modification est en attente et sera réitérée ultérieur ement.	Une erreur s'est produite lors du processus de réplication en lecture. Pour plus d'informations, consultez le message de l'évènement. En outre, consultez la section de dépannage pour les réplicas en lecture de votre moteur de base de données. • Résolution d'un problème de réplica en lecture MariaDB • Résolution d'un problème de réplication de lecture sur un SQL serveur • Résolution d'un problème de réplication de lecture MySQL • Dépannage des réplicas RDS for Oracle

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0253	La base de données utilise le tampon doublewrite. *message*. Pour plus d'informations, consultez la *name* documentation relative aux écritures optimisées RDS.	La fonctionnalité RDS Optimized Writes n'est pas compatible avec la configuration de stockage de l'instance. Pour plus d'informations, consultez Améliorer les performances d'écriture grâce à RDS Optimized Writes for MySQL et Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB. Vous pouvez effectuer une mise à niveau de la configuration du stockage pour activer les écritures optimisées en créant un blue/green déploiement.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0297	La configuration de stockage pour l'instance de base de données *name* prend en charge une taille maximale de 16384 GiB. Effectuez une mise à niveau de la configuration du stockage pour prendre en charge des volumes de stockage supérieurs à 16 384 Gio.	Vous ne pouvez pas augmenter le volume de stockage alloué à l'instance de base de données au-delà de 16 384 Gio. Pour remédier à cette limitation, mettez à niveau la configuration du stockage. Pour plus d'informations, consultez la section Mise à niveau du système de fichiers de stockage pour une instance de base de données.
notification	RDS-EVENT-0298	La configuration de stockage pour l'instance de base de données *name* prend en charge une taille de table maximale de 2 048 GiB. Mettez à niveau la configuration de stockage pour que les tables supérieures à 2 048 Gio soient prises en charge.	Les instances RDS MySQL et MariaDB soumises à cette limitation ne peuvent pas avoir une taille de table supérieure à 2 048 Gio. Pour remédier à cette limitation, mettez à niveau la configuration du stockage. Pour plus d'informations, consultez la section Mise à niveau du système de fichiers de stockage pour une instance de base de données.
notification	RDS-EVENT-0327	Amazon RDS n'a pas pu trouver le secret*SECRET ARN*. *message.*	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0365	Les fichiers de fuseau horaire ont été mis à jour. Redémarrez votre instance RDS pour que les modifications prennent effet.	Aucun
notification	RDS-EVENT-0385	La topologie du cluster est mise à jour.	Des modifications DNS ont été apportées au cluster de base de données pour l'instance de base de données. Cela inclut lorsque de nouvelles instances de base de données sont ajoutées ou supprimées, ou en cas de basculement.
notification	RDS-EVENT-0403	Une charge de travail de base de données fait que le système manque cruellement de mémoire. Pour atténuer le problème, RDS a automatiquement défini la valeur de innodb_buffer_pool_size sur. *amount*	S'applique uniquement aux instances de base de données RDS pour MySQL et RDS pour MariaDB.
notification	RDS-EVENT-0404	Une charge de travail de base de données fait que le système manque cruellement de mémoire. Pour atténuer le problème, RDS a automatiquement défini la valeur de shared_buffers sur. *amount*	S'applique uniquement aux instances de base de données RDS pour PostgreSQL.

Catégorie	ID d'évènement RDS	Message	Remarques
réplica en lecture	RDS-EVENT-0045	La réplication s'est arrêtée.	Ce message apparaît en cas d'erreur lors de la réplication. Pour déterminer le type d'erreur, consultez Résolution d'un problème de réplication de lecture MySQL.
réplica en lecture	RDS-EVENT-0046	Reprise de la réplication pour le réplica en lecture.	Ce message s'affiche lorsque vous créez un réplica en lecture, ou comme message de surveillance lorsque vous confirmez que la réplication fonctionne correctement. Si ce message fait suite à une notification `RDS-EVENT-0045` , la réplication a repris suite à une erreur ou à un arrêt de la réplication.
réplica en lecture	RDS-EVENT-0057	Le streaming de réplication a été suspendu.	Aucun
réplica en lecture	RDS-EVENT-0062	La réplication pour le réplica en lecture a été arrêtée manuellement.	Aucun
réplica en lecture	RDS-EVENT-0063	La réplication depuis une instance autre que RDS a été réinitialisée.	Aucun
réplica en lecture	RDS-EVENT-0202	La création d'un réplica en lecture a échoué.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
réplica en lecture	RDS-EVENT-0233	Le passage à la réplique en lecture a commencé.	Aucun
réplica en lecture	RDS-EVENT-0357	Le canal de réplication *name* a démarré.	Pour plus d'informations sur les canaux de réplication, consultezthe section called “Configuration de la réplication multi-sources”.
réplica en lecture	RDS-EVENT-0358	Le canal de réplication *name* s'est arrêté.	Pour plus d'informations sur les canaux de réplication, consultezthe section called “Configuration de la réplication multi-sources”.
réplica en lecture	RDS-EVENT-0359	Le canal de réplication *name* a été arrêté manuellement.	Pour plus d'informations sur les canaux de réplication, consultezthe section called “Configuration de la réplication multi-sources”.
réplica en lecture	RDS-EVENT-0360	Le canal de réplication *name* a été réinitialisé.	Pour plus d'informations sur les canaux de réplication, consultezthe section called “Configuration de la réplication multi-sources”.
réplica en lecture	RDS-EVENT-0415	Le processus de mise à niveau a repris la réplication sur la réplique lue.	Aucun
réplica en lecture	RDS-EVENT-0416	Le processus de mise à niveau a arrêté la réplication sur la réplique lue.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
récupération	RDS-EVENT-0020	La récupération de l'instance de base de données a démarré. Le temps de récupération varie selon la quantité de données à restaurer.	Aucun
récupération	RDS-EVENT-0021	La récupération de l'instance de base de données est terminée.	Aucun
récupération	RDS-EVENT-0023	Demande de capture instantanée urgente :*message*.	Une sauvegarde manuelle a été demandée, mais Amazon RDS est actuellement en cours de création d'un instantané de base de données. Soumettez à nouveau la demande après qu'Amazon RDS a terminé l'instantané de base de données.
récupération	RDS-EVENT-0052	La récupération de l'instance multi-AZ a démarré.	Le temps de récupération varie selon la quantité de données à restaurer.
récupération	RDS-EVENT-0053	La récupération de l'instance multi-AZ est terminée. En attente de basculement ou d'activation.	Ce message indique qu'Amazon RDS a préparé votre instance de base de données pour lancer un basculement vers l'instance secondaire si nécessaire.

Catégorie	ID d'évènement RDS	Message	Remarques
récupération	RDS-EVENT-0066	L'instance sera dégradée lors du rétablissement de la mise en miroir :. *message*	L'instance de base de données SQL Server est en train de rétablir son miroir. Les performances seront dégradées tant que le miroir n'est pas restauré. Il a été trouvé une base de données avec un modèle de récupérat ion autre que FULL. Le modèle de récupération a été remodifié en FULL et la récupération de la mise en miroir a démarré. (<dbname>: <recovery model found>[,...])”
récupération	RDS-EVENT-0166	*message*	L'instance de base de données RDS Custom se trouve dans le périmètre de prise en charge.
récupération	RDS-EVENT-0361	La restauration de l'instanc e de base de données de secours a commencé.	L'instance de base de données de secours est reconstruite pendant le processus de restaurat ion. Les performances de la base de données sont affectées pendant le processus de restauration.

Catégorie	ID d'évènement RDS	Message	Remarques
récupération	RDS-EVENT-0362	La restauration de l'instance de base de données de secours est terminée.	L'instance de base de données de secours est reconstruite pendant le processus de restauration. Les performances de la base de données sont affectées pendant le processus de restauration.
restauration	RDS-EVENT-0019	Restauré depuis une instance de base de données *name* vers*name*.	L'instance de base de données a été restaurée à partir d'une point-in-time sauvegarde.
sécurité	RDS-EVENT-0068	Déchiffrement du mot de passe de la partition hsm pour mettre à jour l'instance.	RDS déchiffre le mot de passe de AWS CloudHSM partition pour mettre à jour l'instance de base de données. Pour plus d'informations, consultez Chiffrement transparent des données (TDE) des bases de données Oracle avec AWS CloudHSM dans le Guide de l'utilisateur AWS CloudHSM .

Catégorie	ID d'évènement RDS	Message	Remarques
application de correctifs de sécurité	RDS-EVENT-0230	La mise à jour du système est disponible pour votre instance de base de données. Pour obtenir des informations sur l'application des mises à niveau, consultez « Entretien d'une instance de base de données » dans le Guide de l'utilisateur RDS.	Une nouvelle mise à jour du système d'exploitation est disponible. Une nouvelle mise à jour mineure du système d'exploitation est disponible pour votre instance de base de données. Pour obtenir des informations sur l'application de mises à jour, consultez Mises à jour du système d'exploitation pour les instances de base de données RDS.
maintenance	RDS-EVENT-0425	Amazon RDS ne peut pas effectuer la mise à niveau du système d'exploitation car aucune adresse IP n'est disponible dans les sous-réseaux spécifiés. Choisissez des sous-réseaux avec des adresses IP disponibles et réessayez.	Aucun
maintenance	RDS-EVENT-0429	Amazon RDS ne peut pas effectuer la mise à niveau du système d'exploitation en raison d'une capacité insuffisante disponible pour le type d'*type*instance dans la zone de *zone* disponibilité	Aucun

Évènements de groupe de paramètres de base de données

Le tableau suivant affiche la catégorie d'évènement et la liste des évènements lorsqu'un groupe de paramètres de base de données est le type source.

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0037	Paramètre mis *name* à jour *value* avec méthode d'application*method*.	Aucun

Évènements de groupe de sécurité de base de données

Le tableau suivant affiche la catégorie d'évènement et la liste des évènements lorsqu'un groupe de sécurité de base de données est le type source.

> **Note**
>
> Les groupes de sécurité de base de données sont des ressources pour EC2 -Classic. EC2-Classic a été retiré le 15 août 2022. Si vous n'avez pas migré de EC2 -Classic vers un VPC, nous vous recommandons de migrer dès que possible. Pour plus d'informations, consultez Migrer de EC2 -Classic vers un VPC dans le guide de l'utilisateur EC2 Amazon et sur le EC2blog -Classic Networking is Retiring — Here's How to Prepare.

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0038	Modification appliquée au groupe de sécurité.	Aucun
échec	RDS-EVENT-0039	Révocation de l'autorisation en tant que*user*.	Le groupe de sécurité détenu par *user* n'existe pas. L'autorisation pour le

Catégorie	ID d'évènement RDS	Message	Remarques
			groupe de sécurité a été révoquée, car elle n'est pas valide.

Évènements d'instantané de bases de données

Le tableau suivant affiche la catégorie d'évènement et la liste des évènements lorsqu'un instantané de base de données est le type source.

Catégorie	ID d'évènement RDS	Message	Remarques
création	RDS-EVENT-0040	Création d'un instantané manuel.	Aucun
création	RDS-EVENT-0042	Instantané manuel créé.	Aucun
création	RDS-EVENT-0090	Création d'un instantané automatisé.	Aucun
création	RDS-EVENT-0091	Instantané automatisé créé.	Aucun
suppression	RDS-EVENT-0041	Instantané utilisateur supprimé.	Aucun
notification	RDS-EVENT-0059	Copie démarrée de l'instantané *name* depuis la région*name*.	Ceci est une copie d'instantané entre régions.
notification	RDS-EVENT-0060	Copie terminée de l'instantané *name* de la région *name* en *number* quelques minutes.	Ceci est une copie d'instantané entre régions.

Catégorie	ID d'évènement RDS	Message	Remarques
notification	RDS-EVENT-0061	Annulation de la demande de copie instantanée *name* de la région*name*.	Ceci est une copie d'instantané entre régions.
notification	RDS-EVENT-0159	La tâche d'exportation de l'instantané a échoué.	Aucun
notification	RDS-EVENT-0160	La tâche d'exportation de l'instantané a été annulée.	Aucun
notification	RDS-EVENT-0161	La tâche d'exportation de l'instantané est terminée.	Aucun
notification	RDS-EVENT-0196	Copie démarrée du cliché *name* dans la région*name*.	Ceci est une copie d'instantané locale.
notification	RDS-EVENT-0197	Copie terminée de la capture d'écran *name* dans la région*name*.	Ceci est une copie d'instantané locale.
notification	RDS-EVENT-0190	Demande de copie instantanée annulée de *name* dans la région*name*.	Ceci est une copie d'instantané locale.
restauration	RDS-EVENT-0043	Restauré à partir d'un instantané*name*.	Une instance de base de données est en cours de restauration à partir d'un instantané de base de données.

Événements RDS Proxy

Le tableau suivant recense la catégorie d'événement et la liste des événements lorsqu'un proxy RDS Proxy est le type source.

Catégorie	ID d'évènement RDS	Message	Remarques
modification de configuration	RDS-EVENT-0204	Proxy *name* de base de données modifié par RDS.	Aucun
modification de configuration	RDS-EVENT-0207	RDS a modifié le point final du proxy *name* de base de données.	Aucun
modification de configuration	RDS-EVENT-0213	RDS a détecté l'ajout de l'instance de base de données et l'a automatiquement ajoutée au groupe cible du proxy *name* de base de données.	Aucun
modification de configuration	RDS-EVENT-0214	RDS a détecté la suppression de l'instance de base de données *name* et l'a automatiquement supprimée du groupe cible du proxy *name name* de base de données.	Aucun
modification de configuration	RDS-EVENT-0215	RDS a détecté la suppression du cluster de base de données *name* et l'a automatiquement supprimé du groupe cible du proxy *name name* de base de données.	Aucun
création	RDS-EVENT-0203	RDS a créé un proxy *name* de base de données.	Aucun

Catégorie	ID d'évènement RDS	Message	Remarques
création	RDS-EVENT-0206	RDS a créé un point de terminaison *name* pour le proxy *name* de base de données.	Aucun
suppression	RDS-EVENT-0205	RDS a supprimé le proxy *name* de base de données.	Aucun
suppression	RDS-EVENT-0208	RDS a supprimé le point de terminaison *name* pour le proxy *name* de base de données.	Aucun
échec	RDS-EVENT-0243	RDS n'a pas pu fournir de capacité pour le proxy *name* car il n'y a pas assez d'adresses IP disponibl es dans vos sous-rése aux :. *name* Pour résoudre ce problème, veillez à ce que vos sous-réseaux aient le nombre minimum d'adresses IP inutilisées, comme recommandé dans la documentation Proxy RDS.	Pour déterminer le nombre recommandé pour votre classe d'instances, consultez Planification de la capacité des adresses IP.
échec	RDS-EVENT-0275	RDS a limité certaines connexions au proxy de base de données. *name* Le nombre de demandes de connexion simultanées du client au proxy a dépassé la limite.	Aucun

Événements de déploiement bleu/vert

Le tableau suivant indique la catégorie d'événements et la liste des événements dont le type source est un blue/green déploiement.

Pour plus d'informations sur les blue/green déploiements, consultezUtilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Catégorie	ID d'évènement Amazon RDS	Message	Remarques
création	RDS-EVENT-0244	Les tâches de déploiement bleu/vert sont terminées. Vous pouvez apporter plus de modifications aux bases de données de l'environnement vert ou effectuer un basculement du déploiement.	Aucun
échec	RDS-EVENT-0245	La création du blue/green déploiement a échoué car*reason*.	Aucun
suppression	RDS-EVENT-0246	Le déploiement bleu/vert a été supprimé.	Aucun
notification	RDS-EVENT-0247	Basculer entre le mode de démarrage et le mode *blue* de démarrage. *green*	Aucun
notification	RDS-EVENT-0248	Le passage au numérique a été effectué lors blue/green du déploiement.	Aucun
échec	RDS-EVENT-0249	Le passage au numérique a été annulé lors blue/green du déploiement.	Aucun

Catégorie	ID d'évènement Amazon RDS	Message	Remarques
notification	RDS-EVENT-0250	Basculement du cluster de de données de réplicati on principal/en lecture au cluster démarré. *blue green*	Aucun
notification	RDS-EVENT-0251	Passage du cluster de base de principal/de réplicati on en lecture au cluster terminé. *blue green* Renommé *blue blue-old* en et *green* en*blue*.	Aucun
échec	RDS-EVENT-0252	Le passage du cluster de base de données de réplication principal/de lecture au *blue* en raison de. *reason*	Aucun
notification	RDS-EVENT-0307	La synchronisation de séquence pour le passage du *blue* à *green* a été lancée. La commutation lors de l'utilisation de séquences peut entraîner des temps d'arrêt prolongés.	Aucun
notification	RDS-EVENT-0308	La synchronisation de séquence pour le passage du *blue* à *green* est terminée.	Aucun

Catégorie	ID d'évènement Amazon RDS	Message	Remarques
échec	RDS-EVENT-0310	La synchronisation des séquences pour le passage du *blue* à *green* a été annulée car les séquences n'ont pas pu être synchronisées.	Aucun
notification	RDS-EVENT-0405	Vos volumes de stockage sont en cours d'initialisation.	Aucun
notification	RDS-EVENT-0406	Vos volumes de stockage ont été initialisés.	Aucun
notification	RDS-EVENT-0409	*message*	Aucun

Événements de version du moteur personnalisés

Le tableau suivant présente la catégorie d'événement et une liste d'événements lorsqu'une version personnalisée du moteur est le type de source.

Catégorie	ID d'évènement Amazon RDS	Message	Remarques
création	RDS-EVENT-0316	Préparation à la création d'une version personnalisée du moteur*name*. L'ensemble du processus de création peut prendre jusqu'à quatre heures.	Aucun
création	RDS-EVENT-0317	Création d'une version personnalisée du moteur*name*.	Aucun

Catégorie	ID d'évènement Amazon RDS	Message	Remarques
création	RDS-EVENT-0318	Validation de la version *name* personnalisée du moteur.	Aucun
création	RDS-EVENT-0319	La version personnalisée du moteur *name* a été créée avec succès.	Aucun
création	RDS-EVENT-0320	RDS ne peut pas créer de version de moteur personnalisée en *name* raison d'un problème interne. Nous sommes en train de résoudre le problème et nous vous contacterons si nécessaire. Pour obtenir de l'aide supplémentaire, contactez l'assistance Premium AWS.	Aucun
échec	RDS-EVENT-0198	La création de la version personnalisée du moteur a échoué*name*. *message*	*message*Cela inclut des détails sur l'échec, tels que les fichiers manquants.
échec	RDS-EVENT-0277	Échec lors de la suppression de la version personnalisée du moteur*name*. *message*	*message*Cela inclut des détails sur la panne.
restauration	RDS-EVENT-0352	Le nombre maximal de bases de données prises en charge pour point-in-time la restauration a changé.	*message*Il inclut des détails sur l'événement.

Surveillance des fichiers journaux RDSAmazon

Chaque moteur RDS de base de données génère des journaux auxquels vous pouvez accéder à des fins d'audit et de résolution des problèmes. Le type de journaux dépend de votre moteur de base de données.

Vous pouvez accéder aux journaux de base de données pour les instances de base de données à l'aide de, the AWS Command Line Interface (AWS CLI) ou Amazon RDSAPI. AWS Management Console Vous ne pouvez pas afficher, visualiser ou télécharger les journaux des transactions.

Rubriques

- Liste et affichage des fichiers journaux de base de données
- Téléchargement d'un fichier journal de base de données
- Consultation d'un fichier journal de base de données
- Publication des journaux de base de données sur Amazon CloudWatch Logs
- Lire le contenu du fichier journal à l'aide de REST
- Fichiers journaux RDS de base de données Amazon pour DB2
- Fichiers journaux de base de données MariaDB
- Fichiers journaux RDS de base de données Amazon pour Microsoft SQL Server
- Fichiers journaux de base de données MySQL
- Fichiers journaux RDS de base de données Amazon pour Oracle
- Fichiers journaux de base de données RDS for PostgreSQL

Liste et affichage des fichiers journaux de base de données

Vous pouvez consulter les fichiers journaux de base de données pour votre moteur de base de données RDSAmazon à l'aide du AWS Management Console. Vous pouvez répertorier les fichiers journaux disponibles au téléchargement ou à la surveillance en utilisant le AWS CLI ou Amazon RDSAPI.

> **Note**
>
> Si vous ne pouvez pas afficher la liste des fichiers journaux d'une instance de base de données Oracle existanteRDS, redémarrez l'instance pour afficher la liste.

Console

Pour afficher un fichier journal de base de données

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de l'instance de base de données qui contient le fichier journal que vous voulez consulter.
4. Choisissez l'onglet Logs & events (Journaux et événements).
5. Faites défiler jusqu'à la section Journaux.
6. (Facultatif) Entrez un terme de recherche pour filtrer vos résultats.
7. Sélectionnez le journal que vous souhaitez afficher, puis cliquez sur View (Afficher).

AWS CLI

Pour répertorier les fichiers journaux de base de données disponibles pour une instance de base de données, utilisez la AWS CLI `describe-db-log-files`commande.

L'exemple suivant renvoie une liste des fichiers journaux pour une instance DB nommée `my-db-instance`.

Example

```
aws rds describe-db-log-files --db-instance-identifier my-db-instance
```

RDS API

Pour répertorier les fichiers journaux de base de données disponibles pour une instance de base de données, utilisez l'RDSAPI`DescribeDBLogFiles`action Amazon.

Téléchargement d'un fichier journal de base de données

Vous pouvez utiliser le AWS Management Console AWS CLI, ou API pour télécharger un fichier journal de base de données.

Console

Pour télécharger un fichier journal de base de données

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de l'instance de base de données qui contient le fichier journal que vous voulez consulter.
4. Choisissez l'onglet Logs & events (Journaux et événements).
5. Faites défiler jusqu'à la section Journaux.
6. Dans la section Journaux, sélectionnez le bouton en regard du journal que vous voulez télécharger, puis choisissez Télécharger.
7. Ouvrez le menu contextuel (clic droit) pour le lien fourni, puis choisissez Enregistrer le lien sous. Saisissez l'emplacement souhaité pour l'enregistrement du fichier journal, puis cliquez sur Enregistrer.



AWS CLI

Pour télécharger un fichier journal de base de données, utilisez la AWS CLI commande `download-db-log-file-portion`. Par défaut, cette commande télécharge uniquement la portion la plus récente d'un fichier journal. Vous pouvez toutefois télécharger un fichier complet en spécifiant le paramètre `--starting-token 0`.

L'exemple suivant montre comment télécharger l'intégralité du contenu d'un fichier journal appelé log/ERROR .4 et le stocker dans un fichier local appelé errorlog.txt.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds download-db-log-file-portion \
    --db-instance-identifier myexampledb \
    --starting-token 0 --output text \
    --log-file-name log/ERROR.4 > errorlog.txt
```

Dans Windows :

```
aws rds download-db-log-file-portion ^
    --db-instance-identifier myexampledb ^
    --starting-token 0 --output text ^
    --log-file-name log/ERROR.4 > errorlog.txt
```

RDS API

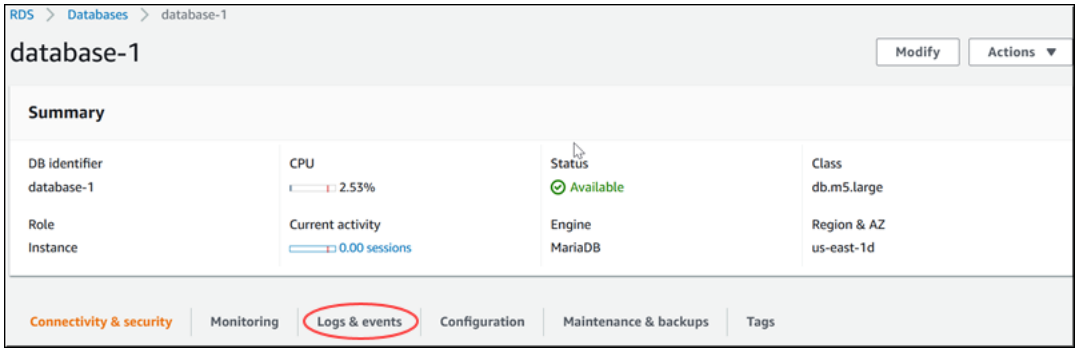
Pour télécharger un fichier journal de base de données, utilisez l'RDSAPIDownloadDBLogFilePortionaction Amazon.

Consultation d'un fichier journal de base de données

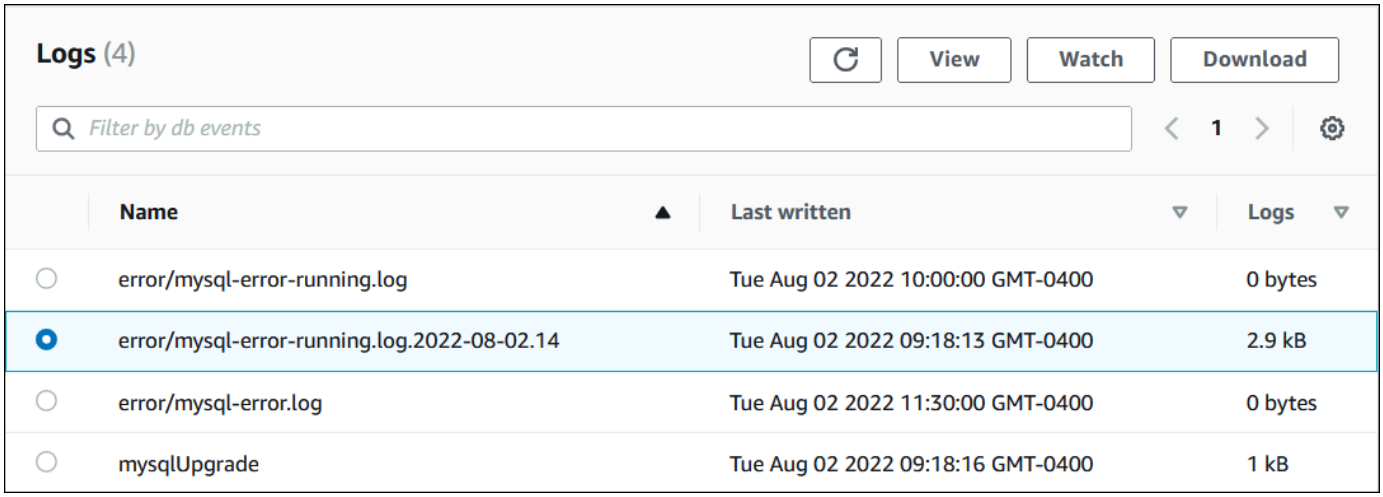
L'observation d'un fichier journal de base de données revient à suivre le fichier sur un système Linux UNIX ou sur un système Linux. Vous pouvez afficher un fichier journal en utilisant la AWS Management Console. RDSactualise la fin du journal toutes les 5 secondes.

Pour consulter un fichier journal de base de données

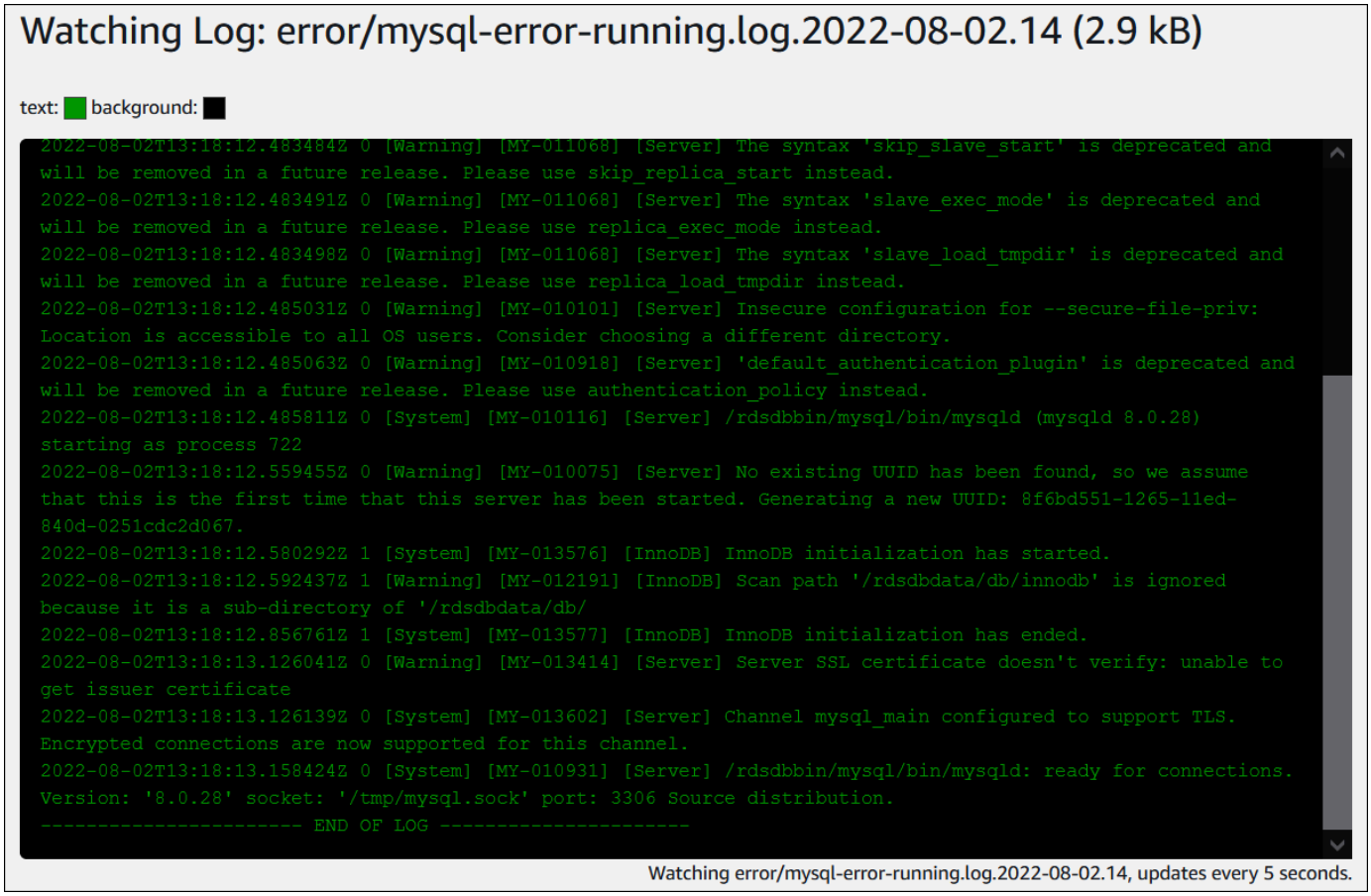
1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de l'instance de base de données qui contient le fichier journal que vous voulez consulter.
4. Choisissez l'onglet Logs & events (Journaux et événements).



5. Dans la section Journaux, choisissez un fichier journal, puis Consulter.



RDSmontre la fin du journal, comme dans mon SQL exemple suivant.



Publication des journaux de base de données sur Amazon CloudWatch Logs

Dans une base de données sur site, les journaux de la base de données résident sur le système de fichiers. Amazon RDS ne fournit pas d'accès hôte aux journaux de la base de données sur le système de fichiers de votre instance de base de données. C'est pourquoi Amazon RDS vous permet d'exporter les journaux de base de données vers Amazon CloudWatch Logs. Avec CloudWatch Logs, vous pouvez effectuer une analyse en temps réel des données du journal. Vous pouvez également stocker les données dans un espace de stockage hautement durable et les gérer à l'aide de l'agent CloudWatch Logs.

Rubriques

- Vue d'ensemble de l'intégration de RDS avec CloudWatch Logs
- Choix des journaux à publier dans CloudWatch Logs
- Spécification des journaux à publier dans CloudWatch Logs

- Recherche et filtrage de vos journaux dans CloudWatch Logs

Vue d'ensemble de l'intégration de RDS avec CloudWatch Logs

Dans CloudWatch Logs, un flux de log est une séquence d'événements de log qui partagent la même source. Chaque source distincte de CloudWatch journaux dans Logs constitue un flux de journaux distinct. Un groupe de journaux est un groupe de flux de journaux qui partagent les mêmes paramètres de conservation, de surveillance et de contrôle d'accès.

Amazon RDS diffuse en continu les enregistrements des journaux de votre instance de base de données vers un groupe de journaux. Par exemple, vous possédez un groupe de journaux `/aws/rds/instance/`*`instance_name`*`/`*`log_type`* pour chaque type de journaux que vous publiez. Ce groupe de journaux se trouve dans la même AWS région que l'instance de base de données qui génère le journal.

AWS conserve les données de journal publiées dans CloudWatch Logs pour une durée indéfinie, sauf si vous spécifiez une période de conservation. Pour plus d'informations, consultez la section Conservation des données du journal des modifications dans CloudWatch les journaux.

Choix des journaux à publier dans CloudWatch Logs

Chaque moteur de base de données RDS prend en charge son propre ensemble de journaux. Pour en savoir plus sur les options de votre moteur de base de données, consultez les rubriques suivantes :

- the section called “Publication de journaux DB2 sur Amazon CloudWatch Logs”
- the section called “Publier des logs MariaDB sur Amazon Logs CloudWatch ”
- the section called “Publication de journaux MySQL sur Amazon CloudWatch Logs”
- the section called “Publication de journaux Oracle sur Amazon CloudWatch Logs”
- the section called “Publication de journaux PostgreSQL sur Amazon Logs CloudWatch ”
- the section called “Publication des journaux SQL du serveur sur Amazon CloudWatch Logs”

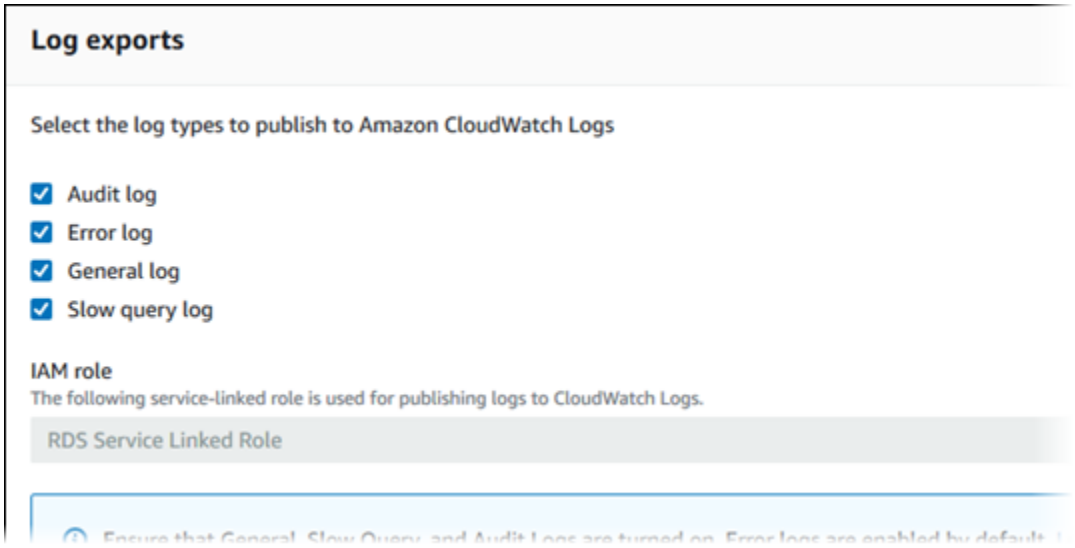
Spécification des journaux à publier dans CloudWatch Logs

Vous spécifiez les journaux à publier dans la console. Assurez-vous que vous disposez d'un rôle lié à un service dans AWS Identity and Access Management (IAM). Pour plus d’informations sur les rôles liés à un service, consultez Utilisation des rôles liés à un service pour Amazon RDS.

Pour spécifier les journaux à publier

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Effectuez l'une des actions suivantes :
   - Choisissez Créer une base de données.
   - Choisissez une base de données dans la liste, puis sélectionnez Modify (Modifier).
4. Dans Logs exports (Exportations de journaux), choisissez les journaux à publier.

   L'exemple suivant indique le journal d'audit, les journaux d'erreurs, le journal général, le journal d' et le journal des requêtes lentes pour une instance de base de RDS pour MySQL.



Recherche et filtrage de vos journaux dans CloudWatch Logs

Vous pouvez rechercher des entrées de journal répondant à des critères spécifiques à l'aide de la console CloudWatch Logs. Vous pouvez accéder aux journaux soit par le biais de la console RDS, qui vous mène à la console CloudWatch Logs, soit directement depuis la console CloudWatch Logs.

Pour rechercher les journaux de votre RDS à l'aide de la console RDS

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez un instance de base de données.
4. Choisissez Configuration.

5. Sous Published logs (Journaux publiés), choisissez le journal de la base de données que vous souhaitez afficher.

Pour effectuer une recherche dans vos journaux RDS à l'aide de la console CloudWatch Logs

1. Ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.
2. Dans le panneau de navigation, choisissez Groupes de journaux.
3. Dans la zone de filtre, entrez **`/aws/rds`**.
4. Pour Log Groups, choisissez le nom du groupe de journaux contenant le flux de journal devant faire l'objet de la recherche.
5. Pour Log Streams, choisissez le nom du flux de journal devant faire l'objet de la recherche.
6. Sous Journal des événements, saisissez la syntaxe du filtre à utiliser.

Pour plus d'informations, consultez la section Recherche et filtrage des données de journal dans le guide de l'utilisateur Amazon CloudWatch Logs. Pour consulter un didacticiel de blog expliquant comment surveiller les journaux RDS, consultez Créer une surveillance proactive des bases de données pour Amazon RDS avec Amazon CloudWatch Logs, AWS Lambda et Amazon SNS.

Lire le contenu du fichier journal à l'aide de REST

Amazon RDS fournit un REST point de terminaison qui permet d'accéder aux fichiers journaux des instances de base de données. Cela est utile si vous devez écrire une application pour diffuser le contenu du fichier RDS journal Amazon.

La syntaxe est la suivante :

```
GET /v13/downloadCompleteLogFile/DBInstanceIdentifier/LogFileName HTTP/1.1
Content-type: application/json
host: rds.region.amazonaws.com
```

Les paramètres suivants sont obligatoires :

- *`DBInstanceIdentifier`*—le nom assigné par le client de l'instance de base de données qui contient le fichier journal que vous souhaitez télécharger.
- *`LogFileName`*—le nom du fichier journal à télécharger.

La réponse contient les contenus du fichier journal demandé, en tant que flux.

L'exemple suivant télécharge le fichier journal nommé log/ ERROR .6 pour l'instance de base de données nommée sample-sql dans la région us-west-2.

```
GET /v13/downloadCompleteLogFile/sample-sql/log/ERROR.6 HTTP/1.1
host: rds.us-west-2.amazonaws.com
X-Amz-Security-Token: AQoDYXdzEIH//////////
wEa0AIXLhngC5zp9CyB1R6abwKrXHVR5efnAVN3XvR7IwqKYalFSn6UyJuEFTft9nObglx4QJ+GXV9cpACkETq=
X-Amz-Date: 20140903T233749Z
X-Amz-Algorithm: AWS4-HMAC-SHA256
X-Amz-Credential: AKIADQKE4SARGYLE/20140903/us-west-2/rds/aws4_request
X-Amz-SignedHeaders: host
X-Amz-Content-SHA256: e3b0c44298fc1c229afbf4c8996fb92427ae41e4649b934de495991b7852b855
X-Amz-Expires: 86400
X-Amz-Signature: 353a4f14b3f250142d9afc34f9f9948154d46ce7d4ec091d0cdabbcf8b40c558
```

Si vous spécifiez une instance de base de données qui n'existe pas, la réponse se compose de l'erreur suivante :

- `DBInstanceNotFound`—*`DBInstanceIdentifier`* ne fait pas référence à une instance de base de données existante. (code HTTP d'état : 404)

Fichiers journaux RDS de base de données Amazon pour DB2

Vous pouvez accéder RDS aux journaux de diagnostic de DB2 et aux journaux de notification à l'aide de la RDS console Amazon AWS CLI, ou RDSAPI. Pour plus d'informations sur l'affichage, le téléchargement ou la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon.

Rubriques

- Programme de rétention
- Publication de journaux DB2 sur Amazon CloudWatch Logs

Programme de rétention

Les fichiers journaux font l'objet d'une rotation chaque jour et chaque fois que votre instance de base de données est redémarrée. Voici le calendrier de conservation RDS des journaux DB2 sur AmazonRDS.

Log type (Type de journal)	Programme de rétention
Journaux de diagnostic	Db2 supprime les journaux en dehors des paramètres de rétention dans la configuration au niveau de l'instance. Amazon RDS définit le `diagsize` paramètre sur 1000.
Notifier les journaux	Db2 supprime les journaux en dehors des paramètres de rétention dans la configuration au niveau de l'instance. Amazon RDS définit le `diagsize` paramètre sur 1000.

Publication de journaux DB2 sur Amazon CloudWatch Logs

Avec RDS for Db2, vous pouvez publier des diagnostics et notifier les événements du journal directement sur Amazon CloudWatch Logs. Analysez les données du journal avec CloudWatch Logs, puis utilisez-les CloudWatch pour créer des alarmes et afficher les métriques.

Avec CloudWatch Logs, vous pouvez effectuer les opérations suivantes :

- Stocker des journaux dans un espace de stockage hautement durable pour lequel vous définissez la période de rétention.
- Chercher et filtrer les données de journaux.
- Partager des données de journaux entre les comptes.
- Exporter des journaux vers Amazon S3.
- Diffusez des données vers Amazon OpenSearch Service.
- Traiter des données de journaux en temps réel avec Amazon Kinesis Data Streams. Pour plus d'informations, consultez le guide du développeur de Working with Amazon CloudWatch Logs in the Amazon Managed Service for Apache Flink for SQL Applications.

Amazon RDS publie chaque journal RDS de base de données DB2 en tant que flux de base de données distinct dans le groupe de journaux. Par exemple, si vous publiez les journaux de diagnostic et les journaux de notifications, les données de diagnostic sont stockées dans un flux de journaux de diagnostic du groupe de `/aws/rds/instance/my_instance/diagnostic` journaux, et les données du journal de notifications sont stockées dans le groupe de `/aws/rds/instance/my_instance/notify` journaux.

> **Note**
>
> La publication RDS des journaux DB2 dans CloudWatch Logs n'est pas activée par défaut. La publication des journaux des statistiques du gestionnaire de mémoire à réglage automatique (STMM) et de l'optimiseur n'est pas prise en charge. La publication RDS de journaux DB2 vers CloudWatch Logs est prise en charge dans toutes les régions, à l'exception de l'Asie-Pacifique (Hong Kong).

Console

RDSPour publier des journaux DB2 dans des CloudWatch journaux à partir du AWS Management Console

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).

4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.

   Vous pouvez choisir diag.log, notify.log ou les deux.

5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Pour publier RDS des journaux pour DB2, vous pouvez utiliser la `modify-db-instance`commande avec les paramètres suivants :

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier des journaux RDS pour DB2 à l'aide des commandes suivantes :

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-to-point-in-time`

Example

L'exemple suivant crée une instance de base de données RDS for DB2 avec la publication CloudWatch des journaux activée. La `--enable-cloudwatch-logs-exports` valeur est un JSON tableau de chaînes pouvant inclure `diag.lognotify.log`, ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
```

```
    --enable-cloudwatch-logs-exports '["diag.log","notify.log"]' \
    --db-instance-class db.m4.large \
    --engine db2-se
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --enable-cloudwatch-logs-exports "[\"diag.log\",\"notify.log\"]" ^
    --db-instance-class db.m4.large ^
    --engine db2-se
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Example

L'exemple suivant modifie une instance de base de données existante RDS pour DB2 afin de publier des fichiers journaux dans Logs. CloudWatch La `--cloudwatch-logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes qui peut inclure `diag.log`, `notify.log` ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["diag.log","notify.log"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration "{\"EnableLogTypes\":[\"diag.log\",
\"notify.log\"]}"
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Example

L'exemple suivant modifie une instance de base de données Db2 existante RDS afin de désactiver la publication de fichiers journaux de diagnostic dans Logs. CloudWatch La `--cloudwatch-logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `DisableLogTypes` et sa valeur est un tableau de chaînes qui peut inclure `diag.log`, `notify.log` ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"DisableLogTypes":["diag.log"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration "{\"DisableLogTypes\":[\"diag.log\"]}"
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Fichiers journaux de base de données MariaDB

Vous pouvez surveiller le journal des erreurs MariaDB, le journal des requêtes lentes, le journal des erreurs d'authentification de la base de données IAM et le journal général. Le journal des erreurs MySQL est généré par défaut. Vous pouvez générer le journal des requêtes lentes et le journal général en définissant les paramètres nécessaires dans votre groupe de paramètres DB. Amazon RDS assura la rotation de tous les fichiers journaux MariaDB. Les intervalles pour chaque type sont donnés ci-dessous.

Vous pouvez surveiller les journaux MariaDB directement via la console Amazon RDS, l'API Amazon RDS, la CLI Amazon RDS ou. AWS SDKs Vous pouvez également accéder aux journaux MariaDB en dirigeant les journaux vers une table de base de données de la base de données principale et interroger cette table. Vous pouvez utiliser l'utilitaire mysqlbinlog pour télécharger un journal binaire.

Pour plus d'informations sur l'affichage, le téléchargement ou la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon.

Rubriques

- Accès aux journaux des erreurs MariaDB
- Accès au journal des requêtes lentes et au journal général MariaDB
- Publier des logs MariaDB sur Amazon Logs CloudWatch
- Rotation et conservation des journaux pour MariaDB
- Gestion des journaux MariaDB sous forme de table
- Configuration de la journalisation binaire MariaDB
- Accès aux journaux binaires MariaDB
- Activation de l'annotation du journal binaire MariaDB

Accès aux journaux des erreurs MariaDB

Le journal des erreurs MariaDB est écrit dans le fichier `<host-name>.err`. Vous pouvez consulter ce fichier à l'aide de la RDS console Amazon. Vous pouvez également récupérer le journal à l'aide d'Amazon RDS API RDSCLI, Amazon ou AWS SDKs. Le fichier `<host-name>.err` est vidé toutes les 5 minutes, et son contenu est ajouté à `mysql-error-running.log`. Le fichier `mysql-error-running.log` fait ensuite l'objet d'une rotation toutes les heures et les fichiers générés toutes les heures au cours des 24 dernières heures sont conservés. L'heure à laquelle il a été généré est

ajoutée à UTC son nom pour chaque fichier journal. Les fichiers journaux comportent également un horodatage permettant de déterminer le moment où les entrées du journal ont été écrites.

MariaDB écrit des informations dans le fichier des erreurs uniquement au moment du démarrage, de l'arrêt et lorsqu'une erreur survient. Une instance de base de données peut fonctionner pendant des heures ou des jours sans qu'aucune nouvelle entrée soit écrite dans le journal des erreurs. Si aucune entrée récente ne figure, cela signifie que le serveur n'a pas rencontré d'erreur justifiant une entrée de journal.

Accès au journal des requêtes lentes et au journal général MariaDB

Le journal des requêtes lentes MariaDB et le journal général peuvent être écrits dans un fichier ou dans une table de base de données en définissant les paramètres nécessaires dans votre groupe de paramètres de base de données. Pour plus d'informations sur la création et la modification d'un groupe de paramètres DB, consultez Groupes de paramètres pour (Amazon RDS). Vous devez définir ces paramètres avant de pouvoir consulter le journal des requêtes lentes ou le journal général dans la RDS console Amazon ou à l'aide d'Amazon RDS API AWS CLI, ou AWS SDKs.

Vous pouvez contrôler la journalisation MariaDB à l'aide des paramètres de cette liste :

- `slow_query_log`ou `log_slow_query` : pour créer le journal des requêtes lentes, définissez ce paramètre sur 1. La valeur par défaut est 0.
- `general_log` : Pour créer le journal général, définir sur 1. La valeur par défaut est 0.
- `long_query_time`ou `log_slow_query_time` : pour éviter que les requêtes rapides ne soient enregistrées dans le journal des requêtes lentes, spécifiez une valeur pour la durée d'exécution des requêtes la plus courte à enregistrer, en secondes. La valeur par défaut est de 10 secondes et la valeur minimum est 0. Si log_output =FILE, vous pouvez spécifier une valeur à virgule flottante correspondant à une résolution de l'ordre de la microseconde. Si log_output =TABLE, vous devez spécifier une valeur entière avec une deuxième résolution. Seules les requêtes dont le temps d'exécution est supérieur à la `log_slow_query_time` valeur `long_query_time` or sont enregistrées. Par exemple, si vous définissez `long_query_time` ou `log_slow_query_time` sur 0,1, toute requête exécutée pendant moins de 100 millisecondes ne sera enregistrée.
- `log_queries_not_using_indexes` : Pour enregistrer toutes les requêtes n'utilisant pas d'index dans le journal des requêtes lentes, définir ce paramètre sur 1. La valeur par défaut est 0. Les requêtes n'utilisant pas d'index sont enregistrées même si la durée de leur exécution est inférieure à la valeur du paramètre `long_query_time`.
- `log_output` *`option`* : Vous pouvez spécifier l'une des options suivantes pour le paramètre `log_output` :

- TABLE(par défaut) — Écrivez des requêtes générales dans la `mysql.general_log` table et des requêtes lentes dans la `mysql.slow_log` table.
- FILE— Rédigez des journaux de requêtes générales et lentes dans le système de fichiers. Les fichiers journaux font l'objet d'une rotation horaire.
- NONE— Désactive la journalisation.

Lorsque la journalisation est activée, Amazon RDS fait pivoter les journaux de table ou supprime les fichiers journaux à intervalles réguliers. Cette précaution permet de limiter la possibilité qu'un fichier journal volumineux ne bloque l'utilisation de la base de données ou n'affecte les performances. Rotation et suppression de l'approche de journalisation `FILE` et `TABLE` comme suit :

- Lorsque la journalisation `FILE` est activée, les fichiers journaux sont examinés toutes les heures et ceux dont l'ancienneté est supérieure à 24 heures sont supprimés. Dans certains cas, la taille des fichiers journaux combinés restant après la suppression peut dépasser le seuil de 2 % de l'espace alloué à une instance de base de données. Dans ces cas, les fichiers journaux les plus volumineux sont supprimés jusqu'à ce que la taille des fichiers journaux ne soit plus supérieure au seuil.
- Lorsque la journalisation de `TABLE` est activée, les journaux des tables font dans certains cas l'objet d'une rotation toutes les 24 heures. Cette rotation se produit si l'espace utilisé par les journaux des tables est supérieur à 20 % de l'espace de stockage alloué. Cela se produit également si la taille de tous les journaux combinés est supérieure à 10 Go. Si l'espace utilisé pour une instance de base de données est supérieur à 90 % de l'espace de stockage alloué à l'instance de base de données, alors les seuils correspondant à la rotation des journaux sont réduits. La rotation des journaux des tables se produit ensuite si l'espace utilisé par les journaux des tables est supérieur à 10 % de l'espace de stockage alloué. Elle se produit également si la taille de tous les journaux combinés est supérieure à 5 Go.

  Lors de la rotation des tables de journaux, la table de journal actuelle est copiée vers une table de journal de sauvegarde et les entrées de la table de journal actuelle sont supprimées. Si la table de journal de sauvegarde existe déjà, elle est supprimée avant que la table de journal actuelle ne soit copiée dans la sauvegarde. Si besoin, vous pouvez interroger la table de journal de sauvegarde. La table de journal de sauvegarde de la table `mysql.general_log` est nommée `mysql.general_log_backup`. La table de journal de sauvegarde de la table `mysql.slow_log` est nommée `mysql.slow_log_backup`.

Vous pouvez effectuer une rotation de la table `mysql.general_log` en appelant la procédure `mysql.rds_rotate_general_log`. Vous pouvez effectuer une rotation de la table `mysql.slow_log` en appelant la procédure `mysql.rds_rotate_slow_log`.

La rotation des journaux des tables est effectuée pendant la mise à niveau de la version d'une base de données.

Amazon RDS enregistre à la fois la rotation `TABLE` des `FILE` journaux et la rotation des journaux lors d'un RDS événement Amazon et vous envoie une notification.

Pour utiliser les journaux depuis la RDS console Amazon RDSAPI, Amazon RDS CLI AWS SDKs, ou définissez le `log_output` paramètre surFILE. A l'instar du journal des erreurs MariaDB, ces fichiers journaux font l'objet d'une rotation horaire. Les fichiers journaux qui ont été générés au cours des dernières 24 heures sont conservés.

Pour plus d'informations sur le journal des requêtes lentes et le journal général, accédez aux rubriques suivantes dans la documentation MariaDB :

- Journal des requêtes lentes
- Journal des requêtes générales

Publier des logs MariaDB sur Amazon Logs CloudWatch

Vous pouvez configurer votre instance de base de données MariaDB pour publier les données de journal dans un groupe de journaux dans Amazon Logs. CloudWatch Avec CloudWatch Logs, vous pouvez effectuer une analyse en temps réel des données du journal, puis les utiliser CloudWatch pour créer des alarmes et afficher des métriques. Vous pouvez utiliser CloudWatch les journaux pour stocker vos enregistrements de journal dans un espace de stockage hautement durable.

Amazon RDS publie chaque journal de base de données MariaDB sous la forme d'un flux de base de données distinct dans le groupe de journaux. Par exemple, supposons que vous configuriez la fonction d'exportation de manière à inclure le journal de requêtes lentes. Les données de requêtes lentes sont ensuite stockées dans un flux de journaux de requêtes lentes dans le groupe de journaux `/aws/rds/instance/my_instance/slowquery`.

Le journal d'erreurs est activé par défaut. Le tableau ci-dessous récapitule les conditions requises pour les autres journaux MariaDB.

Log	Exigence
Journal d'audit	L'instance de base de données doit disposer d'un groupe d'options personnalisées avec l'option `MARIADB_AUDIT_PLUGIN` .
Journal général	L'instance de base de données doit disposer d'un groupe de paramètres personnalisés avec le paramètre `general_log = 1` pour autoriser la journalisation générale.
Journal des requêtes lentes	L'instance de base de données doit utiliser un groupe de paramètres personnalisé avec le réglage des paramètres `slow_query_log = 1` ou `log_slow_query = 1` pour activer le journal des requêtes lentes.
Journal des erreurs d'authentification de base de données IAM	Vous devez activer le type de journal `iam-db-auth-error` pour une instance de base de données en créant ou en modifiant une instance de base de données.
Sortie de journal	L'instance de base de données doit utiliser un groupe de paramètres personnalisé avec le paramètre défini `log_output = FILE` pour écrire des journaux dans le système de fichiers et les publier dans CloudWatch Logs.

console

Pour publier les journaux MariaDB dans Logs depuis CloudWatch la console

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).

4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.
5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Vous pouvez publier un journal MariaDB avec le. AWS CLI Vous pouvez appeler la commande `modify-db-instance` avec les paramètres suivants :

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier des journaux MariaDB en appelant les commandes suivantes : AWS CLI

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-from-s3`
- `restore-db-instance-to-point-in-time`

Exécutez l'une de ces AWS CLI commandes avec les options suivantes :

- `--db-instance-identifier`
- `--enable-cloudwatch-logs-exports`
- `--db-instance-class`
- `--engine`

D'autres options peuvent être nécessaires en fonction de la AWS CLI commande que vous exécutez.

Example

L'exemple suivant modifie une instance de base de données MariaDB existante pour publier des fichiers journaux dans Logs. CloudWatch La valeur `--cloudwatch-logs-export-configuration` n'est pas un objet JSON. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes avec une combinaison quelconque de `audit`, `error`, `general` et `slowquery`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["audit","error","general","slowquery"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["audit","error","general","slowquery"]}'
```

Example

La commande suivante crée une instance de base de données MariaDB et publie les fichiers journaux dans Logs. CloudWatch La valeur `--enable-cloudwatch-logs-exports` est un tableau de chaînes JSON. Les chaînes peuvent être une combinaison de `audit`, `error`, `general` et `slowquery`.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --enable-cloudwatch-logs-exports '["audit","error","general","slowquery"]' \
    --db-instance-class db.m4.large \
    --engine mariadb
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --enable-cloudwatch-logs-exports '["audit","error","general","slowquery"]' ^
    --db-instance-class db.m4.large ^
    --engine mariadb
```

API RDS

Vous pouvez publier des journaux MariaDB avec l'API RDS. Vous pouvez appeler l'opération ModifyDBInstance avec les paramètres suivants :

- DBInstanceIdentifier
- CloudwatchLogsExportConfiguration

> **Note**
>
> Une modification apportée au paramètre CloudwatchLogsExportConfiguration est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, le paramètre ApplyImmediately est sans effet.

Vous pouvez également publier des journaux MariaDB en appelant les opérations d'API RDS suivantes :

- CreateDBInstance
- RestoreDBInstanceFromDBSnapshot
- RestoreDBInstanceFromS3
- RestoreDBInstanceToPointInTime

Exécutez l'une de ces opérations d'API RDS avec les paramètres suivants :

- DBInstanceIdentifier
- EnableCloudwatchLogsExports
- Engine
- DBInstanceClass

D'autres paramètres peuvent être nécessaires en fonction de la AWS CLI commande que vous exécutez.

Rotation et conservation des journaux pour MariaDB

Lorsque la journalisation est activée, Amazon RDS effectue une rotation des journaux des tables ou supprime les fichiers journaux à intervalles réguliers. Cette précaution permet de limiter la possibilité qu'un fichier journal volumineux ne bloque l'utilisation de la base de données ou n'affecte les performances.

Les tailles du journal des requêtes lentes, du journal des erreurs et du journal général MariaDB sont limitées à 2 %maximum de l'espace de stockage alloué à une instance de base de données. Pour respecter ce seuil, les journaux font l'objet d'une rotation automatique toutes les heures et les fichiers dont l'ancienneté est supérieure à 24 heures sont supprimés. Si la taille de l'ensemble des fichiers journaux après la suppression dépasse le seuil, les fichiers journaux les plus volumineux sont supprimés jusqu'à ce que la taille des fichiers journaux ne soit plus supérieure au seuil.

Amazon RDS fait pivoter les fichiers journaux d'erreurs d'authentification de base de données IAM supérieurs à 10 Mo. Amazon RDS supprime les fichiers journaux d'erreurs d'authentification de base de données IAM datant de plus de cinq jours ou de plus de 100 Mo.

Gestion des journaux MariaDB sous forme de table

Vous pouvez diriger les journaux des requêtes générales et lentes vers des tables sur l'instance de base de données. Pour ce faire, créez un groupe de paramètres de base de données et définissez le paramètre de serveur `log_output` sur `TABLE`. Les requêtes générales sont ensuite enregistrées dans la table `mysql.general_log` et les requêtes lentes dans la table `mysql.slow_log`. Vous pouvez interroger les tables pour accéder aux informations des journaux. L'activation de cette journalisation augmente le volume de données écrites dans la base de données, ce qui peut dégrader les performances.

Par défaut, le journal général et le journal des requêtes lentes sont désactivés. Pour activer la journalisation dans les tables, vous devez également définir les paramètres de serveur suivants sur `1` :

- `general_log`
- `slow_query_log` ou `log_slow_query`

Les tables de journaux continuent de grossir jusqu'à ce que les activités de journalisation correspondantes soient désactivées en redéfinissant le paramètre approprié sur `0`. Avec le temps, une grande quantité de données s'accumule et risque d'utiliser une part considérable de l'espace de stockage alloué. Amazon RDS ne vous autorise pas à tronquer les tables de journaux, mais vous pouvez déplacer leur contenu. Lorsque vous procédez à la rotation d'une table, son contenu est enregistré dans une table de sauvegarde et une nouvelle table de journal vide est créée. Vous pouvez effectuer une rotation manuelle des tables de journaux avec les procédures de ligne de commande suivantes, dans lesquelles l'invite de commande est indiquée par `PROMPT>`:

```
PROMPT> CALL mysql.rds_rotate_slow_log;
PROMPT> CALL mysql.rds_rotate_general_log;
```

Pour supprimer totalement les anciennes données et récupérer l'espace de disque, appelez deux fois à la suite la procédure appropriée.

Configuration de la journalisation binaire MariaDB

Le journal binaire est un ensemble de fichiers journaux contenant des informations sur les modifications de données apportées à une instance de serveur MariaDB. Le journal binaire contient des informations telles que les suivantes :

- Événements décrivant les modifications apportées à la base de données telles que la création de tables ou les modifications de lignes
- Informations sur la durée de chaque instruction qui a mis à jour les données
- Événements pour des instructions pouvant mettre à jour des données mais ne l'ayant pas fait

Le journal binaire enregistre les instructions envoyées pendant la réplication. Il est également requis pour certaines opérations de récupération. Pour plus d'informations, consultez Binary Log dans la documentation de MariaDB.

La fonctionnalité de sauvegarde automatique détermine si la journalisation binaire est activée ou désactivée pour MariaDB. Vous avez les options suivantes :

Activer la journalisation binaire

Définissez la période de rétention des sauvegardes sur une valeur positive différente de zéro.

Désactiver la journalisation binaire

Définissez la période de rétention des sauvegardes sur zéro.

Pour de plus amples informations, veuillez consulter Activation des sauvegardes automatiques.

MariaDB sur RDS Amazon prend en charge les formats de journalisation binaire basés sur des lignes, basés sur des instructions et mixtes. Le format de journalisation binaire par défaut est mixte. Pour plus de détails sur les différents formats de journalisation binaire MariaDB, consultez Formats de journalisation binaire dans la documentation MariaDB.

Si vous envisagez d'utiliser la réplication, le format de journalisation binaire est important. Il détermine le dossier de modifications de données qui est enregistré dans la source et envoyés aux cibles de réplication. Pour plus d'informations sur les avantages et les inconvénients des différents formats de journalisation binaire pour la réplication, consultez la section Avantages et inconvénients de la réplication basée sur des instructions et basée sur des lignes dans la section Ma documentation. SQL

> **Important**
>
> Lorsque vous définissez le format de journalisation binaire sur « basé sur les lignes », vous risquez de générer des fichiers journaux binaires très volumineux. Les fichiers de journaux binaires importants réduisent le stockage disponible pour une instance de base de données. Ils peuvent également augmenter le temps nécessaire pour effectuer une opération de restauration d'une instance de base de données.
>
> La réplication basée sur les instructions peut provoquer des incohérences entre l'instance de base de données source et un réplica en lecture. Pour plus d'informations, consultez la section Unsafe Statements for Statement-based Replication dans la documentation MariaDB.
>
> L'activation de la journalisation binaire augmente le nombre d'opérations d'E/S de disque d'écriture sur l'instance de base de données. Vous pouvez surveiller IOPS l'utilisation à l'aide de la `WriteIOPS` CloudWatch métrique.

Pour définir le format de journalisation binaire MariaDB :

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Groupes de paramètres.
3. Choisissez le groupe de paramètres utilisé par l'instance de base de données que vous souhaitez modifier.

Vous ne pouvez pas modifier un groupe de paramètres par défaut. Si l'instance de base de données utilise un groupe de paramètres par défaut, créez un nouveau groupe et associez-le à l'instance.

Pour plus d'informations sur les groupes de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).

4. Sous Parameter group actions (Actions de groupe de paramètres), choisissez Edit (Modifier).
5. Définissez le paramètre `binlog_format` au format de journalisation binaire de votre choix (ROW, STATEMENT ou MIXED).

   Vous pouvez désactiver la journalisation binaire en définissant la période de conservation des sauvegardes d'une instance de base de données sur zéro, mais cela désactive les sauvegardes automatiques quotidiennes. La désactivation des sauvegardes automatiques désactive ou désactive la variable de `log_bin` session. Cela désactive la journalisation binaire sur l'instance RDS de base de données MariaDB, qui à son tour réinitialise `binlog_format` la variable de session à la valeur par défaut de dans la base de données. `ROW` Nous vous recommandons de ne pas désactiver les sauvegardes. Pour plus d'informations sur le paramètre Période de rétention des sauvegardes, consultez Paramètres des instances de base de données.
6. Choisissez Enregistrer les modifications pour enregistrer les mises à jour apportées au groupe de paramètres de base de données.

Comme le `binlog_format` paramètre est dynamique dans RDS MariaDB, il n'est pas nécessaire de redémarrer l'instance de base de données pour que les modifications s'appliquent.

> **Important**
>
> La modification d'un groupe de paramètres de base de données affecte toutes les instances de base de données qui utilisent ce dernier. Si vous souhaitez spécifier différents formats de journalisation binaire pour différentes instances de base de données MariaDB dans AWS une région, les instances de base de données doivent utiliser différents groupes de paramètres de base de données. Ces groupes de paramètres identifient différents formats de journalisation. Affectez le groupe de paramètres de base de données approprié à chaque instance de base de données.

Accès aux journaux binaires MariaDB

Vous pouvez utiliser l'utilitaire mysqlbinlog pour télécharger les journaux binaires au format texte à partir d'instances de base de données MariaDB. Le journal binaire est téléchargé dans votre ordinateur local. Pour plus d'informations sur l'utilisation de l'utilitaire mysqlbinlog, accédez à Utilisation de mysqlbinlog dans la documentation MariaDB.

Pour exécuter l'utilitaire mysqlbinlog sur une RDS instance Amazon, utilisez les options suivantes :

- Spécifiez l'option `--read-from-remote-server`.
- `--host`: Spécifiez le DNS nom à partir du point de terminaison de l'instance.
- `--port` : Spécifiez le port utilisé par l'instance.
- `--user` : Spécifiez un utilisateur MariaDB ayant l'autorisation de réplication esclave.
- `--password` : Spécifiez le mot de passe de l'utilisateur ou omettez la valeur de mot de passe pour que l'utilitaire vous invite à saisir un mot de passe.
- `--result-file` : Spécifiez le fichier local qui recevra la sortie.
- Spécifiez les noms pour un ou plusieurs fichiers journaux binaires. Pour obtenir la liste des journaux disponibles, utilisez la SQL commande SHOW BINARYLOGS.

Pour plus d'informations sur les options mysqlbinlog, accédez aux Options mysqlbinlog dans la documentation MariaDB.

Voici un exemple :

Dans Linux, macOS, ou Unix:

```
mysqlbinlog \
    --read-from-remote-server \
    --host=mariadbinstance1.1234abcd.region.rds.amazonaws.com \
    --port=3306  \
    --user ReplUser \
    --password <password> \
    --result-file=/tmp/binlog.txt
```

Dans Windows:

```
mysqlbinlog ^
    --read-from-remote-server ^
    --host=mariadbinstance1.1234abcd.region.rds.amazonaws.com ^
```

```
    --port=3306  ^
    --user ReplUser ^
    --password <password> ^
    --result-file=/tmp/binlog.txt
```

Amazon purge RDS normalement un journal binaire dès que possible. Toutefois, le journal binaire doit toujours être disponible sur l'instance afin que mysqlbinlog puisse y accéder. Pour spécifier le nombre d'heures pendant lesquelles RDS les journaux binaires doivent être conservés, utilisez la procédure `mysql.rds_set_configuration` stockée. Spécifiez une période suffisamment longue pour télécharger les journaux. Après avoir défini la période de rétention, surveillez l'utilisation du stockage de l'instance de base de données afin de garantir que les journaux binaires conservés n'utilisent pas un espace de stockage trop grand.

L'exemple suivant définit la période de conservation sur 1 jour.

```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

Pour afficher les paramètres actuels, utilisez la procédure stockée `mysql.rds_show_configuration`.

```
call mysql.rds_show_configuration;
```

Activation de l'annotation du journal binaire MariaDB

Dans une instance de base de données MariaDB, vous pouvez utiliser `Annotate_rows` l'événement pour annoter un événement de ligne avec une copie de SQL la requête à l'origine de l'événement de ligne. Cette approche fournit des fonctionnalités similaires à celles de l'activation du `binlog_rows_query_log_events` paramètre sur une instance RDS for My SQL DB.

Vous pouvez activer globalement les annotations des journaux binaires en créant un groupe de paramètres personnalisé et en définissant le paramètre `binlog_annotate_row_events` sur **1**. Vous pouvez également activer les annotations au niveau de la session, en appelant `SET SESSION binlog_annotate_row_events = 1`. Utilisez `replicate_annotate_row_events` pour répliquer les annotations des journaux binaires dans l'instance de réplica si la journalisation binaire est activée pour cette instance. Aucun privilège spécial n'est nécessaire pour utiliser ces paramètres.

L'exemple suivant illustre une transaction basée sur les lignes dans MariaDB. L'utilisation de la journalisation basée sur les lignes est déclenchée en définissant le niveau d'isolement des transactions sur read-committed.

```
CREATE DATABASE IF NOT EXISTS test;
USE test;
CREATE TABLE square(x INT PRIMARY KEY, y INT NOT NULL) ENGINE = InnoDB;
SET SESSION TRANSACTION ISOLATION LEVEL READ COMMITTED;
BEGIN
INSERT INTO square(x, y) VALUES(5, 5 * 5);
COMMIT;
```

Sans les annotations, les entrées du journal binaire pour la transaction ressemblent à :

```
BEGIN
/*!*/;
# at 1163
# at 1209
#150922  7:55:57 server id 1855786460  end_log_pos 1209         Table_map:
 `test`.`square` mapped to number 76
#150922  7:55:57 server id 1855786460  end_log_pos 1247         Write_rows: table id 76
 flags: STMT_END_F
### INSERT INTO `test`.`square`
### SET
###   @1=5
###   @2=25
# at 1247
#150922  7:56:01 server id 1855786460  end_log_pos 1274         Xid = 62
COMMIT/*!*/;
```

L'instruction suivante active les annotations au niveau de la session pour cette même transaction, et les désactive après validation de la transaction :

```
CREATE DATABASE IF NOT EXISTS test;
USE test;
CREATE TABLE square(x INT PRIMARY KEY, y INT NOT NULL) ENGINE = InnoDB;
SET SESSION TRANSACTION ISOLATION LEVEL READ COMMITTED;
SET SESSION binlog_annotate_row_events = 1;
BEGIN;
INSERT INTO square(x, y) VALUES(5, 5 * 5);
COMMIT;
SET SESSION binlog_annotate_row_events = 0;
```

Avec les annotations, les entrées du journal binaire pour la transaction ressemblent à :

```
BEGIN
```

```
/*!*/;
# at 423
# at 483
# at 529
#150922  8:04:24 server id 1855786460  end_log_pos 483  Annotate_rows:
#Q> INSERT INTO square(x, y) VALUES(5, 5 * 5)
#150922  8:04:24 server id 1855786460  end_log_pos 529  Table_map: `test`.`square`
 mapped to number 76
#150922  8:04:24 server id 1855786460  end_log_pos 567  Write_rows: table id 76 flags:
 STMT_END_F
### INSERT INTO `test`.`square`
### SET
###   @1=5
###   @2=25
# at 567
#150922  8:04:26 server id 1855786460  end_log_pos 594  Xid = 88
COMMIT/*!*/;
```

Fichiers journaux RDS de base de données Amazon pour Microsoft SQL Server

Vous pouvez accéder aux journaux d'erreurs, aux journaux des agents, aux fichiers de trace et aux fichiers dump de Microsoft SQL Server à l'aide de la RDS console Amazon AWS CLI, ou RDSAPI. Pour plus d'informations sur l'affichage, le téléchargement ou la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon.

Programme de rétention

Les fichiers journaux font l'objet d'une rotation chaque jour et chaque fois que votre instance de base de données est redémarrée. Le calendrier de conservation des journaux Microsoft SQL Server sur Amazon est le suivantRDS.

Log type (Type de journal)	Programme de rétention
Journaux des erreurs	Au maximum, 30 journaux d'erreurs sont conservés. Amazon RDS peut supprimer les journaux d'erreurs datant de plus de 7 jours.
Journaux de l'agent	Au maximum, 10 journaux de l'agent sont conservés. Amazon RDS peut supprimer les journaux des agents datant de plus de 7 jours.
Fichiers de trace	Les fichiers de trace sont conservés selon la période de rétention des fichiers de trace de votre instance de base de données. La période de rétention par défaut des fichiers de trace est de 7 jours. Pour modifier la période de rétention des fichiers de trace pour votre instance de base de données, consultez Configuration de la période de rétention pour les fichiers de trace et de vidage.
Fichiers de vidage	Les fichiers de vidage sont conservés selon la période de rétention des fichiers de vidage de votre instance de base de données. La période de rétention par défaut des fichiers de vidage est de 7 jours. Pour modifier la période de rétention des fichiers de vidage pour votre instance de base de données, consultez Configuration de la période de rétention pour les fichiers de trace et de vidage.

Affichage du journal des erreurs du SQL serveur à l'aide de la procédure rds_read_error_log

Vous pouvez utiliser la procédure RDS stockée Amazon `rds_read_error_log` pour consulter les journaux d'erreurs et les journaux des agents. Pour de plus amples informations, veuillez consulter Affichage des journaux des erreurs et des agents.

Publication des journaux SQL du serveur sur Amazon CloudWatch Logs

Avec Amazon RDS for SQL Server, vous pouvez publier les erreurs et les événements du journal des agents directement sur Amazon CloudWatch Logs. Analysez les données du journal avec CloudWatch Logs, puis utilisez-les CloudWatch pour créer des alarmes et afficher les métriques.

Avec CloudWatch Logs, vous pouvez effectuer les opérations suivantes :

- Stocker des journaux dans un espace de stockage hautement durable pour lequel vous définissez la période de rétention.
- Chercher et filtrer les données de journaux.
- Partager des données de journaux entre les comptes.
- Exporter des journaux vers Amazon S3.
- Diffusez des données vers Amazon OpenSearch Service.
- Traiter des données de journaux en temps réel avec Amazon Kinesis Data Streams. Pour plus d'informations, consultez le guide du développeur de Working with Amazon CloudWatch Logs in the Amazon Managed Service for Apache Flink for SQL Applications.

Amazon RDS publie chaque journal de base de données SQL du serveur en tant que flux de base de données distinct dans le groupe de journaux. Par exemple, si vous publiez les journaux de l'agent et les journaux des erreurs, les données d'erreur sont stockées dans un flux de journaux d'erreurs du groupe de `/aws/rds/instance/`*`my_instance.node1`*`/error` journaux, et les données du journal de l'agent sont stockées dans le groupe de `/aws/rds/instance/`*`my_instance.node1`*`/agent` journaux.

Pour les instances de base de données Multi-AZ, Amazon RDS publie le journal de base de données sous la forme de deux flux distincts dans le groupe de journaux. Par exemple, si vous publiez les journaux d'erreurs, les données d'erreurs sont stockées dans les flux de journaux d'erreurs `/aws/rds/instance/`*`my_instance.node1`*`/error` et `/aws/rds/instance/`*`my_instance.node2`*`/error` respectivement. Les flux de journaux ne changent pas lors d'un basculement et le flux de

journaux d'erreurs de chaque nœud peut contenir les journaux d'erreurs issus de l'instance principale ou secondaire. Avec Multi-AZ, un flux de journal est automatiquement créé pour stocker les données d'événements telles que les basculements d'instances de base de données. `/aws/rds/instance/my_instance/rds-events`

> **Note**
>
> La publication des journaux SQL du serveur dans CloudWatch Logs n'est pas activée par défaut. La publication de fichiers de trace et de vidage n'est pas prise en charge. La publication des journaux SQL du serveur dans les CloudWatch journaux est prise en charge dans toutes les régions, à l'exception de l'Asie-Pacifique (Hong Kong).

Console

Pour publier les journaux SQL de base de données du serveur dans CloudWatch des journaux à partir du AWS Management Console

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).
4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.

   Vous pouvez choisir Journal de l'agent, Journal des erreursou les deux.
5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Pour publier les journaux SQL du serveur, vous pouvez utiliser la `modify-db-instance`commande avec les paramètres suivants :

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier les journaux SQL du serveur à l'aide des commandes suivantes :

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-to-point-in-time`

Example

L'exemple suivant crée une instance de base de données de SQL serveur avec la publication CloudWatch des journaux activée. La `--enable-cloudwatch-logs-exports` valeur est un JSON tableau de chaînes pouvant inclure `erroragent`, ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --enable-cloudwatch-logs-exports '["error","agent"]' \
    --db-instance-class db.m4.large \
    --engine sqlserver-se
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --enable-cloudwatch-logs-exports "[\"error\",\"agent\"]" ^
    --db-instance-class db.m4.large ^
    --engine sqlserver-se
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Example

L'exemple suivant modifie une instance de base de données de SQL serveur existante pour publier des fichiers CloudWatch journaux dans Logs. La `--cloudwatch-logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes qui peut inclure `error`, `agent` ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":["error","agent"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration "{\"EnableLogTypes\":[\"error\",\"agent\"]}"
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Example

L'exemple suivant modifie une instance de base de données de SQL serveur existante pour désactiver la publication des fichiers journaux de l'agent dans CloudWatch Logs. La `--cloudwatch-logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `DisableLogTypes` et sa valeur est un tableau de chaînes qui peut inclure `error`, `agent` ou les deux.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"DisableLogTypes":["agent"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration "{\"DisableLogTypes\":[\"agent\"]}"
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez éviter les guillemets («) dans le JSON code en les préfixant d'une barre oblique inverse (\).

Fichiers journaux de base de données MySQL

Vous pouvez surveiller les journaux MySQL directement via la console Amazon RDS, l'API Amazon RDS ou AWS CLI. AWS SDKs Vous pouvez également accéder aux journaux MySQL en dirigeant les journaux vers une table de base de données de la base de données principale et interroger cette table. Vous pouvez utiliser l'utilitaire mysqlbinlog pour télécharger un journal binaire.

Pour plus d'informations sur l'affichage, le téléchargement ou la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon.

Rubriques

- Présentation des journaux de base de données RDS for MySQL
- Publication de journaux MySQL sur Amazon CloudWatch Logs
- Envoi de la sortie du SQL journal My à des tables
- Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ
- Configuration de ma journalisation SQL binaire pour les clusters de bases de données multi-AZ
- Accès aux journaux binaires MySQL

Présentation des journaux de base de données RDS for MySQL

Vous pouvez surveiller les types de fichiers journaux RDS for MySQL suivants :

- Journal des erreurs
- Journal des requêtes lentes
- Journal général
- Journal d'audit
- Journal d'instance
- Journal des erreurs d'authentification de base de données IAM

Le journal des erreurs RDS for MySQL est généré par défaut. Vous pouvez générer le journal des requêtes lentes et le journal général en définissant les paramètres nécessaires dans votre groupe de paramètres de base de données.

Rubriques

- Journaux des erreurs RDS for MySQL
- Journal des requêtes lentes et journal général RDS for MySQL
- Journal d'audit MySQL
- Renouvellement et rétention des journaux pour RDS for MySQL
- Limites de taille des journaux de reprise

Journaux des erreurs RDS for MySQL

RDS for MySQL écrit les erreurs dans le fichier `mysql-error.log`. Le nom du fichier journal comporte l'heure à laquelle le fichier a été généré (au format UTC). Les fichiers journaux comportent également un horodatage permettant de déterminer le moment où les entrées du journal ont été écrites.

RDS for MySQL écrit dans le journal des erreurs uniquement au moment du démarrage, de l'arrêt et lorsqu'une erreur survient. Une instance de base de données peut fonctionner pendant des heures ou des jours sans qu'aucune nouvelle entrée soit écrite dans le journal des erreurs. Si aucune entrée récente ne figure, cela signifie que le serveur n'a pas rencontré d'erreur justifiant une entrée de journal.

Par défaut, les journaux des erreurs sont filtrés de sorte que seuls les événements inattendus tels que les erreurs soient affichés. Toutefois, les journaux des erreurs contiennent également des informations supplémentaires sur la base de données, comme la progression des requêtes, qui ne sont pas affichées. Par conséquent, même en l'absence d'erreurs réelles, la taille des journaux des erreurs peut augmenter en raison des activités en cours sur la base de données. Et même si vous pouvez voir une certaine taille en octets ou en kilo-octets pour les journaux d'erreurs dans le AWS Management Console, ils peuvent contenir 0 octet lorsque vous les téléchargez.

RDS for MySQL écrit `mysql-error.log` sur le disque toutes les 5 minutes. Il ajoute le contenu du journal à `mysql-error-running.log`.

RDS for MySQL assure la rotation du fichier `mysql-error-running.log` toutes les heures. Les journaux générés au cours des deux dernières semaines sont conservés.

> **Note**
>
> La période de conservation des journaux est différente pour Amazon RDS et Aurora.

Journal des requêtes lentes et journal général RDS for MySQL

Vous pouvez écrire le journal des requêtes lentes et le journal général RDS for MySQL dans un fichier ou dans une table de base de données. Pour ce faire, définissez les paramètres de votre groupe de paramètres de base de données. Pour plus d'informations sur la création et la modification d'un groupe de paramètres DB, consultez Groupes de paramètres pour (Amazon RDS). Vous devez définir ces paramètres avant de pouvoir consulter le journal des requêtes lentes ou le journal général dans la console Amazon RDS ou à l'aide de l'API Amazon RDS, de la CLI Amazon RDS ou. AWS SDKs

Vous pouvez contrôler la journalisation RDS for MySQL à l'aide des paramètres de cette liste :

- `slow_query_log` : Pour créer le journal des requêtes lentes, définir sur 1. La valeur par défaut est 0.
- `general_log` : Pour créer le journal général, définir sur 1. La valeur par défaut est 0.
- `long_query_time` : Pour empêcher l'enregistrement des requêtes rapides dans le journal des requêtes lentes, indiquez la valeur de la durée d'exécution de requête la plus courte devant être journalisée, en secondes. La valeur par défaut est de 10 secondes et la valeur minimum est 0. Si log_output = FILE, vous pouvez indiquer une valeur à virgule flottante avec une résolution en microseconde. Si log_output = TABLE, vous devez indiquer un nombre entier avec une résolution en seconde. Seules les requêtes dont la durée d'exécution dépasse la valeur `long_query_time` sont journalisées. Par exemple, si vous définissez `long_query_time` sur 0,1, les requêtes s'exécutant pendant moins de 100 millisecondes ne sont pas enregistrées.
- `log_queries_not_using_indexes` : Pour enregistrer toutes les requêtes n'utilisant pas d'index dans le journal des requêtes lentes, définir sur 1. Les requêtes n'utilisant pas d'index sont journalisées même si la durée de leur exécution est inférieure à la valeur du paramètre `long_query_time`. La valeur par défaut est 0.
- `log_output` *`option`* : Vous pouvez spécifier l'une des options suivantes pour le paramètre `log_output`.
  - TABLEAU (par défaut) – Écrit les requêtes générales dans le tableau `mysql.general_log` et les requêtes lentes dans le tableau `mysql.slow_log`.
  - FICHIER – Écrit les fichiers des requêtes générales et lentes dans le fichier système.
  - AUCUNE– Désactive la journalisation.

Pour que les données de requête lentes apparaissent dans Amazon CloudWatch Logs, les conditions suivantes doivent être remplies :

- CloudWatch Les journaux doivent être configurés pour inclure les journaux des requêtes lentes.
- `slow_query_log`doit être activé.
- `log_output` doit être défini sur `FILE`.
- La durée de la requête doit être plus longue que celle configurée pour`long_query_time`.

Pour plus d'informations sur le journal des requêtes lentes et le journal général, accédez aux rubriques suivantes dans la documentation MySQL :

- Journal des requêtes lentes
- Journal des requêtes générales

Journal d'audit MySQL

Pour accéder au journal d'audit, l'instance de base de données doit utiliser un groupe d'options personnalisé avec l'option `MARIADB_AUDIT_PLUGIN`. Pour de plus amples informations, veuillez consulter Support du plugin d'audit MariaDB pour My SQL.

Renouvellement et rétention des journaux pour RDS for MySQL

Lorsque la journalisation est activée, Amazon RDS effectue une rotation des journaux des tables ou supprime les fichiers journaux à intervalles réguliers. Cette précaution permet de limiter la possibilité qu'un fichier journal volumineux ne bloque l'utilisation de la base de données ou n'affecte les performances. RDS for MySQL gère la rotation et la suppression des journaux comme suit :

- Les tailles du journal des requêtes lentes, du journal des erreurs et du journal général MySQL sont limitées à 2 %maximum de l'espace de stockage alloué à une instance de base de données. Pour maintenir ce seuil, les journaux sont automatiquement renouvelés toutes les heures. MySQL supprime les fichiers journaux datant de plus de deux semaines. Si la taille de l'ensemble des fichiers journaux après la suppression dépasse le seuil, les fichiers journaux les plus anciens sont supprimés jusqu'à ce que la taille des fichiers journaux ne soit plus supérieure au seuil.
- Lorsque la journalisation `FILE` est activée, les fichiers journaux sont examinés toutes les heures et ceux datant de plus de deux semaines sont supprimés. Dans certains cas, la taille des fichiers journaux combinés restant après la suppression peut dépasser le seuil de 2 % de l'espace alloué à une instance de base de données. Le cas échéant, les fichiers journaux les plus anciens sont supprimés jusqu'à ce que la taille des fichiers journaux ne soit plus supérieure au seuil.
- Lorsque la journalisation de `TABLE` est activée, les tables des journaux font dans certains cas l'objet d'une rotation toutes les 24 heures. Cette rotation se produit si l'espace utilisé par

les journaux des tables est supérieur à 20 % de l'espace de stockage alloué. Cela se produit également si la taille de tous les journaux combinés est supérieure à 10 Go. Si l'espace utilisé pour une instance de base de données est supérieur à 90 % de l'espace de stockage alloué à l'instance de base de données, alors les seuils correspondant à la rotation des journaux est réduite. La rotation des journaux des tables se produit ensuite si l'espace utilisé par les journaux des tables est supérieur à 10 % de l'espace de stockage alloué. Elle se produit également si la taille de tous les journaux combinés est supérieure à 5 Go. Vous pouvez vous abonner à la catégorie d'`low storage`événements pour être averti lorsque les tables de journalisation sont pivotées afin de libérer de l'espace. Pour de plus amples informations, veuillez consulter Utilisation des notifications d'RDSévénements Amazon.

Lors de la rotation des tables de journaux, la table de journal actuelle est d'abord copiée vers une table de journal de sauvegarde. Les entrées de la table de journal actuelle sont ensuite supprimées. Si la table de journal de sauvegarde existe déjà, elle est supprimée avant que la table de journal actuelle ne soit copiée dans la sauvegarde. Si besoin, vous pouvez interroger la table de journal de sauvegarde. La table de journal de sauvegarde de la table `mysql.general_log` est nommée `mysql.general_log_backup`. La table de journal de sauvegarde de la table `mysql.slow_log` est nommée `mysql.slow_log_backup`.

Vous pouvez effectuer une rotation de la table `mysql.general_log` en appelant la procédure `mysql.rds_rotate_general_log`. Vous pouvez effectuer une rotation de la table `mysql.slow_log` en appelant la procédure `mysql.rds_rotate_slow_log`.

La rotation des journaux des tables est effectuée pendant la mise à niveau de la version d'une base de données.

Pour utiliser les journaux de la console Amazon RDS, de l'API Amazon RDS, de l'interface de ligne de commande Amazon RDS, AWS SDKs ou définissez `log_output` le paramètre sur FILE. A l'instar du journal des erreurs MySQL, ces fichiers journaux font l'objet d'une rotation horaire. Les fichiers journaux qui ont été générés au cours des deux dernières semaines sont conservés. Veuillez noter que la période de rétention est différente pour Amazon RDS et pour Aurora.

Limites de taille des journaux de reprise

Pour les versions 8.0.32 et antérieures de RDS for MySQL, la valeur par défaut de ce paramètre est de 256 Mo. Ce montant est obtenu en multipliant la valeur par défaut du `innodb_log_file_size` paramètre (128 Mo) par la valeur par défaut du `innodb_log_files_in_group` paramètre (2).

Pour plus d'informations, consultez Bonnes pratiques de configuration des paramètres pour Amazon RDS for MySQL, partie 1 : Paramètres liés aux performances.

Pour RDS for MySQL version 8.0.33 et versions mineures supérieures, Amazon RDS utilise le `innodb_redo_log_capacity` paramètre au lieu du paramètre. `innodb_log_file_size` La valeur par défaut du `innodb_redo_log_capacity` paramètre sur Amazon RDS est de 2 Go. Pour plus d'informations, consultez Changements dans MySQL 8.0.30 dans la documentation MySQL.

À partir de MySQL 8.4, Amazon RDS active le `innodb_dedicated_server` paramètre par défaut. Avec le `innodb_dedicated_server` paramètre, le moteur de base de données calcule les `innodb_redo_log_capacity` paramètres `innodb_buffer_pool_size` et. Pour de plus amples informations, veuillez consulter Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL.

Publication de journaux MySQL sur Amazon CloudWatch Logs

Vous pouvez configurer votre instance de base de données MySQL pour publier les données de journal dans un groupe de CloudWatch journaux dans Amazon Logs. Avec CloudWatch Logs, vous pouvez effectuer une analyse en temps réel des données du journal, puis les utiliser CloudWatch pour créer des alarmes et afficher des métriques. Vous pouvez utiliser CloudWatch les journaux pour stocker vos enregistrements de journal dans un espace de stockage hautement durable.

Amazon RDS publie chaque journal de base de données MySQL sous la forme d'un flux de base de données distinct dans le groupe de journaux. Par exemple, si vous configurez la fonction d'exportation de sorte à inclure le journal de requêtes lentes, les données de requêtes lentes sont stockées dans un flux de journal de requêtes lentes dans le groupe de journaux `/aws/rds/instance/`*`my_instance`*`/slowquery`.

Le journal d'erreurs est activé par défaut. Le tableau ci-dessous récapitule les conditions requises pour les autres journaux MySQL.

Log	Exigence
Journal d'audit	L'instance de base de données doit disposer d'un groupe d'options personnalisées avec l'option `MARIADB_AUDIT_PLUGIN` .
Journal général	L'instance de base de données doit disposer d'un groupe de paramètres personnalisés

Log	Exigence
	avec le paramètre `general_log = 1` pour autoriser la journalisation générale.
Journal des requêtes lentes	L'instance de base de données doit disposer d'un groupe de paramètres personnalisés avec le paramètre `slow_query_log = 1` pour autoriser la journalisation de requête lente.
Journal des erreurs d'authentification de base de données IAM	Vous devez activer le type de journal `iam-db-auth-error` pour une instance de base de données en créant ou en modifiant une instance de base de données.
Sortie de journal	L'instance de base de données doit utiliser un groupe de paramètres personnalisé avec le paramètre défini `log_output = FILE` pour écrire des journaux dans le système de fichiers et les publier dans CloudWatch Logs.

console

Pour publier les journaux MySQL dans CloudWatch Logs à l'aide de la console

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).
4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.
5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Vous pouvez publier des journaux MySQL avec le AWS CLI. Vous pouvez appeler la commande `modify-db-instance` avec les paramètres suivants :

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier des journaux MySQL en appelant les AWS CLI commandes suivantes :

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-from-s3`
- `restore-db-instance-to-point-in-time`

Exécutez l'une de ces AWS CLI commandes avec les options suivantes :

- `--db-instance-identifier`
- `--enable-cloudwatch-logs-exports`
- `--db-instance-class`
- `--engine`

D'autres options peuvent être nécessaires en fonction de la AWS CLI commande que vous exécutez.

Example

L'exemple suivant modifie une instance de base de données MySQL existante pour publier des fichiers CloudWatch journaux dans Logs. La valeur `--cloudwatch-logs-export-configuration` n'est pas un objet JSON. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes avec une combinaison quelconque de `audit`, `error`, `general` et `slowquery`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["audit","error","general","slowquery"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["audit","error","general","slowquery"]}'
```

Example

L'exemple suivant crée une instance de base de données MySQL et publie des fichiers CloudWatch journaux dans Logs. La valeur `--enable-cloudwatch-logs-exports` est un tableau de chaînes JSON. Les chaînes peuvent être une combinaison de `audit`, `error`, `general` et `slowquery`.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --enable-cloudwatch-logs-exports '["audit","error","general","slowquery"]' \
    --db-instance-class db.m4.large \
    --engine MySQL
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --enable-cloudwatch-logs-exports '["audit","error","general","slowquery"]' ^
    --db-instance-class db.m4.large ^
    --engine MySQL
```

API RDS

Vous pouvez publier des journaux MySQL avec l'API RDS. Vous pouvez appeler l'action `ModifyDBInstance` avec les paramètres suivants :

- `DBInstanceIdentifier`
- `CloudwatchLogsExportConfiguration`

> **Note**
>
> Une modification apportée au paramètre `CloudwatchLogsExportConfiguration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, le paramètre `ApplyImmediately` est sans effet.

Vous pouvez également publier des journaux MySQL en appelant les opérations d'API RDS suivantes :

- `CreateDBInstance`
- `RestoreDBInstanceFromDBSnapshot`
- `RestoreDBInstanceFromS3`
- `RestoreDBInstanceToPointInTime`

Exécutez l'une de ces opérations d'API RDS avec les paramètres suivants :

- `DBInstanceIdentifier`
- `EnableCloudwatchLogsExports`
- `Engine`
- `DBInstanceClass`

D'autres paramètres peuvent être nécessaires en fonction de la AWS CLI commande que vous exécutez.

Envoi de la sortie du SQL journal My à des tables

Vous pouvez diriger le journal des requêtes lentes et le journal général vers des tables sur l'instance de base de données en créant un groupe de paramètres DB et en définissant le paramètre du serveur `log_output` sur `TABLE`. Les requêtes générales sont ensuite enregistrées dans la table `mysql.general_log` et les requêtes lentes dans la table `mysql.slow_log`. Vous pouvez interroger les tables pour accéder aux informations des journaux. L'activation de cette journalisation augmente le volume de données écrites dans la base de données, ce qui peut dégrader les performances.

Par défaut, le journal général et le journal des requêtes lentes sont désactivés. Afin d'activer la journalisation dans les tables, vous devez également définir les paramètres `general_log` et `slow_query_log` sur `1`.

Les tables de journaux continuent de grossir jusqu'à ce que les activités de journalisation correspondantes soient désactivées en redéfinissant le paramètre approprié sur `0`. Avec le temps, une grande quantité de données s'accumule et risque d'utiliser une part considérable de l'espace de stockage alloué. Amazon RDS ne vous permet pas de tronquer les tables de journaux, mais vous pouvez déplacer leur contenu. Lorsque vous procédez à la rotation d'une table, son contenu est enregistré dans une table de sauvegarde et une nouvelle table de journal vide est créée. Vous pouvez effectuer une rotation manuelle des tables de journaux avec les procédures de ligne de commande suivantes, dans lesquelles l'invite de commande est indiquée par `PROMPT>`:

```
PROMPT> CALL mysql.rds_rotate_slow_log;
PROMPT> CALL mysql.rds_rotate_general_log;
```

Pour supprimer totalement les anciennes données et récupérer l'espace de disque, appelez deux fois à la suite la procédure appropriée.

Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ

Le journal binaire est un jeu de fichiers journaux contenant des informations sur les modifications de données apportées à une instance de serveur MySQL. Le journal binaire contient des informations telles que les suivantes :

- Événements décrivant les modifications apportées à la base de données telles que la création de tables ou les modifications de lignes
- Informations sur la durée de chaque instruction qui a mis à jour les données

- Événements pour des instructions pouvant mettre à jour des données mais ne l'ayant pas fait

Le journal binaire enregistre les instructions envoyées pendant la réplication. Il est également requis pour certaines opérations de récupération. Pour plus d'informations, consultez The Binary Log dans la documentation MySQL.

La fonction de sauvegarde automatisée détermine si la journalisation binaire est activée ou désactivée pour MySQL. Vous avez les options suivantes :

Activer la journalisation binaire

Définissez la période de rétention des sauvegardes sur une valeur positive différente de zéro.

Désactiver la journalisation binaire

Définissez la période de rétention des sauvegardes sur zéro.

Pour de plus amples informations, veuillez consulter Activation des sauvegardes automatiques.

MySQL on Amazon RDS prend en charge les formats de journalisation binaire basés sur les lignes, basés sur les instructions et mixtes. Nous recommandons le format mixte, sauf si vous avez besoin d'un format binlog spécifique. Pour plus de détails sur les différents formats de journaux binaires MySQL, consultez la section Formats de journalisation binaires dans la documentation MySQL.

Si vous prévoyez d'utiliser la réplication, le format de journalisation binaire est important car il détermine le dossier de modifications de données qui est enregistré dans la source et envoyés aux cibles de réplication. Pour plus d'informations sur les avantages et les désavantages des différents formats de journalisation binaire pour la réplication, consultez la section Advantages and Disadvantages of Statement-Based and Row-Based Replication de la documentation MySQL.

> **Important**
>
> Avec MySQL 8.0.34, MySQL a rendu le paramètre obsolète. `binlog_format` Dans les versions ultérieures de MySQL, MySQL prévoit de supprimer le paramètre et de ne prendre en charge que la réplication basée sur les lignes. Par conséquent, nous recommandons d'utiliser la journalisation basée sur les lignes pour les nouvelles configurations de réplication MySQL. Pour plus d'informations, consultez binlog_format dans la documentation MySQL. Les versions 8.0 et 8.4 de MySQL acceptent le paramètre`binlog_format`. Lors de l'utilisation de ce paramètre, MySQL émet un avertissement de dépréciation. Dans une future version majeure, MySQL supprimera le paramètre`binlog_format`.

La réplication basée sur les instructions peut provoquer des incohérences entre l'instance de base de données source et un réplica en lecture. Pour de plus amples informations, veuillez consulter Determination of Safe and Unsafe Statements in Binary Logging dans la documentation MySQL.
L'activation de la journalisation binaire augmente le nombre d'opérations d'I/O d'écriture disque sur l'instance de bases de données. Vous pouvez surveiller l'utilisation des IOPS à l'aide de cette `WriteIOPS` CloudWatch métrique.

Pour définir le format de journalisation binaire MySQL

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez le groupe de paramètres du de base de données, associé au d'instances de base de données, que vous souhaitez modifier.

   Vous ne pouvez pas modifier un groupe de paramètres par défaut. Si l'instance de base de données utilise un groupe de paramètres par défaut, créez un nouveau groupe et associez-le à l'instance.

   Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).
4. Dans Actions, sélectionnez Modifier.
5. Définissez le paramètre `binlog_format` au format de journalisation binaire de votre choix (`ROW`, `STATEMENT` ou `MIXED`).

   Vous pouvez désactiver la journalisation binaire en définissant la période de conservation des sauvegardes d'une instance de base de données sur zéro, mais cela désactive les sauvegardes automatiques quotidiennes. La désactivation des sauvegardes automatiques désactive ou désactive la variable de `log_bin` session. Cela désactive la journalisation binaire sur l'instance de base de données RDS pour MySQL, qui à son tour réinitialise la variable de `binlog_format` session à la valeur par défaut de `ROW` dans la base de données. Nous vous recommandons de ne pas désactiver les sauvegardes. Pour plus d'informations sur le paramètre Période de rétention des sauvegardes, consultez Paramètres des instances de base de données.
6. Choisissez Save changes (Enregistrer les modifications)pour enregistrer les mises à jour apportées au groupe de paramètres de base de données.

Comme le `binlog_format` paramètre est dynamique dans RDS pour MySQL, il n'est pas nécessaire de redémarrer l'instance de base de données pour que les modifications s'appliquent. (Notez que dans Aurora MySQL, ce paramètre est statique. Pour plus d'informations, consultez Configuration de la journalisation binaire Aurora MySQL.)

> **Important**
>
> La modification d'un groupe de paramètres de base de données affecte toutes les instances de base de données qui utilisent ce dernier. Si vous souhaitez spécifier différents formats de journalisation binaire pour différentes instances de base de données MySQL dans une AWS région, les instances de base de données doivent utiliser différents groupes de paramètres de base de données. Ces groupes de paramètres identifient différents formats de journalisation. Affectez le groupe de paramètres de base de données approprié à chaque instance de base de données.

Configuration de ma journalisation SQL binaire pour les clusters de bases de données multi-AZ

La journalisation binaire dans les clusters de base de données Amazon RDS for My SQL Multi-AZ enregistre toutes les modifications de base de données afin de prendre en charge la réplication, point-in-time la restauration et l'audit. Dans les clusters de bases de données multi-AZ, les journaux binaires synchronisent les nœuds secondaires avec le nœud principal, garantissant ainsi la cohérence des données entre les zones de disponibilité et permettant des basculements fluides.

Pour optimiser la journalisation binaire, Amazon RDS prend en charge la compression des transactions des journaux binaires, ce qui réduit les exigences de stockage pour les journaux binaires et améliore l'efficacité de la réplication.

Rubriques

- Compression des transactions du journal binaire pour les clusters de bases de données multi-AZ
- Configuration de la compression des transactions du journal binaire pour les clusters de bases de données multi-AZ

Compression des transactions du journal binaire pour les clusters de bases de données multi-AZ

La compression des transactions du journal binaire utilise l'algorithme zstd pour réduire la taille des données de transaction stockées dans les journaux binaires. Lorsqu'il est activé, le moteur My

SQL database compresse les charges utiles des transactions en un seul événement, minimisant ainsi les frais d'E/S et de stockage. Cette fonctionnalité améliore les performances des bases de données, réduit la taille des journaux binaires et optimise l'utilisation des ressources pour la gestion et la réplication des journaux dans les clusters de bases de données multi-AZ.

Amazon RDS fournit une compression des transactions du journal binaire RDS pour les clusters de base de données My SQL Multi-AZ via les paramètres suivants :

- `binlog_transaction_compression`— Lorsque cette option est activée (1), le moteur de base de données compresse les charges utiles des transactions et les écrit dans le journal binaire sous la forme d'un événement unique. Cela réduit l'utilisation du stockage et les frais d'E/S. Le paramètre est désactivé par défaut.
- `binlog_transaction_compression_level_zstd`— Configure le niveau de compression zstd pour les transactions du journal binaire. Des valeurs plus élevées augmentent le taux de compression, ce qui réduit encore les besoins en stockage tout en augmentant CPU l'utilisation de la mémoire pour la compression. La valeur par défaut est 3, avec une plage comprise entre 1 et 22.

Ces paramètres vous permettent d'affiner la compression des journaux binaires en fonction des caractéristiques de la charge de travail et de la disponibilité des ressources. Pour plus d'informations, consultez la section Compression des transactions du journal binaire dans la section Ma SQL documentation.

La compression des transactions dans les journaux binaires présente les principaux avantages suivants :

- La compression réduit la taille des journaux binaires, en particulier pour les charges de travail comportant des transactions importantes ou des volumes d'écriture élevés.
- Les journaux binaires de plus petite taille réduisent la surcharge du réseau et des E/S, améliorant ainsi les performances de réplication.
- Le `binlog_transaction_compression_level_zstd` paramètre permet de contrôler le compromis entre le taux de compression et la consommation de ressources.

Configuration de la compression des transactions du journal binaire pour les clusters de bases de données multi-AZ

Pour configurer la compression des transactions du journal binaire pour un cluster de base de données RDS for My SQL Multi-AZ, modifiez les paramètres de cluster appropriés en fonction de vos exigences de charge de travail.

Console

Pour activer la compression des transactions du journal binaire

1. Modifiez le groupe de paramètres du cluster de base de données pour définir le `binlog_transaction_compression` paramètre sur`1`.
2. (Facultatif) Ajustez la valeur du `binlog_transaction_compression_level_zstd` paramètre en fonction de vos exigences en matière de charge de travail et de la disponibilité des ressources.

Pour de plus amples informations, veuillez consulter the section called “Modification de paramètres dans un groupe de paramètres de cluster de base de données”.

AWS CLI

Pour configurer la compression des transactions du journal binaire à l'aide de AWS CLI, utilisez la commande modify-db-cluster-parameter-group.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-cluster-parameter-group \
  --db-cluster-parameter-group-name your-cluster-parameter-group \
  --parameters
 "ParameterName=binlog_transaction_compression,ParameterValue=1,ApplyMethod=pending-
reboot"
```

Dans Windows:

```
aws rds modify-db-cluster-parameter-group ^
  --db-cluster-parameter-group-name your-cluster-parameter-group ^
```

```
  --parameters
 "ParameterName=binlog_transaction_compression,ParameterValue=1,ApplyMethod=pending-
reboot"
```

RDS API

Pour configurer la compression des transactions du journal binaire à l'aide d'Amazon RDSAPI, utilisez l'`ModifyDBClusterParameterGroup`opération.

Accès aux journaux binaires MySQL

Vous pouvez utiliser l'utilitaire mysqlbinlog pour télécharger ou diffuser des journaux binaires à partir des instances de base de données RDS for MySQL. Le journal binaire est téléchargé dans votre ordinateur local et vous pouvez effectuer des actions comme relire le journal à l'aide de l'utilitaire mysql. Pour plus d'informations sur l'utilisation de l'utilitaire mysqlbinlog, consultez Using mysqlbinlog to back up binary log files (Utilisation de mysqlbinlog pour sauvegarder les fichiers journaux binaires) dans la documentation MySQL.

Pour exécuter à nouveau l'utilitaire mysqlbinlog sur une instance Amazon RDS, utilisez les options suivantes :

- `--read-from-remote-server` : obligatoire.
- `--host` : le nom DNS du point de terminaison de l'instance.
- `--port` : le port utilisé par l'instance.
- `--user` : un utilisateur MySQL ayant l'autorisation `REPLICATION SLAVE`.
- `--password` : le mot de passe de l'utilisateur MySQL ou omettez la valeur de mot de passe pour que l'utilitaire vous invite à saisir un mot de passe.
- `--raw` : téléchargez le fichier au format binaire.
- `--result-file` : le fichier local qui recevra la sortie brute.
- `--stop-never` : diffusez les fichiers journaux binaires.
- `--verbose` : lorsque vous utilisez le format binlog `ROW`, incluez cette option pour afficher les événements de ligne sous forme d'instructions pseudo-SQL. Pour plus d'informations sur l'option `--verbose`, consultez mysqlbinlog row event display (Affichage d'événements de ligne mysqlbinlog) dans la documentation MySQL.
- Spécifiez les noms pour un ou plusieurs fichiers journaux binaires. Pour obtenir la liste des journaux disponibles, utilisez la commande SQL `SHOW BINARY LOGS`.

Pour plus d'informations sur les options mysqlbinlog, consultez mysqlbinlog — Utility for processing binary log files (mysqlbinlog : utilitaire de traitement des fichiers journaux binaires) dans la documentation MySQL.

Les exemples suivants montrent comment utiliser l'utilitaire mysqlbinlog.

Dans Linux, macOS, ou Unix:

```
mysqlbinlog \
    --read-from-remote-server \
    --host=MySQLInstance1.cg034hpkmmjt.region.rds.amazonaws.com \
    --port=3306  \
    --user ReplUser \
    --password \
    --raw \
    --verbose \
    --result-file=/tmp/ \
    binlog.00098
```

Dans Windows:

```
mysqlbinlog ^
    --read-from-remote-server ^
    --host=MySQLInstance1.cg034hpkmmjt.region.rds.amazonaws.com ^
    --port=3306  ^
    --user ReplUser ^
    --password ^
    --raw ^
    --verbose ^
    --result-file=/tmp/ ^
    binlog.00098
```

Les journaux binaires doivent rester disponibles sur l'instance de base de données pour que l'utilitaire mysqlbinlog puisse y accéder. Pour garantir leur disponibilité, utilisez la procédure mysql.rds_set_configuration stockée et spécifiez une période suffisante pour que vous puissiez télécharger les journaux. Si cette configuration n'est pas définie, Amazon RDS purge les journaux binaires dès que possible, ce qui entraîne des lacunes dans les journaux binaires récupérés par l'utilitaire mysqlbinlog.

L'exemple suivant définit la période de conservation sur 1 jour.

```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

Pour afficher les paramètres actuels, utilisez la procédure stockée mysql.rds_show_configuration.

```
call mysql.rds_show_configuration;
```

Fichiers journaux RDS de base de données Amazon pour Oracle

Vous pouvez accéder aux journaux d'alertes, aux fichiers d'audit et aux fichiers de trace Oracle à l'aide de la RDS console Amazon ouAPI. Pour plus d'informations sur l'affichage, le téléchargement ou la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon.

Les fichiers d'audit Oracle fournis sont les fichiers d'audit Oracle standard. Amazon RDS prend en charge la fonctionnalité d'audit fin d'Oracle (FGA). Cependant, l'accès au journal ne permet pas d'accéder aux FGA événements stockés dans la `SYS.FGA_LOG$` table et accessibles via la `DBA_FGA_AUDIT_TRAIL` vue.

L'`DescribeDBLogFiles`APIopération qui répertorie les fichiers journaux Oracle disponibles pour une instance de base de données ignore le `MaxRecords` paramètre et renvoie jusqu'à 1 000 enregistrements. L'appel est renvoyé `LastWritten` sous forme de POSIX date en millisecondes.

Rubriques

- Programme de rétention
- Utilisation des fichiers de trace Oracle
- Publication de journaux Oracle sur Amazon CloudWatch Logs
- Accès aux journaux d'alertes et aux journaux des auditeurs

Programme de rétention

Le moteur de base de données Oracle peut procéder à la rotation des fichiers journaux s'ils deviennent très volumineux. Pour conserver les fichiers d'audit ou de trace, vous devez les télécharger. Si vous stockez les fichiers localement, vous réduisez vos coûts RDS de stockage sur Amazon et vous libérez de l'espace pour vos données.

Le tableau suivant indique le calendrier de conservation des journaux d'alertes, des fichiers d'audit et des fichiers de trace Oracle sur AmazonRDS.

Log type (Type de journal)	Programme de rétention
Journaux des alertes	Le journal des alertes textuelles fait l'objet d'une rotation quotidienne avec une rétention de 30 jours gérée par Amazon. RDS Le journal des XML

Log type (Type de journal)	Programme de rétention
	alertes est conservé pendant au moins sept jours. Vous pouvez accéder à ce journal grâce à la vue `ALERTLOG`.
Fichiers d'audit	La période de rétention par défaut des fichiers d'audit est de 7 jours. Amazon RDS peut supprimer des fichiers d'audit datant de plus de sept jours.
Fichiers de trace	La période de rétention par défaut des fichiers de trace est de 7 jours. Amazon RDS peut supprimer les fichiers de trace datant de plus de sept jours.
Journaux de l'écouteur	La période de rétention par défaut pour les journaux d'écouteur est de 7 jours. Amazon RDS peut supprimer les journaux d'écoute datant de plus de sept jours.

> **Note**
>
> Les fichiers d'audit et les fichiers de trace partagent la même configuration de rétention.

Utilisation des fichiers de trace Oracle

Vous trouverez ci-dessous des descriptions des RDS procédures Amazon relatives à la création, à l'actualisation, à l'accès et à la suppression de fichiers de trace.

Rubriques

- Liste de fichiers
- Génération de fichiers de trace et suivi d'une session
- Récupération de fichiers de trace
- Purge des fichiers de trace

Liste de fichiers

Vous pouvez utiliser l'une ou l'autre de deux procédures permettent d'accéder à tous les fichiers dans le chemin `background_dump_dest`. La première actualise la liste de l'ensemble des fichiers actuellement dans `background_dump_dest`.

```
EXEC rdsadmin.manage_tracefiles.refresh_tracefile_listing;
```

Une fois la liste actualisée, interrogez la vue suivante pour accéder aux résultats.

```
SELECT * FROM rdsadmin.tracefile_listing;
```

Vous pouvez également utiliser `FROM table` pour diffuser des données non relationnelles dans un format de table afin de répertorier le contenu de l'annuaire de bases de données.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir('BDUMP'));
```

La requête suivante affiche le texte d'un fichier journal.

```
SELECT text FROM
 TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP','alert_dbname.log.date'));
```

Sur une réplique en lecture, obtenez le nom du BDUMP répertoire en `V$DATABASE.DB_UNIQUE_NAME` interrogeant. Si le nom unique est`DATABASE_B`, le BDUMP répertoire l'est`BDUMP_B`. L'exemple suivant demande le BDUMP nom d'une réplique, puis utilise ce nom pour interroger le contenu de`alert_DATABASE.log.2020-06-23`.

```
SELECT 'BDUMP' || (SELECT regexp_replace(DB_UNIQUE_NAME,'.*(_[A-Z])', '\1') FROM V
$DATABASE) AS BDUMP_VARIABLE FROM DUAL;

BDUMP_VARIABLE
--------------
BDUMP_B

SELECT TEXT FROM
 table(rdsadmin.rds_file_util.read_text_file('BDUMP_B','alert_DATABASE.log.2020-06-23'));
```

Génération de fichiers de trace et suivi d'une session

Comme il n'existe aucune restriction`ALTER SESSION`, de nombreuses méthodes standard de génération de fichiers de trace dans Oracle restent disponibles pour une RDS instance de base de données Amazon. Les procédures suivantes sont fournies pour les fichiers de trace pour lesquels l'accès doit être élargi.

Méthode Oracle	RDSMéthode Amazon
`oradebug hanganalyze 3`	`EXEC rdsadmin.manage_tracefiles.hanganalyze;`
`oradebug dump systemstate 266`	`EXEC rdsadmin.manage_tracefiles.dump_systemstate;`

Vous pouvez utiliser de nombreuses méthodes standard pour suivre les sessions individuelles connectées à une instance de base de données Oracle sur AmazonRDS. Pour activer le suivi d'une session, vous pouvez exécuter des sous-programmes dans des SQL packages PL/ fournis par Oracle, tels que `DBMS_SESSION` et. `DBMS_MONITOR` Pour plus d'informations, consultez Activation du suivi d'une session dans la documentation d'Oracle.

Récupération de fichiers de trace

Vous pouvez récupérer n'importe quel fichier de trace en `background_dump_dest` utilisant une SQL requête standard sur une table RDS externe gérée par Amazon. Pour utiliser cette méthode, vous devez exécuter la procédure définissant l'emplacement de cette table sur le fichier de trace spécifique.

Par exemple, vous pouvez utiliser la vue `rdsadmin.tracefile_listing` indiquée ci-dessus pour répertorier tous les fichiers de trace sur le système. Vous pouvez ensuite définir la vue `tracefile_table` afin qu'elle pointe vers le fichier de trace souhaité à l'aide de la procédure suivante.

```
EXEC
 rdsadmin.manage_tracefiles.set_tracefile_table_location('CUST01_ora_3260_SYSTEMSTATE.trc');
```

L'exemple suivant créé une table externe selon le schéma actuel. L'emplacement est défini sur le fichier fourni. Vous pouvez récupérer le contenu dans un fichier local à l'aide d'une SQL requête.

```
SPOOL /tmp/tracefile.txt
SELECT * FROM tracefile_table;
SPOOL OFF;
```

Purge des fichiers de trace

Les fichiers de trace peuvent s'accumuler et consommer de l'espace sur le disque. Amazon RDS purge les fichiers de trace par défaut et les fichiers journaux datant de plus de sept jours. Vous pouvez afficher et définir la période de rétention des fichiers à l'aide de la procédure `show_configuration`. Vous devez exécuter la commande `SET SERVEROUTPUT ON` afin de pouvoir afficher les résultats de la configuration.

L'exemple suivant affiche la période de rétention des fichiers de trace actuelle, puis définit une nouvelle période.

```
# Show the current tracefile retention
SQL> EXEC rdsadmin.rdsadmin_util.show_configuration;
NAME:tracefile retention
VALUE:10080
DESCRIPTION:tracefile expiration specifies the duration in minutes before tracefiles in
 bdump are automatically deleted.

# Set the tracefile retention to 24 hours:
SQL> EXEC rdsadmin.rdsadmin_util.set_configuration('tracefile retention',1440);
SQL> commit;

#show the new tracefile retention
SQL> EXEC rdsadmin.rdsadmin_util.show_configuration;
NAME:tracefile retention
VALUE:1440
DESCRIPTION:tracefile expiration specifies the duration in minutes before tracefiles in
 bdump are automatically deleted.
```

En complément du processus de purge périodique, vous pouvez supprimer manuellement des fichiers de `background_dump_dest`. L'exemple suivant montre comment purger tous les fichiers datant de plus de cinq minutes.

```
EXEC rdsadmin.manage_tracefiles.purge_tracefiles(5);
```

Vous pouvez également purger tous les fichiers correspondant à un modèle spécifique (dans ce cas, n'incluez pas l'extension de fichier, par exemple .trc). L'exemple suivant montre comment purger tous les fichiers commençant par `SCHPOC1_ora_5935`.

```
EXEC rdsadmin.manage_tracefiles.purge_tracefiles('SCHPOC1_ora_5935');
```

Publication de journaux Oracle sur Amazon CloudWatch Logs

Vous pouvez configurer votre instance de base de données Oracle RDS pour publier les données de journal dans un groupe de CloudWatch journaux dans Amazon Logs. Avec CloudWatch Logs, vous pouvez analyser les données du journal et les utiliser CloudWatch pour créer des alarmes et afficher des métriques. Vous pouvez utiliser CloudWatch les journaux pour stocker vos enregistrements de journal dans un espace de stockage hautement durable.

Amazon RDS publie chaque journal de base de données Oracle en tant que flux de base de données distinct dans le groupe de journaux. Par exemple, si vous configurez la fonction d'exportation de sorte à inclure le journal d'audit, les données d'audit sont stockées dans un flux de journal d'audit dans le groupe de journaux `/aws/rds/instance/my_instance/audit`. Le tableau suivant récapitule les conditions requises RDS pour qu'Oracle publie des CloudWatch journaux sur Amazon Logs.

Nom du journal	Exigence	Par défaut
Journal des alertes	Aucune. Vous ne pouvez pas désactiver ce journal.	Activées
Journal de suivi	Définissez le `trace_enabled` paramètre sur TRUE ou laissez-le sur sa valeur par défaut.	`TRUE`
Journal d'audit	Définissez le `audit_trail` paramètre sur l'une des valeurs autorisées suivantes : `{ none | os | db [, extended] | xml [, extended] }`	`none`
Journal d'écoute	Aucune. Vous ne pouvez pas désactiver ce journal.	Activées

Nom du journal	Exigence	Par défaut
Journal d'Oracle Management Agent	Aucune. Vous ne pouvez pas désactiver ce journal.	Activées

Le journal d'Oracle Management Agent contient les groupes de journaux présentés dans le tableau suivant.

Nom du journal	CloudWatch groupe de journaux
emctl.log	oemagent-emctl
emdctlj.log	oemagent-emdctlj
gcagent.log	oemagent-gcagent
gcagent_errors.log	oemagent-gcagent-errors
emagent.nohup	oemagent-emagent-nohup
secure.log	oemagent-secure

Pour plus d'informations, consultez Localisation des fichiers de suivi et des fichiers journaux de Management Agent dans la documentation Oracle.

Console

Pour publier les journaux Oracle DB dans CloudWatch Logs à partir du AWS Management Console

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).
4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.
5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Pour publier les journaux Oracle vous pouvez utiliser la commande `modify-db-instance` avec les paramètres suivants :

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier des journaux Oracle en utilisant les commandes suivantes :

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-from-s3`
- `restore-db-instance-to-point-in-time`

Example

L'exemple suivant crée une instance de base de données Oracle avec la publication CloudWatch des journaux activée. La `--cloudwatch-logs-export-configuration` valeur est un JSON tableau de chaînes. Les chaînes peuvent être une combinaison de `alert`, `audit`, `listener` et `trace`.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration
 '["trace","audit","alert","listener","oemagent"]' \
    --db-instance-class db.m5.large \
    --allocated-storage 20 \
    --engine oracle-ee \
    --engine-version 19.0.0.0.ru-2024-04.rur-2024-04.r1 \
    --license-model bring-your-own-license \
```

```
    --master-username myadmin \
    --manage-master-user-password
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration trace alert audit listener oemagent ^
    --db-instance-class db.m5.large ^
    --allocated-storage 20 ^
    --engine oracle-ee ^
    --engine-version 19.0.0.0.ru-2024-04.rur-2024-04.r1 ^
    --license-model bring-your-own-license ^
    --master-username myadmin ^
    --manage-master-user-password
```

Example

L'exemple suivant modifie une instance de base de données Oracle existante pour publier des fichiers CloudWatch journaux dans Logs. La `--cloudwatch-logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes avec une combinaison quelconque de `alert`, `audit`, `listener` et `trace`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["trace","alert","audit","listener","oemagent"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration EnableLogTypes=\"trace\",\"alert\",\"audit
\",\"listener\",\"oemagent\"
```

Example

L'exemple suivant modifie une instance de base de données Oracle existante pour désactiver la publication des fichiers journaux d'audit et d'écoute dans Logs. CloudWatch La `--cloudwatch-`

`logs-export-configuration` valeur est un JSON objet. La clé pour cet objet est `DisableLogTypes` et sa valeur est un tableau de chaînes avec une combinaison quelconque de `alert`, `audit`, `listener` et `trace`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"DisableLogTypes":["audit","listener"]}'
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration DisableLogTypes=\"audit\",\"listener\"
```

RDS API

Vous pouvez publier les journaux Oracle DB à l'aide du RDSAPI. Vous pouvez appeler l'action `ModifyDBInstance` avec les paramètres suivants :

- `DBInstanceIdentifier`
- `CloudwatchLogsExportConfiguration`

> **Note**
>
> Une modification apportée au paramètre `CloudwatchLogsExportConfiguration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, le paramètre `ApplyImmediately` est sans effet.

Vous pouvez également publier des journaux Oracle en effectuant les RDS API opérations suivantes :

- `CreateDBInstance`
- `RestoreDBInstanceFromDBSnapshot`
- `RestoreDBInstanceFromS3`
- `RestoreDBInstanceToPointInTime`

Exécutez l'une de ces RDS API opérations avec les paramètres suivants :

- `DBInstanceIdentifier`
- `EnableCloudwatchLogsExports`
- `Engine`
- `DBInstanceClass`

D'autres paramètres peuvent être nécessaires en fonction de l'RDSopération que vous exécutez.

Accès aux journaux d'alertes et aux journaux des auditeurs

Vous pouvez consulter le journal des alertes à l'aide de la RDS console Amazon. Vous pouvez également utiliser la SQL déclaration suivante.

```
SELECT message_text FROM alertlog;
```

Accédez au journal de l'écouteur à l'aide d'Amazon CloudWatch Logs.

> **Note**
>
> Oracle fait pivoter les journaux d'alerte et d'écoute lorsqu'ils dépassent 10 Mo, date à laquelle ils ne sont plus disponibles sur Amazon RDS Views.

Fichiers journaux de base de données RDS for PostgreSQL

Vous pouvez surveiller les types de fichiers journaux suivants :

- Journal PostgreSQL
- Mettre à niveau le journal
- Journal des erreurs d'authentification de base de données IAM

> **Note**
>
> Pour activer les journaux d'erreurs d'authentification de base de données IAM, vous devez d'abord activer l'authentification de base de données IAM pour votre instance de base de données Aurora . Pour plus d'informations sur l'activation de l'authentification de base de données IAM, consultezActivation et désactivation de l'authentification de base de données IAM.

RDS for PostgreSQL consigne les activités de base de données dans le fichier journal PostgreSQL par défaut. Pour une instance de base de données PostgreSQL sur site, ces messages sont stockés localement dans `log/postgresql.log`. Pour une instance de base de données RDS for PostgreSQL, le fichier journal est disponible sur l'instance Amazon RDS. Ces journaux sont également accessibles via le AWS Management Console, où vous pouvez les consulter ou les télécharger. Le niveau de journalisation par défaut capture les échecs de connexion, les erreurs de serveur fatales, les blocages et les échecs de requête.

Pour plus d'informations sur l'affichage, le téléchargement et la consultation des journaux de base de données basés sur des fichiers, consultez Surveillance des fichiers journaux RDSAmazon. Pour en savoir plus sur les journaux PostgreSQL, consultez la section Working with Amazon RDS and Aurora PostgreSQL logs: Part 1 (Utilisation des journaux Amazon RDS et Aurora PostgreSQL : Partie 1) et Working with Amazon RDS and Aurora PostgreSQL logs: Part 2 (Utilisation des journaux Amazon RDS et Aurora PostgreSQL : Partie 2).

Outre les journaux PostgreSQL standard décrits dans cette rubrique, RDS for PostgreSQL prend également en charge l'extension PostgreSQL Audit (`pgAudit`). La plupart des secteurs réglementés et des agences gouvernementales doivent conserver un journal d'audit ou une piste d'audit des modifications apportées aux données afin de se conformer aux exigences légales. Pour plus d'informations sur l'installation et l'utilisation de pgAudit, consultez Utilisation pgAudit pour enregistrer l'activité de la base de données.

Rubriques

- Paramètres de connexion à pour PostgreSQL
- Activer la journalisation des requêtes pour votre instance de base de données RDS for PostgreSQL.
- Publication de journaux PostgreSQL sur Amazon Logs CloudWatch

Paramètres de connexion à pour PostgreSQL

Vous pouvez personnaliser le comportement de journalisation de votre instance de base de données RDS for PostgreSQL en modifiant divers paramètres. Dans le tableau suivant, vous trouverez les paramètres qui affectent combien de temps les journaux sont stockés, quand effectuer la rotation du journal et s'il convient de fournir en sortie le journal au format CSV (valeurs séparées par des virgules). Vous pouvez également trouver le texte de sortie envoyé à STDERR, entre autres paramètres. Pour modifier les valeurs des paramètres modifiables, utilisez un groupe de paramètres de base de données pour votre Instance RDS for PostgreSQL. Pour plus d'informations, consultez Groupes de paramètres de base de données pour les instances de base de données RDSAmazon.

Paramètre	Par défaut	Description
log_destination	stderr	Définit le format de sortie pour le journal. La valeur par défaut est `stderr`, mais vous pouvez également spécifier une valeur séparée par des virgules (CSV) en ajoutant `csvlog` au paramètre. Pour de plus amples informations, veuillez consulter Définition de la destination du journal (`stderr`, `csvlog`).
log_filename	postgresql.log.%Y-%m-%d-%H	Spécifie le modèle du nom de fichier journal. Outre la valeur par défaut, ce paramètre prend en charge `postgresql.log.%Y-%m-%d` et `postgresql.log.%Y-%m-%d-%H%M` pour le modèle de nom de fichier.
log_line_prefix	%t:%r:%u@%d:[%p]:	Définit le préfixe pour chaque ligne de journal qui est écrite sur `stderr`, afin de noter l'heure (%t), l'hôte distant (%r), l'utilisateur (%u), la

Paramètre	Par défaut	Description
		base de données (%d) et l'ID du processus (%p).
log_rotation_age	60	Minutes après lesquelles la rotation automatiq ue du fichier journal est effectuée. Vous pouvez modifier cette valeur entre 1 et 1 440 minutes. Pour de plus amples informations, veuillez consulter Définition de la rotation des fichiers journaux.
log_rotation_size	–	Taille (en Ko) à laquelle la rotation automatique du fichier journal est effectuée. Par défaut, ce paramètre n'est pas utilisé car les journaux sont pivotés en fonction de `log_rotation_age` ce paramètre. Pour en savoir plus, veuillez consulter la section Définition de la rotation des fichiers journaux.
rds.log_retention_ period	4320	Les journaux PostgreSQL plus anciens que le nombre de minutes spécifié sont supprimés. La valeur par défaut de 4 320 minutes supprime les fichiers journaux après trois jours. Pour de plus amples informations, veuillez consulter Définition de la période de conservation des journaux.

Pour identifier les problèmes d'application, vous pouvez rechercher dans le journal les échecs de requête, les échecs de connexion, les interblocages et les erreurs fatales du serveur. Par exemple, supposons que vous avez converti une application héritée d'Oracle vers Amazon RDS for PostgreSQL, mais que certaines requêtes n'ont pas été converties correctement. Ces requêtes mal formatées génèrent des messages d'erreur que vous pouvez trouver dans les journaux pour aider à identifier les problèmes. Pour plus d'informations sur la journalisation des requêtes, consultez Activer la journalisation des requêtes pour votre instance de base de données RDS for PostgreSQL..

Dans les rubriques suivantes, vous pouvez trouver des informations sur la manière de définir les différents paramètres qui contrôlent les détails de base de vos journaux PostgreSQL.

Rubriques

- Définition de la période de conservation des journaux
- Définition de la rotation des fichiers journaux
- Définition de la destination du journal (stderr, csvlog)
- Compréhension du paramètre log_line_prefix

Définition de la période de conservation des journaux

Le paramètre `rds.log_retention_period` indique la durée pendant laquelle votre instance de base de données RDS for PostgreSQL conserve ses fichiers journaux. La valeur par défaut est de 3 jours (4 320 minutes), mais vous pouvez définir une valeur comprise entre 1 jour (1 440 minutes) et 7 jours (10 080 minutes). Assurez-vous que votre instance de base de données RDS for PostgreSQL dispose d'un espace de stockage suffisant pour contenir les fichiers journaux pendant cette période.

Nous vous recommandons de publier régulièrement vos CloudWatch journaux sur Amazon Logs afin de pouvoir consulter et analyser les données système longtemps après leur suppression de votre cluster de base de données . Instance de base de données RDS for PostgreSQL. Pour plus d'informations, consultez Publication de journaux PostgreSQL sur Amazon Logs CloudWatch .

Définition de la rotation des fichiers journaux

Amazon RDS crée de nouveaux fichiers journaux toutes les heures, par défaut. Le timing est contrôlé par le paramètre `log_rotation_age`. Ce paramètre a une valeur par défaut de 60 (minutes), mais vous pouvez le régler sur une valeur comprise entre 1 minute et 24 heures (1 440 minutes). Au moment de la rotation, un fichier journal distinct est créé. Le fichier est nommé selon le modèle spécifié par le paramètre `log_filename`.

Les fichiers journaux peuvent également faire l'objet d'une rotation en fonction de leur taille, comme indiqué dans le paramètre `log_rotation_size`. Ce paramètre indique que le journal doit faire l'objet d'une rotation lorsqu'il atteint la taille spécifiée (en kilo-octets). Pour une instance de base de données RDS for PostgreSQL, la valeur `log_rotation_size` n'est pas définie, c'est-à-dire qu'il n'y a pas de valeur spécifiée. Toutefois, vous pouvez définir ce paramètre entre 0 et 2 097 151 Ko (kilo-octets).

Les noms de fichier journal sont basés sur le modèle de nom de fichier spécifié dans le paramètre `log_filename`. Les valeurs disponibles pour ce paramètre sont les suivantes :

- `postgresql.log.%Y-%m-%d` : format par défaut du nom de fichier journal. Inclut l'année, le mois et la date dans le nom du fichier journal.
- `postgresql.log.%Y-%m-%d-%H` – Inclut l'heure dans le format du nom de fichier journal.

Pour plus d'informations, consultez `log_rotation_age` et `log_rotation_size` dans la documentation de PostgreSQL.

Définition de la destination du journal (**`stderr`**, **`csvlog`**)

Par défaut, Amazon RDS PostgreSQL génère des journaux au format d'erreur standard (stderr). Ce format correspond au réglage par défaut du paramètre `log_destination`. Chaque message est préfixé selon le modèle spécifié dans le paramètre `log_line_prefix`. Pour de plus amples informations, veuillez consulter Compréhension du paramètre log_line_prefix.

RDS for PostgreSQL peut également générer les journaux au format `csvlog`. La valeur `csvlog` permet d'analyser les données du journal en tant que données CSV (valeurs séparées par des virgules). Par exemple, supposons que vous utilisez l'extension `log_fdw` pour travailler avec vos journaux en tant que tables externes. La table externe créée sur les fichiers journaux `stderr` contient une seule colonne avec les données des événements de journal. En ajoutant `csvlog` au paramètre `log_destination`, vous obtenez le fichier journal au format CSV avec des démarcations pour les différentes colonnes de la table externe. Vous pouvez désormais trier et analyser vos journaux plus facilement. Pour savoir comment utiliser `log_fdw` avec `csvlog`, consultez Utilisation de l'extension log_fdw pour accéder au journal de base de données en utilisant SQL.

Si vous spécifiez `csvlog` pour ce paramètre, sachez que les deux fichiers `stderr` et `csvlog` sont générés. Assurez-vous de surveiller le stockage consommé par les journaux, en tenant compte de `rds.log_retention_period` et des autres paramètres qui affectent le stockage et la rotation des journaux. Utiliser `stderr` et `csvlog` fait plus que doubler le stockage consommé par les journaux.

Si vous ajoutez `csvlog` à `log_destination` et que vous souhaitez revenir au paramètre `stderr` seul, vous devez réinitialiser le paramètre. Pour ce faire, ouvrez la console Amazon RDS, puis ouvrez le groupe de paramètres personnalisé de la base de données pour votre instance. Choisissez le paramètre `log_destination`, choisissez Edit parameter (Modifier le paramètre), puis Reset (Réinitialiser).

Pour plus d'informations sur la configuration de la journalisation, consultez Working with Amazon RDS and Aurora PostgreSQL logs: Part 1 (Utiliser les journaux d'Amazon RDS et Aurora PostgreSQL : partie 1).

Compréhension du paramètre log_line_prefix

Le format du `stderr` journal préfixe chaque message de journal avec les détails spécifiés par le `log_line_prefix` paramètre. La valeur par défaut est :

```
%t:%r:%u@%d:[%p]:t
```

À partir de la version 16 d'Aurora PostgreSQL, vous pouvez également choisir :

```
%m:%r:%u@%d:[%p]:%l:%e:%s:%v:%x:%c:%q%a
```

Chaque entrée de journal envoyée à stderr inclut les informations suivantes en fonction de la valeur sélectionnée :

- `%t`— Heure de saisie du journal sans millisecondes
- `%m`— Heure de saisie du journal en millisecondes
- `%r` : adresse de l'hôte distant
- `%u@%d` : nom d'utilisateur @ nom de base de données
- `[%p]` : ID de processus si disponible
- `%l`— Numéro de ligne de journal par session
- `%e`— Code d'erreur SQL
- `%s`— Horodatage de début du processus
- `%v`— Identifiant de transaction virtuel
- `%x`— Numéro de transaction
- `%c`— Identifiant de session
- `%q`— Fin de session
- `%a`— Nom de l'application

Activer la journalisation des requêtes pour votre instance de base de données RDS for PostgreSQL.

Vous pouvez collecter des informations plus détaillées sur les activités de votre base de données, notamment les requêtes, les requêtes en attente de verrouillage, les points de contrôle et de nombreux autres détails en définissant certains des paramètres répertoriés dans le tableau suivant. Cette rubrique se concentre sur la journalisation des requêtes.

Paramètre	Par défaut	Description
log_connections	–	Enregistre toutes les connexions réussies.
log_disconnections	–	Journalise la fin de chaque session et sa durée.
log_checkpoints	1	Enregistre chaque point de contrôle.
log_lock_waits	–	Enregistre les longs temps d'attente pour l'acquisition d'un verrou. Ce paramètre n'est pas défini par défaut.
log_min_duration_sample	–	(ms) Définit la durée minimum d'exécution au-delà de laquelle les instructions sont journalisées. La taille de l'échantillon est définie à l'aide du paramètre `log_statement_sample_rate` .
log_min_duration_statement	–	Toute instruction SQL exécutée au moins pendant la durée spécifiée ou plus est journalisée. Ce paramètre n'est pas défini par défaut. L'activation de ce paramètre peut vous aider à identifier les requêtes non optimisées.
log_statement	–	Définit le type d'instructions enregistrées. Par défaut, ce paramètre n'est pas défini, mais vous pouvez le modifier pour `all`, `ddl` ou `mod` afin de spécifier les types d'instructions SQL que vous souhaitez journaliser. Si vous spécifiez autre chose que `none` pour ce paramètre, vous devez également prendre des mesures

Paramètre	Par défaut	Description
		supplémentaires pour empêcher l'exposition des mots de passe dans les fichiers journaux. Pour de plus amples informations, veuillez consulter Atténuation du risque d'exposition des mots de passe lors de l'utilisation de la journalisation de requêtes.
log_statement_sample_rate	–	Le pourcentage d'instructions dépassant la durée spécifiée dans `log_min_duration_sample` pour être journalisées, exprimé sous la forme d'une valeur à virgule flottante comprise entre 0,0 et 1,0.
log_statement_stats	–	Ecrit les statistiques de performance cumulées dans le journal du serveur.

Utilisation de la journalisation pour détecter les requêtes lentes

Vous pouvez journaliser les instructions et les requêtes SQL pour favoriser la recherche des requêtes lentes. Vous pouvez activer cette fonctionnalité en modifiant les valeurs des paramètres `log_statement` et `log_min_duration`, comme indiqué dans cette section. Avant d'activer la journalisation des requêtes pour votre instance de base de données RDS for PostgreSQL, vous devez être conscient de l'exposition possible à des mots de passe dans les journaux et de la manière d'atténuer les risques. Pour de plus amples informations, veuillez consulter Atténuation du risque d'exposition des mots de passe lors de l'utilisation de la journalisation de requêtes.

Vous trouverez ci-dessous des informations de référence sur les paramètres `log_statement` et `log_min_duration`.

log_statement

Ce paramètre indique le type d'instructions SQL qui doivent être envoyées au journal. La valeur par défaut est `none`. Si vous remplacez ce paramètre par `all`, `ddl` ou `mod`, veillez à prendre les mesures recommandées pour réduire le risque d'exposition des mots de passe dans les journaux. Pour de plus amples informations, veuillez consulter Atténuation du risque d'exposition des mots de passe lors de l'utilisation de la journalisation de requêtes.

Tout

Journalise toutes les instructions. Ce paramètre est recommandé à des fins de débogage.

ddl

Journalise toutes les instructions DDL (Data Definition Language), telles que CREATE, ALTER, DROP, etc.

mod

Journalise toutes les instructions DDL et les instructions de langage de manipulation des données (DML) telles que INSERT, UPDATE et DELETE, qui modifient les données.

none

Aucune instruction SQL n'est journalisée. Nous recommandons ce paramètre pour éviter le risque d'exposer des mots de passe dans les journaux.

log_min_duration_statement

Toute instruction SQL exécutée au moins pendant la durée spécifiée ou plus est journalisée. Ce paramètre n'est pas défini par défaut. L'activation de ce paramètre peut vous aider à identifier les requêtes non optimisées.

–1–2147483647

Le nombre de millisecondes (ms) d'exécution pendant lequel une instruction est journalisée.

Configurer la journalisation des requêtes

Ces étapes supposent que votre L'instance de base de données RDS for PostgreSQL utilise un groupe de paramètres de base de données personnalisé.

1. Définissez le paramètre `log_statement` sur `all`. L'exemple suivant illustre les informations écrites dans le fichier `postgresql.log` avec cette définition de paramètre.

```
2022-10-05 22:05:52 UTC:52.95.4.1(11335):postgres@labdb:[3639]:LOG: statement:
 SELECT feedback, s.sentiment,s.confidence
FROM support,aws_comprehend.detect_sentiment(feedback, 'en') s
ORDER BY s.confidence DESC;
2022-10-05 22:05:52 UTC:52.95.4.1(11335):postgres@labdb:[3639]:LOG: QUERY
 STATISTICS
```

```
2022-10-05 22:05:52 UTC:52.95.4.1(11335):postgres@labdb:[3639]:DETAIL: ! system
 usage stats:
! 0.017355 s user, 0.000000 s system, 0.168593 s elapsed
! [0.025146 s user, 0.000000 s system total]
! 36644 kB max resident size
! 0/8 [0/8] filesystem blocks in/out
! 0/733 [0/1364] page faults/reclaims, 0 [0] swaps
! 0 [0] signals rcvd, 0/0 [0/0] messages rcvd/sent
! 19/0 [27/0] voluntary/involuntary context switches
2022-10-05 22:05:52 UTC:52.95.4.1(11335):postgres@labdb:[3639]:STATEMENT: SELECT
 feedback, s.sentiment,s.confidence
FROM support,aws_comprehend.detect_sentiment(feedback, 'en') s
ORDER BY s.confidence DESC;
2022-10-05 22:05:56 UTC:52.95.4.1(11335):postgres@labdb:[3639]:ERROR: syntax error
 at or near "ORDER" at character 1
2022-10-05 22:05:56 UTC:52.95.4.1(11335):postgres@labdb:[3639]:STATEMENT: ORDER BY
 s.confidence DESC;
----------------------- END OF LOG ----------------------
```

2. Définissez le paramètre `log_min_duration_statement`. L'exemple suivant illustre les informations écrites dans le fichier `postgresql.log` lorsque le paramètre est défini sur 1.

   Les requêtes qui dépassent la durée spécifiée dans le paramètre `log_min_duration_statement` sont enregistrées. Vous en trouverez un exemple ci-dessous. Vous pouvez consulter le fichier journal de votre instance de base données RDS for PostgreSQL dans la console Amazon RDS.

   ```
   2022-10-05 19:05:19 UTC:52.95.4.1(6461):postgres@labdb:[6144]:LOG: statement: DROP
    table comments;
   2022-10-05 19:05:19 UTC:52.95.4.1(6461):postgres@labdb:[6144]:LOG: duration:
    167.754 ms
   2022-10-05 19:08:07 UTC::@:[355]:LOG: checkpoint starting: time
   2022-10-05 19:08:08 UTC::@:[355]:LOG: checkpoint complete: wrote 11 buffers
    (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=1.013 s, sync=0.006 s,
    total=1.033 s; sync files=8, longest=0.004 s, average=0.001 s; distance=131028 kB,
    estimate=131028 kB
   ----------------------- END OF LOG ----------------------
   ```

Atténuation du risque d'exposition des mots de passe lors de l'utilisation de la journalisation de requêtes

Nous vous recommandons de garder `log_statement` sur `none` pour éviter de dévoiler les mots de passe. Si vous avez réglé `log_statement` sur `all`, `ddl` ou `mod`, nous vous recommandons de suivre une ou plusieurs des étapes suivantes.

- Pour le client, chiffrez les informations sensibles. Pour plus d'informations, consultez Options de chiffrement dans la documentation PostgreSQL. Utilisez les options `ENCRYPTED` (et `UNENCRYPTED`) des instructions `CREATE` et `ALTER`. Pour plus d'informations, consultez CREATE USER dans la documentation PostgreSQL.
- Pour votre instance de base de données RDS for PostgreSQL, configurez et utilisez l'extension PostgreSQL Auditing (pgAudit). Cette extension supprime les informations sensibles dans les instructions CREATE et ALTER envoyées au journal. Pour de plus amples informations, veuillez consulter Utilisation pgAudit pour enregistrer l'activité de la base de données.
- Limitez l'accès aux CloudWatch journaux.
- Utilisez des mécanismes d'authentification plus forts tels que IAM.

Publication de journaux PostgreSQL sur Amazon Logs CloudWatch

Pour stocker vos enregistrements de journal PostgreSQL dans un espace de stockage hautement durable, vous pouvez utiliser Amazon Logs. CloudWatch Avec CloudWatch Logs, vous pouvez également effectuer une analyse en temps réel des données des journaux et les utiliser CloudWatch pour consulter les métriques et créer des alarmes. Par exemple, si vous définissez `log_statement` sur `ddl`, vous pouvez configurer une alarme pour vous alerter chaque fois qu'une instruction DDL est exécutée. Vous pouvez choisir de télécharger vos journaux PostgreSQL dans Logs pendant le processus de création CloudWatch de votre instance de base de données RDS pour PostgreSQL. Si vous avez choisi de ne pas télécharger de journaux à ce moment-là, vous pouvez modifier ultérieurement votre instance pour commencer à charger les journaux à partir de ce moment. En d'autres termes, les journaux existants ne sont pas chargés. Seuls les nouveaux journaux sont chargés lorsqu'ils sont créés sur votre instance de base de données RDS for PostgreSQL modifiée.

Toutes les versions de RDS pour PostgreSQL actuellement disponibles prennent en charge la publication de fichiers journaux dans Logs. CloudWatch Pour plus d'informations, consultez Mises à jour d'Amazon RDS for PostgreSQL dans les notes de mise à jour d'Amazon RDS for PostgreSQL.

Pour utiliser les CloudWatch journaux, configurez votre instance de base de données RDS pour PostgreSQL afin de publier les données des journaux dans un groupe de journaux.

Vous pouvez publier les types de CloudWatch journaux suivants dans Logs for RDS pour PostgreSQL :

- Journal PostgreSQL
- Mettre à niveau le journal
- Journal des erreurs d'authentification de base de données IAM

Une fois la configuration terminée, Amazon RDS publie les événements du journal dans les flux de journaux au sein d'un groupe de CloudWatch journaux. Par exemple, les données de journaux PostgreSQL sont stockés dans le groupe de journaux `/aws/rds/instance/my_instance/postgresql`. Pour consulter vos journaux, ouvrez la CloudWatch console à l'adresse https://console.aws.amazon.com/cloudwatch/.

console

Pour publier les journaux PostgreSQL dans Logs CloudWatch à l'aide de la console

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données que vous souhaitez modifier, puis choisissez Modifier.
4. Dans la section Exportations de journaux, choisissez les journaux que vous souhaitez commencer à publier dans CloudWatch Logs.

   La section Exportations de journaux n'est disponible que pour les versions de PostgreSQL qui prennent en charge la publication dans Logs. CloudWatch
5. Choisissez Continuer, puis Modifier l'instance de base de données sur la page récapitulative.

AWS CLI

Vous pouvez publier des journaux PostgreSQL à l'aide du. AWS CLI Vous pouvez appeler la commande `modify-db-instance` avec les paramètres suivants.

- `--db-instance-identifier`
- `--cloudwatch-logs-export-configuration`

> **Note**
>
> Une modification apportée à l'option `--cloudwatch-logs-export-configuration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, les options `--apply-immediately` et `--no-apply-immediately` sont sans effet.

Vous pouvez également publier des journaux PostgreSQL en appelant les commandes de CLI suivantes :

- `create-db-instance`
- `restore-db-instance-from-db-snapshot`
- `restore-db-instance-to-point-in-time`

Exécutez l'une de ces commandes de CLI avec les options suivantes :

- `--db-instance-identifier`
- `--enable-cloudwatch-logs-exports`
- `--db-instance-class`
- `--engine`

D'autres options peuvent être requises en fonction de la commande de CLI que vous exécutez.

Example Modifier une instance pour publier des journaux dans CloudWatch Logs

L'exemple suivant modifie une instance de base de données PostgreSQL existante pour publier des fichiers journaux dans Logs. CloudWatch La valeur `--cloudwatch-logs-export-configuration` n'est pas un objet JSON. La clé pour cet objet est `EnableLogTypes` et sa valeur est un tableau de chaînes avec une combinaison quelconque de `postgresql` et `upgrade`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":["postgresql",
 "upgrade"]}'
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --cloudwatch-logs-export-configuration '{"EnableLogTypes":
["postgresql","upgrade"]}'
```

Example Création d'une instance pour publier des journaux dans CloudWatch Logs

L'exemple suivant crée une instance de base de données PostgreSQL et publie des fichiers journaux dans Logs. CloudWatch La valeur `--enable-cloudwatch-logs-exports` est un tableau de chaînes JSON. Les chaînes peuvent être une combinaison quelconque de `postgresql` et `upgrade`.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --enable-cloudwatch-logs-exports '["postgresql","upgrade"]' \
    --db-instance-class db.m4.large \
    --engine postgres
```

Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --enable-cloudwatch-logs-exports '["postgresql","upgrade"]' ^
    --db-instance-class db.m4.large ^
    --engine postgres
```

API RDS

Vous pouvez publier des journaux PostgreSQL avec l'API RDS. Vous pouvez appeler l'action `ModifyDBInstance` avec les paramètres suivants :

- `DBInstanceIdentifier`
- `CloudwatchLogsExportConfiguration`

> **Note**
>
> Une modification apportée au paramètre `CloudwatchLogsExportConfiguration` est toujours appliquée immédiatement à l'instance de base de données. Par conséquent, le paramètre `ApplyImmediately` est sans effet.

Vous pouvez également publier des journaux PostgreSQL en appelant les opérations d'API RDS suivantes :

- `CreateDBInstance`
- `RestoreDBInstanceFromDBSnapshot`
- `RestoreDBInstanceToPointInTime`

Exécutez l'une de ces opérations d'API RDS avec les paramètres suivants :

- `DBInstanceIdentifier`
- `EnableCloudwatchLogsExports`
- `Engine`
- `DBInstanceClass`

D'autres paramètres peuvent être requis en fonction de l'opération que vous exécutez.

Surveillance (Amazon RDS API appelle) AWS CloudTrail

AWS CloudTrail est un AWS service qui vous aide à auditer votre AWS compte. AWS CloudTrail est activé pour votre AWS compte lorsque vous le créez. Pour plus d'informations CloudTrail, consultez le guide de AWS CloudTrail l'utilisateur.

Rubriques

- CloudTrail intégration avec Amazon RDS
- Entrées du fichier RDS journal Amazon

CloudTrail intégration avec Amazon RDS

Toutes les RDS actions Amazon sont enregistrées par CloudTrail. CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans Amazon RDS.

CloudTrail événements

CloudTrail capture API les appels pour Amazon RDS sous forme d'événements. Un événement représente une demande individuelle émise à partir d'une source quelconque et comprend des informations sur l'action demandée, la date et l'heure de l'action, les paramètres de la demande, etc. Les événements incluent les appels depuis la RDS console Amazon et les appels de code vers les RDS API opérations Amazon.

RDS L'activité d'Amazon est enregistrée dans un CloudTrail événement dans l'historique des événements. Vous pouvez utiliser la CloudTrail console pour consulter les 90 derniers jours d'APIactivité et d'événements enregistrés dans une AWS région. Pour plus d'informations, consultez la section Affichage des événements avec l'historique des CloudTrail événements.

CloudTrail sentiers

Pour un enregistrement continu des événements de votre AWS compte, y compris des événements relatifs à Amazon RDS, créez un historique. Un suivi est une configuration qui permet de transmettre des événements à un compartiment Amazon S3 spécifique. CloudTrail fournit généralement les fichiers journaux dans les 15 minutes suivant l'activité du compte.

> **Note**
>
> Si vous ne configurez pas de suivi, vous pouvez toujours consulter les événements les plus récents dans la CloudTrail console dans Historique des événements.

Vous pouvez créer deux types de sentiers pour un AWS compte : un sentier qui s'applique à toutes les régions ou un sentier qui s'applique à une région. Par défaut, lorsque vous créez un journal de suivi dans la console, il s’applique à toutes les régions .

En outre, vous pouvez configurer d'autres AWS services pour analyser plus en détail les données d'événements collectées dans les CloudTrail journaux et agir en conséquence. Pour plus d’informations, consultez :

- Présentation de la création d’un journal d’activité
- CloudTrail services et intégrations pris en charge
- Configuration des SNS notifications Amazon pour CloudTrail
- Réception de fichiers CloudTrail journaux de plusieurs régions et réception de fichiers CloudTrail journaux de plusieurs comptes

Entrées du fichier RDS journal Amazon

CloudTrail les fichiers journaux contiennent une ou plusieurs entrées de journal. CloudTrail les fichiers journaux ne constituent pas une trace ordonnée des API appels publics, ils n'apparaissent donc pas dans un ordre spécifique.

L'exemple suivant montre une entrée de CloudTrail journal illustrant l'`CreateDBInstance`action.

```
{
    "eventVersion": "1.04",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "AKIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::123456789012:user/johndoe",
        "accountId": "123456789012",
        "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
        "userName": "johndoe"
    },
```

```
    "eventTime": "2018-07-30T22:14:06Z",
    "eventSource": "rds.amazonaws.com",
    "eventName": "CreateDBInstance",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "aws-cli/1.15.42 Python/3.6.1 Darwin/17.7.0 botocore/1.10.42",
    "requestParameters": {
        "enableCloudwatchLogsExports": [
            "audit",
            "error",
            "general",
            "slowquery"
        ],
        "dBInstanceIdentifier": "test-instance",
        "engine": "mysql",
        "masterUsername": "myawsuser",
        "allocatedStorage": 20,
        "dBInstanceClass": "db.m1.small",
        "masterUserPassword": "****"
    },
    "responseElements": {
        "dBInstanceArn": "arn:aws:rds:us-east-1:123456789012:db:test-instance",
        "storageEncrypted": false,
        "preferredBackupWindow": "10:27-10:57",
        "preferredMaintenanceWindow": "sat:05:47-sat:06:17",
        "backupRetentionPeriod": 1,
        "allocatedStorage": 20,
        "storageType": "standard",
        "engineVersion": "8.0.28",
        "dbInstancePort": 0,
        "optionGroupMemberships": [
            {
                "status": "in-sync",
                "optionGroupName": "default:mysql-8-0"
            }
        ],
        "dBParameterGroups": [
            {
                "dBParameterGroupName": "default.mysql8.0",
                "parameterApplyStatus": "in-sync"
            }
        ],
        "monitoringInterval": 0,
        "dBInstanceClass": "db.m1.small",
```

```
        "readReplicaDBInstanceIdentifiers": [],
        "dBSubnetGroup": {
            "dBSubnetGroupName": "default",
            "dBSubnetGroupDescription": "default",
            "subnets": [
                {
                    "subnetAvailabilityZone": {"name": "us-east-1b"},
                    "subnetIdentifier": "subnet-cbfff283",
                    "subnetStatus": "Active"
                },
                {
                    "subnetAvailabilityZone": {"name": "us-east-1e"},
                    "subnetIdentifier": "subnet-d7c825e8",
                    "subnetStatus": "Active"
                },
                {
                    "subnetAvailabilityZone": {"name": "us-east-1f"},
                    "subnetIdentifier": "subnet-6746046b",
                    "subnetStatus": "Active"
                },
                {
                    "subnetAvailabilityZone": {"name": "us-east-1c"},
                    "subnetIdentifier": "subnet-bac383e0",
                    "subnetStatus": "Active"
                },
                {
                    "subnetAvailabilityZone": {"name": "us-east-1d"},
                    "subnetIdentifier": "subnet-42599426",
                    "subnetStatus": "Active"
                },
                {
                    "subnetAvailabilityZone": {"name": "us-east-1a"},
                    "subnetIdentifier": "subnet-da327bf6",
                    "subnetStatus": "Active"
                }
            ],
            "vpcId": "vpc-136a4c6a",
            "subnetGroupStatus": "Complete"
        },
        "masterUsername": "myawsuser",
        "multiAZ": false,
        "autoMinorVersionUpgrade": true,
        "engine": "mysql",
        "cACertificateIdentifier": "rds-ca-2015",
```

```
        "dbiResourceId": "db-ETDZIIXHEWY5N7GXVC4SH7H5IA",
        "dBSecurityGroups": [],
        "pendingModifiedValues": {
            "masterUserPassword": "****",
            "pendingCloudwatchLogsExports": {
                "logTypesToEnable": [
                    "audit",
                    "error",
                    "general",
                    "slowquery"
                ]
            }
        },
        "dBInstanceStatus": "creating",
        "publiclyAccessible": true,
        "domainMemberships": [],
        "copyTagsToSnapshot": false,
        "dBInstanceIdentifier": "test-instance",
        "licenseModel": "general-public-license",
        "iAMDatabaseAuthenticationEnabled": false,
        "performanceInsightsEnabled": false,
        "vpcSecurityGroups": [
            {
                "status": "active",
                "vpcSecurityGroupId": "sg-f839b688"
            }
        ]
    },
    "requestID": "daf2e3f5-96a3-4df7-a026-863f96db793e",
    "eventID": "797163d3-5726-441d-80a7-6eeb7464acd4",
    "eventType": "AwsApiCall",
    "recipientAccountId": "123456789012"
}
```


Comme indiqué dans l'élément `userIdentity` de l'exemple précédent, chaque événement ou entrée de journal contient des informations sur la personne qui a généré la demande. Les informations relatives à l'identité permettent de déterminer les éléments suivants :

- Si la demande a été effectuée avec les informations d'identification utilisateur racine ou IAM.
- Si la demande a été effectuée avec les informations d’identification de sécurité temporaires d’un rôle ou d’un utilisateur fédéré.
- Si la demande a été faite par un autre AWS service.

Pour plus d'informations sur le`userIdentity`, consultez l'CloudTrail userIdentity élément. Pour plus d'informations sur `CreateDBInstance` les actions et les autres RDS actions Amazon, consultez la RDSAPIréférence Amazon.

Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données

En utilisant les flux d'activité de base de données, vous pouvez surveiller en temps quasi réel les flux d'activité de base de données.

Rubriques

- Présentation des flux d'activité de base de données
- Configuration d'audit unifié pour Oracle Database
- Configuration de la politique d'audit pour Amazon RDS pour Microsoft SQL Server
- Démarrage d'un flux d'activité de base de données
- Modifier le flux d'activité d'une base de données pour Amazon RDS
- Obtention de l'état d'un flux d'activité de base de données
- Arrêt d'un flux d'activité de base de données
- Surveillance des flux d'activité de base de données
- Exemples de politiques IAM pour les flux d'activité des bases de données

Présentation des flux d'activité de base de données

En tant qu'administrateur de base de données Amazon RDS, vous devez protéger votre base de données et satisfaire aux exigences en matière de conformité et de réglementation. Une politique consiste à intégrer les flux d'activités de base de données avec vos outils de surveillance. De cette façon, vous surveillez et définissez des alarmes pour l'activité d'audit dans votre base de données de .

Les menaces de sécurité sont à la fois externes et internes. Pour vous protéger contre des menaces internes, vous pouvez contrôler l'accès administrateur aux flux de données à l'aide de la fonction Database Activity Streams. Amazon RDS DBAs n'a pas accès à la collecte, à la transmission, au stockage et au traitement des flux.

Table des matières

- Fonctionnement des flux d'activité de base de données
- Audit dans Oracle Database et la base de données Microsoft SQL Server
  - Audit unifié dans Oracle Database

- Audit dans Microsoft SQL Server
- Champs d'audit non natifs pour Oracle Database et SQL Server
- Remplacement de groupe de paramètres de base de données
- Mode asynchrone pour les flux d'activité de base de données
- Exigences et limites pour les flux d'activité de base de données
- Disponibilité des régions et des versions
- Classes d'instance de base de données prises en charge pour les flux d'activité de base de données

Fonctionnement des flux d'activité de base de données

Amazon RDS envoie (push) les activités vers un flux de données Amazon Kinesis en temps quasi réel. Le flux Kinesis est créé automatiquement. Kinesis vous permet de configurer des AWS services tels qu'Amazon Data Firehose, de consommer le flux et AWS Lambda de stocker les données.

> **Important**
>
> L'utilisation de la fonction de flux d'activité de base de données dans Amazon RDS est gratuite, mais Amazon Kinesis facture un flux de données. Pour plus d'informations, consultez la Tarification d'Amazon Kinesis Data Streams.

Vous pouvez configurer les applications de gestion de la conformité pour qu'elles consomment les flux d'activité des bases de données. Ces applications peuvent utiliser le flux pour générer des alertes et auditer l'activité sur votre base de données.

Amazon RDS prend en charge les flux d'activité des bases de données dans les déploiements multi-AZ. Dans ce cas, les flux d'activité de la base de données vérifient à la fois les instances principales et les instances en veille.

Audit dans Oracle Database et la base de données Microsoft SQL Server

L'audit est la surveillance et l'enregistrement d'actions de base de données configurées. Amazon RDS ne capture aucune activité de base de données par défaut. Vous créez et gérez vous-même les politiques d'audit dans votre base de données.

Rubriques

- Audit unifié dans Oracle Database
- Audit dans Microsoft SQL Server
- Champs d'audit non natifs pour Oracle Database et SQL Server
- Remplacement de groupe de paramètres de base de données

Audit unifié dans Oracle Database

Dans une base de données Oracle, une politique d'audit unifié est un groupe nommé de paramètres d'audit que vous pouvez utiliser pour auditer un aspect du comportement utilisateur. Une politique peut être aussi simple que l'audit des activités d'un seul utilisateur. Vous pouvez également créer des politiques d'audit complexes qui utilisent des conditions.

Une base de données Oracle écrit des enregistrements d'audit, dont des enregistrements d'audit SYS, dans la trace d'audit unifié. Par exemple, si une erreur survient pendant une instruction `INSERT`, un audit standard indique le numéro d'erreur et le code SQL qui a été exécuté. La trace d'audit se trouve dans une table en lecture seule dans le schéma `AUDSYS`. Pour accéder à ces enregistrements, interrogez la vue du dictionnaire de données `UNIFIED_AUDIT_TRAIL`.

Généralement, vous configurez les flux d'activité de base de données comme suit :

1. Créez une politique d'audit Oracle Database à l'aide de la commande `CREATE AUDIT POLICY`.

   Oracle Database génère des enregistrements d'audit.
2. Activez la politique d'audit à l'aide de la commande `AUDIT POLICY`.
3. Configurer les flux d'activité de base de données.

   Seules les activités qui correspondent aux politiques d'audit d'Oracle Database sont capturées et envoyées au flux de données Amazon Kinesis. Lorsque les flux d'activité de base de données sont activés, un administrateur de base de données Oracle ne peut pas modifier la politique d'audit ou supprimer des journaux d'audit.

Pour en savoir plus sur les politiques d'audit unifié, consultez About Auditing Activities with Unified Audit Policies and AUDIT dans Oracle Database Security Guide.

Audit dans Microsoft SQL Server

Database Activity Stream utilise SQLAudit une fonctionnalité pour auditer la base de données SQL Server.

L'instance RDS pour SQL Server contient les éléments suivants :

- Audit de serveur – L'audit SQL Server collecte une instance unique d'actions au niveau du serveur ou de la base de données, ainsi qu'un groupe d'actions à surveiller. Les audits au niveau du serveur `RDS_DAS_AUDIT` et `RDS_DAS_AUDIT_CHANGES` sont gérés par RDS.
- Spécification d'audit de serveur – La spécification d'audit de serveur enregistre les événements au niveau du serveur. Vous pouvez modifier la spécification `RDS_DAS_SERVER_AUDIT_SPEC`. Cette spécification est liée à l'audit du serveur `RDS_DAS_AUDIT`. La spécification `RDS_DAS_CHANGES_AUDIT_SPEC` est gérée par RDS.
- Spécification d'audit de base de données – La spécification d'audit de base de données enregistre les événements au niveau du serveur. Vous pouvez créer une spécification d'audit de base de données `RDS_DAS_DB_<name>` et la lier à l'audit de serveur `RDS_DAS_AUDIT`.

Vous pouvez configurer les flux d'activité de base de données à l'aide de la console ou de l'interface de ligne de commande. Généralement, vous configurez les flux d'activité de base de données comme suit :

1. (Facultatif) Créez une spécification d'audit de base de données à l'aide de la commande `CREATE DATABASE AUDIT SPECIFICATION` et associez-la à l'audit de serveur `RDS_DAS_AUDIT`.
2. (Facultatif) Modifiez la spécification d'audit de serveur à l'aide de la commande `ALTER SERVER AUDIT SPECIFICATION` et définissez les politiques.
3. Activez les politiques d'audit de base de données et de serveur. Par exemple :

   ```
   ALTER DATABASE AUDIT SPECIFICATION [<Your database specification>] WITH
   (STATE=ON)

   ALTER SERVER AUDIT SPECIFICATION [RDS_DAS_SERVER_AUDIT_SPEC] WITH
   (STATE=ON)
   ```

4. Configurer les flux d'activité de base de données.

   Seules les activités qui correspondent aux politiques d'audit de serveur et de base de données sont capturées et envoyées au flux de données Amazon Kinesis. Quand les flux d'activité de base de données sont activés et que les politiques sont verrouillées, un administrateur de base de données ne peut pas modifier la politique d'audit ni supprimer des journaux d'audit.

> **Important**
>
> Si la spécification d'audit de base de données pour une base de données spécifique est activée et que la politique est à l'état verrouillé, la base de données ne peut pas être supprimée.

Pour plus d'informations sur l'audit SQL Server, consultez Composants d'audit SQL Server dans la documentation sur Microsoft SQL Server.

Champs d'audit non natifs pour Oracle Database et SQL Server

Lorsque vous démarrez un flux d'activité de base de données, chaque événement de base de données génère un événement de flux d'activité correspondant. Par exemple, un utilisateur de base de données peut exécuter des instructions `SELECT` et `INSERT`. La base de données audite ces événements et les envoie à un flux de données Amazon Kinesis Data Stream.

Les événements sont représentés dans le flux comme des objets JSON. Un objet JSON contient un `DatabaseActivityMonitoringRecord`, qui contient un tableau `databaseActivityEventList`. Les champs prédéfinis dans le tableau sont `class`, `clientApplication` et `command`.

Par défaut, un flux d'activité n'inclut pas de champs d'audit natifs du moteur. Vous pouvez configurer Amazon RDS pour Oracle et SQL Server de sorte qu'il inclue ces champs supplémentaires dans l'objet JSON `engineNativeAuditFields`.

Dans Oracle Database, la plupart des événements dans la trace d'audit unifié sont mappés à des champs dans le flux d'activité de données RDS. Par exemple, le champ `UNIFIED_AUDIT_TRAIL.SQL_TEXT` dans un audit mappe au champ `commandText` dans un flux d'activité de base de données. Toutefois, des champs d'audit d'Oracle Database tels que `OS_USERNAME` ne mappent pas à des champs prédéfinis dans un flux d'activité de base de données.

Dans SQL Server, la plupart des champs de l'événement enregistrés par le système SQLAudit correspondent aux champs du flux d'activité de la base de données RDS. Par exemple, le champ `code` issu de `sys.fn_get_audit_file` dans l'audit est mappé sur le champ `commandText` dans un flux d'activité de base de données. Toutefois, les champs d'audit de base de données SQL Server, tels que `permission_bitmask`, ne sont pas mappés sur les champs prédéfinis dans un flux d'activité de base de données.

Pour plus d'informations sur databaseActivityEvent List, consultezdatabaseActivityEventAfficher le tableau JSON pour les flux d'activité de base de données.

Remplacement de groupe de paramètres de base de données

En règle générale, vous activez l'audit unifié dans RDS for Oracle en attachant un groupe de paramètres. Toutefois, les flux d'activité de base de données nécessitent une configuration supplémentaire. Pour améliorer votre expérience client, Amazon RDS procède comme suit :

- Si vous activez un flux d'activité, RDS pour Oracle ignore les paramètres d'audit dans le groupe de paramètres.
- Si vous désactivez un flux d'activité, RDS pour Oracle cesse d'ignorer les paramètres d'audit.

Le flux d'activité de base de données pour SQL Server est indépendant des paramètres que vous définissez dans l'option d'audit SQL.

Mode asynchrone pour les flux d'activité de base de données

Les flux d'activité dans Amazon RDS sont toujours asynchrones. Quand une session de base de données génère un événement de flux d'activité, la session revient immédiatement aux activités normales. En arrière-plan, Amazon RDS transforme l'événement de flux d'activité en un enregistrement durable.

Si une erreur se produit dans la tâche en arrière-plan, Amazon RDS génère un événement. Cet événement indique le début et la fin de toute fenêtre de temps au cours de laquelle des enregistrements d'événement de flux d'activité ont pu être perdus. Le mode asynchrone favorise les performances de la base de données plutôt que la précision du flux d'activité.

Exigences et limites pour les flux d'activité de base de données

Dans RDS, les flux d'activité de base de données présentent les limites et les exigences suivantes :

- Amazon Kinesis est nécessaire pour les flux d'activité des bases de données.
- AWS Key Management Service (AWS KMS) est obligatoire pour les flux d'activité de base de données car ils sont toujours chiffrés.
- L'application d'un chiffrement supplémentaire à votre flux de données Amazon Kinesis est incompatible avec les flux d'activité de base de données, qui sont déjà chiffrés avec votre AWS KMS clé.

- Vous créez et gérez vous-même les politiques d'audit. Contrairement à Amazon Aurora, RDS for Oracle ne capture aucune activité de base de données par défaut.
- Vous créez et gérez vous-même les politiques d'audit ou les spécifications. Contrairement à Amazon Aurora, Amazon RDS ne capture aucune activité de base de données par défaut.
- Dans un déploiement multi-AZ, démarrez le flux d'activité de base de données uniquement sur l'instance de base de données principale. Le flux d'activité vérifie automatiquement les instances de base de données principales et en veille. Aucune étape supplémentaire n'est requise lors d'un basculement.
- Le fait de renommer une instance de base de données ne crée pas un nouveau flux Kinesis.
- CDBs ne sont pas pris en charge par RDS pour Oracle.
- Les réplicas en lecture ne sont pas pris en charge.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions avec les flux d'activité des bases de données, consultez Régions et moteurs de base de données pris en charge pour les flux d'activité des bases de données sur Amazon RDS.

Classes d'instance de base de données prises en charge pour les flux d'activité de base de données

Pour RDS pour Oracle, vous pouvez utiliser des flux d'activité de base de données avec les classes d'instances de base de données suivantes :

- db.m4.*large
- db.m5.*large
- db.m5d.*large
- db.m6i.*large
- db.r4.*large
- db.r5.*large
- db.r5.*large.tpc*.mem*x
- db.r5b.*large
- db.r5b.*large.tpc*.mem*x

- db.r5d.*large
- db.r6i.*large
- db.r6i. *grand.tpc*.mem*x
- db.x2idn.*large
- db.x2iedn.*large
- db.x2iezn.*large
- db.z1d.*large

Pour RDS pour SQL Server, vous pouvez utiliser des flux d'activité de base de données avec les classes d'instances de base de données suivantes :

- db.m4.*large
- db.m5.*large
- db.m5d.*large
- db.m6i.*large
- db.r4.*large
- db.r5.*large
- db.r5b.*large
- db.r5d.*large
- db.r6i.*large
- db.x1e.*large
- db.x2iedn.*large
- db.z1d.*large

Pour plus d'informations sur les types de classes d'instances , consultez Classes d'instances de base de données .

Configuration d'audit unifié pour Oracle Database

Lorsque vous configurez un audit unifié pour une utilisation avec des flux d'activité de base de données, les situations suivantes peuvent se présenter :

- L'audit unifié est configuré pour votre base de données Oracle

Dans ce cas, créez de nouvelles politiques avec la commande `CREATE AUDIT POLICY`, puis activez-les avec la commande `AUDIT POLICY`. L'exemple suivant crée et active une politique pour surveiller les utilisateurs disposant de privilèges et de rôles spécifiques.

```
CREATE AUDIT POLICY table_pol
PRIVILEGES CREATE ANY TABLE, DROP ANY TABLE
ROLES emp_admin, sales_admin;

AUDIT POLICY table_pol;
```

Pour obtenir des instructions complètes, consultez Configuring Audit Policies dans la documentation Oracle Database.

- L'audit unifié est configuré pour votre base de données Oracle

  Lorsque vous activez un flux d'activité de base de données, RDS pour Oracle efface automatiquement les données d'audit existantes. Il révoque également les privilèges de trace d'audit. RDS for Oracle ne peut plus effectuer les opérations suivantes :

  - Purger les enregistrements de journal d'activité d'audit unifié.
  - Ajouter, supprimer ou modifier la politique d'audit unifié.
  - Mettre à jour le dernier horodatage archivé.

  > **Important**
  >
  > Nous vous recommandons fortement de sauvegarder vos données d'audit avant d'activer un flux d'activité de base de données.

  Pour obtenir une description de la vue `UNIFIED_AUDIT_TRAIL`, consultez UNIFIED_AUDIT_TRAIL. Si vous possédez un compte auprès d'Oracle Support, consultez How To Purge The UNIFIED AUDIT TRAIL.

Configuration de la politique d'audit pour Amazon RDS pour Microsoft SQL Server

L'audit SQL du serveur est effectué sur une instance de base de données Server`RDS_DAS_AUDIT`, qui est géré par AmazonRDS. Vous pouvez définir les politiques d'enregistrement des événements

de serveur dans la spécification d'audit de serveur RDS_DAS_SERVER_AUDIT_SPEC. Vous pouvez créer une spécification d'audit de base de données, telle que RDS_DAS_DB_<name>, et définir les politiques d'enregistrement des événements de base de données. Pour obtenir la liste des groupes d'actions d'audit au niveau du serveur et de la base de données, consultez SQL la section Groupes d'actions et actions d'audit du serveur dans la documentation Microsoft SQL Server.

La politique de serveur par défaut surveille uniquement les échecs de connexion et les modifications apportées aux spécifications d'audit de base de données ou de serveur pour les flux d'activité de base de données.

Les limites de l'audit et des spécifications d'audit sont les suivantes :

- Vous ne pouvez pas modifier les spécifications d'audit de serveur ou de base de données lorsque le flux d'activité de base de données est à l'état verrouillé.
- Vous ne pouvez pas modifier la spécification RDS_DAS_AUDIT d'audit de serveur.
- Vous ne pouvez pas modifier l'audit SQL du serveur RDS_DAS_CHANGES ni les spécifications d'audit du serveur associéesRDS_DAS_CHANGES_AUDIT_SPEC.
- Lors de la création d'une spécification d'audit de base de données, vous devez utiliser le format RDS_DAS_DB_<name>, par exemple RDS_DAS_DB_databaseActions.

> **Important**
>
> Pour les classes d'instances plus petites, nous vous recommandons de ne pas auditer toutes les données, mais seulement les données requises. Cela permet de réduire l'impact des flux d'activité de base de données sur les performances de ces classes d'instance.

L'exemple de code suivant modifie la spécification d'audit de serveur RDS_DAS_SERVER_AUDIT_SPEC et audite toute déconnexion et toute action de connexion réussie :

```
ALTER SERVER AUDIT SPECIFICATION [RDS_DAS_SERVER_AUDIT_SPEC]
      WITH (STATE=OFF);
ALTER SERVER AUDIT SPECIFICATION [RDS_DAS_SERVER_AUDIT_SPEC]
      ADD (LOGOUT_GROUP),
      ADD (SUCCESSFUL_LOGIN_GROUP)
      WITH (STATE = ON );
```

L'exemple de code suivant crée une spécification d'audit de base de données `RDS_DAS_DB_database_spec` et l'attache à l'audit de serveur `RDS_DAS_AUDIT` :

```
USE testDB;
CREATE DATABASE AUDIT SPECIFICATION [RDS_DAS_DB_database_spec]
     FOR SERVER AUDIT [RDS_DAS_AUDIT]
     ADD ( INSERT, UPDATE, DELETE
          ON testTable BY testUser )
     WITH (STATE = ON);
```

Une fois les spécifications d'audit configurées, veillez à ce que les spécifications `RDS_DAS_SERVER_AUDIT_SPEC` et `RDS_DAS_DB_<name>` soient définies sur l'état `ON`. Elles peuvent désormais envoyer les données d'audit à votre flux d'activité de base de données.

Démarrage d'un flux d'activité de base de données

Lorsque vous démarrez un flux d'activité pour l'instance de base de données, chaque événement d'activité de base de données que vous avez configuré dans la politique d'audit génère un événement de flux d'activité. Des commandes SQL telles que `CONNECT` et `SELECT` génèrent des événements d'accès. Des commandes SQL telles que `CREATE` et `INSERT` génèrent des événements de modification.

> **Important**
>
> L'activation d'un flux d'activité pour une instance de base de données Oracle efface les données d'audit existantes. Il révoque également les privilèges de trace d'audit. Quand le flux est activé, RDS for Oracle ne peut plus effectuer les opérations suivantes :
>
> - Purger les enregistrements de journal d'activité d'audit unifié.
> - Ajouter, supprimer ou modifier la politique d'audit unifié.
> - Mettre à jour le dernier horodatage archivé.

Console

Pour démarrer un flux d'activité de base de données

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).

3. Choisissez l'instance de base de données Amazon RDS sur laquelle vous souhaitez démarrer un flux d'activité. Dans un déploiement Multi-AZ, démarrez le flux uniquement sur l'instance principale. Le flux d'activité vérifie à la fois l'instance principale et l'instance en veille.
4. Pour Actions, choisissez Start activity stream (Démarrer le flux d'activité).

   La *name* fenêtre Démarrer le flux d'activité de la base de données : apparaît, où se *name* trouve votre RDS.
5. Définissez les paramètres suivants :

   - Pour AWS KMS key, choisissez une clé dans la liste des AWS KMS keys.

     Amazon RDS utilise la clé KMS pour chiffrer la clé qui va à son tour chiffrer l'activité de base de données. Choisissez une clé KMS différente de la clé par défaut. Pour plus d'informations sur les clés de chiffrement et AWS KMS voir Qu'est-ce que c'est AWS Key Management Service ? dans le Guide AWS Key Management Service du développeur.
   - Pour Événements d'activité de base de données, choisissez Activer les champs d'audit natifs au moteur pour inclure les champs d'audit spécifiques du moteur.
   - Choisissez Immédiatement.

     Lorsque vous choisissez Immédiatement, le instance RDSredémarre tout de suite. Si vous choisissez Pendant la prochaine fenêtre de maintenance, le instance RDS ne redémarre pas tout de suite. Dans ce cas, le flux d'activité de base de données ne démarre pas avant la prochaine fenêtre de maintenance.
6. Choisissez Démarrer le flux d'activité de base de données.

   Le statut pour la base de données indique que le flux d'activité démarre.

> **Note**
>
> Si l'erreur `You can't start a database activity stream in this configuration` s'affiche, vérifiez Classes d'instance de base de données prises en charge pour les flux d'activité de base de données pour voir si votre instance RDS utilise une classe d'instance prise en charge.

AWS CLI

Pour démarrer des flux d'activité de base de données pour de base de données (une instance de base de données), configurez de base de données (la base de données) à l'aide de la start-activity-stream AWS CLI commande.

- `--resource-arn` *arn* – Spécifie l'Amazon Resource Name (ARN) du instance de base de données.
- `--kms-key-id` *key* – Spécifie l'identificateur de clé KMS pour le chiffrement des messages dans le flux d'activité de base de données. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias du AWS KMS key.
- `--engine-native-audit-fields-included` – Inclut des champs d'audit spécifiques du moteur dans le flux de données. Pour exclure ces champs, spécifiez `--no-engine-native-audit-fields-included` (par défaut).

L'exemple suivant démarre un flux d'activité de base de données pour une instance de base de données en mode asynchrone.

Pour LinuxmacOS, ou Unix :

```
aws rds start-activity-stream \
    --mode async \
    --kms-key-id my-kms-key-arn \
    --resource-arn my-instance-arn \
    --engine-native-audit-fields-included \
    --apply-immediately
```

Dans Windows :

```
aws rds start-activity-stream ^
    --mode async ^
    --kms-key-id my-kms-key-arn ^
    --resource-arn my-instance-arn ^
    --engine-native-audit-fields-included ^
    --apply-immediately
```

Amazon RDS API

Pour démarrer des flux d'activité de base de données pour une de bases de données, configurez à l'aide de l'StartActivityStreamopération.

Appelez l'action avec les paramètres ci-dessous :

- `Region`
- `KmsKeyId`
- `ResourceArn`
- `Mode`
- `EngineNativeAuditFieldsIncluded`

Modifier le flux d'activité d'une base de données pour Amazon RDS

Vous souhaiterez peut-être personnaliser votre politique RDS d'audit Amazon au démarrage de votre flux d'activité. Si vous ne voulez pas perdre de temps et de données en arrêtant votre flux d'activité, vous pouvez modifier l'état de la politique d'audit en choisissant l'un des paramètres suivants :

Locked (default) [Verrouillé (par défaut)]

Les politiques d'audit de votre base de données sont en lecture seule.

Unlocked (Débloqué)

Les politiques d'audit de votre base de données sont en lecture/écriture.

La procédure de base est la suivante :

1. Modifiez l'état de la politique d'audit pour qu'elle soit déverrouillée.
2. Personnalisez votre politique d'audit.
3. Modifier l'état de la politique d'audit pour qu'elle soit verrouillée.

Console

Pour modifier l'état de la politique d'audit de votre flux d'activité

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Pour Actions, sélectionnez Modify database activity stream (Modifier le flux d'activité de la base de données).

Le flux d'activité de modification de la base de données : ***name***une fenêtre apparaît, où *name* est votre RDS instance.

4. Choisissez l'une des options suivantes :

   Locked (Verrouillée)

   Lorsque vous verrouillez votre politique d'audit, elle passe en lecture seule. Vous ne pouvez pas modifier votre politique d'audit à moins de déverrouiller la politique ou d'arrêter le flux d'activité.

   Unlocked (Débloqué)

   Lorsque vous déverrouillez votre politique d'audit, elle passe en lecture/écriture. Vous pouvez modifier votre politique d'audit pendant que le flux d'activité est lancé.

5. Sélectionnez Modify DB activity stream (Modifier le flux d'activité de la base de données).

   L'état de la base de RDS données Amazon du de bases de données indique Configuration du flux d'activité.

6. (Facultatif) Choisissez le lien de l'instance de base de données. Choisissez ensuite l'onglet Configuration.

   Le champ Audit policy status (Statut de la politique d'audit) affiche l'une des valeurs suivantes :

   - Locked (Verrouillée)
   - Unlocked (Débloqué)
   - Locking policy (Politique de verrouillage)
   - Unlocking policy (Politique de déverrouillage)

AWS CLI

Pour modifier l'état du flux d'activité de l'instance de base de données, utilisez la modify-activity-stream AWS CLI commande.

Option	Obligatoire ?	Description
`--resource-arn` *my-instance-ARN*	Oui	Le nom de ressource Amazon (ARN) de votre instance de RDS base de données.
`--audit-policy-state`	Non	Le nouvel état de la politique d'audit pour le flux d'activité de la base de données sur votre instance : `locked` ou `unlocked`.

L'exemple suivant déverrouille la politique d'audit pour le flux d'activité démarré le *my-instance-ARN*.

Dans Linux, macOS, ou Unix:

```
aws rds modify-activity-stream \
    --resource-arn my-instance-ARN \
    --audit-policy-state unlocked
```

Dans Windows:

```
aws rds modify-activity-stream ^
    --resource-arn my-instance-ARN ^
    --audit-policy-state unlocked
```

L'exemple suivant décrit l'instance *my-instance*. L'échantillon de sortie partiel indique que la politique d'audit est déverrouillée.

```
aws rds describe-db-instances --db-instance-identifier my-instance

{
    "DBInstances": [
        {
            ...
            "Engine": "oracle-ee",
            ...
            "ActivityStreamStatus": "started",
            "ActivityStreamKmsKeyId": "ab12345e-1111-2bc3-12a3-ab1cd12345e",
            "ActivityStreamKinesisStreamName": "aws-rds-das-db-
AB1CDEFG23GHIJK4LMNOPQRST",
```

```
            "ActivityStreamMode": "async",
            "ActivityStreamEngineNativeAuditFieldsIncluded": true,
            "ActivityStreamPolicyStatus": "unlocked",
            ...
        }
    ]
}
```

RDS API

Pour modifier l'état de la politique du flux d'activité de votre base de données, utilisez l'ModifyActivityStreamopération.

Appelez l'action avec les paramètres ci-dessous :

- `AuditPolicyState`
- `ResourceArn`

Obtention de l'état d'un flux d'activité de base de données

Vous pouvez obtenir le statut d'un flux d'activité pour votre instance de base de données Amazon RDS en utilisant la console ou AWS CLI.

Console

Obtention de l'état d'un flux d'activité de base de données

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis le lien du instance de base de données.
3. Choisissez l'onglet Configuration et cochez l'option Flux d'activité de base de données pour obtenir l'état.

AWS CLI

Vous pouvez obtenir la configuration du flux d'activité pour une instance de base de données en réponse à une demande CLI describe-db-instances.

L'exemple suivant décrit *`my-instance`*.

```
aws rds --region my-region describe-db-instances --db-instance-identifier my-db
```

Voici un exemple de réponse JSON. Les champs suivants s'affichent :

- `ActivityStreamKinesisStreamName`
- `ActivityStreamKmsKeyId`
- `ActivityStreamStatus`
- `ActivityStreamMode`
- `ActivityStreamPolicyStatus`

```
{
    "DBInstances": [
        {
            ...
            "Engine": "oracle-ee",
            ...
            "ActivityStreamStatus": "starting",
            "ActivityStreamKmsKeyId": "ab12345e-1111-2bc3-12a3-ab1cd12345e",
            "ActivityStreamKinesisStreamName": "aws-rds-das-db-
AB1CDEFG23GHIJK4LMNOPQRST",
            "ActivityStreamMode": "async",
            "ActivityStreamEngineNativeAuditFieldsIncluded": true,
            "ActivityStreamPolicyStatus": locked",
            ...
        }
    ]
}
```

API RDS

Vous pouvez obtenir la configuration du flux d'activité pour une base de données en réponse à une opération DescribeDBInstances.

Arrêt d'un flux d'activité de base de données

Vous pouvez arrêter un flux d'activité à partir de la console ou d AWS CLI.

Si vous supprimez votre instance de base de données Amazon RDS, le flux d'activité est arrêté et le flux Amazon Kinesis sous-jacent est supprimé automatiquement.

Console

Pour désactiver un flux d'activité

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez une base de données pour lequel vous souhaitez arrêter le flux d'activité de base de données.
4. Pour Actions, choisissez Stop activity stream (Arrêter le flux d'activité). La fenêtre Database Activity Stream (Flux d'activité de base de données) apparaît.

    a. Choisissez Immédiatement.

        Lorsque vous choisissez Immédiatement, le instance RDSredémarre tout de suite. Si vous choisissez Pendant la prochaine fenêtre de maintenance, le instance RDS ne redémarre pas tout de suite. Dans ce cas, le flux d'activité de base de données ne s'arrête pas avant la prochaine fenêtre de maintenance.

    b. Choisissez Continuer.

AWS CLI

Pour arrêter les flux d'activité de base de données pour de base de données (votre base de données), configurez l' de base de données à l'aide de la AWS CLI commande stop-activity-stream. Identifiez la région AWS pour le instance de base de données avec le paramètre `--region`. Le paramètre `--apply-immediately` est facultatif.

Pour LinuxmacOS, ou Unix :

```
aws rds --region MY_REGION \
    stop-activity-stream \
    --resource-arn MY_DB_ARN \
    --apply-immediately
```

Dans Windows :

```
aws rds --region MY_REGION ^
    stop-activity-stream ^
    --resource-arn MY_DB_ARN ^
    --apply-immediately
```

API RDS

Pour arrêter les flux d'activité de base de données pour de base de données (votre base de données), configurez l'instance de base de données du à l'aide de l'StopActivityStreamopération. Identifiez la région AWS pour le instance de base de données avec le paramètre `Region`. Le paramètre `ApplyImmediately` est facultatif.

Surveillance des flux d'activité de base de données

Les flux d'activité de base de données surveillent et rapportent les activités. Le flux d'activité est collecté et transmis à Amazon Kinesis. Depuis Kinesis, vous pouvez surveiller le flux d'activité ou d'autres services et applications peuvent utiliser le flux d'activité pour une analyse plus approfondie. Vous pouvez trouver le nom du flux Kinesis sous-jacent à l'aide de la AWS CLI commande ou de l'RDSAPIopération.

Amazon RDS gère le flux Kinesis pour vous comme suit :

- Amazon RDS crée automatiquement le flux Kinesis avec une période de rétention de 24 heures.
- Amazon RDS redimensionne le flux Kinesis si nécessaire.
- Si vous arrêtez le flux d'activité de base de données ou supprimez de base de données ou l'instance de base de données, Amazon RDS supprime le flux Kinesis.

Les catégories d'activité suivantes sont surveillées et incluses dans le journal d'audit de flux d'activité :

- SQLcommandes — Toutes les SQL commandes sont auditées, ainsi que des instructions préparées, des fonctions intégrées et des fonctions dans PL/SQL. Les appels aux procédures stockées sont vérifiés. Toutes SQL les déclarations émises dans le cadre de procédures ou de fonctions stockées sont également auditées.
- Autres informations de base de données — L'activité surveillée inclut l'SQLinstruction complète, le nombre de lignes affectées provenant des DML commandes, les objets consultés et le nom unique de la base de données. Les flux d'activité de base de données surveillent également les variables de liaison et les paramètres des procédures stockées.

> **Important**
>
> Le SQL texte complet de chaque déclaration est visible dans le journal d'audit du flux d'activité, y compris les données sensibles. Toutefois, les mots de passe des utilisateurs de

base de données sont supprimés si Oracle peut les déterminer à partir du contexte, comme dans l'SQLinstruction suivante.

```
ALTER ROLE role-name WITH password
```

- Informations de connexion – L'activité surveillée inclut les informations de session et de réseau, l'ID de processus serveur et les codes de sortie.

Si un flux d'activité échoue lors de la surveillance de votre instance de base de données, vous en êtes averti par le biais d'RDSévénements.

Dans les sections suivantes, vous pouvez accéder aux flux d'activité des bases de données, les auditer et les traiter.

Rubriques

- Accès à un flux d'activité depuis Amazon Kinesis
- Contenu du journal d'audit et exemples de flux d'activité de base de données
- databaseActivityEventAfficher le tableau JSON pour les flux d'activité de base de données
- Traitement d'un flux d'activité de base de données à l'aide du AWS SDK

Accès à un flux d'activité depuis Amazon Kinesis

Lorsque vous activez un flux d'activité pour une base de données, un flux Kinesis est créé pour vous. Depuis Kinesis, vous pouvez surveiller l'activité de votre base de données en temps réel. Pour effectuer des analyses plus poussées de l'activité de base de données, vous pouvez connecter votre flux Kinesis à des applications grand public. Vous pouvez également connecter le flux à des applications de gestion de la conformité telles que des SecureSphere bases de données d'Imperva.

Vous pouvez accéder à votre flux Kinesis depuis la RDS console ou depuis la console Kinesis.

Pour accéder à un flux d'activité depuis Kinesis à l'aide de la console RDS

1. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l' de bases de données sur laquelle vous avez démarré un flux d'activité.
4. Choisissez Configuration.

5. Sous Database activity stream (Flux d'activité de la base de données), choisissez le lien sous Kinesis stream (Flux Kinesis).
6. Dans la console Kinesis, choisissez Monitoring (Surveillance) pour commencer à observer l'activité de la base de données.

Pour accéder à un flux d'activité depuis Kinesis avec la console Kinesis

1. Ouvrez la console Kinesis à l'adresse https://console.aws.amazon.com/kinesis.
2. Choisissez votre flux d'activité dans la liste des flux Kinesis.

   Le nom d'un flux d'activité comprend le préfixe `aws-rds-das-db-` suivi de l'ID de ressource de la base de données. Voici un exemple.

   ```
   aws-rds-das-db-NHVOV4PCLWHGF52NP
   ```

   Pour utiliser la RDS console Amazon afin de trouver l'ID de ressource de base de données et de la base de données, choisissez votre de base de données dans la liste des bases de données, puis cliquez sur l'onglet Configuration.

   AWS CLI Pour trouver le nom complet du flux Kinesis d'un flux d'activité, utilisez une CLIrequête et notez la valeur de `ActivityStreamKinesisStreamName` dans la réponse.
3. Choisissez Surveillance pour commencer à observer l'activité de base de données.

Pour plus d'informations sur l'utilisation d'Amazon Kinesis, consultez la section En quoi consiste le service Amazon Kinesis Data Streams ?.

Contenu du journal d'audit et exemples de flux d'activité de base de données

Les événements surveillés sont représentés dans le flux d'activité de base de données sous la forme de chaînes JSON. La structure se compose d'un objet JSON contenant un `DatabaseActivityMonitoringRecord`, qui contient lui-même un tableau des événements d'activité `databaseActivityEventList`.

> **Note**
>
> Pour les flux d'activité de base de données, le tableau `paramList` JSON n'inclut pas les valeurs nulles des applications Hibernate.

Rubriques

- Exemples de journaux d'audit de flux d'activité
- DatabaseActivityMonitoringRecordsObjet JSON
- databaseActivityEvents Objet JSON

Exemples de journaux d'audit de flux d'activité

Vous trouverez ci-après des exemples de journaux d'audits JSON déchiffrés d'enregistrements d'événements d'activité.

Example Enregistrement d'événement d'activité d'une instruction CONNECT SQL

L'enregistrement d'événement d'activité suivant indique une connexion à l'aide d'une instruction SQL `CONNECT` (`command`) par un client léger JDBC (`clientApplication`) pour votre base de données Oracle.


```
{
    "class": "Standard",
    "clientApplication": "JDBC Thin Client",
    "command": "LOGON",
    "commandText": null,
    "dbid": "0123456789",
    "databaseName": "ORCL",
    "dbProtocol": "oracle",
    "dbUserName": "TEST",
    "endTime": null,
    "errorMessage": null,
    "exitCode": 0,
    "logTime": "2021-01-15 00:15:36.233787",
    "netProtocol": "tcp",
    "objectName": null,
    "objectType": null,
    "paramList": [],
    "pid": 17904,
    "remoteHost": "123.456.789.012",
    "remotePort": "25440",
    "rowCount": null,
    "serverHost": "987.654.321.098",
    "serverType": "oracle",
    "serverVersion": "19.0.0.0.ru-2020-01.rur-2020-01.r1.EE.3",
    "serviceName": "oracle-ee",
```

```
    "sessionId": 987654321,
    "startTime": null,
    "statementId": 1,
    "substatementId": null,
    "transactionId": "0000000000000000",
    "engineNativeAuditFields": {
        "UNIFIED_AUDIT_POLICIES": "TEST_POL_EVERYTHING",
        "FGA_POLICY_NAME": null,
        "DV_OBJECT_STATUS": null,
        "SYSTEM_PRIVILEGE_USED": "CREATE SESSION",
        "OLS_LABEL_COMPONENT_TYPE": null,
        "XS_SESSIONID": null,
        "ADDITIONAL_INFO": null,
        "INSTANCE_ID": 1,
        "DBID": 123456789
        "DV_COMMENT": null,
        "RMAN_SESSION_STAMP": null,
        "NEW_NAME": null,
        "DV_ACTION_NAME": null,
        "OLS_PROGRAM_UNIT_NAME": null,
        "OLS_STRING_LABEL": null,
        "RMAN_SESSION_RECID": null,
        "OBJECT_PRIVILEGES": null,
        "OLS_OLD_VALUE": null,
        "XS_TARGET_PRINCIPAL_NAME": null,
        "XS_NS_ATTRIBUTE": null,
        "XS_NS_NAME": null,
        "DBLINK_INFO": null,
        "AUTHENTICATION_TYPE": "(TYPE\u003d(DATABASE));(CLIENT ADDRESS\u003d((ADDRESS
\u003d(PROTOCOL\u003dtcp)(HOST\u003d205.251.233.183)(PORT\u003d25440))));",
        "OBJECT_EDITION": null,
        "OLS_PRIVILEGES_GRANTED": null,
        "EXCLUDED_USER": null,
        "DV_ACTION_OBJECT_NAME": null,
        "OLS_LABEL_COMPONENT_NAME": null,
        "EXCLUDED_SCHEMA": null,
        "DP_TEXT_PARAMETERS1": null,
        "XS_USER_NAME": null,
        "XS_ENABLED_ROLE": null,
        "XS_NS_ATTRIBUTE_NEW_VAL": null,
        "DIRECT_PATH_NUM_COLUMNS_LOADED": null,
        "AUDIT_OPTION": null,
        "DV_EXTENDED_ACTION_CODE": null,
        "XS_PACKAGE_NAME": null,
```

```
        "OLS_NEW_VALUE": null,
        "DV_RETURN_CODE": null,
        "XS_CALLBACK_EVENT_TYPE": null,
        "USERHOST": "a1b2c3d4e5f6.amazon.com",
        "GLOBAL_USERID": null,
        "CLIENT_IDENTIFIER": null,
        "RMAN_OPERATION": null,
        "TERMINAL": "unknown",
        "OS_USERNAME": "sumepate",
        "OLS_MAX_READ_LABEL": null,
        "XS_PROXY_USER_NAME": null,
        "XS_DATASEC_POLICY_NAME": null,
        "DV_FACTOR_CONTEXT": null,
        "OLS_MAX_WRITE_LABEL": null,
        "OLS_PARENT_GROUP_NAME": null,
        "EXCLUDED_OBJECT": null,
        "DV_RULE_SET_NAME": null,
        "EXTERNAL_USERID": null,
        "EXECUTION_ID": null,
        "ROLE": null,
        "PROXY_SESSIONID": 0,
        "DP_BOOLEAN_PARAMETERS1": null,
        "OLS_POLICY_NAME": null,
        "OLS_GRANTEE": null,
        "OLS_MIN_WRITE_LABEL": null,
        "APPLICATION_CONTEXTS": null,
        "XS_SCHEMA_NAME": null,
        "DV_GRANTEE": null,
        "XS_COOKIE": null,
        "DBPROXY_USERNAME": null,
        "DV_ACTION_CODE": null,
        "OLS_PRIVILEGES_USED": null,
        "RMAN_DEVICE_TYPE": null,
        "XS_NS_ATTRIBUTE_OLD_VAL": null,
        "TARGET_USER": null,
        "XS_ENTITY_TYPE": null,
        "ENTRY_ID": 1,
        "XS_PROCEDURE_NAME": null,
        "XS_INACTIVITY_TIMEOUT": null,
        "RMAN_OBJECT_TYPE": null,
        "SYSTEM_PRIVILEGE": null,
        "NEW_SCHEMA": null,
        "SCN": 5124715
    }
```

```
}
```

L'enregistrement d'événement d'activité suivant indique un échec de connexion pour votre base de données SQL Server.

```
{
    "type": "DatabaseActivityMonitoringRecord",
    "clusterId": "",
    "instanceId": "db-4JCWQLUZVFYP7DIWP6JVQ77O3Q",
    "databaseActivityEventList": [
        {
            "class": "LOGIN",
            "clientApplication": "Microsoft SQL Server Management Studio",
            "command": "LOGIN FAILED",
            "commandText": "Login failed for user 'test'. Reason: Password did not
 match that for the login provided. [CLIENT: local-machine]",
            "databaseName": "",
            "dbProtocol": "SQLSERVER",
            "dbUserName": "test",
            "endTime": null,
            "errorMessage": null,
            "exitCode": 0,
            "logTime": "2022-10-06 21:34:42.7113072+00",
            "netProtocol": null,
            "objectName": "",
            "objectType": "LOGIN",
            "paramList": null,
            "pid": null,
            "remoteHost": "local machine",
            "remotePort": null,
            "rowCount": 0,
            "serverHost": "172.31.30.159",
            "serverType": "SQLSERVER",
            "serverVersion": "15.00.4073.23.v1.R1",
            "serviceName": "sqlserver-ee",
            "sessionId": 0,
            "startTime": null,
            "statementId": "0x1eb0d1808d34a94b9d3dcf5432750f02",
            "substatementId": 1,
            "transactionId": "0",
            "type": "record",
            "engineNativeAuditFields": {
                "target_database_principal_id": 0,
```

```
                "target_server_principal_id": 0,
                "target_database_principal_name": "",
                "server_principal_id": 0,
                "user_defined_information": "",
                "response_rows": 0,
                "database_principal_name": "",
                "target_server_principal_name": "",
                "schema_name": "",
                "is_column_permission": false,
                "object_id": 0,
                "server_instance_name": "EC2AMAZ-NFUJJNO",
                "target_server_principal_sid": null,
                "additional_information": "<action_info "xmlns=\"http://
schemas.microsoft.com/sqlserver/2008/sqlaudit_data\"><pooled_connection>0</
pooled_connection><error>0x00004818</error><state>8</state><address>local machine</
address><PasswordFirstNibbleHash>B</PasswordFirstNibbleHash></action_info>"-->,
                "duration_milliseconds": 0,
                "permission_bitmask": "0x00000000000000000000000000000000",
                "data_sensitivity_information": "",
                "session_server_principal_name": "",
                "connection_id": "98B4F537-0F82-49E3-AB08-B9D33B5893EF",
                "audit_schema_version": 1,
                "database_principal_id": 0,
                "server_principal_sid": null,
                "user_defined_event_id": 0,
                "host_name": "EC2AMAZ-NFUJJNO"
            }
        }
    ]
}
```

> **Note**
>
> Si un flux d'activité de base de données n'est pas activé, le dernier champ du document JSON est `"engineNativeAuditFields": { }`.

Example Registre d'événement d'activité d'une instruction CREATE TABLE

L'exemple suivant montre un événement `CREATE TABLE` pour votre base de données Oracle.

```
{
```

```
    "class": "Standard",
    "clientApplication": "sqlplus@ip-12-34-5-678 (TNS V1-V3)",
    "command": "CREATE TABLE",
    "commandText": "CREATE TABLE persons(\n    person_id NUMBER GENERATED BY DEFAULT AS
 IDENTITY,\n    first_name VARCHAR2(50) NOT NULL,\n    last_name VARCHAR2(50) NOT NULL,
\n    PRIMARY KEY(person_id)\n)",
    "dbid": "0123456789",
    "databaseName": "ORCL",
    "dbProtocol": "oracle",
    "dbUserName": "TEST",
    "endTime": null,
    "errorMessage": null,
    "exitCode": 0,
    "logTime": "2021-01-15 00:22:49.535239",
    "netProtocol": "beq",
    "objectName": "PERSONS",
    "objectType": "TEST",
    "paramList": [],
    "pid": 17687,
    "remoteHost": "123.456.789.0",
    "remotePort": null,
    "rowCount": null,
    "serverHost": "987.654.321.01",
    "serverType": "oracle",
    "serverVersion": "19.0.0.0.ru-2020-01.rur-2020-01.r1.EE.3",
    "serviceName": "oracle-ee",
    "sessionId": 1234567890,
    "startTime": null,
    "statementId": 43,
    "substatementId": null,
    "transactionId": "090011007F0D0000",
    "engineNativeAuditFields": {
        "UNIFIED_AUDIT_POLICIES": "TEST_POL_EVERYTHING",
        "FGA_POLICY_NAME": null,
        "DV_OBJECT_STATUS": null,
        "SYSTEM_PRIVILEGE_USED": "CREATE SEQUENCE, CREATE TABLE",
        "OLS_LABEL_COMPONENT_TYPE": null,
        "XS_SESSIONID": null,
        "ADDITIONAL_INFO": null,
        "INSTANCE_ID": 1,
        "DV_COMMENT": null,
        "RMAN_SESSION_STAMP": null,
        "NEW_NAME": null,
        "DV_ACTION_NAME": null,
```

```
        "OLS_PROGRAM_UNIT_NAME": null,
        "OLS_STRING_LABEL": null,
        "RMAN_SESSION_RECID": null,
        "OBJECT_PRIVILEGES": null,
        "OLS_OLD_VALUE": null,
        "XS_TARGET_PRINCIPAL_NAME": null,
        "XS_NS_ATTRIBUTE": null,
        "XS_NS_NAME": null,
        "DBLINK_INFO": null,
        "AUTHENTICATION_TYPE": "(TYPE\u003d(DATABASE));(CLIENT ADDRESS\u003d((PROTOCOL
\u003dbeq)(HOST\u003d123.456.789.0)));",
        "OBJECT_EDITION": null,
        "OLS_PRIVILEGES_GRANTED": null,
        "EXCLUDED_USER": null,
        "DV_ACTION_OBJECT_NAME": null,
        "OLS_LABEL_COMPONENT_NAME": null,
        "EXCLUDED_SCHEMA": null,
        "DP_TEXT_PARAMETERS1": null,
        "XS_USER_NAME": null,
        "XS_ENABLED_ROLE": null,
        "XS_NS_ATTRIBUTE_NEW_VAL": null,
        "DIRECT_PATH_NUM_COLUMNS_LOADED": null,
        "AUDIT_OPTION": null,
        "DV_EXTENDED_ACTION_CODE": null,
        "XS_PACKAGE_NAME": null,
        "OLS_NEW_VALUE": null,
        "DV_RETURN_CODE": null,
        "XS_CALLBACK_EVENT_TYPE": null,
        "USERHOST": "ip-10-13-0-122",
        "GLOBAL_USERID": null,
        "CLIENT_IDENTIFIER": null,
        "RMAN_OPERATION": null,
        "TERMINAL": "pts/1",
        "OS_USERNAME": "rdsdb",
        "OLS_MAX_READ_LABEL": null,
        "XS_PROXY_USER_NAME": null,
        "XS_DATASEC_POLICY_NAME": null,
        "DV_FACTOR_CONTEXT": null,
        "OLS_MAX_WRITE_LABEL": null,
        "OLS_PARENT_GROUP_NAME": null,
        "EXCLUDED_OBJECT": null,
        "DV_RULE_SET_NAME": null,
        "EXTERNAL_USERID": null,
        "EXECUTION_ID": null,
```

```
        "ROLE": null,
        "PROXY_SESSIONID": 0,
        "DP_BOOLEAN_PARAMETERS1": null,
        "OLS_POLICY_NAME": null,
        "OLS_GRANTEE": null,
        "OLS_MIN_WRITE_LABEL": null,
        "APPLICATION_CONTEXTS": null,
        "XS_SCHEMA_NAME": null,
        "DV_GRANTEE": null,
        "XS_COOKIE": null,
        "DBPROXY_USERNAME": null,
        "DV_ACTION_CODE": null,
        "OLS_PRIVILEGES_USED": null,
        "RMAN_DEVICE_TYPE": null,
        "XS_NS_ATTRIBUTE_OLD_VAL": null,
        "TARGET_USER": null,
        "XS_ENTITY_TYPE": null,
        "ENTRY_ID": 12,
        "XS_PROCEDURE_NAME": null,
        "XS_INACTIVITY_TIMEOUT": null,
        "RMAN_OBJECT_TYPE": null,
        "SYSTEM_PRIVILEGE": null,
        "NEW_SCHEMA": null,
        "SCN": 5133083
    }
}
```

L'exemple suivant montre un événement `CREATE TABLE` pour votre base de données SQL Server.

```
{
    "type": "DatabaseActivityMonitoringRecord",
    "clusterId": "",
    "instanceId": "db-4JCWQLUZVFYP7DIWP6JVQ77O3Q",
    "databaseActivityEventList": [
        {
            "class": "SCHEMA",
            "clientApplication": "Microsoft SQL Server Management Studio - Query",
            "command": "ALTER",
            "commandText": "Create table [testDB].[dbo].[TestTable2](\r\ntextA
 varchar(6000),\r\n    textB varchar(6000)\r\n)",
            "databaseName": "testDB",
            "dbProtocol": "SQLSERVER",
            "dbUserName": "test",
```

```
            "endTime": null,
            "errorMessage": null,
            "exitCode": 1,
            "logTime": "2022-10-06 21:44:38.4120677+00",
            "netProtocol": null,
            "objectName": "dbo",
            "objectType": "SCHEMA",
            "paramList": null,
            "pid": null,
            "remoteHost": "local machine",
            "remotePort": null,
            "rowCount": 0,
            "serverHost": "172.31.30.159",
            "serverType": "SQLSERVER",
            "serverVersion": "15.00.4073.23.v1.R1",
            "serviceName": "sqlserver-ee",
            "sessionId": 84,
            "startTime": null,
            "statementId": "0x5178d33d56e95e419558b9607158a5bd",
            "substatementId": 1,
            "transactionId": "4561864",
            "type": "record",
            "engineNativeAuditFields": {
                "target_database_principal_id": 0,
                "target_server_principal_id": 0,
                "target_database_principal_name": "",
                "server_principal_id": 2,
                "user_defined_information": "",
                "response_rows": 0,
                "database_principal_name": "dbo",
                "target_server_principal_name": "",
                "schema_name": "",
                "is_column_permission": false,
                "object_id": 1,
                "server_instance_name": "EC2AMAZ-NFUJJNO",
                "target_server_principal_sid": null,
                "additional_information": "",
                "duration_milliseconds": 0,
                "permission_bitmask": "0x00000000000000000000000000000000",
                "data_sensitivity_information": "",
                "session_server_principal_name": "test",
                "connection_id": "EE1FE3FD-EF2C-41FD-AF45-9051E0CD983A",
                "audit_schema_version": 1,
                "database_principal_id": 1,
```

```
                "server_principal_sid":
 "0x01050000000000051500000bdc2795e2d0717901ba6998cf4010000",
                "user_defined_event_id": 0,
                "host_name": "EC2AMAZ-NFUJJNO"
            }
        }
    ]
}
```

Example Registre d'événement d'activité d'une instruction SELECT

L'exemple suivant montre un événement SELECT pour votre base de données Oracle.

```
{
    "class": "Standard",
    "clientApplication": "sqlplus@ip-12-34-5-678 (TNS V1-V3)",
    "command": "SELECT",
    "commandText": "select count(*) from persons",
    "databaseName": "1234567890",
    "dbProtocol": "oracle",
    "dbUserName": "TEST",
    "endTime": null,
    "errorMessage": null,
    "exitCode": 0,
    "logTime": "2021-01-15 00:25:18.850375",
    "netProtocol": "beq",
    "objectName": "PERSONS",
    "objectType": "TEST",
    "paramList": [],
    "pid": 17687,
    "remoteHost": "123.456.789.0",
    "remotePort": null,
    "rowCount": null,
    "serverHost": "987.654.321.09",
    "serverType": "oracle",
    "serverVersion": "19.0.0.0.ru-2020-01.rur-2020-01.r1.EE.3",
    "serviceName": "oracle-ee",
    "sessionId": 1080639707,
    "startTime": null,
    "statementId": 44,
    "substatementId": null,
    "transactionId": null,
    "engineNativeAuditFields": {
        "UNIFIED_AUDIT_POLICIES": "TEST_POL_EVERYTHING",
```

```
        "FGA_POLICY_NAME": null,
        "DV_OBJECT_STATUS": null,
        "SYSTEM_PRIVILEGE_USED": null,
        "OLS_LABEL_COMPONENT_TYPE": null,
        "XS_SESSIONID": null,
        "ADDITIONAL_INFO": null,
        "INSTANCE_ID": 1,
        "DV_COMMENT": null,
        "RMAN_SESSION_STAMP": null,
        "NEW_NAME": null,
        "DV_ACTION_NAME": null,
        "OLS_PROGRAM_UNIT_NAME": null,
        "OLS_STRING_LABEL": null,
        "RMAN_SESSION_RECID": null,
        "OBJECT_PRIVILEGES": null,
        "OLS_OLD_VALUE": null,
        "XS_TARGET_PRINCIPAL_NAME": null,
        "XS_NS_ATTRIBUTE": null,
        "XS_NS_NAME": null,
        "DBLINK_INFO": null,
        "AUTHENTICATION_TYPE": "(TYPE\u003d(DATABASE));(CLIENT ADDRESS\u003d((PROTOCOL
\u003dbeq)(HOST\u003d123.456.789.0)));",
        "OBJECT_EDITION": null,
        "OLS_PRIVILEGES_GRANTED": null,
        "EXCLUDED_USER": null,
        "DV_ACTION_OBJECT_NAME": null,
        "OLS_LABEL_COMPONENT_NAME": null,
        "EXCLUDED_SCHEMA": null,
        "DP_TEXT_PARAMETERS1": null,
        "XS_USER_NAME": null,
        "XS_ENABLED_ROLE": null,
        "XS_NS_ATTRIBUTE_NEW_VAL": null,
        "DIRECT_PATH_NUM_COLUMNS_LOADED": null,
        "AUDIT_OPTION": null,
        "DV_EXTENDED_ACTION_CODE": null,
        "XS_PACKAGE_NAME": null,
        "OLS_NEW_VALUE": null,
        "DV_RETURN_CODE": null,
        "XS_CALLBACK_EVENT_TYPE": null,
        "USERHOST": "ip-12-34-5-678",
        "GLOBAL_USERID": null,
        "CLIENT_IDENTIFIER": null,
        "RMAN_OPERATION": null,
        "TERMINAL": "pts/1",
```

```
        "OS_USERNAME": "rdsdb",
        "OLS_MAX_READ_LABEL": null,
        "XS_PROXY_USER_NAME": null,
        "XS_DATASEC_POLICY_NAME": null,
        "DV_FACTOR_CONTEXT": null,
        "OLS_MAX_WRITE_LABEL": null,
        "OLS_PARENT_GROUP_NAME": null,
        "EXCLUDED_OBJECT": null,
        "DV_RULE_SET_NAME": null,
        "EXTERNAL_USERID": null,
        "EXECUTION_ID": null,
        "ROLE": null,
        "PROXY_SESSIONID": 0,
        "DP_BOOLEAN_PARAMETERS1": null,
        "OLS_POLICY_NAME": null,
        "OLS_GRANTEE": null,
        "OLS_MIN_WRITE_LABEL": null,
        "APPLICATION_CONTEXTS": null,
        "XS_SCHEMA_NAME": null,
        "DV_GRANTEE": null,
        "XS_COOKIE": null,
        "DBPROXY_USERNAME": null,
        "DV_ACTION_CODE": null,
        "OLS_PRIVILEGES_USED": null,
        "RMAN_DEVICE_TYPE": null,
        "XS_NS_ATTRIBUTE_OLD_VAL": null,
        "TARGET_USER": null,
        "XS_ENTITY_TYPE": null,
        "ENTRY_ID": 13,
        "XS_PROCEDURE_NAME": null,
        "XS_INACTIVITY_TIMEOUT": null,
        "RMAN_OBJECT_TYPE": null,
        "SYSTEM_PRIVILEGE": null,
        "NEW_SCHEMA": null,
        "SCN": 5136972
    }
}
```

L'exemple suivant montre un événement SELECT pour votre base de données SQL Server.

```
{
    "type": "DatabaseActivityMonitoringRecord",
    "clusterId": "",
```

```
    "instanceId": "db-4JCWQLUZVFYP7DIWP6JVQ77O3Q",
    "databaseActivityEventList": [
        {
            "class": "TABLE",
            "clientApplication": "Microsoft SQL Server Management Studio - Query",
            "command": "SELECT",
            "commandText": "select * from [testDB].[dbo].[TestTable]",
            "databaseName": "testDB",
            "dbProtocol": "SQLSERVER",
            "dbUserName": "test",
            "endTime": null,
            "errorMessage": null,
            "exitCode": 1,
            "logTime": "2022-10-06 21:24:59.9422268+00",
            "netProtocol": null,
            "objectName": "TestTable",
            "objectType": "TABLE",
            "paramList": null,
            "pid": null,
            "remoteHost": "local machine",
            "remotePort": null,
            "rowCount": 0,
            "serverHost": "172.31.30.159",
            "serverType": "SQLSERVER",
            "serverVersion": "15.00.4073.23.v1.R1",
            "serviceName": "sqlserver-ee",
            "sessionId": 62,
            "startTime": null,
            "statementId": "0x03baed90412f564fad640ebe51f89b99",
            "substatementId": 1,
            "transactionId": "4532935",
            "type": "record",
            "engineNativeAuditFields": {
                "target_database_principal_id": 0,
                "target_server_principal_id": 0,
                "target_database_principal_name": "",
                "server_principal_id": 2,
                "user_defined_information": "",
                "response_rows": 0,
                "database_principal_name": "dbo",
                "target_server_principal_name": "",
                "schema_name": "dbo",
                "is_column_permission": true,
                "object_id": 581577110,
```

```
                "server_instance_name": "EC2AMAZ-NFUJJNO",
                "target_server_principal_sid": null,
                "additional_information": "",
                "duration_milliseconds": 0,
                "permission_bitmask": "0x00000000000000000000000000000001",
                "data_sensitivity_information": "",
                "session_server_principal_name": "test",
                "connection_id": "AD3A5084-FB83-45C1-8334-E923459A8109",
                "audit_schema_version": 1,
                "database_principal_id": 1,
                "server_principal_sid":
 "0x0105000000000005150000000bdc2795e2d0717901ba6998cf4010000",
                "user_defined_event_id": 0,
                "host_name": "EC2AMAZ-NFUJJNO"
            }
        }
    ]
}
```

DatabaseActivityMonitoringRecordsObjet JSON

Les enregistrements d'événement d'activité de base de données se trouvent dans un objet JSON qui contient les informations suivantes.

Champ JSON	Type de données	Description
`type`	chaîne	Type de l'enregistrement JSON. La valeur est `DatabaseActivityMonitoringRecords` .
`version`	chaîne	Version des enregistrements de surveillance d'activité de base de données. La base de données Oracle utilise la version 1.3 et SQL Server utilise la version 1.4. Ces versions du moteur introduisent l'objet JSON `engineNativeAuditFields` .

Champ JSON	Type de données	Description
databaseActivityEvents	chaîne	Objet JSON qui contient les événements d'activité.
key	chaîne	Clé de chiffrement que vous utilisez pour déchiffrer databaseActivityEventTableau JSON de liste

databaseActivityEvents Objet JSON

L'objet JSON `databaseActivityEvents` contient les informations suivantes.

Champs de niveau supérieur dans l'enregistrement JSON

Chaque événement du journal d'audit est encapsulé dans un enregistrement au format JSON. Cet enregistrement contient les champs suivants.

type

Ce champ a toujours la valeur `DatabaseActivityMonitoringRecords`.

version ;

Ce champ représente la version du contrat ou du protocole de données de flux d'activité de base de données. Il définit les champs disponibles.

databaseActivityEvents

Chaîne chiffrée représentant un ou plusieurs événements d'activité. Elle est représentée sou la forme d'un tableau base64 octets. Lorsque vous déchiffrez la chaîne, le résultat est un enregistrement au format JSON avec des champs comme ceux des exemples de cette section.

key

Clé de données chiffrée utilisée pour chiffrer la chaîne `databaseActivityEvents`. Il s'agit du même AWS KMS key que celui que vous avez fourni lorsque vous avez démarré le flux d'activité de la base de données.

L'exemple suivant illustre le format de cet enregistrement.

```
{
  "type":"DatabaseActivityMonitoringRecords",
  "version":"1.3",
  "databaseActivityEvents":"encrypted audit records",
  "key":"encrypted key"
}
```

```
          "type":"DatabaseActivityMonitoringRecords",
          "version":"1.4",
          "databaseActivityEvents":"encrypted audit records",
          "key":"encrypted key"
```

Pour déchiffrer le contenu du champ `databaseActivityEvents`, procédez comme suit :

1. Déchiffrez la valeur dans le champ JSON `key` à l'aide de la clé KMS que vous avez fournie lors du démarrage du flux d'activité de base de données. Cette opération renvoie la clé de chiffrement des données en texte clair.
2. Décodez en base64 la valeur dans le champ JSON `databaseActivityEvents` pour obtenir le texte chiffré, au format binaire, de la charge utile d'audit.
3. Déchiffrez le chiffrement binaire avec la clé de chiffrement de données que vous avez décodée au cours de la première étape.
4. Décompressez la charge utile déchiffrée.
   - La charge utile chiffrée se trouve dans le champ `databaseActivityEvents`.
   - Le champ `databaseActivityEventList` contient un tableau d'enregistrements d'audits. Les champs `type` du tableau peuvent être `record` ou `heartbeat`.

L'enregistrement d'événement d'activité du journal d'audit est un objet JSON qui contient les informations suivantes.

Champ JSON	Type de données	Description
`type`	chaîne	Type de l'enregistrement JSON. La valeur est `DatabaseActivityMonitoringRecord` .

Champ JSON	Type de données	Description
`instanceId`	chaîne	Identificateur de ressource d'instance de base de données. Il correspond à l'attribut d'instance de base de données `DbiResourceId` .
databaseActivityEventTableau JSON de liste	chaîne	Tableau d'enregistrements d'audits d'activité ou de messages de pulsations.

databaseActivityEventAfficher le tableau JSON pour les flux d'activité de base de données

La charge utile du journal d'audit est un tableau JSON `databaseActivityEventList` chiffré. Ci-dessous, le tableau répertorie par ordre alphabétique les champs de chaque événement d'activité dans le tableau `DatabaseActivityEventList` déchiffré d'un journal d'audit.

Lorsque l'audit unifié est activé dans Oracle Database, les enregistrements d'audit sont renseignés dans cette nouvelle trace d'audit. La vue `UNIFIED_AUDIT_TRAIL` affiche les enregistrements d'audit sous forme de tableau en extrayant les enregistrements d'audit de la trace d'audit. Lorsque vous démarrez un flux d'activité de base de données, une colonne dans `UNIFIED_AUDIT_TRAIL` mappe à un champ dans le tableau `databaseActivityEventList`.

> **Important**
>
> Il se peut que la structure d'événement change. Il se peut qu'Amazon RDS ajoute de nouveaux champs aux événements d'activité à l'avenir. Dans les applications qui analysent les données JSON, assurez-vous que votre code peut ignorer ou prendre les mesures appropriées pour les noms de champs inconnus.

databaseActivityEventChamps de liste pour Amazon RDS for Oracle

Les `databaseActivityEventList` champs suivants concernent Amazon RDS for Oracle.

Champ	Type de donné	Source	Description
`class`	chaîn	Colonne `AUDIT_TYPE` dans `UNIFIED_AUDIT_TRAIL`	La classe d'un événement d'activité. Celle-ci correspond à la colonne `AUDIT_TYPE` dans la vue `UNIFIED_AUDIT_TRAIL`. Les valeurs valides pour Amazon RDS for Oracle sont les suivantes : • `Standard` • `FineGrainedAudit` • `XS` • `Database Vault` • `Label Security` • `RMAN_AUDIT` • `Datapump` • `Direct path API` Pour plus d'informations, consultez UNIFIED_AUDIT_TRAIL dans la documentation Oracle.
`clientApplication`	chaîn	`CLIENT_PROGRAM_NAME` dans `UNIFIED_AUDIT_TRAIL`	Application utilisée par le client pour se connecter, telle que signalée par le client. Le client n'a pas à fournir cette information, la valeur peut être « null ». Un exemple de

Champ	Type de donne	Source	Description
			valeur est `JDBC Thin Client`.
`command`	chaîn	Colonne `ACTION_NAME` dans `UNIFIED_AUDIT_TRAIL`	Nom de l'action exécutée par l'utilisateur. Pour comprendre l'action complète, lisez le nom de la commande et la valeur `AUDIT_TYPE` . Un exemple de valeur est `ALTER DATABASE`.
`commandText`	chaîn	Colonne `SQL_TEXT` dans `UNIFIED_AUDIT_TRAIL`	Instruction SQL associée à l'événement. Un exemple de valeur est `ALTER DATABASE BEGIN BACKUP`.
`databaseName`	chaîn	Colonne `NAME` dans `V$DATABASE`	Nom de la base de données.
`dbid`	nomb	Colonne `DBID` dans `UNIFIED_AUDIT_TRAIL`	Identificateur numérique de la base de données. Un exemple de valeur est `1559204751` .
`dbProtocol`	chaîn	N/A	Protocole de la base de données. Dans cette version bêta, la valeur est `oracle`.

Champ	Type de donne	Source	Description
`dbUserName`	chaîn	Colonne `DBUSERNAME` dans `UNIFIED_AUDIT_TRAIL`	Nom de l'utilisateur de la base de données dont les actions ont été auditées. Un exemple de valeur est `RDSADMIN`.
`endTime`	chaîn	N/A	Ce champ n'est pas utilisé pour RDS for Oracle. Sa valeur est toujours null.

Champ	Type de donne	Source	Description

`engineNativeAuditFields`	objet	`UNIFIED_AUDIT_TRAIL`	Par défaut, cet objet est vide. Lorsque vous démarrez le flux d'activité avec l'option `--engine-native-audit-fields-included` , cet objet inclut les colonnes et leurs valeurs suivantes :
			
			`ADDITIONAL_INFO`
			`APPLICATION_CONTEXTS`
			`AUDIT_OPTION`
			`AUTHENTICATION_TYPE`
			`CLIENT_IDENTIFIER`
			`CURRENT_USER`
			`DBLINK_INFO`
			`DBPROXY_USERNAME`
			`DIRECT_PATH_NUM_COLUMNS_LOADED`
			`DP_BOOLEAN_PARAMETERS1`
			`DP_TEXT_PARAMETERS1`
			`DV_ACTION_CODE`
			`DV_ACTION_NAME`
			`DV_ACTION_OBJECT_NAME`
			`DV_COMMENT`
			`DV_EXTENDED_ACTION_CODE`
			`DV_FACTOR_CONTEXT`
			`DV_GRANTEE`
			`DV_OBJECT_STATUS`
			`DV_RETURN_CODE`

Champ	Type de donne	Source	Description
			DV_RULE_SET_NAME ENTRY_ID EXCLUDED_OBJECT EXCLUDED_SCHEMA EXCLUDED_USER EXECUTION_ID EXTERNAL_USERID FGA_POLICY_NAME GLOBAL_USERID INSTANCE_ID KSACL_SER VICE_NAME KSACL_SOURCE_LOCA TION KSACL_USER_NAME NEW_NAME NEW_SCHEMA OBJECT_EDITION OBJECT_PRIVILEGES OLS_GRANTEE OLS_LABEL_COM PONENT_NAME OLS_LABEL_COMPO NENT_TYPE OLS_MAX_READ_LABEL OLS_MAX_WRITE_LA BEL OLS_MIN_WRITE_ LABEL OLS_NEW_VALUE OLS_OLD_VALUE OLS_PARENT_ GROUP_NAME OLS_POLICY_NAME OLS_PRIVILEGES_GRA NTED OLS_PRIVILEGE S_USED

Champ	Type de donne	Source	Description
			OLS_PROGRAM_UNIT_NAME OLS_STRING_LABEL OS_USERNAME PROTOCOL_ACTION_NAME PROTOCOL_MESSAGE PROTOCOL_RETURN_CODE PROTOCOL_SESSION_ID PROTOCOL_USERHOST PROXY_SESSIONID RLS_INFO RMAN_DEVICE_TYPE RMAN_OBJECT_TYPE RMAN_OPERATION RMAN_SESSION_RECID RMAN_SESSION_STAMP ROLE SCN SYSTEM_PRIVILEGE SYSTEM_PRIVILEGE_USED TARGET_USER TERMINAL UNIFIED_AUDIT_POLICIES USERHOST XS_CALLBACK_EVENT_TYPE XS_COOKIE XS_DATASEC_POLICY_NAME XS_ENABLED_ROLE XS_ENTITY_TYPE XS_INACTIVITY_TIMEOUT XS_NS_ATTRIBUTE

Champ	Type de donne	Source	Description
			`XS_NS_ATTRI BUTE_NEW_VAL` `XS_NS_ATTRIBUT E_OLD_VAL` `XS_NS_NAME` `XS_PACKAGE_NAME` `XS_PROCEDURE_NAME` `XS_PROXY_USER_NAME` `XS_SCHEMA_NAME` `XS_SESSIONID` `XS_TARGET_PRINC IPAL_NAME` `XS_USER_NAME` Pour plus d'informations, consultez UNIFIED_A UDIT_TRAIL dans la documentation Oracle Database.
`errorMessage`	chaîn	N/A	Ce champ n'est pas utilisé pour RDS for Oracle. Sa valeur est toujours null.
`exitCode`	nomb	Colonne `RETURN_CO DE` dans `UNIFIED_A UDIT_TRAIL`	Code d'erreur Oracle Database généré par l'action. Si l'action a réussi, la valeur est `0`.

Champ	Type de donne	Source	Description
logTime	chaîn	Colonne EVENT_TIMESTAMP_UTC dans UNIFIED_AUDIT_TRAIL	Horodatage de la création de l'entrée de trace d'audit. Un exemple de valeur est 2020-11-27 06:56:14.981404 .
netProtocol	chaîn	Colonne AUTHENTICATION_TYPE dans UNIFIED_AUDIT_TRAIL	Protocole de communication réseau. Un exemple de valeur est TCP.
objectName	chaîn	Colonne OBJECT_NAME dans UNIFIED_AUDIT_TRAIL	Nom de l'objet affecté par l'action. Un exemple de valeur est employees .
objectType	chaîn	Colonne OBJECT_SCHEMA dans UNIFIED_AUDIT_TRAIL	Nom de schéma de l'objet affecté par l'action. Un exemple de valeur est hr.
paramList	liste	Colonne SQL_BINDS dans UNIFIED_AUDIT_TRAIL	La liste des variables de liaison éventuelles associées à SQL_TEXT. Un exemple de valeur est parameter_1,parameter_2 .
pid	nomb	Colonne OS_PROCESS dans UNIFIED_AUDIT_TRAIL	Identificateur de processus du système d'exploitation du processus de base de données Oracle. Un exemple de valeur est 22396.

Champ	Type de donne	Source	Description
`remoteHost`	chaîn	Colonne `AUTHENTICATION_TYPE` dans `UNIFIED_AUDIT_TRAIL`	Soit l'adresse IP du client, soit le nom de l'hôte à partir duquel la session a été générée. Un exemple de valeur est `123.456.789.123` .
`remotePort`	chaîn	Colonne `AUTHENTICATION_TYPE` dans `UNIFIED_AUDIT_TRAIL`	Numéro de port du client. Une valeur typique dans les environnements Oracle Database est `1521`.
`rowCount`	nomb	N/A	Ce champ n'est pas utilisé pour RDS for Oracle. Sa valeur est toujours null.
`serverHost`	chaîn	Hôte de base de données	Adresse IP de l'hôte du serveur de base de données. Un exemple de valeur est `123.456.789.123` .
`serverType`	chaîn	N/A	Type de serveur de base de données. La valeur est toujours `ORACLE`.

Champ	Type de donne	Source	Description
`serverVersion`	chaîn	Hôte de base de données	Version d'Amazon RDS for Oracle, Release Update (RU) et Release Update Revision (RUR). Un exemple de valeur est `19.0.0.0.ru-2020-01.rur-2020-01.r1.EE.3` .
`serviceName`	chaîn	Hôte de base de données	Nom du service. Un exemple de valeur est `oracle-ee` .
`sessionId`	nomb	Colonne `SESSIONID` dans `UNIFIED_AUDIT_TRAIL`	Identificateur de session de l'audit. Par exemple : `1894327130` .
`startTime`	chaîn	N/A	Ce champ n'est pas utilisé pour RDS for Oracle. Sa valeur est toujours null.
`statementId`	nomb	Colonne `STATEMENT_ID` dans `UNIFIED_AUDIT_TRAIL`	ID numérique pour chaque exécution d'instruction. Une instruction peut provoquer de nombreuses actions. Un exemple de valeur est `142197`.
`substatementId`	N/A	N/A	Ce champ n'est pas utilisé pour RDS for Oracle. Sa valeur est toujours null.

Champ	Type de donne	Source	Description
`transactionId`	chaîn	Colonne `TRANSACTION_ID` dans `UNIFIED_AUDIT_TRAIL`	L'identificateur de la transaction dans laquelle l'objet est modifié. Un exemple de valeur est `02000800D5030000` .

databaseActivityEventChamps de liste pour Amazon RDS for SQL Server

Les `databaseActivityEventList` champs suivants concernent Amazon RDS for SQL Server.

Champ	Type de donnée	Source	Description
`class`	chaîne	`sys.fn_get_audit_file.class_type` mappé sur `sys.dm_audit_class_type_map.class_type_desc`	La classe d'un événement d'activité. Pour plus d'informations, consultez Audit SQL Server (moteur de base de données) dans la documentation Microsoft.
`clientApplication`	chaîne	`sys.fn_get_audit_file.application_name`	Application à laquelle le client se connecte, comme indiqué par le client (SQL Server versions 14 et ultérieures). Ce champ a la valeur null dans SQL Server version 13.
`command`	chaîne	`sys.fn_get_audit_file.action_id` mappé sur `sys.dm_audit_actions.name`	Catégorie générale de l'instruction SQL. La valeur de ce champ dépend de la valeur de la classe.
`commandText`	chaîne	`sys.fn_get_audit_file.statement`	Ce champ indique l'instruction SQL.

Champ	Type de donnée	Source	Description
`databaseName`	chaîne	`sys.fn_get_audit_file.database_name`	Nom du moteur de la base de données.
`dbProtocol`	chaîne	N/A	Protocole de la base de données. Cette valeur est `SQLSERVER` .
`dbUserName`	chaîne	`sys.fn_get_audit_file.server_principal_name`	L'utilisateur de la base de données pour l'authentification du client.
`endTime`	chaîne	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`engineNativeAuditFields`	objet	Chaque champ dans `sys.fn_get_audit_file` qui n'est pas répertorié dans cette colonne.	Par défaut, cet objet est vide. Lorsque vous démarrez le flux d'activité avec l'option `--engine-native-audit-fields-included` , cet objet inclut d'autres champs d'audit de moteur natifs, qui ne sont pas renvoyés par cette carte JSON.
`errorMessage`	chaîne	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.

Champ	Type de donnée	Source	Description
`exitCode`	entier	`sys.fn_get_audit_file.succeeded`	Indique si l'action qui a démarré l'événement a réussi. Ce champ ne peut pas avoir la valeur null. Pour tous les événements, à l'exception des événements de connexion, ce champ indique si la vérification des autorisations a réussi ou échoué, mais pas si l'opération a réussi ou échoué. Les valeurs sont les suivantes : • `0` – Échec • `1` – Réussite
`logTime`	chaîne	`sys.fn_get_audit_file.event_time`	Horodatage de l'événement enregistré par le serveur SQL Server.
`netProtocol`	chaîne	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`objectName`	chaîne	`sys.fn_get_audit_file.object_name`	Nom de l'objet de base de données si l'instruction SQL agit sur un objet.
`objectType`	chaîne	`sys.fn_get_audit_file.class_type` mappé sur `sys.dm_audit_class_type_map.class_type_desc`	Type de l'objet de base de données si l'instruction SQL agit sur un type d'objet.

Champ	Type de donnée	Source	Description
`paramList`	chaîne	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`pid`	entier	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`remoteHost`	chaîne	`sys.fn_get_audit_file.client_ip`	L'adresse IP ou le nom d'hôte du client qui a émis l'instruction SQL (SQL Server versions 14 et ultérieures). Ce champ a la valeur null dans SQL Server version 13.
`remotePort`	entier	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`rowCount`	entier	`sys.fn_get_audit_file.affected_rows`	Nombre de lignes de la table affectées par l'instruction SQL (SQL Server versions 14 et ultérieures). Ce champ figure dans SQL Server version 13.
`serverHost`	chaîne	Hôte de base de données	Adresse IP du serveur de base de données hôte.
`serverType`	chaîne	N/A	Type de serveur de base de données. La valeur est `SQLSERVER` .

Champ	Type de donnée	Source	Description
`serverVersion`	chaîne	Hôte de base de données	Version du serveur de base de données, par exemple 15.00.4073.23.v1.R1 pour SQL Server 2017.
`serviceName`	chaîne	Hôte de base de données	Nom du service. Un exemple de valeur est `sqlserver-ee` .
`sessionId`	entier	`sys.fn_get_audit_file.session_id`	Identificateur unique de la session.
`startTime`	chaîne	N/A	Ce champ n'est pas utilisé par Amazon RDS pour SQL Server et la valeur est null.
`statementId`	chaîne	`sys.fn_get_audit_file.sequence_group_id`	Identificateur unique de l'instruction SQL du client. L'identificateur est différent pour chaque événement généré. Un exemple de valeur est `0x38eaf4156267184094bb82071aaab644` .
`substatementId`	entier	`sys.fn_get_audit_file.sequence_number`	Identificateur permettant de déterminer le numéro de séquence d'une instruction. Cet identificateur est utile quand des enregistrements volumineux sont divisés en plusieurs enregistrements.
`transactionId`	entier	`sys.fn_get_audit_file.transaction_id`	Identificateur d'une transaction. S'il n'existe pas de transactions actives, la valeur est zéro.

Champ	Type de donnée	Source	Description
`type`	chaîne	Flux d'activité de base de données généré	Type d'événement. Les valeurs sont `record` ou `heartbeat` .

Traitement d'un flux d'activité de base de données à l'aide du AWS SDK

Vous pouvez traiter un flux d'activité par programmation à l'aide du. AWS SDK Les exemples suivants sont des exemples Java et Python entièrement fonctionnels d'utilisation des enregistrements de flux d'activité de base de données pour une activation basée sur des instances.

Java

```
import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.net.InetAddress;
import java.nio.ByteBuffer;
import java.nio.charset.StandardCharsets;
import java.security.NoSuchAlgorithmException;
import java.security.NoSuchProviderException;
import java.security.Security;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.UUID;
import java.util.zip.GZIPInputStream;

import javax.crypto.Cipher;
import javax.crypto.NoSuchPaddingException;
import javax.crypto.spec.SecretKeySpec;

import com.amazonaws.auth.AWSStaticCredentialsProvider;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.encryptionsdk.AwsCrypto;
import com.amazonaws.encryptionsdk.CryptoInputStream;
import com.amazonaws.encryptionsdk.jce.JceMasterKey;
import
 com.amazonaws.services.kinesis.clientlibrary.exceptions.InvalidStateException;
```

```
import com.amazonaws.services.kinesis.clientlibrary.exceptions.ShutdownException;
import com.amazonaws.services.kinesis.clientlibrary.exceptions.ThrottlingException;
import com.amazonaws.services.kinesis.clientlibrary.interfaces.IRecordProcessor;
import
 com.amazonaws.services.kinesis.clientlibrary.interfaces.IRecordProcessorCheckpointer;
import
 com.amazonaws.services.kinesis.clientlibrary.interfaces.IRecordProcessorFactory;
import
 com.amazonaws.services.kinesis.clientlibrary.lib.worker.InitialPositionInStream;
import
 com.amazonaws.services.kinesis.clientlibrary.lib.worker.KinesisClientLibConfiguration;
import com.amazonaws.services.kinesis.clientlibrary.lib.worker.ShutdownReason;
import com.amazonaws.services.kinesis.clientlibrary.lib.worker.Worker;
import com.amazonaws.services.kinesis.clientlibrary.lib.worker.Worker.Builder;
import com.amazonaws.services.kinesis.model.Record;
import com.amazonaws.services.kms.AWSKMS;
import com.amazonaws.services.kms.AWSKMSClientBuilder;
import com.amazonaws.services.kms.model.DecryptRequest;
import com.amazonaws.services.kms.model.DecryptResult;
import com.amazonaws.util.Base64;
import com.amazonaws.util.IOUtils;
import com.google.gson.Gson;
import com.google.gson.GsonBuilder;
import com.google.gson.annotations.SerializedName;
import org.bouncycastle.jce.provider.BouncyCastleProvider;

public class DemoConsumer {

    private static final String STREAM_NAME = "aws-rds-das-[instance-external-
resource-id]"; // aws-rds-das-db-ABCD123456
    private static final String APPLICATION_NAME = "AnyApplication"; //unique
 application name for dynamo table generation that holds kinesis shard tracking
    private static final String AWS_ACCESS_KEY =
 "[AWS_ACCESS_KEY_TO_ACCESS_KINESIS]";
    private static final String AWS_SECRET_KEY =
 "[AWS_SECRET_KEY_TO_ACCESS_KINESIS]";
    private static final String RESOURCE_ID = "[external-resource-id]"; // db-
ABCD123456
    private static final String REGION_NAME = "[region-name]"; //us-east-1, us-
east-2...
    private static final BasicAWSCredentials CREDENTIALS = new
 BasicAWSCredentials(AWS_ACCESS_KEY, AWS_SECRET_KEY);
    private static final AWSStaticCredentialsProvider CREDENTIALS_PROVIDER = new
 AWSStaticCredentialsProvider(CREDENTIALS);
```

```
    private static final AwsCrypto CRYPTO = new AwsCrypto();
    private static final AWSKMS KMS = AWSKMSClientBuilder.standard()
            .withRegion(REGION_NAME)
            .withCredentials(CREDENTIALS_PROVIDER).build();

    class Activity {
        String type;
        String version;
        String databaseActivityEvents;
        String key;
    }

    class ActivityEvent {
        @SerializedName("class") String _class;
        String clientApplication;
        String command;
        String commandText;
        String databaseName;
        String dbProtocol;
        String dbUserName;
        String endTime;
        String errorMessage;
        String exitCode;
        String logTime;
        String netProtocol;
        String objectName;
        String objectType;
        List<String> paramList;
        String pid;
        String remoteHost;
        String remotePort;
        String rowCount;
        String serverHost;
        String serverType;
        String serverVersion;
        String serviceName;
        String sessionId;
        String startTime;
        String statementId;
        String substatementId;
        String transactionId;
        String type;
    }
```

```
   class ActivityRecords {
       String type;
       String clusterId; // note that clusterId will contain an empty string on RDS
Oracle and RDS SQL Server
       String instanceId;
       List<ActivityEvent> databaseActivityEventList;
   }

   static class RecordProcessorFactory implements IRecordProcessorFactory {
       @Override
       public IRecordProcessor createProcessor() {
           return new RecordProcessor();
       }
   }

   static class RecordProcessor implements IRecordProcessor {

       private static final long BACKOFF_TIME_IN_MILLIS = 3000L;
       private static final int PROCESSING_RETRIES_MAX = 10;
       private static final long CHECKPOINT_INTERVAL_MILLIS = 60000L;
       private static final Gson GSON = new
GsonBuilder().serializeNulls().create();

       private static final Cipher CIPHER;
       static {
           Security.insertProviderAt(new BouncyCastleProvider(), 1);
           try {
               CIPHER = Cipher.getInstance("AES/GCM/NoPadding", "BC");
           } catch (NoSuchAlgorithmException | NoSuchPaddingException |
NoSuchProviderException e) {
               throw new ExceptionInInitializerError(e);
           }
       }

       private long nextCheckpointTimeInMillis;

       @Override
       public void initialize(String shardId) {
       }

       @Override
       public void processRecords(final List<Record> records, final
IRecordProcessorCheckpointer checkpointer) {
```

```
            for (final Record record : records) {
                processSingleBlob(record.getData());
            }

            if (System.currentTimeMillis() > nextCheckpointTimeInMillis) {
                checkpoint(checkpointer);
                nextCheckpointTimeInMillis = System.currentTimeMillis() +
CHECKPOINT_INTERVAL_MILLIS;
            }
        }

        @Override
        public void shutdown(IRecordProcessorCheckpointer checkpointer,
ShutdownReason reason) {
            if (reason == ShutdownReason.TERMINATE) {
                checkpoint(checkpointer);
            }
        }

        private void processSingleBlob(final ByteBuffer bytes) {
            try {
                // JSON $Activity
                final Activity activity = GSON.fromJson(new String(bytes.array(),
StandardCharsets.UTF_8), Activity.class);

                // Base64.Decode
                final byte[] decoded =
Base64.decode(activity.databaseActivityEvents);
                final byte[] decodedDataKey = Base64.decode(activity.key);

                Map<String, String> context = new HashMap<>();
                context.put("aws:rds:db-id", RESOURCE_ID);

                // Decrypt
                final DecryptRequest decryptRequest = new DecryptRequest()

.withCiphertextBlob(ByteBuffer.wrap(decodedDataKey)).withEncryptionContext(context);
                final DecryptResult decryptResult = KMS.decrypt(decryptRequest);
                final byte[] decrypted = decrypt(decoded,
getByteArray(decryptResult.getPlaintext()));

                // GZip Decompress
                final byte[] decompressed = decompress(decrypted);
                // JSON $ActivityRecords
```

```
                final ActivityRecords activityRecords = GSON.fromJson(new
String(decompressed, StandardCharsets.UTF_8), ActivityRecords.class);

                // Iterate throught $ActivityEvents
                for (final ActivityEvent event :
activityRecords.databaseActivityEventList) {
                    System.out.println(GSON.toJson(event));
                }
            } catch (Exception e) {
                // Handle error.
                e.printStackTrace();
            }
        }

        private static byte[] decompress(final byte[] src) throws IOException {
            ByteArrayInputStream byteArrayInputStream = new
ByteArrayInputStream(src);
            GZIPInputStream gzipInputStream = new
GZIPInputStream(byteArrayInputStream);
            return IOUtils.toByteArray(gzipInputStream);
        }

        private void checkpoint(IRecordProcessorCheckpointer checkpointer) {
            for (int i = 0; i < PROCESSING_RETRIES_MAX; i++) {
                try {
                    checkpointer.checkpoint();
                    break;
                } catch (ShutdownException se) {
                    // Ignore checkpoint if the processor instance has been shutdown
(fail over).
                    System.out.println("Caught shutdown exception, skipping
checkpoint." + se);
                    break;
                } catch (ThrottlingException e) {
                    // Backoff and re-attempt checkpoint upon transient failures
                    if (i >= (PROCESSING_RETRIES_MAX - 1)) {
                        System.out.println("Checkpoint failed after " + (i + 1) +
"attempts." + e);
                        break;
                    } else {
                        System.out.println("Transient issue when checkpointing -
attempt " + (i + 1) + " of " + PROCESSING_RETRIES_MAX + e);
                    }
                } catch (InvalidStateException e) {
```

```
                    // This indicates an issue with the DynamoDB table (check for
table, provisioned IOPS).
                    System.out.println("Cannot save checkpoint to the DynamoDB table
used by the Amazon Kinesis Client Library." + e);
                    break;
                }
                try {
                    Thread.sleep(BACKOFF_TIME_IN_MILLIS);
                } catch (InterruptedException e) {
                    System.out.println("Interrupted sleep" + e);
                }
            }
        }
    }

    private static byte[] decrypt(final byte[] decoded, final byte[] decodedDataKey)
throws IOException {
        // Create a JCE master key provider using the random key and an AES-GCM
encryption algorithm
        final JceMasterKey masterKey = JceMasterKey.getInstance(new
SecretKeySpec(decodedDataKey, "AES"),
                "BC", "DataKey", "AES/GCM/NoPadding");
        try (final CryptoInputStream<JceMasterKey> decryptingStream =
CRYPTO.createDecryptingStream(masterKey, new ByteArrayInputStream(decoded));
             final ByteArrayOutputStream out = new ByteArrayOutputStream()) {
            IOUtils.copy(decryptingStream, out);
            return out.toByteArray();
        }
    }

    public static void main(String[] args) throws Exception {
        final String workerId = InetAddress.getLocalHost().getCanonicalHostName() +
":" + UUID.randomUUID();
        final KinesisClientLibConfiguration kinesisClientLibConfiguration =
                new KinesisClientLibConfiguration(APPLICATION_NAME, STREAM_NAME,
CREDENTIALS_PROVIDER, workerId);

kinesisClientLibConfiguration.withInitialPositionInStream(InitialPositionInStream.LATEST);
        kinesisClientLibConfiguration.withRegionName(REGION_NAME);
        final Worker worker = new Builder()
                .recordProcessorFactory(new RecordProcessorFactory())
                .config(kinesisClientLibConfiguration)
                .build();
```

```
        System.out.printf("Running %s to process stream %s as worker %s...\n",
 APPLICATION_NAME, STREAM_NAME, workerId);

        try {
            worker.run();
        } catch (Throwable t) {
            System.err.println("Caught throwable while processing data.");
            t.printStackTrace();
            System.exit(1);
        }
        System.exit(0);
    }

    private static byte[] getByteArray(final ByteBuffer b) {
        byte[] byteArray = new byte[b.remaining()];
        b.get(byteArray);
        return byteArray;
    }
}
```

Python

```
import base64
import json
import zlib
import aws_encryption_sdk
from aws_encryption_sdk import CommitmentPolicy
from aws_encryption_sdk.internal.crypto import WrappingKey
from aws_encryption_sdk.key_providers.raw import RawMasterKeyProvider
from aws_encryption_sdk.identifiers import WrappingAlgorithm, EncryptionKeyType
import boto3

REGION_NAME = '<region>'                    # us-east-1
RESOURCE_ID = '<external-resource-id>'      # db-ABCD123456
STREAM_NAME = 'aws-rds-das-' + RESOURCE_ID  # aws-rds-das-db-ABCD123456

enc_client =
 aws_encryption_sdk.EncryptionSDKClient(commitment_policy=CommitmentPolicy.FORBID_ENCRYPT_AL

class MyRawMasterKeyProvider(RawMasterKeyProvider):
    provider_id = "BC"

    def __new__(cls, *args, **kwargs):
```

```
        obj = super(RawMasterKeyProvider, cls).__new__(cls)
        return obj

    def __init__(self, plain_key):
        RawMasterKeyProvider.__init__(self)
        self.wrapping_key =
 WrappingKey(wrapping_algorithm=WrappingAlgorithm.AES_256_GCM_IV12_TAG16_NO_PADDING,
                                        wrapping_key=plain_key,
 wrapping_key_type=EncryptionKeyType.SYMMETRIC)

    def _get_raw_key(self, key_id):
        return self.wrapping_key


def decrypt_payload(payload, data_key):
    my_key_provider = MyRawMasterKeyProvider(data_key)
    my_key_provider.add_master_key("DataKey")
    decrypted_plaintext, header = enc_client.decrypt(
        source=payload,

 materials_manager=aws_encryption_sdk.materials_managers.default.DefaultCryptoMaterialsManag
    return decrypted_plaintext


def decrypt_decompress(payload, key):
    decrypted = decrypt_payload(payload, key)
    return zlib.decompress(decrypted, zlib.MAX_WBITS + 16)


def main():
    session = boto3.session.Session()
    kms = session.client('kms', region_name=REGION_NAME)
    kinesis = session.client('kinesis', region_name=REGION_NAME)

    response = kinesis.describe_stream(StreamName=STREAM_NAME)
    shard_iters = []
    for shard in response['StreamDescription']['Shards']:
        shard_iter_response = kinesis.get_shard_iterator(StreamName=STREAM_NAME,
 ShardId=shard['ShardId'],

 ShardIteratorType='LATEST')
        shard_iters.append(shard_iter_response['ShardIterator'])

    while len(shard_iters) > 0:
```

```
        next_shard_iters = []
        for shard_iter in shard_iters:
            response = kinesis.get_records(ShardIterator=shard_iter, Limit=10000)
            for record in response['Records']:
                record_data = record['Data']
                record_data = json.loads(record_data)
                payload_decoded =
 base64.b64decode(record_data['databaseActivityEvents'])
                data_key_decoded = base64.b64decode(record_data['key'])
                data_key_decrypt_result =
 kms.decrypt(CiphertextBlob=data_key_decoded,

 EncryptionContext={'aws:rds:db-id': RESOURCE_ID})
                print (decrypt_decompress(payload_decoded,
 data_key_decrypt_result['Plaintext']))
            if 'NextShardIterator' in response:
                next_shard_iters.append(response['NextShardIterator'])
        shard_iters = next_shard_iters


if __name__ == '__main__':
    main()
```

Exemples de politiques IAM pour les flux d'activité des bases de données

Tout utilisateur disposant des privilèges de rôle AWS Identity and Access Management (IAM) appropriés pour les flux d'activité de base de données peut créer, démarrer, arrêter et modifier les paramètres des flux d'activité pour une instance de base de données de de base de données. Ces actions sont consignées dans le journal d'audit du flux. Pour des raisons de conformité optimales, nous vous recommandons de ne pas octroyer ces privilèges à DBAs.

Vous devrez paramétrer les accès aux flux d'activité de base de données à l'aide de politiques IAM. Pour plus d'informations sur l'authentification Amazon RDS, consultez Gestion des identités et des accès pour Amazon RDS. Pour plus d'informations sur la création des stratégies IAM, consultez Création et utilisation d'une politique IAM pour l'accès à une base de données IAM.

Example politique pour autoriser la configuration de Database Activity Streams

Pour donner aux utilisateurs des accès précis en vue de modifier les flux d'activité, utilisez les clés de contexte d'opération spécifiques au service `rds:StartActivityStream` et

`rds:StopActivityStream` dans une stratégie IAM. L'exemple de politique IAM suivant autorise un utilisateur ou un rôle à configurer des flux d'activité.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ConfigureActivityStreams",
            "Effect": "Allow",
            "Action": [
                "rds:StartActivityStream",
                "rds:StopActivityStream"
            ],
            "Resource": "*"
        }
    ]
}
```

Example politique pour autoriser le démarrage de Database Activity Streams

L'exemple de politique IAM suivant autorise un utilisateur ou un rôle à démarrer des flux d'activité.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"AllowStartActivityStreams",
            "Effect":"Allow",
            "Action":"rds:StartActivityStream",
            "Resource":"*"
        }
    ]
}
```

Example politique pour autoriser l'arrêt de Database Activity Streams

L'exemple de politique IAM suivant autorise un utilisateur ou un rôle à arrêter des flux d'activité.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"AllowStopActivityStreams",
            "Effect":"Allow",
            "Action":"rds:StopActivityStream",
            "Resource":"*"
        }
     ]
}
```

Example politique pour refuser le démarrage de Database Activity Streams

La politique IAM suivante empêche un utilisateur ou un rôle de démarrer des flux d'activité.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"DenyStartActivityStreams",
            "Effect":"Deny",
            "Action":"rds:StartActivityStream",
            "Resource":"*"
        }
     ]
}
```

Example politique pour refuser l'arrêt de Database Activity Streams

La politique IAM suivante empêche un utilisateur ou un rôle d'arrêter des flux d'activité.

JSON

```
{
    "Version":"2012-10-17",
    "Statement":[
        {
            "Sid":"DenyStopActivityStreams",
            "Effect":"Deny",
            "Action":"rds:StopActivityStream",
            "Resource":"*"
        }
    ]
}
```

Surveillance des menaces avec Amazon GuardDuty RDS Protection

Amazon GuardDuty est un service de détection des menaces qui aide à protéger vos comptes, vos conteneurs, vos charges de travail et les données de votre AWS environnement. À l'aide de modèles d'apprentissage automatique (ML) et de capacités de détection des anomalies et des menaces, vous surveillez GuardDuty en permanence les différentes sources de journaux et l'activité d'exécution afin d'identifier et de hiérarchiser les risques de sécurité potentiels et les activités malveillantes dans votre environnement.

GuardDuty RDS Protection analyse et établit le profil des événements de connexion pour détecter les menaces d'accès potentielles à vos bases de données Amazon RDS. Lorsque vous activez la protection RDS, GuardDuty consomme les événements de connexion RDS de vos bases de données RDS. RDS Protection surveille ces événements et en établit le profil pour détecter les menaces potentielles de l'intérieur ou des acteurs externes.

Pour plus d'informations sur l'activation de la protection GuardDuty RDS, consultez la section Protection GuardDuty RDS dans le guide de GuardDuty l'utilisateur Amazon.

Lorsque RDS Protection détecte une menace potentielle, telle qu'un schéma inhabituel de tentatives de connexion réussies ou échouées, GuardDuty génère une nouvelle découverte contenant des informations détaillées sur la base de données potentiellement compromise. Vous pouvez consulter les détails de la recherche dans la section récapitulative des résultats de la GuardDuty console Amazon. Les détails des résultats varient en fonction du type de découverte. Les principaux détails, le type de ressource et le rôle de la ressource, déterminent le type d'informations disponibles pour toute recherche. Pour plus d'informations sur les informations couramment disponibles pour les résultats et les types de résultats, consultez les sections Détails de la recherche et Types de résultats GuardDuty RDS Protection respectivement dans le guide de l' GuardDuty utilisateur Amazon.

Vous pouvez activer ou désactiver la fonction de protection RDS Compte AWS partout Région AWS où elle est disponible. Lorsque la protection RDS n'est pas activée, GuardDuty elle ne détecte pas les bases de données RDS potentiellement compromises et ne fournit pas de détails sur la compromission.

Un GuardDuty compte existant peut activer RDS Protection avec une période d'essai de 30 jours. Pour un nouveau GuardDuty compte, la protection RDS est déjà activée et incluse dans la période d'essai gratuite de 30 jours. Pour plus d'informations, consultez la section Estimation GuardDuty des coûts dans le guide de GuardDuty l'utilisateur Amazon.

Pour plus d'informations sur l' Régions AWS endroit où la protection RDS GuardDuty n'est pas encore prise en charge, consultez la section Disponibilité des fonctionnalités spécifiques à chaque région dans le guide de l'utilisateur Amazon GuardDuty .

Pour plus d'informations sur les versions de base de données RDS prises en charge par GuardDuty RDS Protection, consultez la section Bases de données Amazon Aurora, Amazon RDS et Aurora Limitless prises en charge dans le guide de l'utilisateur Amazon. GuardDuty

Amazon RDS Custom

Amazon RDS Custom automatise les tâches et les opérations d'administration des bases de données. Amazon RDS Custom vous permet, en tant qu'administrateur de base de données, d'accéder à et de personnaliser votre environnement de base de données et votre système d'exploitation. Avec RDS Custom, vous pouvez personnaliser pour répondre aux exigences des applications héritées, personnalisées et compilées.

Pour consulter les derniers webinaires et blogs sur RDS Custom, consultez Ressources Amazon RDS Custom.

Rubriques

- Relever le défi de la personnalisation des bases de données
- Modèle de gestion et avantages pour Amazon RDS Custom
- Architecture Amazon RDS Custom
- Sécurité dans Amazon RDS Custom
- Utilisation de RDS Custom for Oracle
- Utilisation de RDS Custom for SQL Server

Relever le défi de la personnalisation des bases de données

Amazon RDS Custom apporte les avantages d'Amazon RDS sur un marché qui ne peut pas facilement passer à un service entièrement géré en raison des personnalisations requises avec des applications tierces. Amazon RDS Custom économise du temps administratif, est durable et évolue avec votre entreprise.

Si vous avez besoin d'une gestion complète de la base de données et du système d'exploitation AWS, nous vous recommandons Amazon RDS. Si vous avez besoin de droits d'administration sur la base de données et le système d'exploitation sous-jacent pour rendre des applications dépendantes disponibles, Amazon RDS Custom est le meilleur choix. Si vous souhaitez assumer l'entière responsabilité de la gestion et avez simplement besoin d'un service informatique géré, la meilleure option consiste à gérer vous-même vos bases de données commerciales sur Amazon EC2.

Pour offrir une expérience de service géré, Amazon RDS ne vous permet pas d'accéder à l'hôte sous-jacent. Amazon RDS restreint également l'accès à certaines procédures et objets qui requièrent

des privilèges avancés. Toutefois, pour certaines applications, vous devrez peut-être effectuer des opérations en tant qu'utilisateur de système d'exploitation privilégié.

Par exemple, il se peut que vous ayez besoin de faire ce qui suit.

- Installer des correctifs et des packages personnalisés de base de données et d'OS.
- Configurer des paramètres de base de données
- Configurer des systèmes de fichiers pour partager des fichiers directement avec leurs applications.

Auparavant, si vous deviez personnaliser votre application, vous deviez déployer votre base de données sur site ou sur Amazon EC2. Dans ce cas, vous assumez la plupart ou la totalité de la responsabilité de la gestion des bases de données, comme le résume le tableau suivant.

Fonctionnalité	Responsabilité sur site	EC2 Responsabilité d'Amazon	Responsabilité Amazon RDS
Optimisation des applications	Client	Client	Client
Mise à l'échelle	Client	Client	AWS
Haute disponibilité	Client	Client	AWS
Sauvegardes de base de données	Client	Client	AWS
Correction de logiciel de base de données	Client	Client	AWS
Installation de logiciels de base de données	Client	Client	AWS
Correction du système d'exploitation	Client	Client	AWS
Installation du système d'exploitation	Client	Client	AWS

Fonctionnalité	Responsabilité sur site	EC2 Responsabilité d'Amazon	Responsabilité Amazon RDS
Maintenance des serveurs	Client	AWS	AWS
Cycle de vie du matériel	Client	AWS	AWS
Alimentation, réseau et refroidissement	Client	AWS	AWS

Lorsque vous gérez vous-même un logiciel de base de données, vous gagnez en contrôle, mais vous êtes également plus sujet aux erreurs des utilisateurs. Par exemple, lorsque vous effectuez des modifications manuellement, vous risquez de provoquer accidentellement des interruptions d'application. Vous pouvez passer des heures à vérifier chaque modification pour identifier et résoudre un problème. Idéalement, vous souhaitez disposer d'un service de base de données géré qui automatise les tâches d'administrateur de base de données courantes, mais prend également en charge l'accès privilégié à la base de données et au système d'exploitation sous-jacent.

Modèle de gestion et avantages pour Amazon RDS Custom

Amazon RDS Custom est un service de base de données géré destiné aux applications héritées, personnalisées et empaquetées nécessitant un accès à l'environnement de base de données et au système d'exploitation sous-jacents. RDS Custom automatise la configuration, le fonctionnement et le dimensionnement des bases de données AWS Cloud tout en vous donnant accès à la base de données et au système d'exploitation sous-jacent. Avec cet accès, vous pouvez configurer des paramètres, installer des correctifs et activer des fonctionnalités natives pour répondre aux exigences de l'application dépendante. Avec RDS Custom, vous pouvez exécuter la charge de travail de votre base de données à l'aide du AWS Management Console ou du AWS CLI.

Actuellement, RDS Custom prend en charge uniquement les moteurs de base de données Oracle Database et Microsoft SQL Server.

Rubriques

- Modèle de responsabilité partagée dans RDS Custom
- Périmètre de prise en charge et configurations non prises en charge dans RDS Custom

- Principaux avantages de RDS Custom

Modèle de responsabilité partagée dans RDS Custom

Avec RDS Custom, vous utilisez les fonctionnalités gérées d'Amazon RDS, mais vous gérez l'hôte et personnalisez le système d'exploitation comme vous le faites dans Amazon. EC2 Vous assumez des responsabilités supplémentaires en matière de gestion de bases de données au-delà de ce que vous faites dans Amazon RDS. Vous avez ainsi plus de contrôle sur la gestion des bases de données et des instances de base de données que dans Amazon RDS, tout en bénéficiant de l'automatisation RDS.

La responsabilité partagée signifie ce qui suit :

1. Vous êtes propriétaire d'une partie du processus lorsque vous utilisez une fonctionnalité RDS Custom.

   Par exemple, dans RDS Custom for Oracle, vous décidez quels correctifs de base de données Oracle utiliser et quand les appliquer à vos instances de base de données.
2. Vous devez vous assurer que toutes les personnalisations apportées aux fonctionnalités RDS Custom fonctionnent correctement.

   Pour vous protéger contre les personnalisations non valides, RDS Custom dispose d'un logiciel d'automatisation qui s'exécute en dehors de votre instance de base de données. Si votre EC2 instance Amazon sous-jacente est défectueuse, RDS Custom tente de résoudre ces problèmes automatiquement en redémarrant ou en remplaçant l'instance. EC2 La seule modification visible par l'utilisateur est une nouvelle adresse IP. Pour de plus amples informations, veuillez consulter Remplacement de l'hôte Amazon RDS Custom.

Le tableau suivant détaille le modèle de responsabilité partagée pour les différentes fonctionnalités RDS Custom.

Fonctionnalité	EC2 Responsabilité d'Amazon	Responsabilité Amazon RDS	Responsabilité RDS Custom for Oracle	Responsabilité RDS Custom for SQL Server
Optimisation des applications	Client	Client	Client	Client

Fonctionnalité	EC2 Responsabilité d'Amazon	Responsabilité Amazon RDS	Responsabilité RDS Custom for Oracle	Responsabilité RDS Custom for SQL Server
Mise à l'échelle	Client	AWS	Partagé	Partagé
Haute disponibilité	Client	AWS	AWS	AWS
Sauvegardes de base de données	Client	AWS	Partagé	AWS
Correction de logiciel de base de données	Client	AWS	Partagé	AWS pour RPEV, client pour CEV 1
Installation de logiciels de base de données	Client	AWS	Partagé	AWS pour RPEV, client pour CEV 1
Correction du système d'exploitation	Client	AWS	Client	AWS pour RPEV, client pour CEV 1
Installation du système d'exploitation	Client	AWS	Partagé	AWS
Maintenance des serveurs	AWS	AWS	AWS	AWS
Cycle de vie du matériel	AWS	AWS	AWS	AWS
Alimentation, réseau et refroidissement	AWS	AWS	AWS	AWS

[1] Une version de moteur personnalisée (CEV) est un instantané de volume binaire d'une version de base de données et d'Amazon Machine Image (AMI). Une version du moteur fournie par RDS (RPEV) est l'installation par défaut d'Amazon Machine Image (AMI) et de Microsoft SQL Server.

Vous pouvez créer une instance de base de données RDS Custom à l'aide de Microsoft SQL Server. Dans ce cas :

- Vous pouvez choisir entre deux modèles de licence : License Included (LI) et Bring Your Own Media (BYOM).
- Avec LI, vous n'avez pas besoin d'acheter des licences SQL Server séparément. AWS détient la licence du logiciel de base de données SQL Server.
- Avec BYOM, vous fournissez et installez vos propres fichiers binaires et licences Microsoft SQL Server.

Vous pouvez créer une instance de base de données RDS Custom à l'aide d'Oracle Database. Dans ce cas, procédez comme suit :

- Gérez vos propres médias.

  Lorsque vous utilisez RDS Custom, vous chargez vos propres fichiers et correctifs d'installation de base de données. Vous créez une version de moteur personnalisée (CEV) à partir de ces fichiers. Vous pouvez ensuite créer une instance de base de données RDS Custom à l'aide de cette CEV.
- Gérez vos propres licences.

  Vous apportez vos propres licences Oracle Database et gérez vous-même les licences.

Périmètre de prise en charge et configurations non prises en charge dans RDS Custom

RDS Custom offre une fonctionnalité de surveillance appelée périmètre de prise en charge. Cette fonctionnalité garantit que votre environnement d'hôte et de base de données est correctement configuré. Si vous apportez une modification entraînant la sortie de votre instance de base de données du périmètre de prise en charge, RDS Custom fait passer le statut de l'instance à `unsupported-configuration` jusqu'à ce que vous résolviez manuellement les problèmes de configuration. Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.

Principaux avantages de RDS Custom

RDS Custom vous permet d'effectuer les actions suivantes :

- Automatiser de nombreuses tâches administratives identiques à celles d'Amazon RDS, notamment les suivantes :
  - Gestion des cycles de vie des bases de données
  - Sauvegardes et point-in-time restaurations automatisées (PITR)
  - Surveillance de l'état des instances de base de données personnalisées RDS et observation des modifications apportées à l'infrastructure, au système d'exploitation et aux processus de base de données.
  - Notification ou action visant à résoudre les problèmes en fonction de la perturbation de l'instance de base de données
- Installer des applications tierces.

  Vous pouvez installer un logiciel pour exécuter des applications et des agents personnalisés. Étant donné que vous disposez d'un accès privilégié à l'hôte, vous pouvez modifier les systèmes de fichiers pour prendre en charge des applications héritées.
- Installer des correctifs personnalisés.

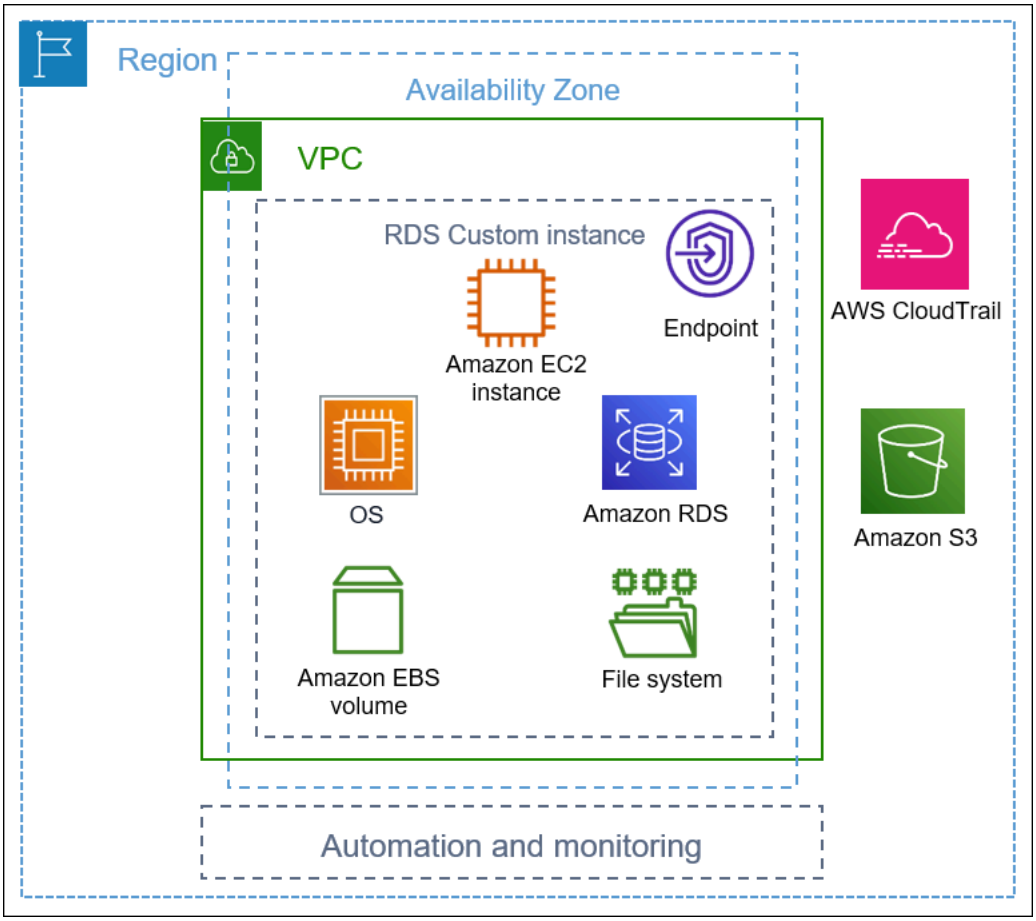
  Vous pouvez appliquer des correctifs de base de données personnalisés ou modifier des packages de système d'exploitation sur vos instances de base de données RDS Custom.
- Placer une base de données sur site avant de la déplacer vers un service entièrement géré.

  Si vous gérez votre propre base de données sur site, vous pouvez placer la base de données sur RDS Custom telle quelle. Une fois que vous vous êtes familiarisé avec l'environnement cloud, vous pouvez migrer votre base de données vers une instance de base de données Amazon RDS entièrement gérée.
- Créer votre propre automatisation.

  Vous pouvez créer, planifier et exécuter des scripts d'automatisation personnalisés pour les outils de création de rapports, de gestion ou de diagnostic.

Architecture Amazon RDS Custom

L'architecture Amazon RDS Custom est basée sur Amazon RDS, avec des différences importantes. Le diagramme suivant montre les composants clés de l'architecture RDS Custom.



Rubriques

- VPC
- Automatisation et surveillance RDS Custom
- Amazon S3
- AWS CloudTrail

VPC

Comme dans Amazon RDS, votre instance de base de données RDS Custom réside dans un cloud privé virtuel (VPC).



Votre instance de base de données RDS Custom comprend les composants principaux suivants :

- EC2 Instance Amazon
- Point de terminaison d'instance
- Système d'exploitation installé sur l' EC2 instance Amazon
- Stockage Amazon EBS, qui contient tous les systèmes de fichiers supplémentaires

Automatisation et surveillance RDS Custom

RDS Custom dispose d'un logiciel d'automatisation qui s'exécute en dehors de l'instance de base de données. Ce logiciel communique avec les agents de l'instance de base de données et avec d'autres composants de l'environnement RDS Custom global.

Les fonctions de surveillance et de récupération de RDS Custom offrent des fonctionnalités similaires à celles d'Amazon RDS. Par défaut, RDS Custom est en mode d'automatisation complète. Le logiciel d'automatisation a les principales responsabilités suivantes :

- Collecte de mesures et envoi de notifications
- Récupération automatique des instances

L'une des principales responsabilités de RDS Custom Automation est de répondre aux problèmes liés à votre EC2 instance Amazon. Pour diverses raisons, l'hôte peut se dégrader ou devenir inaccessible. RDS Custom résout ces problèmes en redémarrant ou en remplaçant l'instance Amazon. EC2

Rubriques

- Remplacement de l'hôte Amazon RDS Custom
- Périmètre de prise en charge RDS Custom

Remplacement de l'hôte Amazon RDS Custom

Si l' EC2 hôte Amazon est défaillant, RDS Custom tente de le redémarrer. Si cet effort échoue, RDS Custom utilise la même fonctionnalité d'arrêt et de démarrage incluse dans Amazon EC2. La seule modification visible par le client lorsqu'un hôte est remplacé est une nouvelle adresse IP publique.

Rubriques

- Arrêt et démarrage de l'hôte
- Effets du remplacement de l'hôte
- Bonnes pratiques pour les EC2 hébergeurs Amazon

Arrêt et démarrage de l'hôte

RDS Custom effectue automatiquement les étapes suivantes, sans qu'une intervention de l'utilisateur ne soit nécessaire :

1. Arrête l' EC2 hôte Amazon.

   L' EC2 instance effectue un arrêt normal et cesse de fonctionner. Les volumes Amazon EBS restent attachés à l'instance et leurs données persistent. Les données stockées dans les volumes de stockage d'instances (non pris en charge sur RDS Custom) ou la RAM de l'ordinateur hôte sont perdues.

   Pour plus d'informations, consultez la section Arrêter et démarrer votre instance dans le guide de EC2 l'utilisateur Amazon.
2. Démarre l' EC2 hôte Amazon.

   L' EC2 instance migre vers un nouveau matériel hôte sous-jacent. Dans certains cas, l'instance de base de données RDS Custom reste sur l'hôte d'origine.

Effets du remplacement de l'hôte

Dans RDS Custom, vous disposez d'un contrôle absolu sur le volume du périphérique racine et les volumes de stockage Amazon EBS. Le volume racine peut contenir des données et des configurations importantes que vous ne voulez pas perdre.

RDS Custom for Oracle conserve toutes les données de base de données et client après l'opération, y compris celles du volume racine. Aucune intervention de l'utilisateur n'est requise. Sur RDS Custom for SQL Server, les données de base de données sont conservées, mais toutes les données présentes sur le `C:` disque, y compris le système d'exploitation et les données client, sont perdues.

Après le processus de remplacement de l'hôte, l' EC2 hôte Amazon dispose d'une nouvelle adresse IP publique. Le nouvel hôte conserve les données et les métadonnées indiquées dans le tableau suivant.

Métadonnées ou données	RDS Custom pour la rétention d'Oracle	RDS Custom pour la rétention de SQL Server
EC2 ID d'instance	Oui	Non
EC2 métadonnées de l'instance	Oui	Oui
Données du volume de stockage de données	Oui	Oui
Données du volume racine	Oui	Non
Adresses IP privées	Oui	Oui
Adresses IP Elastic	Oui	Oui

Bonnes pratiques pour les EC2 hébergeurs Amazon

La fonctionnalité de remplacement de l' EC2 hôte Amazon couvre la majorité des scénarios de EC2 défaillance d'Amazon. Nous vous recommandons de respecter les bonnes pratiques ci-dessous :

- Avant de modifier votre configuration ou le système d'exploitation, sauvegardez vos données. Si le volume racine ou le système d'exploitation est endommagé, le remplacement de l'hôte ne peut pas

le réparer. Vos seules options sont la restauration à partir d'un instantané de base de données ou point-in-time la restauration.

- N'arrêtez pas ou ne mettez pas hors service manuellement l' EC2 hôte Amazon physique. En effet, ces deux actions placent l'instance en dehors du périmètre de prise en charge de RDS Custom.
- (RDS Custom pour SQL Server) Si vous attachez des volumes supplémentaires à l' EC2 hôte Amazon, configurez-les pour qu'ils soient remontés au redémarrage. Si l'hôte est endommagé, il se peut que RDS Custom l'arrête et le démarre automatiquement.

Périmètre de prise en charge RDS Custom

RDS Custom offre une fonctionnalité de surveillance supplémentaire appelée le périmètre de support. Cette surveillance supplémentaire garantit que votre instance de base de données personnalisée RDS utilise une AWS infrastructure, un système d'exploitation et une base de données pris en charge.

Le périmètre de prise en charge vérifie que votre instance de base de données est conforme aux exigences répertoriées dans Correction des configurations non prises en charge dans RDS Custom for Oracle et Correction des configurations non prises en charge dans RDS Custom for SQL Server. Si l'une de ces exigences n'est pas remplie, RDS Custom considère que votre instance de base de données se trouve en dehors du périmètre de prise en charge.

Rubriques

- Configurations non prises en charge dans RDS Custom
- Résolution des problèmes de configurations non prises en charge

Configurations non prises en charge dans RDS Custom

Lorsque votre instance de base de données est en dehors du périmètre de prise en charge, RDS Custom fait passer le statut de l'instance de base de données à `unsupported-configuration` et envoie des notifications d'événement. Une fois que vous avez corrigé les problèmes de configuration, RDS Custom rétablit le statut de l'instance de base de données sur `available`.

Lorsque votre instance de base de données a le statut `unsupported-configuration`, ce qui suit est vrai :

- Votre base de données est accessible. Il y a une exception à cela : quand l'instance de base de données a le statut `unsupported-configuration` parce que la base de données s'arrête de façon inattendue.
- Vous ne pouvez pas modifier votre instance de base de données.
- Vous ne pouvez pas réaliser d'instantanés de bases de données.
- Les sauvegardes automatiques ne sont pas créées.
- Pour les instances de base de données RDS Custom pour SQL Server uniquement, RDS Custom ne remplace pas l' EC2instance Amazon sous-jacente si elle est altérée. Pour plus d'informations sur le remplacement de l'hôte, consultez Remplacement de l'hôte Amazon RDS Custom.
- Vous pouvez supprimer votre instance de base de données, mais la plupart des autres opérations d'API RDS Custom ne sont pas disponibles.
- RDS Custom continue de prendre en charge la point-in-time restauration (PITR) en archivant les fichiers de journalisation et en les téléchargeant sur Amazon S3. La récupération ponctuelle avec un statut `unsupported-configuration` diffère selon les manières suivantes :
  - La récupération ponctuelle peut prendre beaucoup de temps pour restaurer complètement une nouvelle instance de base de données RDS Custom. Cette situation tient au fait que vous ne pouvez pas réaliser d'instantanés automatisés ni manuels lorsque l'instance a le statut `unsupported-configuration`.
  - La restauration à un instant dans le passé doit lire d'autres journaux de reprise à partir de l'instantané le plus récent réalisé avant que l'instance ne passe à l'état `unsupported-configuration`.
  - Dans certains cas, l'instance de base de données a le statut `unsupported-configuration` parce que vous avez apporté une modification empêchant le téléchargement des fichiers journaux redo archivés. Les exemples incluent l'arrêt de l' EC2 instance, l'arrêt de l'agent personnalisé RDS et le détachement de volumes EBS. Dans de tels cas, la récupération ponctuelle (PITR) ne peut pas restaurer l'instance de base de données à l'heure de restauration la plus récente.

Résolution des problèmes de configurations non prises en charge

RDS Custom fournit des conseils de résolution de problèmes pour le statut `unsupported-configuration`. Bien que certaines instructions s'appliquent à RDS Custom for Oracle et à RDS Custom for SQL Server, d'autres conseils dépendent de votre moteur de base de données. Pour accéder aux informations spécifiques de résolution de problèmes, consultez les rubriques suivantes :

- Correction des configurations non prises en charge dans RDS Custom for Oracle
- Correction des configurations non prises en charge dans RDS Custom for SQL Server

Amazon S3

Si vous utilisez RDS Custom for Oracle, vous chargez le support d'installation dans un compartiment Amazon S3 créé par l'utilisateur. RDS Custom for Oracle utilise les médias de ce compartiment pour créer une version personnalisée du moteur (CEV). Une CEV est un instantané de volume binaire d'une version de base de données et d'Amazon Machine Image (AMI). À partir de la CEV, vous pouvez créer une instance de base de données RDS Custom. Pour de plus amples informations, veuillez consulter Utilisation de versions de moteurs personnalisées pour Amazon RDS Custom for Oracle.

Pour RDS Custom for Oracle et RDS Custom for SQL Server, RDS Custom crée automatiquement un compartiment Amazon S3 préfixé par la chaîne `do-not-delete-rds-custom-`. RDS Custom utilise le compartiment S3 `do-not-delete-rds-custom-` pour stocker les types de fichiers suivants :

- AWS CloudTrail journaux pour le parcours créé par RDS Custom
- Artefacts du périmètre de support (voir Périmètre de prise en charge RDS Custom).
- Fichiers de journal de reprise de la base de données (RDS Custom for Oracle uniquement)
- Journaux de transactions (RDS Custom for SQL Server uniquement)
- Artefacts de version du moteur personnalisé (RDS Custom for Oracle uniquement)

RDS Custom crée le compartiment S3 `do-not-delete-rds-custom-` lorsque vous créez l'une des ressources suivantes :

- Votre première CEV pour RDS Custom for Oracle
- Votre première instance de base de données pour RDS Custom for SQL Server

RDS Custom crée un compartiment pour chaque combinaison des éléments suivants :

- Compte AWS ID
- Type de moteur (RDS Custom for Oracle ou RDS Custom for SQL Server)
- Région AWS

Par exemple, si vous créez RDS Custom pour Oracle CEVs dans un seul compartiment Région AWS, il existe un seul `do-not-delete-rds-custom-` compartiment. Si vous créez plusieurs instances RDS Custom pour SQL Server et qu'elles résident dans des instances différentes Régions AWS, un `do-not-delete-rds-custom-` compartiment existe dans chacune Région AWS. Si vous créez une instance RDS Custom pour Oracle et deux instances RDS Custom pour SQL Server en une seule instance Région AWS, deux `do-not-delete-rds-custom-` compartiments existent.

AWS CloudTrail

RDS Custom crée automatiquement un AWS CloudTrail parcours dont le nom commence `do-not-delete-rds-custom-` par. Le périmètre de support RDS Custom s'appuie sur les événements survenus CloudTrail pour déterminer si vos actions affectent l'automatisation de RDS Custom. Pour de plus amples informations, veuillez consulter Résolution des problèmes de configurations non prises en charge.

RDS Custom crée le journal de suivi lorsque vous créez votre première instance de base de données. RDS Custom crée un journal de suivi pour chaque combinaison des éléments suivants :

- Compte AWS ID
- Type de moteur (RDS Custom for Oracle ou RDS Custom for SQL Server)
- Région AWS

Lorsque vous supprimez une instance de base de données personnalisée RDS, le code CloudTrail correspondant à cette instance n'est pas automatiquement supprimé. Dans ce cas, les données non supprimées Compte AWS continuent de vous être facturées CloudTrail. RDS Custom n'est pas responsable de la suppression de cette ressource. Pour savoir comment supprimer CloudTrail manuellement le, consultez la section Supprimer une trace dans le guide de AWS CloudTrail l'utilisateur.

Sécurité dans Amazon RDS Custom

Familiarisez-vous avec sur les considérations de sécurité pour RDS Custom.

Pour plus d'informations sur la sécurité de RDS Custom, consultez les rubriques suivantes.

- Sécurisation de votre compartiment Amazon S3 contre le problème de l'adjoint confus
- Rotation des informations d'identification RDS Custom for Oracle pour les programmes de conformité

Comment RDS Custom gère les tâches en votre nom en toute sécurité

RDS Custom utilise les outils et techniques suivants pour exécuter en toute sécurité des opérations en votre nom :

AWSServiceRoleForRDSCustom rôle lié au service

Un rôle lié à un service est prédéfini par le service et inclut toutes les autorisations requises par le service pour appeler d'autres Services AWS en votre nom. Pour RDS Custom, `AWSServiceRoleForRDSCustom` est un rôle lié à un service, défini selon le principe du moindre privilège. RDS Custom utilise les autorisations définies dans `AmazonRDSCustomServiceRolePolicy`, qui constitue la politique attachée à ce rôle, pour effectuer la plupart des tâches de provisionnement et toutes les tâches de gestion hors hôte. Pour plus d'informations, consultez Amazon RDSCustom ServiceRolePolicy.

Lorsqu'il exécute des tâches sur l'hôte, RDS Custom Automation utilise les informations d'identification du rôle lié au service pour exécuter des commandes à l'aide de. AWS Systems Manager Vous pouvez auditer l'historique des commandes via l'historique des commandes de Systems Manager et AWS CloudTrail. Systems Manager se connecte à votre instance de base de données RDS Custom à l'aide de votre configuration réseau. Pour de plus amples informations, veuillez consulter Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.

Informations d'identification IAM temporaires

Lors du provisionnement ou de la suppression de ressources, RDS Custom utilise parfois des informations d'identification temporaires dérivées des informations d'identification du principal IAM appelant. Ces informations d'identification IAM sont limitées par les politiques IAM attachées

à ce principal et expirent une fois l'opération terminée. Pour en savoir plus sur les autorisations requises pour les principaux IAM qui utilisent RDS Custom, consultez Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM.

Profil d' EC2 instance Amazon

Un profil d' EC2 instance est un conteneur pour un rôle IAM que vous pouvez utiliser pour transmettre des informations de rôle à une EC2 instance. Une EC2 instance sous-tend une instance de base de données personnalisée RDS. Vous fournissez un profil d'instance lorsque vous créez une instance de base de données RDS Custom. RDS Custom utilise les informations d'identification du profil d' EC2 instance lorsqu'il exécute des tâches de gestion basées sur l'hôte, telles que des sauvegardes. Pour de plus amples informations, veuillez consulter Créer manuellement votre profil d'instance et de rôle IAM.

Paire de clés SSH

Lorsque RDS Custom crée l' EC2 instance qui sous-tend une instance de base de données, il crée une paire de clés SSH en votre nom. La clé utilise le préfixe de dénomination `do-not-delete-rds-custom-ssh-privatekey-db-` `ourds-custom!oracle-do-not-delete-`*`db_resource_id-uuid`*`-ssh-privatekey`. AWS Secrets Manager stocke cette clé privée SSH en tant que secret dans votre Compte AWS. Amazon RDS ne stocke pas ces informations d'identification, n'y accède pas et ne les utilise pas. Pour plus d'informations, consultez les paires de EC2 clés Amazon et les instances Linux.

Certificats SSL

Les instances de base de données RDS Custom ne prennent pas en charge les certificats SSL gérés. Si vous souhaitez déployer SSL, vous pouvez gérer vous-même les certificats SSL dans votre propre portefeuille et créer un écouteur SSL pour sécuriser les connexions entre la base de données cliente ou pour la réplication de base de données. Pour en savoir plus, consultez Configuring Transport Layer Security Authentication dans la documentation Oracle Database.

Sécurisation de votre compartiment Amazon S3 contre le problème de l'adjoint confus

Lorsque vous créez une version RDS personnalisée du moteur Amazon Custom pour Oracle (CEV) ou une instance de base de données RDS personnalisée pour le SQL serveur, RDS Custom crée un compartiment Amazon S3. Le compartiment S3 stocke des fichiers tels que CEV des artefacts,

des journaux de rétablissement (transactions), des éléments de configuration pour le périmètre de support et AWS CloudTrail des journaux.

Vous pouvez rendre ces compartiments S3 plus sûrs en utilisant les clés de contexte de condition globale pour éviter le problème de l'adjoint confus. Pour de plus amples informations, veuillez consulter Prévention des problèmes d'adjoint confus entre services.

L'exemple RDS Custom for Oracle suivant montre l'utilisation des clés de contexte de condition `aws:SourceAccount` globale `aws:SourceArn` et des clés de contexte dans une politique de compartiment S3. Pour RDS Custom for Oracle, assurez-vous d'inclure les noms des ressources Amazon (ARNs) pour les instances de base de données CEVs et les instances de base de données. Pour RDS Custom for SQL Server, assurez-vous d'inclure les instances ARN de base de données.

```
...
{
  "Sid": "AWSRDSCustomForOracleInstancesObjectLevelAccess",
  "Effect": "Allow",
  "Principal": {
     "Service": "custom.rds.amazonaws.com"
  },
  "Action": [
     "s3:GetObject",
     "s3:GetObjectVersion",
     "s3:DeleteObject",
     "s3:DeleteObjectVersion",
     "s3:GetObjectRetention",
     "s3:BypassGovernanceRetention"
  ],
  "Resource": "arn:aws:s3:::do-not-delete-rds-custom-123456789012-us-east-2-c8a6f7/
RDSCustomForOracle/Instances/*",
  "Condition": {
     "ArnLike": {
        "aws:SourceArn": [
            "arn:aws:rds:us-east-2:123456789012:db:*",
            "arn:aws:rds:us-east-2:123456789012:cev:*/*"
        ]
     },
     "StringEquals": {
        "aws:SourceAccount": "123456789012"
    }
  }
},
```

```
...
```

Rotation des informations d'identification RDS Custom for Oracle pour les programmes de conformité

Certains programmes de conformité exigent que les informations d'identification des utilisateurs de la base de données soient modifiées régulièrement, par exemple tous les 90 jours. RDS Custom for Oracle alterne automatiquement les informations d'identification de certains utilisateurs de la base de données prédéfinis.

Rubriques

- Rotation automatique des informations d'identification pour les utilisateurs prédéfinis
- Instructions pour la rotation des informations d'identification des utilisateurs
- Rotation manuelle des informations d'identification des utilisateurs

Rotation automatique des informations d'identification pour les utilisateurs prédéfinis

Si votre instance de base de données RDS Custom for Oracle est hébergée sur Amazon RDS, les informations d'identification des utilisateurs Oracle prédéfinis suivants changent automatiquement tous les 30 jours. Les informations d'identification des utilisateurs précédents se trouvent dans AWS Secrets Manager.

Utilisateur de la base de donnée	Créé par	Versions de moteur prises en charge	Remarques
SYS	Oracle	custom-oracle-ee custom-oracle-ee-cdb custom-oracle-se2 custom-oracle-se2 cdb	
SYSTEM	Oracle	custom-oracle-ee custom-oracle-ee-cdb	

Utilisateur de la base de donnée	Créé par	Versions de moteur prises en charge	Remarques
		custom-oracle-se2 custom-oracle-se2 cdb	
`RDSADMIN`	RDS	custom-oracle-ee custom-oracle-se2	
`C##RDSADMIN`	RDS	custom-oracle-ee-cdb custom-oracle-se2 cdb	Les noms d'utilisateur dotés d'un `C##` préfixe n'existent que dans CDBs. Pour plus d'informations CDBs, consultez Présentation de l'architecture Amazon RDS Custom for Oracle.
`RDS_DATAGUARD`	RDS	custom-oracle-ee	Cet utilisateur existe uniquement dans les réplicas en lecture, les bases de données sources pour les réplicas en lecture et les bases de données que vous avez migrées physiquement vers RDS Custom à l'aide d'Oracle Data Guard.

Utilisateur de la base de donnée	Créé par	Versions de moteur prises en charge	Remarques
`C##RDS_DATAGUARD`	RDS	custom-oracle-ee-cdb	Cet utilisateur existe uniquement dans les réplicas en lecture, les bases de données sources pour les réplicas en lecture et les bases de données que vous avez migrées physiquement vers RDS Custom à l'aide d'Oracle Data Guard. Les noms d'utilisateur dotés d'un `C##` préfixe n'existent que dans CDBs. Pour plus d'informations CDBs, consultez Présentation de l'architecture Amazon RDS Custom for Oracle.

Une instance de base de données RDS Custom for Oracle que vous avez configurée manuellement en tant que base de données de secours fera exception à la rotation automatique des informations d'identification. RDS n'alterne les informations d'identification que pour les réplicas en lecture que vous avez créés à l'aide de la commande d'interface de ligne de commande `create-db-instance-read-replica` ou de l'API `CreateDBInstanceReadReplica`.

Instructions pour la rotation des informations d'identification des utilisateurs

Pour vous assurer que vos informations d'identification changent en fonction de votre programme de conformité, tenez compte des instructions suivantes :

- Si votre instance de base de données alterne automatiquement les informations d'identification, ne modifiez ni ne supprimez manuellement un secret, un fichier de mots de passe ou un mot de passe pour les utilisateurs répertoriés dans Utilisateurs Oracle prédéfinis. Sinon, RDS Custom risque de placer votre instance de base de données en dehors du périmètre de support, ce qui suspend la rotation automatique.

- L'utilisateur principal RDS n'est pas prédéfini. Vous êtes donc responsable de modifier le mot de passe manuellement ou de configurer la rotation automatique dans Secrets Manager. Pour plus d'informations, consultez Rotation AWS Secrets Manager des secrets.

Rotation manuelle des informations d'identification des utilisateurs

Pour les catégories de bases de données suivantes, RDS ne modifie pas automatiquement les informations d'identification des utilisateurs répertoriés dans Utilisateurs Oracle prédéfinis :

- Base de données que vous avez configurée manuellement pour fonctionner en tant que base de données de secours.
- Bases de données sur site.
- Instance de base de données située en dehors du périmètre de support ou dans un état dans lequel l'automatisation personnalisée RDS ne peut pas s'exécuter. Dans ce cas, RDS Custom n'effectue pas non plus une rotation des clés.

Si votre base de données appartient à l'une des catégories précédentes, vous devez effectuer une rotation manuelle de vos informations d'identification utilisateur.

Pour effectuer une rotation manuelle des informations d'identification utilisateur pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans Dases de données, assurez-vous que RDS ne sauvegarde pas actuellement votre instance de base de données ou n'effectue aucune opération telle que la configuration de la haute disponibilité.
3. Sur la page de détails de la base de données, choisissez Configuration et notez l'ID de ressource de l'instance de base de données. Vous pouvez également utiliser la AWS CLI commande`describe-db-instances`.
4. Ouvrez la console Secrets Manager à l'adresse https://console.aws.amazon.com/secretsmanager/.
5. Dans le champ de recherche, entrez l'ID de ressource de votre base de données et recherchez un secret en utilisant l'une des conventions de dénomination suivantes :

```
do-not-delete-rds-custom-resource_id-uuid
```

```
rds-custom!oracle-do-not-delete-resource_id-uuid
```

Ce secret enregistre le mot de passe pour RDSADMIN, SYS et SYSTEM. Les exemples de clés suivants concernent l'instance de base de données avec l'ID de ressource db-ABCDEFG12HIJKLNMNOPQRS3TUVWX et l'123456UUID :

```
do-not-delete-rds-custom-db-ABCDEFG12HIJKLNMNOPQRS3TUVWX-123456
rds-custom!oracle-do-not-delete-db-ABCDEFG12HIJKLNMNOPQRS3TUVWX-123456
```

> **Important**
>
> Si votre instance de base de données est un réplica en lecture et utilise le moteur custom-oracle-ee-cdb, deux secrets existent avec le suffixe *db-resource_id-uuid*, l'un pour l'utilisateur principal et l'autre pour RDSADMIN, SYS et SYSTEM. Pour trouver le secret correct, exécutez la commande suivante sur l'hôte :
>
> ```
> cat /opt/aws/rdscustomagent/config/database_metadata.json | python3 -c
>  "import sys,json; print(json.load(sys.stdin)['dbMonitoringUserPassword'])"
> ```
>
> L'attribut dbMonitoringUserPassword indique le secret pour RDSADMIN, SYS et SYSTEM.

6. Si votre instance de base de données existe dans une configuration Oracle Data Guard, recherchez un secret en utilisant l'une des conventions de dénomination suivantes :

```
do-not-delete-rds-custom-resource_id-uuid-dg
rds-custom!oracle-do-not-delete-resource_id-uuid-dg
```

Ce secret enregistre le mot de passe pour RDS_DATAGUARD. Les exemples de clés suivants concernent l'instance de base de données avec l'ID de ressource de base de données **db-ABCDEFG12HIJKLNMNOPQRS3TUVWX** et l'UUID 789012 :

```
do-not-delete-rds-custom-db-ABCDEFG12HIJKLNMNOPQRS3TUVWX-789012-dg
rds-custom!oracle-do-not-delete-db-ABCDEFG12HIJKLNMNOPQRS3TUVWX-789012-dg
```

7. Pour tous les utilisateurs de base de données répertoriés dans Utilisateurs Oracle prédéfinis, mettez à jour les mots de passe en suivant les instructions de la section Modifier un AWS Secrets Manager secret.

8. Si votre base de données est une base de données autonome ou une base de données source dans une configuration Oracle Data Guard :

   a. Démarrez votre client Oracle SQL et connectez-vous en tant que SYS.

   b. Exécutez une instruction SQL sous la forme suivante pour chaque utilisateur de base de données répertorié dans Utilisateurs Oracle prédéfinis :

   ```
   ALTER USER user-name IDENTIFIED BY pwd-from-secrets-manager ACCOUNT UNLOCK;
   ```

   Par exemple, si le nouveau mot de passe pour RDSADMIN enregistré dans Secrets Manager est pwd-123, exécutez l'instruction suivante :

   ```
   ALTER USER RDSADMIN IDENTIFIED BY pwd-123 ACCOUNT UNLOCK;
   ```

9. Si votre instance de base de données exécute Oracle Database 12c version 1 (12.1) et qu'elle est gérée par Oracle Data Guard, copiez manuellement le fichier de mots de passe (orapw) de l'instance de base de données principale vers chaque instance de base de données de secours.

   Si votre instance de base de données est hébergée sur Amazon RDS, l'emplacement du fichier de mots de passe est /rdsdbdata/config/orapw. Pour les bases de données qui ne sont pas hébergées dans Amazon RDS, l'emplacement par défaut est $ORACLE_HOME/dbs/orapw$ORACLE_SID sous Linux et UNIX, et %ORACLE_HOME%\database\PWD%ORACLE_SID%.ora sous Windows.

Utilisation de RDS Custom for Oracle

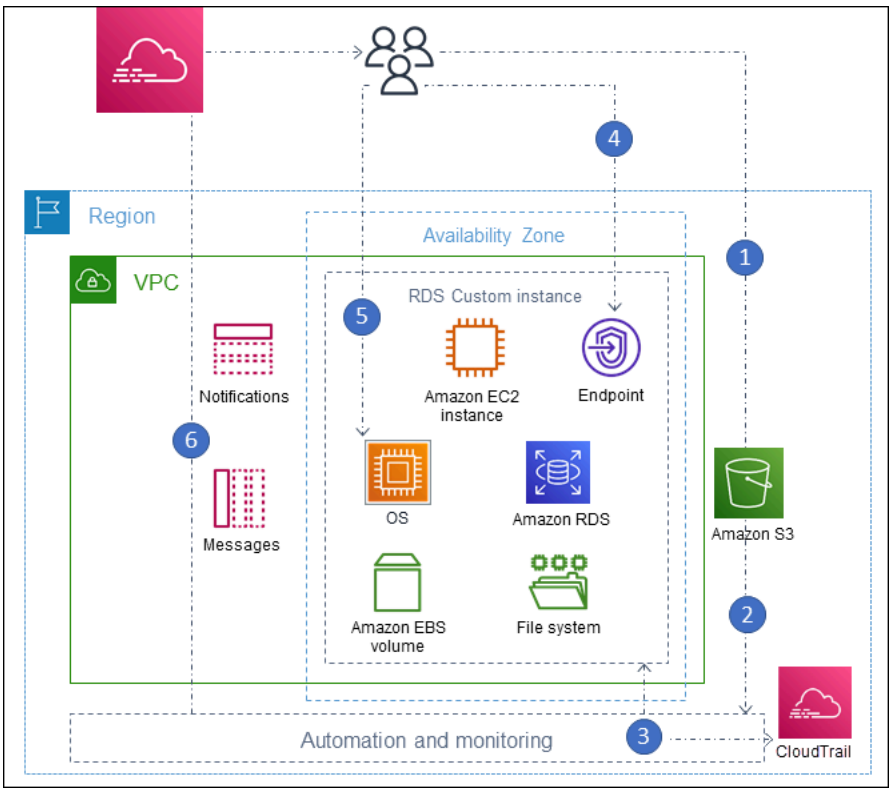
Vous trouverez ci-dessous des instructions pour la création, la gestion et la maintenance de vos instances de base de données RDS Custom for Oracle.

Rubriques

- Flux de travail RDS Custom for Oracle
- Architecture de base de données pour Amazon RDS Custom for Oracle
- Disponibilité des fonctionnalités et support pour RDS Custom pour Oracle
- Exigences et limites de RDS Custom for Oracle
- Configuration de votre environnement pour Amazon RDS Custom for Oracle
- Utilisation de versions de moteurs personnalisées pour Amazon RDS Custom for Oracle
- Configuration d'une instance de base de données pour Amazon RDS Custom for Oracle
- Gestion d'une instance de base de données Amazon RDS Custom pour Oracle
- Gestion d'un déploiement multi-AZ pour RDS Custom for Oracle
- Utilisation de répliques Oracle pour RDS Custom for Oracle
- Sauvegarde et restauration d'une instance de base de données Amazon RDS Custom pour Oracle
- Utilisation de groupes d'options dans RDS Custom for Oracle
- Migration d'une base de données locale vers RDS Custom for Oracle
- Mise à niveau d'une instance de base de données pour Amazon RDS Custom for Oracle
- Résolution des problèmes de base de données pour Amazon RDS Custom for Oracle
- Problèmes connus relatifs à Amazon RDS Custom pour Oracle

Flux de travail RDS Custom for Oracle

Le diagramme suivant illustre le flux de travail typique de RDS Custom for Oracle.



La procédure est la suivante :

1. Chargez votre logiciel de base de données dans votre compartiment Amazon S3.

   Pour de plus amples informations, veuillez consulter Étape 3 : Charger vos fichiers d'installation dans Amazon S3.
2. Créez une version de moteur personnalisée (CEV) RDS Custom for Oracle à partir de votre média.

   Choisissez l'architecture CDB ou l'architecture traditionnelle non CDB. Pour de plus amples informations, veuillez consulter Création d'un CEV.
3. Créez une instance de base de données RDS Custom for Oracle à partir d'une CEV.

   Pour de plus amples informations, veuillez consulter Création d'une instance de base de données RDS Custom for Oracle.
4. Connectez votre application au point de terminaison de l'instance de base de données.

   Pour plus d'informations, consultez Connexion à votre instance de base de données RDS Custom à l'aide de SSH et Connexion à votre instance de base de données RDS personnalisée à l'aide du gestionnaire de session.
5. (Facultatif) Accédez à l'hôte pour personnaliser votre logiciel.

6. Surveillez les notifications et les messages générés par l'automatisation de RDS Custom.

Fichiers d'installation de base de données

Votre responsabilité pour les supports est une différence majeure entre Amazon RDS et RDS Custom. Amazon RDS, qui est un service entièrement géré, fournit l'Amazon Machine Image (AMI) et le logiciel de base de données. Le logiciel de base de données Amazon RDS est préinstallé. Il vous suffit donc de choisir un moteur et une version de base de données, et de créer votre base de données.

Pour RDS Custom, vous fournissez vos propres supports. Lorsque vous créez une version de moteur personnalisée, RDS Custom installe le support que vous fournissez. Le support RDS Custom contient les fichiers d'installation et les correctifs de votre base de données. Ce modèle de service s'appelle Bring Your Own Media (BYOM).

Versions de moteurs personnalisées pour RDS Custom for Oracle

Une version du moteur personnalisée RDS Custom for Oracle (CEV) est un instantané de volume binaire d'une version de base de données et d'une AMI. Par défaut, RDS Custom for Oracle utilise l'AMI la plus récente mise à disposition par Amazon EC2 . Vous pouvez également choisir de réutiliser une AMI existante.

Manifeste CEV

Après avoir téléchargé les fichiers d'installation de la base de données Oracle depuis Oracle, vous les chargez dans un compartiment Amazon S3. Lorsque vous créez votre version CEV, vous spécifiez les noms de fichier dans un document JSON appelé manifeste CEV. RDS Custom for Oracle utilise les fichiers spécifiés et l'image AMI pour créer votre version CEV.

RDS Custom for Oracle fournit des modèles de manifeste JSON avec les fichiers .zip recommandés pour chaque version d'Oracle Database prise en charge. Par exemple, le modèle suivant concerne le 19.17.0.0.0 RU.

```
{
    "mediaImportTemplateVersion": "2020-08-14",
    "databaseInstallationFileNames": [
        "V982063-01.zip"
    ],
    "opatchFileNames": [
```

```
        "p6880880_190000_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames": [
        "p34419443_190000_Linux-x86-64.zip",
        "p34411846_190000_Linux-x86-64.zip"
    ],
    "otherPatchFileNames": [
        "p28852325_190000_Linux-x86-64.zip",
        "p29997937_190000_Linux-x86-64.zip",
        "p31335037_190000_Linux-x86-64.zip",
        "p32327201_190000_Linux-x86-64.zip",
        "p33613829_190000_Linux-x86-64.zip",
        "p34006614_190000_Linux-x86-64.zip",
        "p34533061_190000_Linux-x86-64.zip",
        "p34533150_190000_Generic.zip",
        "p28730253_190000_Linux-x86-64.zip",
        "p29213893_1917000DBRU_Generic.zip",
        "p33125873_1917000DBRU_Linux-x86-64.zip",
        "p34446152_1917000DBRU_Linux-x86-64.zip"
    ]
}
```

Vous pouvez également spécifier les paramètres d'installation dans le manifeste JSON. Par exemple, vous pouvez définir des valeurs autres que celles par défaut pour la base Oracle, le répertoire d'origine Oracle Home, ainsi que l'ID et le nom de l' UNIX/Linux utilisateur et du groupe. Pour de plus amples informations, veuillez consulter Champs JSON dans le manifeste CEV.

Format de nommage CEV

Nommez votre CEV RDS Custom for Oracle à l'aide d'une chaîne spécifiée par le client. Le format de nom est le suivant, en fonction de la version d'Oracle Database :

- `19.`*`customized_string`*
- `18.`*`customized_string`*
- `12.2.`*`customized_string`*
- `12.1.`*`customized_string`*

Vous pouvez utiliser de 1 à 50 caractères alphanumériques, des traits de soulignement, des tirets et des points. Par exemple, vous pouvez nommer votre CEV `19.my_cev1`.

Architecture multilocataire Oracle dans RDS Custom for Oracle

L'architecture multilocataire Oracle permet à une base de données Oracle de fonctionner en tant que base de données de conteneur (CDB). Un CDB inclut zéro, une ou plusieurs bases de données enfichables créées par le client (). PDBs Une PDB est une collection portable de schémas et d'objets qui apparaît à une application en tant que base de données non CDB traditionnelle.

Lorsque vous créez un CEV RDS Custom for Oracle, vous indiquez une architecture CDB ou non CDB. Vous pouvez créer une CDB RDS Custom for Oracle uniquement quand la version CEV que vous avez utilisée pour la créer utilise l'architecture multilocataire Oracle. Pour de plus amples informations, veuillez consulter Utilisation de versions de moteurs personnalisées pour Amazon RDS Custom for Oracle.

Création d'une instance de base de données RDS Custom for Oracle

Une fois que vous avez créé la version CEV, elle est disponible. Vous pouvez en créer plusieurs CEVs, et vous pouvez créer plusieurs instances de base de données RDS Custom pour Oracle à partir de n'importe quel CEV. Vous pouvez également modifier le statut d'une CEV pour la rendre disponible ou inactive.

Vous pouvez créer votre instance de base de données RDS Custom pour Oracle avec l'architecture mutualisée Oracle (`custom-oracle-ee-cdb`ou type de `custom-oracle-se2-cdb` moteur) ou avec l'architecture traditionnelle non CDB (`custom-oracle-ee`ou `custom-oracle-se2` type de moteur). Lorsque vous créez une base de données de conteneurs (CDB), elle contient une base de données enfichable (PDB) et un contenur initial de PDB. Vous pouvez en créer d'autres PDBs manuellement à l'aide d'Oracle SQL.

Pour créer votre instance de base de données RDS Custom for Oracle, utilisez la commande `create-db-instance`. Dans cette commande, spécifiez la CEV à utiliser. La procédure est similaire à la création d'une instance de base de données Amazon RDS. Toutefois, certains paramètres sont différents. Pour de plus amples informations, veuillez consulter Configuration d'une instance de base de données pour Amazon RDS Custom for Oracle.

Connexion de la base de données

Comme avec une instance de base de données Amazon RDS, une instance de base de données RDS Custom réside dans un cloud privé virtuel (VPC). Votre application se connecte à la base de données Oracle à l'aide d'un écouteur Oracle.

Si votre base de données est une CDB, vous pouvez utiliser l'écouteur `L_RDSCDB_001` pour vous connecter à la racine CDB et à une PDB. Si vous connectez une base de données non-CDB à une CDB, assurez-vous de définir `USE_SID_AS_SERVICE_LISTENER = ON` de manière à ce que les applications migrées conservent les mêmes paramètres.

Lorsque vous vous connectez à une base de données non-CDB, l'utilisateur principal est l'utilisateur de la base de données non-CDB. Lorsque vous vous connectez à une CDB, l'utilisateur principal est l'utilisateur de la PDB. Pour vous connecter à la racine CDB, connectez-vous à l'hôte, démarrez un client SQL et créez un utilisateur administratif à l'aide de commandes SQL.

Personnalisation de RDS Custom

Vous pouvez accéder à l'hôte RDS Custom pour installer ou personnaliser le logiciel. Pour éviter les conflits entre vos modifications et l'automatisation RDS Custom, vous pouvez suspendre l'automatisation pendant une période spécifiée. Pendant cette période, RDS Custom n'effectue pas de surveillance ou de récupération d'instance. À la fin de la période, RDS Custom reprend l'automatisation complète. Pour de plus amples informations, veuillez consulter Suspension et reprise de votre RDS instance de base de données personnalisée.

Architecture de base de données pour Amazon RDS Custom for Oracle

RDS Custom for Oracle prend en charge à la fois l'architecture multilocataire et non multilocataire Oracle.

Rubriques

- Architectures de base de données Oracle prises en charge
- Types de moteur pris en charge
- Fonctionnalités prises en charge dans l'architecture mutualisée Oracle

Architectures de base de données Oracle prises en charge

L'architecture multilocataire Oracle, également appelée architecture CDB, permet à une base de données Oracle de fonctionner comme une base de données de conteneur (CDB). Une CDB inclut des bases de données enfichables (PDB). Une PDB est une collection de schémas et d'objets qui apparaît à une application en tant que base de données Oracle traditionnelle. Pour plus d'informations, consultez Introduction à l'architecture multilocataire (langue française non garantie) dans le Guide de l'administrateur d'Oracle Multitenant.

Les architectures CDB et non CDB s'excluent mutuellement. Si une base de données Oracle n'est pas une CDB, c'est une base de données non CDB et elle ne peut donc pas contenir de PDB. Dans RDS Custom for Oracle, seul Oracle Database 19c prend en charge l'architecture CDB. Ainsi, si vous créez des instances de bases de données à l'aide de versions de base de données Oracle antérieures, vous ne pouvez créer que des bases de données non CDB. Pour plus d'informations, consultez Considérations relatives à l'architecture multilocataire.

Types de moteur pris en charge

Lorsque vous créez une instance Amazon RDS Custom pour Oracle CEV ou DB, choisissez un type de moteur CDB ou un type de moteur non CDB :

- `custom-oracle-ee-cdb` et `custom-oracle-se2-cdb`

  Ces types de moteurs définissent l'architecture mutualisée d'Oracle. Cette option est disponible uniquement pour Oracle Database 19c. Lorsque vous créez une instance de base de données RDS for Oracle à l'aide de l'architecture multilocataire, votre CDB inclut les conteneurs suivants :

  - Racine CDB (`CDB$ROOT`)
  - Conteneur initial de PDB (`PDB$SEED`)

  - PDB initiale

    Vous pouvez créer d'autres PDB à l'aide de la commande Oracle SQL `CREATE PLUGGABLE DATABASE`. Vous ne pouvez pas utiliser les API RDS pour créer ou supprimer des PDB.
- `custom-oracle-ee` et `custom-oracle-se2`

  Ces types de moteurs spécifient l'architecture traditionnelle non CDB. Une base de données non-CDB ne peut pas contenir de bases de données enfichables (PDB).

Pour plus d’informations, consultez Considérations relatives à l'architecture multilocataire.

Fonctionnalités prises en charge dans l'architecture mutualisée Oracle

Une instance de CDB RDS Custom for Oracle prend en charge les fonctionnalités suivantes :

- Sauvegardes
- Restauration et point-time-restore (PITR) à partir de sauvegardes
- Réplicas en lecture
- Mises à niveau de version mineure.

Disponibilité des fonctionnalités et support pour RDS Custom pour Oracle

Dans cette rubrique, vous trouverez un résumé de la disponibilité et du support des fonctionnalités RDS Custom for Oracle à des fins de référence rapide.

Rubriques

- Région AWS et prise en charge des versions de base de données pour RDS Custom pour Oracle
- Support des versions de base de données pour RDS Custom pour Oracle
- Prise en charge en matière d'édition et de licence pour RDS Custom for Oracle
- Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle
- Support des groupes d'options pour RDS Custom pour Oracle

Région AWS et prise en charge des versions de base de données pour RDS Custom pour Oracle

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour plus d'informations sur la disponibilité par version et par région de RDS Custom for Oracle, consultez Régions et moteurs de base de données pris en charge pour RDS Custom.

Support des versions de base de données pour RDS Custom pour Oracle

RDS Custom for Oracle prend en charge les versions de base de données Oracle suivantes :

- Oracle Database 19c
- Oracle Database 18c
- Oracle Database 12c version 2 (12.2)
- Oracle Database 12c version 1 (12.1)

Prise en charge en matière d'édition et de licence pour RDS Custom for Oracle

RDS Custom pour Oracle prend en charge les éditions Enterprise (EE) et Standard Edition 2 (SE2) sur le modèle BYOL.

Notez les limites suivantes pour l'édition Standard 2 :

- Oracle Data Guard n'est pas pris en charge. Vous ne pouvez donc pas créer de répliques de lecture Oracle.
- Vous ne pouvez utiliser que des classes d'instance de base de données dont la valeur est inférieure ou égale à 16 v CPUs (jusqu'à 4 x large).
- Une instance CDB sur Standard Edition 2 prend en charge un maximum de 3 bases de données mutualisées.
- Vous ne pouvez pas migrer des données entre Enterprise Edition et Standard Edition 2.

Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle

RDS Custom for Oracle prend en charge les classes d'instances de base de données suivantes. Si vous créez une instance de base de données sur Standard Edition 2, vous ne pouvez utiliser que des classes d'instance de 16 v ou moins CPUs (jusqu'à 4 fois plus grandes).

Type	Size
db.r7i	`db.r7i.large |db.r7i.xlarge |db.r7i.2xlarge |db.r7i.4xlarge |db.r7i.8xlarge |db.r7i.12xlarge |db.r7i.16xlarge |db.r7i.24xlarge |db.r7i.48xlarge`
db.r6i	`db.r6i.large |db.r6i.xlarge |db.r6i.2xlarge |db.r6i.4xlarge |db.r6i.8xlarge |db.r6i.12xlarge |db.r6i.16xlarge |db.r6i.24xlarge |db.r6i.32xlarge`
db.r5b	`db.r5b.large |db.r5b.xlarge |db.r5b.2xlarge |db.r5b.4xlarge |db.r5b.8xlarge |db.r5b.12xlarge |db.r5b.16xlarge |db.r5b.24xlarge`
db.r5	`db.r5.large |db.r5.xlarge |db.r5.2xlarge |db.r5.4xlarge | db.r5.8xlarge |db.r5.12xlarge |db.r5.16xlarge |db.r5.24xlarge`
db.x2iec	`db.x2iedn.xlarge |db.x2iedn.2xlarge |db.x2iedn.4xlarge | db.x2iedn.8xlarge |db.x2iedn.16xlarge |db.x2iedn.24xlarge | db.x2iedn.32xlarge`

Type	Size
db.x2iez	db.x2iezn.2xlarge |db.x2iezn.4xlarge |db.x2iezn.6xlarge | db.x2iezn.8xlarge |db.x2iezn.12xlarge
db.m7i	db.m7i.large |db.m7i.xlarge |db.m7i.2xlarge |db.m7i.4xlarge |db.m7i.8xlarge |db.m7i.12xlarge |db.m7i.16xlarge |db.m7i.24xlarge |db.m7i.48xlarge
db.m6i	db.m6i.large |db.m6i.xlarge |db.m6i.2xlarge |db.m6i.4xlarge |db.m6i.8xlarge |db.m6i.12xlarge |db.m6i.16xlarge |db.m6i.24xlarge |db.m6i.32xlarge
db.m5	db.m5.large |db.m5.xlarge |db.m5.2xlarge |db.m5.4xlarge | db.m5.8xlarge |db.m5.12xlarge |db.m5.16xlarge |db.m5.24xlarge
db.t3	db.t3.medium |db.t3.large |db.t3.xlarge |db.t3.2xlarge

Support des groupes d'options pour RDS Custom pour Oracle

Vous pouvez spécifier un groupe d'options lorsque vous créez ou modifiez une instance de base de données RDS Custom pour Oracle. Pour de plus amples informations, veuillez consulter Utilisation de groupes d'options dans RDS Custom for Oracle.

Exigences et limites de RDS Custom for Oracle

Dans cette rubrique, vous trouverez un résumé des exigences et de la disponibilité des fonctionnalités Amazon RDS Custom for Oracle pour une référence rapide.

Rubriques

- Exigences générales pour RDS Custom for Oracle
- Limites générales pour RDS Custom for Oracle
- Limitations du CEV et de l'AMI pour RDS Custom pour Oracle
- Paramètres non pris en charge pour créer et modifier des flux de travail
- Quotas d'instance de base de données pour votre Compte AWS

Exigences générales pour RDS Custom for Oracle

Assurez-vous de respecter les exigences suivantes pour Amazon RDS Custom pour Oracle :

- Vous avez accès à My Oracle Support et à Oracle Software Delivery Cloud pour télécharger la liste des fichiers d'installation et des correctifs pris en charge pour RDS Custom for Oracle. Si vous utilisez un correctif inconnu, la création d'une version de moteur personnalisée (CEV) échoue. Dans ce cas, contactez l'équipe du service clientèle RDS Custom pour demander l'ajout du correctif manquant. Pour de plus amples informations, veuillez consulter Étape 2 : Télécharger des fichiers d'installation de votre base de données et des correctifs à partir d'Oracle Software Delivery Cloud.
- Vous avez accès à Amazon S3. Vous avez besoin de ce service pour les raisons suivantes :
  - Vous chargez vos fichiers d'installation Oracle dans des compartiments S3. Vous utilisez les fichiers d'installation chargés lors de la création de votre CEV RDS Custom.
  - RDS Custom for Oracle utilise des scripts téléchargés à partir de compartiments S3 définis en interne pour effectuer des actions sur vos instances de base de données. Ces scripts sont nécessaires à l'intégration et à l'automatisation de RDS Custom.
  - RDS Custom for Oracle télécharge certains fichiers vers des compartiments S3 situés dans votre compte client. Ces compartiments utilisent le format de dénomination suivant : `do-not-delete-rds-custom-` *`account_id`* `-` *`region`* `-` *`uuid`*. Par exemple, vous pouvez avoir un compartiment nommé `do-not-delete-rds-custom-123456789012-us-east-1-12a3b4`.

  Pour plus d'informations, consultez Étape 3 : Charger vos fichiers d'installation dans Amazon S3 et Création d'un CEV.

- Vous utilisez les classes d'instances de base de données répertoriées dans le Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle document pour créer votre RDS Custom pour les instances de base de données Oracle.
- Vos instances de base de données RDS Custom pour Oracle exécutent Oracle Linux 8 (recommandé) ou Oracle Linux 7. Si vous avez besoin d'Oracle Linux 7, contactez Support. Pour de plus amples informations, veuillez consulter Considérations relatives aux mises à niveau des bases de données RDS Custom for Oracle.
- Vous spécifiez les disques SSD gp2, gp3 ou io1 pour le stockage Amazon EBS. La taille de stockage maximale est de 64 TiB.
- Vous disposez d'une AWS KMS clé pour créer une instance de base de données personnalisée RDS pour Oracle. Pour de plus amples informations, veuillez consulter Étape 1 : Créer ou réutiliser une clé de chiffrement symétrique AWS KMS.
- Vous disposez du rôle AWS Identity and Access Management (IAM) et du profil d'instance requis pour créer des instances de base de données RDS Custom pour Oracle. Pour de plus amples informations, veuillez consulter Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.
- L'utilisateur AWS Identity and Access Management (IAM) qui crée une instance de base de données personnalisée CEV ou RDS dispose des autorisations requises pour IAM et CloudTrail Amazon S3.

  Pour de plus amples informations, veuillez consulter Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM.
- Vous fournissez votre propre configuration de cloud privé virtuel (VPC) et de groupe de sécurité. Pour de plus amples informations, veuillez consulter Étape 6 : Configuration de votre VPC pour RDS Custom pour Oracle.
- Vous fournissez une configuration réseau que RDS Custom for Oracle peut utiliser pour accéder à d'autres Services AWS. Pour voir les conditions requises spécifiques, consultez Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.

Limites générales pour RDS Custom for Oracle

Les limites suivantes s'appliquent à RDS Custom for Oracle :

- Vous ne pouvez pas modifier l'identifiant d'instance de base de données d'une instance de base de données RDS Custom for Oracle existante.
- Vous ne pouvez pas spécifier l'architecture mutualisée Oracle pour aucune version, à l'exception d'Oracle Database 19c.

- Vous ne pouvez pas créer plusieurs bases de données Oracle sur une seule instance de base de données RDS Custom for Oracle.
- Vous ne pouvez pas arrêter votre instance de base de données RDS Custom for Oracle ou son EC2 instance Amazon sous-jacente. La facturation d'une instance de base de données RDS Custom for Oracle ne peut pas être arrêtée.
- Vous ne pouvez pas utiliser la gestion automatique de la mémoire partagée car RDS Custom for Oracle prend uniquement en charge la gestion automatique de la mémoire. Pour plus d'informations, consultez la section Automatic Memory Management (Gestion automatique de la mémoire) dans Oracle Database Administrator's Guide (Guide de l'administrateur des bases de données Oracle).
- Veillez à ne pas modifier `DB_UNIQUE_NAME` pour l'instance de base de données principale. La modification du nom entraîne le blocage de toute opération de restauration.
- Vous ne pouvez pas créer plus de 20 copies instantanées en même temps dans la même région.
- Vous ne pouvez pas utiliser l'`describe-reserved-db-instances`API pour RDS Custom pour les instances de base de données Oracle.

Pour connaître les limites spécifiques à la modification d'une instance de base données RDS Custom for Oracle, consultez Modification de votre instance de base de données RDS personnalisée pour Oracle. Pour connaître les limites de réplication, consultez Limites générales pour la réplication RDS Custom for Oracle.

Limitations du CEV et de l'AMI pour RDS Custom pour Oracle

Les limites suivantes s'appliquent à RDS Custom pour Oracle CEVs et AMIs :

- Vous ne pouvez pas fournir votre propre AMI à utiliser dans un RDS Custom pour Oracle CEV. Vous pouvez spécifier soit l'AMI par défaut, qui utilise Oracle Linux 8, soit une AMI qui a déjà été utilisée par un RDS Custom pour Oracle CEV.

  > **Note**
  >
  > RDS Custom for Oracle publie une nouvelle AMI par défaut lorsque des vulnérabilités et des risques courants sont découverts. Aucun horaire fixe n'est disponible ou garanti. RDS Custom for Oracle a tendance à publier une nouvelle AMI par défaut tous les 30 jours.

- Vous ne pouvez pas modifier une CEV pour utiliser une autre AMI.

- Vous ne pouvez pas créer une instance CDB à partir d'un CEV qui utilise les types de `custom-oracle-se2` moteurs `custom-oracle-ee` OR. Le CEV doit utiliser `custom-oracle-ee-cdb` ou`custom-oracle-se2-cdb`.
- RDS Custom for Oracle ne vous permet actuellement pas de mettre à niveau le système d'exploitation de votre instance de base de données RDS Custom for Oracle avec des appels d'API RDS. Pour contourner le problème, vous pouvez mettre à jour votre système d'exploitation manuellement à l'aide de la commande suivante :`sudo yum update --security`.

Paramètres non pris en charge pour créer et modifier des flux de travail

Lorsque vous créez ou modifiez une instance de base de données RDS Custom pour Oracle, vous ne pouvez pas effectuer les opérations suivantes :

- Modifier le nombre de cœurs d'UC et de threads par cœur sur la classe d'instance de base de données.
- Activer la scalabilité automatique du stockage.
- Définir la période de rétention des sauvegardes sur `0`.
- Configurer l'authentification Kerberos.
- Indiquez votre propre groupe de paramètres de base de données ou groupe d'options.
- Activer l'option Performance Insights.
- Activer la mise à niveau automatique des versions mineures

Quotas d'instance de base de données pour votre Compte AWS

Assurez-vous que le nombre combiné d'instances de base de données RDS Custom et Amazon RDS ne dépasse pas votre limite de quota. Par exemple, si votre quota pour Amazon RDS est de 40 instances de base de données, vous pouvez avoir 20 instances de base de données RDS Custom for Oracle et 20 instances de base de données Amazon RDS.

Configuration de votre environnement pour Amazon RDS Custom for Oracle

Avant de créer une instance de base de données Amazon RDS Custom for Oracle, effectuez les tâches suivantes.

Rubriques

- Étape 1 : Créer ou réutiliser une clé de chiffrement symétrique AWS KMS
- Étape 2 : Téléchargez et installez AWS CLI
- Étape 3 : Extraire les CloudFormation modèles pour RDS Custom pour Oracle
- Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle
- Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM
- Étape 6 : Configuration de votre VPC pour RDS Custom pour Oracle

Étape 1 : Créer ou réutiliser une clé de chiffrement symétrique AWS KMS

Les clés gérées par le client se trouvent AWS KMS keys dans votre AWS compte que vous créez, détenez et gérez. Une clé KMS de chiffrement symétrique gérée par le client est requise pour RDS Custom. Lorsque vous créez une instance de base de données RDS Custom for Oracle, vous indiquez l'identificateur de clé KMS. Pour de plus amples informations, veuillez consulter Configuration d'une instance de base de données pour Amazon RDS Custom for Oracle.

Vous avez les options suivantes :

- Si vous possédez déjà une clé KMS gérée par le client Compte AWS, vous pouvez l'utiliser avec RDS Custom. Aucune action supplémentaire n'est nécessaire.
- Si vous avez déjà créé une clé KMS de chiffrement symétrique gérée par le client pour un moteur RDS Custom différent, vous pouvez réutiliser la même clé KMS. Aucune action supplémentaire n'est nécessaire.
- Si votre compte ne contient pas encore de clés de chiffrement KMS symétriques gérées par le client, créez-en une en suivant les instructions de la section Creating keys (Création de clés) dans le Guide du développeur AWS Key Management Service .
- Si vous créez une instance de base de données personnalisée CEV ou RDS et que votre clé KMS se trouve dans une autre Compte AWS, assurez-vous d'utiliser le. AWS CLI Vous ne pouvez pas utiliser la AWS console avec des clés KMS entre comptes.

> **Important**
>
> RDS Custom ne prend pas en charge les clés KMS AWS gérées.

Assurez-vous que votre clé de chiffrement symétrique autorise l'accès au rôle (IAM) `kms:Decrypt` et aux `kms:GenerateDataKey` opérations du rôle AWS Identity and Access Management (IAM) dans votre profil d'instance IAM. Si votre compte contient une nouvelle clé de chiffrement symétrique, aucune modification n'est requise. Sinon, veillez à ce que la stratégie de votre clé de chiffrement symétrique puisse fournir l'accès à ces opérations.

Pour de plus amples informations, veuillez consulter Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.

Pour plus d'informations sur la configuration d'IAM pour RDS Custom for Oracle, consultez Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.

Étape 2 : Téléchargez et installez AWS CLI

AWS vous fournit une interface de ligne de commande pour utiliser les fonctionnalités personnalisées de RDS. Vous pouvez utiliser la version 1 ou 2 d' AWS CLI.

Pour plus d'informations sur le téléchargement et l'installation du AWS CLI, voir Installation ou mise à jour de la dernière version du AWS CLI.

Ignorez cette étape si l'une des conditions suivantes est vraie :

- Vous prévoyez d'accéder à RDS Custom uniquement à partir du AWS Management Console.
- Vous avez déjà téléchargé le AWS CLI pour Amazon RDS ou un autre moteur de base de données personnalisé RDS.

Étape 3 : Extraire les CloudFormation modèles pour RDS Custom pour Oracle

Pour simplifier la configuration, nous vous recommandons vivement d'utiliser des AWS CloudFormation modèles pour créer des CloudFormation piles. Si vous prévoyez de configurer IAM et votre VPC manuellement, ignorez cette étape.

Rubriques

- Étape 3a : Téléchargez les fichiers CloudFormation modèles
- Étape 3b : Extraire le custom-oracle-iam fichier .json
- Étape 3c : Extraire custom-vpc.json

Étape 3a : Téléchargez les fichiers CloudFormation modèles

Un CloudFormation modèle est une déclaration des AWS ressources qui constituent une pile. Le modèle est stocké sous la forme d'un fichier JSON.

Pour télécharger les fichiers CloudFormation modèles

1. Ouvrez le menu contextuel (clic droit) du lien custom-oracle-iam.zip et choisissez Enregistrer le lien sous.
2. Enregistrez le fichier sur votre ordinateur.
3. Répétez les étapes précédentes pour le lien custom-vpc.zip.

   Si vous avez déjà configuré votre VPC pour RDS Custom, ignorez cette étape.

Étape 3b : Extraire le custom-oracle-iam fichier .json

Ouvrez le `custom-oracle-iam.zip` fichier que vous avez téléchargé, puis extrayez-le`custom-oracle-iam.json`. Le début du fichier se présente comme suit.

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Parameters": {
    "EncryptionKey": {
      "Type": "String",
      "Default": "*",
      "Description": "KMS Key ARN for encryption of data managed by RDS Custom and by
 DB Instances."
    }
  },
  "Resources": {
    "RDSCustomInstanceServiceRole": {
      "Type": "AWS::IAM::Role",
      "Properties": {
        "RoleName": { "Fn::Sub": "AWSRDSCustomInstanceRole-${AWS::Region}" },
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
```

```
          "Statement": [
            {
              "Action": "sts:AssumeRole",
              "Effect": "Allow",
              "Principal": {
                "Service": "ec2.amazonaws.com"
              }
            }
          ]
        },...
```

Étape 3c : Extraire custom-vpc.json

> **Note**
>
> Si vous avez déjà configuré un VPC existant pour RDS Custom for Oracle, ignorez cette étape. Pour de plus amples informations, veuillez consulter Configurez votre VPC manuellement pour RDS Custom for Oracle.

Ouvrez `custom-vpc.zip` le fichier que vous avez téléchargé, puis extrayez-le. `custom-vpc.json` Le début du fichier se présente comme suit.

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Parameters": {
    "PrivateVpc": {
      "Type": "AWS::EC2::VPC::Id",
      "Description": "Private VPC Id to use for RDS Custom DB Instances"
    },
    "PrivateSubnets": {
      "Type": "List<AWS::EC2::Subnet::Id>",
      "Description": "Private Subnets to use for RDS Custom DB Instances"
    },
    "RouteTable": {
      "Type": "String",
      "Description": "Route Table that must be associated with the PrivateSubnets and
 used by S3 VPC Endpoint",
      "AllowedPattern": "rtb-[0-9a-z]+"
    }
  },
  "Resources": {
```

```
    "DBSubnetGroup": {
      "Type": "AWS::RDS::DBSubnetGroup",
      "Properties": {
        "DBSubnetGroupName": "rds-custom-private",
        "DBSubnetGroupDescription": "RDS Custom Private Network",
        "SubnetIds": {
          "Ref": "PrivateSubnets"
        }
      }
    },...
```

Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle

Vous utilisez un rôle IAM ou un utilisateur IAM (appelé entité IAM) pour créer une instance de base de données RDS Custom à l'aide de la console ou de l' AWS CLI. Cette entité IAM doit disposer des autorisations nécessaires à la création d'instances.

Vous pouvez configurer IAM à l'aide de l'une CloudFormation ou l'autre d'étapes manuelles.

> **Important**
>
> Nous vous recommandons vivement de configurer votre environnement RDS Custom pour Oracle à l'aide AWS CloudFormation de. Cette technique est la plus simple et la moins sujette aux erreurs.

Rubriques

- Configurez IAM à l'aide de CloudFormation
- Créer manuellement votre profil d'instance et de rôle IAM

Configurez IAM à l'aide de CloudFormation

Lorsque vous utilisez le CloudFormation modèle pour IAM, il crée les ressources requises suivantes :

- Un profil d'instance nommé `AWSRDSCustomInstanceProfile-`*`region`*
- Un nom de rôle de service nommé `AWSRDSCustomInstanceRole-`*`region`*
- Une politique d'accès nommée `AWSRDSCustomIamRolePolicy` qui est attachée au rôle de service

Pour configurer IAM à l'aide de CloudFormation

1. Ouvrez la CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Lancez l'assistant de création de piles et choisissez Créer une pile.
3. Sur la page Create stack (Créer une pile), procédez de la manière suivante :
   a. Pour Préparer le modèle, choisissez Le modèle est prêt.
   b. Pour Source du modèle, choisissez Charger un fichier de modèle.
   c. Pour Choisir un fichier, naviguez jusqu'à, puis choisissez custom-oracle-iam.json.
   d. Choisissez Suivant.
4. Sur la page Spécifier les détails de la pile, procédez comme suit :
   a. Dans le champ Nom de la pile, saisissez **`custom-oracle-iam`**.
   b. Choisissez Next (Suivant).
5. Sur la page Configurer les options de pile, choisissez Suivant.
6. Sur la page Review (Vérifier) custom-oracle-iam, procédez comme suit :
   a. Cochez la case Je comprends qu' AWS CloudFormation peut créer des ressources IAM avec des noms personnalisés.
   b. Sélectionnez Envoyer.

   CloudFormation crée les rôles IAM requis par RDS Custom for Oracle. Dans le panneau de gauche, lorsque custom-oracle-iamCREATE_COMPLETE est affiché, passez à l'étape suivante.
7. Dans le panneau de gauche, choisissez custom-oracle-iam. Dans le volet de droite, procédez de la façon suivante :
   a. Choisissez Informations sur la pile. Votre pile possède un identifiant au format arn:aws:cloudformation : ::stack//. *`region`* *`account-no`* custom-oracle-iam *`identifier`*
   b. Sélectionnez Ressources. Vous devez voir ce qui suit :
      - Un profil d'instance nommé AWSRDSCustomInstanceProfile- ***`region`***
      - Un rôle de service nommé AWSRDSCustomInstanceRole- ***`region`***

      Lorsque vous créez votre instance de base de données RDS Custom, vous devez fournir l'ID de profil d'instance.

Créer manuellement votre profil d'instance et de rôle IAM

La configuration est plus simple lorsque vous utilisez CloudFormation. Toutefois, vous pouvez aussi configurer IAM manuellement. Pour une configuration manuelle, procédez comme suit :

- Étape 1 : création du rôle IAM AWSRDSCustomInstanceRoleForRdsCustomInstance.
- Étape 2 : ajouter une politique d'accès à AWSRDSCustom InstanceRoleForRdsCustomInstance.
- Étape 2 : ajouter une politique d'accès à AWSRDSCustom InstanceRoleForRdsCustomInstance.
- Étape 4 : Ajouter AWSRDSCustom InstanceRoleForRdsCustomInstance à AWSRDSCustom InstanceProfile.

Étape 1 : création du rôle IAM AWSRDSCustomInstanceRoleForRdsCustomInstance

Au cours de cette étape, vous créez le rôle à l'aide du format de dénomination `AWSRDSCustomInstanceRole-`*`region`*. En utilisant la politique de confiance, Amazon EC2 peut assumer ce rôle. L'exemple suivant suppose que vous avez défini la variable d'environnement `$REGION` sur la Région AWS dans laquelle vous souhaitez créer votre instance de base de données.

```
aws iam create-role \
  --role-name AWSRDSCustomInstanceRole-$REGION \
  --assume-role-policy-document '{
    "Version": "2012-10-17",
      "Statement": [
        {
          "Action": "sts:AssumeRole",
          "Effect": "Allow",
          "Principal": {
              "Service": "ec2.amazonaws.com"
          }
        }
      ]
    }'
```

Étape 2 : ajouter une politique d'accès à AWSRDSCustom InstanceRoleForRdsCustomInstance

Lorsque vous intégrez une stratégie en ligne à un rôle IAM, celle-ci est utilisée comme une partie de la stratégie d'accès du rôle (autorisations). Vous créez la `AWSRDSCustomIamRolePolicy` politique qui permet EC2 à Amazon d'envoyer et de recevoir des messages et d'effectuer diverses actions.

L'exemple suivant crée la stratégie d'accès nommée `AWSRDSCustomIamRolePolicy`, et l'ajoute au rôle IAM `AWSRDSCustomInstanceRole-`*`region`*. Cet exemple suppose que vous avez défini les variables d'environnement suivantes :

`$REGION`

Définissez cette variable sur celle Région AWS dans laquelle vous prévoyez de créer votre instance de base de données.

`$ACCOUNT_ID`

Définissez cette variable sur votre Compte AWS numéro.

`$KMS_KEY`

Définissez cette variable sur l'Amazon Resource Name (ARN) de la AWS KMS key que vous souhaitez utiliser pour vos instances de base de données RDS Custom. Pour spécifier d'autres clés KMS, ajoutez-les à la section `Resources` de l'ID d'instruction (Sid) 11.

```
aws iam put-role-policy \
  --role-name AWSRDSCustomInstanceRole-$REGION \
  --policy-name AWSRDSCustomIamRolePolicy \
  --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "1",
            "Effect": "Allow",
            "Action": [
                "ssm:DescribeAssociation",
                "ssm:GetDeployablePatchSnapshotForInstance",
                "ssm:GetDocument",
                "ssm:DescribeDocument",
                "ssm:GetManifest",
                "ssm:GetParameter",
                "ssm:GetParameters",
                "ssm:ListAssociations",
                "ssm:ListInstanceAssociations",
                "ssm:PutInventory",
                "ssm:PutComplianceItems",
                "ssm:PutConfigurePackageResult",
                "ssm:UpdateAssociationStatus",
                "ssm:UpdateInstanceAssociationStatus",
```

```
                "ssm:UpdateInstanceInformation",
                "ssm:GetConnectionStatus",
                "ssm:DescribeInstanceInformation",
                "ssmmessages:CreateControlChannel",
                "ssmmessages:CreateDataChannel",
                "ssmmessages:OpenControlChannel",
                "ssmmessages:OpenDataChannel"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "2",
            "Effect": "Allow",
            "Action": [
                "ec2messages:AcknowledgeMessage",
                "ec2messages:DeleteMessage",
                "ec2messages:FailMessage",
                "ec2messages:GetEndpoint",
                "ec2messages:GetMessages",
                "ec2messages:SendReply"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "3",
            "Effect": "Allow",
            "Action": [
                "logs:PutRetentionPolicy",
                "logs:PutLogEvents",
                "logs:DescribeLogStreams",
                "logs:DescribeLogGroups",
                "logs:CreateLogStream",
                "logs:CreateLogGroup"
            ],
            "Resource": [
                "arn:aws:logs:'$REGION':'$ACCOUNT_ID':log-group:rds-custom-instance*"
            ]
        },
        {
            "Sid": "4",
```

```
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetObjectVersion"
            ],
            "Resource": [
                "arn:aws:s3:::do-not-delete-rds-custom-*/*"
            ]
        },
        {
            "Sid": "5",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": [
                "*"
            ],
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": [
                        "RDSCustomForOracle/Agent"
                    ]
                }
            }
        },
        {
            "Sid": "6",
            "Effect": "Allow",
            "Action": [
                "events:PutEvents"
            ],
            "Resource": [
                "*"
            ]
        },
        {
            "Sid": "7",
            "Effect": "Allow",
            "Action": [
                "secretsmanager:GetSecretValue",
                "secretsmanager:DescribeSecret"
            ],
```

```
            "Resource": [
                "arn:aws:secretsmanager:'$REGION':'$ACCOUNT_ID':secret:do-not-delete-
rds-custom-*",
                "arn:aws:secretsmanager:'$REGION':'$ACCOUNT_ID':secret:rds-custom!
oracle-do-not-delete-*"
            ]
        },
        {
           "Sid": "8",
           "Effect": "Allow",
           "Action": [
             "s3:ListBucketVersions"
           ],
           "Resource": [
             "arn:aws:s3:::do-not-delete-rds-custom-*"
           ]
         },
         {
            "Sid": "9",
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": [
                "arn:aws:ec2:*:*:instance/*",
                "arn:aws:ec2:*:*:volume/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:ResourceTag/AWSRDSCustom": "custom-oracle"
                }
            }
          },
          {
            "Sid": "10",
            "Effect": "Allow",
            "Action": "ec2:CreateSnapshots",
            "Resource": [
                "arn:aws:ec2:*::snapshot/*"
            ]
          },
          {
            "Sid": "11",
            "Effect": "Allow",
            "Action": [
              "kms:Decrypt",
```

```
              "kms:GenerateDataKey"
            ],
            "Resource": [
              "arn:aws:kms:'$REGION':'$ACCOUNT_ID':key/'$KMS_KEY'"
            ]
          },
          {
            "Sid": "12",
            "Effect": "Allow",
            "Action": "ec2:CreateTags",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ec2:CreateAction": [
                        "CreateSnapshots"
                    ]
                }
            }
        },
        {
            "Sid": "13",
            "Effect": "Allow",
            "Action": [
                "sqs:SendMessage",
                "sqs:ReceiveMessage",
                "sqs:DeleteMessage",
                "sqs:GetQueueUrl"
            ],
            "Resource": "arn:aws:sqs:'$REGION':'$ACCOUNT_Id':do-not-delete-rds-custom-
*",
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/AWSRDSCustom": "custom-oracle"
                }
            }
        }
    ]
}'
```

Étape 3 : Création du profil d'instance personnalisé RDS AWSRDSCustom InstanceProfile

Un profil d'instance est un conteneur qui inclut un rôle IAM unique. RDS Custom utilise le profil d'instance pour transmettre le rôle à l'instance.

Si vous utilisez l'interface de ligne de commande pour créer un rôle, vous devez créer le rôle et le profil d'instance sous la forme d'actions distinctes et leur attribuer éventuellement des noms différents. Créez votre profil d'instance IAM comme suit, en le nommant avec le format `AWSRDSCustomInstanceProfile-`*`region`*. L'exemple suivant suppose que vous avez défini la variable d'environnement `$REGION` sur celle Région AWS dans laquelle vous souhaitez créer votre instance de base de données.

```
aws iam create-instance-profile \
    --instance-profile-name AWSRDSCustomInstanceProfile-$REGION
```

Étape 4 : Ajouter AWSRDSCustom InstanceRoleForRdsCustomInstance à AWSRDSCustom InstanceProfile

Ajoutez votre rôle IAM au profil d'instance que vous avez créé précédemment. L'exemple suivant suppose que vous avez défini la variable d'environnement `$REGION` sur celle Région AWS dans laquelle vous souhaitez créer votre instance de base de données.

```
aws iam add-role-to-instance-profile \
    --instance-profile-name AWSRDSCustomInstanceProfile-$REGION \
    --role-name AWSRDSCustomInstanceRole-$REGION
```

Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM

Assurez-vous que le principal IAM (utilisateur ou rôle) qui crée l'instance de base de données personnalisée CEV ou RDS applique l'une des politiques suivantes :

- La stratégie `AdministratorAccess`
- La `AmazonRDSFullAccess` politique avec les autorisations requises pour Amazon S3 et CEV AWS KMS, ainsi que pour la création d'instances de base de données

Rubriques

- Autorisations IAM requises pour Amazon S3 et AWS KMS
- Autorisations IAM requises pour créer une CEV
- Autorisations IAM requises pour créer une instance de base de données depuis une CEV

Autorisations IAM requises pour Amazon S3 et AWS KMS

Pour créer CEVs des instances de base de données RDS Custom pour Oracle, votre principal IAM doit accéder à Amazon S3 et. AWS KMS L'exemple de politique JSON suivant accorde les autorisations requises.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CreateS3Bucket",
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:PutBucketPolicy",
                "s3:PutBucketObjectLockConfiguration",
                "s3:PutBucketVersioning"
            ],
            "Resource": "arn:aws:s3:::do-not-delete-rds-custom-*"
        },
        {
            "Sid": "CreateKmsGrant",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
}
```

Pour plus d'informations sur l'autorisation `kms:CreateGrant`, consultez AWS KMS key gestion.

Autorisations IAM requises pour créer une CEV

Pour créer un CEV, votre principal IAM a besoin des autorisations supplémentaires suivantes :

```
s3:GetObjectAcl
```

```
s3:GetObject
s3:GetObjectTagging
s3:ListBucket
mediaimport:CreateDatabaseBinarySnapshot
```

L'exemple de politique JSON suivant accorde les autorisations supplémentaires nécessaires pour accéder au bucket *my-custom-installation-files* et à son contenu.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AccessToS3MediaBucket",
            "Effect": "Allow",
            "Action": [
                "s3:GetObjectAcl",
                "s3:GetObject",
                "s3:GetObjectTagging",
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::my-custom-installation-files",
                "arn:aws:s3:::my-custom-installation-files/*"
            ]
        },
        {
            "Sid": "PermissionForByom",
            "Effect": "Allow",
            "Action": [
                "mediaimport:CreateDatabaseBinarySnapshot"
            ],
            "Resource": "*"
        }
    ]
}
```

Vous pouvez également accorder des autorisations similaires pour Simple Storage Service (Amazon S3) aux comptes appelants à l'aide d'une politique de compartiment S3.

Autorisations IAM requises pour créer une instance de base de données depuis une CEV

Pour créer une instance de base de données RDS Custom pour Oracle à partir d'un CEV existant, le principal IAM a besoin des autorisations supplémentaires suivantes.

```
iam:SimulatePrincipalPolicy
cloudtrail:CreateTrail
cloudtrail:StartLogging
```

L'exemple de politique JSON suivant accorde les autorisations nécessaires pour valider un rôle IAM et journaliser des informations dans un AWS CloudTrail.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ValidateIamRole",
            "Effect": "Allow",
            "Action": "iam:SimulatePrincipalPolicy",
            "Resource": "*"
        },
        {
            "Sid": "CreateCloudTrail",
            "Effect": "Allow",
            "Action": [
                "cloudtrail:CreateTrail",
                "cloudtrail:StartLogging"
            ],
            "Resource": "arn:aws:cloudtrail:*:*:trail/do-not-delete-rds-custom-*"
        }
    ]
}
```

Étape 6 : Configuration de votre VPC pour RDS Custom pour Oracle

Votre instance de base de données personnalisée RDS se trouve dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC, tout comme une instance Amazon ou une EC2 instance Amazon RDS. Vous fournissez et configurez votre propre VPC. Contrairement à RDS Custom for SQL Server,

RDS Custom for Oracle ne crée pas de liste de contrôle d'accès ni de groupes de sécurité. Vous devez associer votre propre groupe de sécurité, vos sous-réseaux et vos tables de routage.

Vous pouvez configurer votre cloud privé virtuel (VPC) à l'aide de l'un CloudFormation ou l'autre processus manuel.

> **Important**
>
> Nous vous recommandons vivement de configurer votre environnement RDS Custom pour Oracle à l'aide AWS CloudFormation de. Cette technique est la plus simple et la moins sujette aux erreurs.

Rubriques

- Configurez votre VPC à l'aide de CloudFormation (recommandé)
- Configurez votre VPC manuellement pour RDS Custom for Oracle

Configurez votre VPC à l'aide de CloudFormation (recommandé)

Si vous avez déjà configuré votre VPC pour un autre moteur RDS Custom et que vous voulez réutiliser le VPC existant, ignorez cette étape. Cette section suppose ce qui suit :

- Vous l'avez déjà utilisé CloudFormation pour créer votre profil et votre rôle d'instance IAM.
- Vous connaissez l'ID de votre table de routage.

  Pour qu'une instance de base de données soit privée, elle doit se trouver dans un sous-réseau privé. Pour qu'un sous-réseau soit privé, il ne doit pas être associé à une table de routage comportant une passerelle Internet par défaut. Pour plus d'informations, consultez Configuration des tables de routage dans le Guide de l'utilisateur d'Amazon VPC.

Lorsque vous utilisez le CloudFormation modèle pour votre VPC, il crée les ressources suivantes :

- Un VPC privé
- Un groupe de sous-réseaux nommé `rds-custom-private`
- Les points de terminaison VPC suivants, avec lesquels votre instance de base de données communique, dépendent : Services AWS
  - `com.amazonaws.region.ec2messages`

- com.amazonaws.*region*.events
- com.amazonaws.*region*.logs
- com.amazonaws.*region*.monitoring
- com.amazonaws.*region*.s3
- com.amazonaws.*region*.secretsmanager
- com.amazonaws.*region*.ssm
- com.amazonaws.*region*.ssmmessages

Si vous créez un déploiement multi-AZ :

- com.amazonaws.*region*.sqs

> **Note**
>
> Pour une configuration réseau complexe avec des comptes existants, nous vous recommandons de configurer manuellement l'accès aux services dépendants s'il n'existe pas encore d'accès. Pour de plus amples informations, veuillez consulter Assurez-vous que votre VPC peut accéder aux personnes dépendantes Services AWS.

Pour configurer votre VPC à l'aide de CloudFormation

1. Ouvrez la CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Lancez l'assistant Créer une pile et choisissez Créer une pile, puis Avec de nouvelles ressources (standard).
3. Sur la page Créer une pile, procédez de la manière suivante :
   a. Pour Préparer le modèle, choisissez Le modèle est prêt.
   b. Pour Source du modèle, choisissez Charger un fichier de modèle.
   c. Pour Choisir un fichier, accédez à `custom-vpc.json` et sélectionnez ce fichier.
   d. Choisissez Suivant.
4. Sur la page Spécifier les détails de la pile, procédez comme suit :
   a. Dans le champ Nom de la pile, saisissez **`custom-vpc`**.
   b. Pour Paramètres, sélectionnez les sous-réseaux privés à utiliser pour les instances de base de données RDS Custom.

c. Choisissez l'ID du VPC privé à utiliser pour les instances de base de données RDS Custom.

d. Indiquez la table de routage associée aux sous-réseaux privés.

e. Choisissez Next (Suivant).

5. Sur la page Configurer les options de pile, choisissez Suivant.

6. Sur la page Vérifier custom-vpc, choisissez Envoyer.

   CloudFormation configure votre VPC privé. Dans le volet de gauche, lorsque custom-vpc affiche CREATE_COMPLETE, passez à l'étape suivante.

7. (Facultatif) Vérifiez les détails de votre VPC. Dans le volet Piles, choisissez custom-vpc. Dans le volet de droite, procédez de la façon suivante :

   a. Choisissez Informations sur la pile. Votre stack possède un identifiant au format arn:aws:cloudformation : ::stack/custom-vpc/. *region account-no identifier*

   b. Sélectionnez Ressources. Vous devriez voir un groupe de sous-réseaux nommé rds-custom-privateet plusieurs points de terminaison VPC utilisant le format de dénomination vpce-. ***string*** Chaque point de terminaison correspond à un Service AWS point de terminaison avec lequel RDS Custom doit communiquer. Pour de plus amples informations, veuillez consulter Assurez-vous que votre VPC peut accéder aux personnes dépendantes Services AWS.

   c. Choisissez Parameters (Paramètres). Vous devriez voir les sous-réseaux privés, le VPC privé et la table de routage que vous avez spécifiés lors de la création de la pile. Lorsque vous créez une instance de base de données, vous devez fournir l'ID du VPC et le groupe de sous-réseaux.

Configurez votre VPC manuellement pour RDS Custom for Oracle

Au lieu d'automatiser la création de VPC AWS CloudFormation avec, vous pouvez configurer votre VPC manuellement. Cette option peut être la meilleure lorsque vous disposez d'une configuration réseau complexe qui utilise des ressources existantes.

Rubriques

- Assurez-vous que votre VPC peut accéder aux personnes dépendantes Services AWS
- Configuration du service des métadonnées d'instance

Assurez-vous que votre VPC peut accéder aux personnes dépendantes Services AWS

RDS Custom envoie la communication de votre instance de base de données vers d'autres Services AWS. Assurez-vous que les services suivants sont accessibles depuis le sous-réseau dans lequel vous créez vos instances de base de données personnalisées RDS :

- Amazon CloudWatch (`com.amazonaws.`*`region`*`.monitoring`)
- Amazon CloudWatch Logs (`com.amazonaws.`*`region`*`.logs`)
- CloudWatch Événements Amazon (`com.amazonaws.`*`region`*`.events`)
- Amazon EC2 (`com.amazonaws.`*`region`*`.ec2etcom.amazonaws.`*`region`*`.ec2messages`)
- Amazon EventBridge (`com.amazonaws.`*`region`*`.events`)
- Amazon S3 (`com.amazonaws.`*`region`*`.s3`)
- AWS Secrets Manager (`com.amazonaws.`*`region`*`.secretsmanager`)
- AWS Systems Manager (`com.amazonaws.`*`region`*`.ssmetcom.amazonaws.`*`region`*`.ssmmessages`)

Si vous créez des déploiements multi-AZ

- Service de file d'attente Amazon Simple (`com.amazonaws.`*`region`*`.sqs`)

Si RDS Custom ne parvient pas à communiquer avec les services nécessaires, il publie les événements suivants :

```
Database instance in incompatible-network. SSM Agent connection not available. Amazon
 RDS can't connect to the dependent AWS services.
```

```
Database instance in incompatible-network. Amazon RDS can't connect to dependent AWS
 services. Make sure port 443 (HTTPS) allows outbound connections, and try again.
 "Failed to connect to the following services: s3 events"
```

Pour éviter les `incompatible-network` erreurs, assurez-vous que les composants VPC impliqués dans la communication entre votre instance de base de données personnalisée RDS Services AWS répondent aux exigences suivantes :

- L'instance de base de données peut établir des connexions sortantes sur le port 443 vers d'autres Services AWS.

- Le VPC autorise les réponses entrantes aux demandes provenant de votre instance de base de données RDS Custom.
- RDS Custom peut correctement résoudre les noms de domaine des points de terminaison pour chaque Service AWS.

Si vous avez déjà configuré un VPC pour un moteur de base de données RDS Custom différent, vous pouvez réutiliser ce VPC et ignorer ce processus.

Configuration du service des métadonnées d'instance

Assurez-vous que votre instance peut effectuer les opérations suivantes :

- Accédez au service de métadonnées d'instance à l'aide du service de métadonnées d'instance version 2 (IMDSv2).
- Autoriser les communications sortantes via le port 80 (HTTP) vers l'adresse IP de la liaison IMDS.
- Demandez les métadonnées de l'instance `http://169.254.169.254` à partir du IMDSv2 lien.

Pour plus d'informations, consultez la section Utilisation IMDSv2 dans le guide de EC2 l'utilisateur Amazon.

RDS Custom for Oracle Automation s'utilise IMDSv2 par défaut, en le configurant `HttpTokens=enabled` sur l' EC2 instance Amazon sous-jacente. Cependant, vous pouvez l'utiliser IMDSv1 si vous le souhaitez. Pour plus d'informations, consultez Configurer les options de métadonnées de l'instance dans le guide de EC2 l'utilisateur Amazon.

Utilisation de versions de moteurs personnalisées pour Amazon RDS Custom for Oracle

Une version cd moteur personnalisée (CEV) pour Amazon RDS Custom for Oracle est un instantané de volume binaire d'un moteur de base de données et d'une Amazon Machine Image (AMI) spécifique. Par défaut, RDS Custom for Oracle utilise la dernière AMI disponible gérée par RDS Custom, mais vous pouvez spécifier une AMI utilisée dans une CEV précédente. Vous stockez les fichiers d'installation de vos bases de données dans Amazon S3. RDS Custom utilise les fichiers d'installation et l'image AMI pour créer votre version CEV pour vous.

Rubriques

- Préparation de la création d'une CEV
- Création d’un CEV
- Modification de l'état de la CEV
- Affichage des CEV informations relatives à Amazon RDS Custom pour Oracle
- Suppression d'une CEV

Préparation de la création d'une CEV

Pour créer un CEV, accédez aux fichiers d'installation et aux correctifs qui sont stockés dans votre compartiment Amazon S3 pour l'une des versions suivantes :

- Oracle Database 19c
- Oracle Database 18c
- Oracle Database 12c version 2 (12.2)
- Oracle Database 12c version 1 (12.1)

Par exemple, vous pouvez utiliser le mois d'avril 2021 RU/RUR pour Oracle Database 19c, ou toute combinaison valide de fichiers d'installation et de correctifs. Pour plus d'informations sur les régions et les versions prises en charge par RDS Custom for Oracle, consultez RDS Custom avec RDS for Oracle.

Rubriques

- Étape 1 (Facultative) : Télécharger les modèles de manifeste

- Étape 2 : Télécharger des fichiers d'installation de votre base de données et des correctifs à partir d'Oracle Software Delivery Cloud
- Étape 3 : Charger vos fichiers d'installation dans Amazon S3
- Étape 4 (facultative) : partagez votre support d'installation dans S3 entre Comptes AWS
- Étape 5 : Préparer le manifeste CEV
- Étape 6 (Facultative) : Valider le manifeste CEV
- Étape 7 : Ajouter les autorisations IAM nécessaires

Étape 1 (Facultative) : Télécharger les modèles de manifeste

Un manifeste CEV est un document JSON qui inclut la liste des fichiers .zip d'installation de base de données pour votre CEV. Pour créer une CEV, procédez comme suit :

1. Identifiez les fichiers d'installation de la base de données Oracle que vous souhaitez inclure dans votre CEV.
2. Téléchargez les fichiers d'installation.
3. Créez un manifeste JSON répertoriant les fichiers d'installation.

RDS Custom for Oracle fournit des modèles de manifeste JSON avec les fichiers .zip recommandés pour chaque version d'Oracle Database prise en charge. Par exemple, le modèle suivant concerne le 19.17.0.0.0 RU.

```
{
    "mediaImportTemplateVersion": "2020-08-14",
    "databaseInstallationFileNames": [
        "V982063-01.zip"
    ],
    "opatchFileNames": [
        "p6880880_190000_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames": [
        "p34419443_190000_Linux-x86-64.zip",
        "p34411846_190000_Linux-x86-64.zip"
    ],
    "otherPatchFileNames": [
        "p28852325_190000_Linux-x86-64.zip",
        "p29997937_190000_Linux-x86-64.zip",
```

```
        "p31335037_190000_Linux-x86-64.zip",
        "p32327201_190000_Linux-x86-64.zip",
        "p33613829_190000_Linux-x86-64.zip",
        "p34006614_190000_Linux-x86-64.zip",
        "p34533061_190000_Linux-x86-64.zip",
        "p34533150_190000_Generic.zip",
        "p28730253_190000_Linux-x86-64.zip",
        "p29213893_1917000DBRU_Generic.zip",
        "p33125873_1917000DBRU_Linux-x86-64.zip",
        "p34446152_1917000DBRU_Linux-x86-64.zip"
    ]
}
```

Chaque modèle est associé à un fichier readme qui inclut des instructions pour télécharger les correctifs, URLs les fichiers .zip et les checksums des fichiers. Vous pouvez utiliser ces modèles tels quels ou les modifier avec vos propres correctifs. Pour consulter les modèles, téléchargez le fichier custom-oracle-manifest.zip sur votre disque local, puis ouvrez-le à l'aide d'une application d'archivage de fichiers. Pour de plus amples informations, veuillez consulter Étape 5 : Préparer le manifeste CEV.

Étape 2 : Télécharger des fichiers d'installation de votre base de données et des correctifs à partir d'Oracle Software Delivery Cloud

Lorsque vous avez identifié les fichiers d'installation que vous souhaitez pour votre CEV, téléchargez-les sur votre système local. Les fichiers d'installation et les correctifs d'installation d'Oracle Database sont hébergés sur Oracle Software Delivery Cloud. Chaque CEV nécessite une version de base, telle qu'Oracle Database 19c ou Oracle Database 12c version 2 (12.2) et une liste de correctifs facultative.

Pour télécharger les fichiers d'installation de la base de données pour Oracle Database

1. Accédez à https://edelivery.oracle.com/ et connectez-vous.
2. Dans le champ de recherche, saisissez **`Oracle Database Enterprise Edition`** ou **`Oracle Database Standard Edition 2`** puis choisissez Rechercher.
3. Choisissez l'une des versions de base suivantes :

Version de base de données	Enterprise Edition	Standard Edition 2
Oracle Database 19c	DLP : Oracle Database 19c Enterprise Edition 19.3.0.0.0	DLP : Oracle Database 19c Standard Edition 2 19.3.0.0.

Version de base de données	Enterprise Edition	Standard Edition 2
	(Oracle Database Enterprise Edition)	0 (Oracle Database Standard Edition 2)
Oracle Database 18c	DLP : Oracle Database 18c Enterprise Edition 18.0.0.0.0 (Oracle Database Enterprise Edition)	DLP : Oracle Database Standard Edition 2 18.0.0.0.0 (Oracle Database Standard Edition 2)
Oracle Database 12c version 2 (12.2.0.1)	DLP : Oracle Database 12c Enterprise Edition 12.2.0.1.0 (Oracle Database Enterprise Edition)	DLP : Oracle Database Standard Edition 2 12.2.0.1.0 (Oracle Database Standard Edition 2)
Oracle Database 12c version 1 (12.1.0.2)	DLP : Oracle Database 12c Enterprise Edition 12.1.0.2.0 (Oracle Database Enterprise Edition)	DLP : Oracle Database Standard Edition 2 12.1.0.2.0 (Oracle Database Standard Edition 2)

4. Choisissez Continuer.
5. Désélectionnez la case à cocher Download Queue (Queue de téléchargement).
6. Choisissez l'option qui correspond à votre version de base :
   - Oracle Database 19.3.0.0.0 : version à long terme..
   - Oracle Database 18.0.0.0.0
   - Oracle Database 12.2.0.1.0.
   - Oracle Database 12.1.0.2.0.
7. Choisissez Linux x86-64 dans Platform/Languages (Plateforme/Langues).
8. Choisissez Continuer, puis signez le contrat de licence Oracle.
9. Choisissez le fichier .zip qui correspond à votre version de base de données :

Publication et édition de la base de données	Fichiers ZIP	Le hachage SHA-256
19c EE et SE2	V982063-01.zip	BA8329C757133DA313ED3B6D7F86C5AC42CD9970A28BF2E6233F3235233AA8D8
18c EE et SE2	V978967-01.zip	C96A4FD768787AF98272008833FE10B172691CF84E42816B138C12D4DE63AB96
12.2.0.1 EE et SE2	V839960-01.zip	96ED97D21F15C1AC0CCE3749DA6C3DAC7059BB60672D76B008103FC754D22DDE
12.1.0.2 VOIR	V46095-01_1of2.zip V46095-01_2of2.zip	31FDC2AF41687B4E547A3A18F796424D8C1AF36406D2160F65B0AF6A9CD47355 pour V46095-01_1of2.zip 03DA14F5E875304B28F0F3BB02AF0EC33227885B99C9865DF70749D1E220ACCD pour V46095-01_2of2.zip
12,10.2 SE2	V77388-01_1of2.zip V77388-01_2of2.zip	73873369753230F5A0921F95ACEADB591388CB06ED72A7F3AEA7BCBCEA2403BC pour V77388-01_1of2.zip 2492E1BE1E3E3531DA83D0843C09C08E435AC8CEFD9A00C0DF56BE4F15CEEBF3 pour V77388-01_2of2.zip

10. Téléchargez les correctifs Oracle souhaités depuis `updates.oracle.com` ou `support.oracle.com` vers votre système local. Vous pouvez trouver URLs les correctifs aux emplacements suivants :

    - Les fichiers readme dans le fichier .zip que vous avez téléchargé dans Étape 1 (Facultative) : Télécharger les modèles de manifeste

- Les correctifs sont répertoriés dans chaque dernière mise à jour (RU) dans Release notes for Amazon Relational Database Service (Amazon RDS) for Oracle (Notes de mise à jour d'Amazon Relational Database Service (Amazon RDS) for Oracle).

Étape 3 : Charger vos fichiers d'installation dans Amazon S3

Chargement de vos fichiers d'installation et de fichiers correctifs Oracle dans Amazon S3 en utilisant la AWS CLI. Le compartiment S3 qui contient vos fichiers d'installation doit se trouver dans la même AWS région que votre CEV.

Les exemples de cette section utilisent les espaces réservés suivants :

- *`install-or-patch-file.zip`* – Fichier multimédia d'installation Oracle. Par exemple, p32126828_190000_Linux-x86-64.zip est un correctif.
- *`amzn-s3-demo-destination-bucket`* – Votre compartiment Amazon S3 désigné pour vos fichiers d'installation chargés.
- *`123456789012/cev1`* : préfixe facultatif dans votre compartiment Simple Storage Service (Amazon S3).
- *`amzn-s3-demo-source-bucket`* : compartiment Simple Storage Service (Amazon S3) dans lequel vous pouvez éventuellement déposer des fichiers.

Rubriques

- Étape 3a : Vérifiez que votre compartiment S3 est dans le bon Région AWS
- Étape 3b : Assurez-vous que votre politique de compartiment S3 dispose des autorisations appropriées
- Étape 3c : Téléchargez vos fichiers à l'aide des commandes cp ou sync
- Étape 3d : Répertoriez les fichiers de votre compartiment S3

Étape 3a : Vérifiez que votre compartiment S3 est dans le bon Région AWS

Vérifiez que votre compartiment S3 se trouve dans la AWS région dans laquelle vous prévoyez d'exécuter la `create-custom-db-engine-version` commande.

```
aws s3api get-bucket-location --bucket amzn-s3-demo-destination-bucket
```

Étape 3b : Assurez-vous que votre politique de compartiment S3 dispose des autorisations appropriées

Vous pouvez créer un CEV à partir de zéro ou à partir d'un CEV source. Si vous envisagez de créer un nouveau CEV à partir de la source CEVs, assurez-vous que votre politique de compartiment S3 dispose des autorisations appropriées :

1. Identifiez le compartiment S3 réservé par RDS Custom. Le nom du compartiment est au format `do-not-delete-rds-custom-`*`account`*`-`*`region`*`-`*`string`*. Par exemple, le nom du compartiment peut être `do-not-delete-rds-custom-`*`123456789012`*`-`*`us-east-1`*`-`*`abc123EXAMPLE`*.
2. Assurez-vous que l'autorisation suivante est ajoutée à votre politique de compartiment S3. Remplacez `do-not-delete-rds-custom-`*`123456789012`*`-`*`us-east-1`*`-`*`abc123EXAMPLE`* par le nom de votre compartiment.

```
{
    "Sid": "AWSRDSCustomForOracleCustomEngineVersionGetObject",
    "Effect": "Allow",
    "Principal": {
        "Service": "custom.rds.amazonaws.com"
    },
    "Action": [
        "s3:GetObject",
        "s3:GetObjectTagging"
    ],
    "Resource": "arn:aws:s3:::do-not-delete-rds-custom-123456789012-us-
east-1-abc123EXAMPLE/CustomEngineVersions/*"
}, ...
```

Étape 3c : Téléchargez vos fichiers à l'aide des commandes cp ou sync

Choisissez l'une des options suivantes :

- Utilisez `aws s3 cp` pour charger un fichier .zip unique.

  Chargez chaque fichier .zip d'installation séparément. Ne combinez pas les fichiers .zip en un seul fichier .zip.
- Utilisez `aws s3 sync` pour charger un répertoire.

Example

L'exemple suivant permet de charger *install-or-patch-file.zip* vers le répertoire *123456789012/cev1* dans le compartiment Amazon S3 RDS Custom. Exécutez une commande `aws s3` séparée pour chaque fichier .zip que vous souhaitez charger.

Pour LinuxmacOS, ou Unix :

```
aws s3 cp install-or-patch-file.zip \
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Dans Windows :

```
aws s3 cp install-or-patch-file.zip ^
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Example

L'exemple suivant télécharge les fichiers de votre *cev1* dossier local vers le *123456789012/cev1* dossier de votre compartiment Amazon S3.

Pour LinuxmacOS, ou Unix :

```
aws s3 sync cev1 \
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Dans Windows :

```
aws s3 sync cev1 ^
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Example

L'exemple suivant permet de charger tous les fichiers dans *amzn-s3-demo-source-bucket* vers le dossier *123456789012/cev1* de votre compartiment Simple Storage Service (Amazon S3).

Pour LinuxmacOS, ou Unix :

```
aws s3 sync s3://amzn-s3-demo-source-bucket/ \
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Dans Windows :

```
aws s3 sync s3://amzn-s3-demo-source-bucket/ ^
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Étape 3d : Répertoriez les fichiers de votre compartiment S3

L'exemple suivant utilise la commande `s3 ls` pour répertorier les fichiers de votre compartiment Custom Amazon S3.

```
aws s3 ls \
    s3://amzn-s3-demo-destination-bucket/123456789012/cev1/
```

Étape 4 (facultative) : partagez votre support d'installation dans S3 entre Comptes AWS

Aux fins de cette section, le compartiment Amazon S3 qui contient vos fichiers d'installation Oracle chargés est votre compartiment média. Votre organisation peut en utiliser plusieurs Comptes AWS dans un Région AWS. Si tel est le cas, vous souhaiterez peut-être en utiliser un Compte AWS pour remplir votre compartiment multimédia et un autre Compte AWS pour le créer CEVs. Si vous n'avez pas l'intention de partager votre compartiment média, passez à la section suivante.

Cette section suppose ce qui suit :

- Vous pouvez accéder au compte qui a créé votre compartiment multimédia et à un autre compte dans lequel vous souhaitez le créer CEVs.
- Vous avez l'intention de créer CEVs en un seul Région AWS. Si vous avez l'intention d'utiliser plusieurs régions, créez un compartiment média dans chaque région.
- Vous utilisez le CLI. Si vous utilisez la console Amazon S3, adaptez les étapes suivantes.

Pour configurer votre compartiment multimédia afin de le partager entre Comptes AWS

1. Connectez-vous à Compte AWS celui qui contient le compartiment S3 dans lequel vous avez chargé votre support d'installation.
2. Commencez par un modèle de politique JSON vierge ou une politique existante que vous pouvez adapter.

   La commande suivante permet de récupérer une politique existante et de l'enregistrer sous*my-policy.json*. Dans cet exemple, le compartiment S3 contenant vos fichiers d'installation est nommé*amzn-s3-demo-bucket*.

```
aws s3api get-bucket-policy \
    --bucket amzn-s3-demo-bucket \
    --query Policy \
    --output text > my-policy.json
```

3. Modifiez les autorisations du compartiment média comme suit :

   - Dans l'élément `Resource` de votre modèle, indiquez le compartiment S3 dans lequel vous avez chargé vos fichiers d'installation de la base de données Oracle.
   - Dans l'`Principal`élément, spécifiez le ARNs pour tout Comptes AWS ce que vous souhaitez utiliser pour créer CEVs. Vous pouvez ajouter la racine, un utilisateur ou un rôle à la liste des autorisations du compartiment S3. Pour plus d'informations, consultez Identifiants IAM dans le AWS Identity and Access Management Guide de l'utilisateur .

   JSON

```
{
    "Version": "2008-10-17",
    "Statement": [
        {
            "Sid": "GrantAccountsAccess",
            "Effect": "Allow",
            "Principal": {
                "AWS": [
                    "arn:aws:iam::111122223333:root",
                    "arn:aws:iam::444455556666:user/user-name-with-path",
                    "arn:aws:iam::123456789012:role/role-name-with-path"
                ]
            },
            "Action": [
                "s3:GetObject",
                "s3:GetObjectAcl",
                "s3:GetObjectTagging",
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
            "Resource": [
                "arn:aws:s3:::amzn-s3-demo-bucket",
                "arn:aws:s3:::amzn-s3-demo-bucket/*"
            ]
```

```
        }
    ]
}
```

4. Attachez la politique à votre compartiment de médias.

   Dans l'exemple suivant, *amzn-s3-demo-bucket* il s'agit du nom du compartiment S3 qui contient vos fichiers d'installation et *my-policy.json* du nom de votre fichier JSON.

   ```
   aws s3api put-bucket-policy \
       --bucket amzn-s3-demo-bucket \
       --policy file://my-policy.json
   ```

5. Connectez-vous à un Compte AWS dans lequel vous avez l'intention de créer CEVs.
6. Vérifiez que ce compte peut accéder au compartiment multimédia dans le compartiment Compte AWS qui l'a créé.

   ```
   aws s3 ls --query "Buckets[].Name"
   ```

   Pour obtenir plus d'informations, consultez la section aws s3 ls dans la référence de commande AWS CLI .
7. Créez un CEV en suivant les étapes indiquées dans Création d'un CEV.

Étape 5 : Préparer le manifeste CEV

Un manifeste CEV est un document JSON qui inclut les éléments suivants :

- (Obligatoire) La liste des fichiers .zip d'installation que vous avez chargés sur Amazon S3. RDS Custom applique les correctifs dans l'ordre dans lequel ils sont listés dans le manifeste.
- (Facultatif) Paramètres d'installation qui définissent des valeurs autres que celles par défaut pour la base Oracle, le répertoire d'origine Oracle Home, ainsi que l'ID et le nom de l' UNIX/Linux utilisateur et du groupe. Sachez que vous ne pouvez pas modifier les paramètres d'installation d'une version CEV existante ou d'une instance de base de données existante. Vous ne pouvez pas non plus effectuer une mise à niveau d'une version CEV vers une autre lorsque les paramètres d'installation ont des paramètres différents.

Pour obtenir des exemples de manifestes CEV, consultez les modèles JSON dans que vous avez téléchargés dans Étape 1 (Facultative) : Télécharger les modèles de manifeste. Vous pouvez également examiner les exemples dans Exemples de manifeste CEV.

Rubriques

- Champs JSON dans le manifeste CEV
- Création du manifeste CEV
- Exemples de manifeste CEV

Champs JSON dans le manifeste CEV

Le tableau suivant décrit les champs JSON dans le fichier manifeste.

Champ JSON	Description
`MediaImportTemplateVersion`	Version du manifeste CEV. La date doit être au format `YYYY-MM-DD` .
`databaseInstallationFileNames`	Liste ordonnée des fichiers d'installation de la base de données.
`opatchFileNames`	Liste ordonnée des OPatch programmes d'installation utilisés pour le moteur de base de données Oracle. Une seule valeur est valide. Les valeurs de `opatchFileNames` doivent commencer par `p6880880_` .
`psuRuPatchFileNames`	Les correctifs PSU et RU pour cette base de données. **Important** Si vous incluez `psuRuPatchFileNames` , la valeur `opatchFileNames` est obligatoire. Les valeurs de `opatchFileNames` doivent commencer par `p6880880_` .

Champ JSON	Description
`OtherPatchFileNames`	Les correctifs qui ne figurent pas dans la liste des correctifs PSU et RU. RDS Custom applique ces correctifs après avoir appliqué les correctifs PSU et RU. **Important** Si vous incluez `OtherPatchFileNames` , la valeur `opatchFileNames` est obligatoire. Les valeurs de `opatchFileNames` doivent commencer par `p6880880_` .

Champ JSON	Description
`installationParameters`	Paramètres autres que ceux par défaut pour la base Oracle, le répertoire d'origine Oracle Home, ainsi que l'ID et le nom de l' UNIX/Linux utilisateur et du groupe. Vous pouvez définir les paramètres suivants : oracleBase Le répertoire dans lequel vos fichiers binaires Oracle sont installés. Il s'agit du point de montage du volume binaire qui stocke vos fichiers. Le répertoire de base Oracle peut inclure plusieurs répertoires de base de base de données Oracle. Par exemple, si `/home/oracle/oracle.19.0.0.0.ru-2020-04.rur-2020-04.r1.EE.1` est l'un de vos répertoires de base de base de données Oracle, `/home/oracle` est le répertoire de base Oracle. Un répertoire de base Oracle spécifié par l'utilisateur n'est pas un lien symbolique. Si vous ne spécifiez pas la base Oracle, le répertoire par défaut est `/rdsdbbin` . oracleHome Le répertoire dans lequel vos fichiers binaires de base de données Oracle sont installés. Par exemple, si vous spécifiez `/home/oracle/` comme base Oracle, vous pouvez spécifier `/home/oracle/oracle.19.0.0.0.ru-2020-04.rur-2020-04.r1.EE.1/` comme répertoire de base de base de données Oracle. Un répertoire de base de base de données Oracle spécifié par l'utilisateur n'est pas un lien symbolique. La valeur de répertoire de base de base de données Oracle est référencée par la variable d'environnement `$ORACLE_HOME` . Si vous ne spécifiez pas le répertoire de base de base de données Oracle, le format de nommage par défaut

Champ JSON	Description
	est `/rdsdbbin/oracle.` *`major-engine-version`* `.custom.r1.` *`engine-edition`* `.1`. unixUname Nom de l'utilisateur UNIX propriétaire du logiciel Oracle. RDS Custom endosse l'identité de cet utilisateur lors de l'exécution de commandes de base de données locales. Si vous spécifiez à la fois `unixUid` et `unixUname` , RDS Custom crée l'utilisateur s'il n'existe pas, puis lui attribue l'UID s'il n'est pas identique à l'UID initial. Le nom d'utilisateur par défaut est `rdsdb`. unixUid ID (UID) de l'utilisateur UNIX propriétaire du logiciel Oracle. Si vous spécifiez à la fois `unixUid` et `unixUname` , RDS Custom crée l'utilisateur s'il n'existe pas, puis lui attribue l'UID s'il n'est pas identique à l'UID initial. L'UID par défaut est `61001`. Il s'agit de l'UID de l'utilisateur `rdsdb`. unixGroupName Nom du groupe UNIX. L'utilisateur UNIX propriétaire du logiciel Oracle appartient à ce groupe. Le nom de groupe par défaut est `rdsdb`. unixGroupId ID du groupe UNIX auquel l'utilisateur UNIX appartient. L'ID de groupe par défaut est `1000`. Il s'agit de l'ID du groupe `rdsdb`.

Chaque version d'Oracle Database possède une liste différente de fichiers d'installation pris en charge. Lorsque vous créez votre manifeste CEV, veillez à ne spécifier que les fichiers pris en charge

par RDS Custom for Oracle. Sinon, la création du CEV échoue et renvoie une erreur. Les correctifs répertoriés dans Release notes for Amazon Relational Database Service (Amazon RDS) for Oracle (Notes de mise à jour d'Amazon Relational Database Service (Amazon RDS) for Oracle) sont pris en charge.

Création du manifeste CEV

Pour créer un manifeste CEV

1. Dressez la liste de tous les fichiers d'installation que vous prévoyez d'appliquer, dans l'ordre dans lequel vous voulez les appliquer.
2. Corrélez les fichiers d'installation avec les champs JSON décrits dans Champs JSON dans le manifeste CEV.
3. Effectuez l'une des actions suivantes :
   - Créez le manifeste CEV sous la forme d'un fichier texte JSON.
   - Modifiez le modèle de manifeste CEV lorsque vous créez le CEV dans la console. Pour de plus amples informations, veuillez consulter Création d'un CEV.

Exemples de manifeste CEV

Les exemples suivants montrent des fichiers manifestes CEV pour différentes versions de la base de données Oracle. Si vous incluez un champ JSON dans votre manifeste, assurez-vous qu'il n'est pas vide. Par exemple, le manifeste CEV suivant n'est pas valide car `otherPatchFileNames` est vide.

```
{
    "mediaImportTemplateVersion": "2020-08-14",
    "databaseInstallationFileNames": [
        "V982063-01.zip"
    ],
    "opatchFileNames": [
        "p6880880_190000_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames": [
        "p32126828_190000_Linux-x86-64.zip"
    ],
    "otherPatchFileNames": [
    ]
}
```

Rubriques

- Sample CEV manifest for Oracle Database 12c Release 1 (12.1)
- Sample CEV manifest for Oracle Database 12c Release 2 (12.2)
- Sample CEV manifest for Oracle Database 18c
- Sample CEV manifest for Oracle Database 19c

Example Exemple de manifeste CEV pour Oracle Database 12c Version 1 (12.1)

Dans l'exemple suivant pour le PSU de juillet 2021 pour Oracle Database 12c Release 1 (12.1), RDS Custom applique les correctifs dans l'ordre indiqué. Ainsi, RDS Custom applique p32768233, puis p32876425, puis p18759211, etc. L'exemple définit de nouvelles valeurs pour l'utilisateur et le groupe UNIX/Linux, ainsi que pour le répertoire de base de base de données Oracle et la base Oracle.

```
{
    "mediaImportTemplateVersion":"2020-08-14",
    "databaseInstallationFileNames":[
        "V46095-01_1of2.zip",
        "V46095-01_2of2.zip"
    ],
    "opatchFileNames":[
        "p6880880_121010_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames":[
        "p32768233_121020_Linux-x86-64.zip"
    ],
    "otherPatchFileNames":[
        "p32876425_121020_Linux-x86-64.zip",
        "p18759211_121020_Linux-x86-64.zip",
        "p19396455_121020_Linux-x86-64.zip",
        "p20875898_121020_Linux-x86-64.zip",
        "p22037014_121020_Linux-x86-64.zip",
        "p22873635_121020_Linux-x86-64.zip",
        "p23614158_121020_Linux-x86-64.zip",
        "p24701840_121020_Linux-x86-64.zip",
        "p25881255_121020_Linux-x86-64.zip",
        "p27015449_121020_Linux-x86-64.zip",
        "p28125601_121020_Linux-x86-64.zip",
        "p28852325_121020_Linux-x86-64.zip",
        "p29997937_121020_Linux-x86-64.zip",
        "p31335037_121020_Linux-x86-64.zip",
```

```
        "p32327201_121020_Linux-x86-64.zip",
        "p32327208_121020_Generic.zip",
        "p17969866_12102210119_Linux-x86-64.zip",
        "p20394750_12102210119_Linux-x86-64.zip",
        "p24835919_121020_Linux-x86-64.zip",
        "p23262847_12102201020_Linux-x86-64.zip",
        "p21171382_12102201020_Generic.zip",
        "p21091901_12102210720_Linux-x86-64.zip",
        "p33013352_12102210720_Linux-x86-64.zip",
        "p25031502_12102210720_Linux-x86-64.zip",
        "p23711335_12102191015_Generic.zip",
        "p19504946_121020_Linux-x86-64.zip"
    ],
    "installationParameters": {
        "unixGroupName": "dba",
        "unixGroupId": 12345,
        "unixUname": "oracle",
        "unixUid": 12345,
        "oracleHome": "/home/oracle/oracle.12.1.0.2",
        "oracleBase": "/home/oracle"
    }
}
```

Example Exemple de manifeste CEV pour Oracle Database 12c Version 2 (12.2)

Dans l'exemple suivant pour le PSU d'octobre 2021 pour Oracle Database 12c Version 2 (12.2), RDS Custom applique p33261817, puis p33192662, puis p29213893, et ainsi de suite. L'exemple définit de nouvelles valeurs pour l'utilisateur et le groupe UNIX/Linux, ainsi que pour le répertoire de base de base de données Oracle et la base Oracle.

```
{
    "mediaImportTemplateVersion":"2020-08-14",
    "databaseInstallationFileNames":[
        "V839960-01.zip"
    ],
    "opatchFileNames":[
        "p6880880_122010_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames":[
        "p33261817_122010_Linux-x86-64.zip"
    ],
    "otherPatchFileNames":[
        "p33192662_122010_Linux-x86-64.zip",
```

```
        "p29213893_122010_Generic.zip",
        "p28730253_122010_Linux-x86-64.zip",
        "p26352615_12201211019DBOCT2021RU_Linux-x86-64.zip",
        "p23614158_122010_Linux-x86-64.zip",
        "p24701840_122010_Linux-x86-64.zip",
        "p25173124_122010_Linux-x86-64.zip",
        "p25881255_122010_Linux-x86-64.zip",
        "p27015449_122010_Linux-x86-64.zip",
        "p28125601_122010_Linux-x86-64.zip",
        "p28852325_122010_Linux-x86-64.zip",
        "p29997937_122010_Linux-x86-64.zip",
        "p31335037_122010_Linux-x86-64.zip",
        "p32327201_122010_Linux-x86-64.zip",
        "p32327208_122010_Generic.zip"
    ],
    "installationParameters": {
        "unixGroupName": "dba",
        "unixGroupId": 12345,
        "unixUname": "oracle",
        "unixUid": 12345,
        "oracleHome": "/home/oracle/oracle.12.2.0.1",
        "oracleBase": "/home/oracle"
    }
}
```

Example Exemple de manifeste CEV pour Oracle Database 18c

Dans l'exemple suivant pour le PSU d'octobre 2021 pour Oracle Database 18c, RDS Custom applique p32126855, puis p28730253, puis p27539475, et ainsi de suite. L'exemple définit de nouvelles valeurs pour l'utilisateur et le groupe UNIX/Linux, ainsi que pour le répertoire de base de base de données Oracle et la base Oracle.

```
{
    "mediaImportTemplateVersion":"2020-08-14",
    "databaseInstallationFileNames":[
        "V978967-01.zip"
    ],
    "opatchFileNames":[
        "p6880880_180000_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames":[
        "p32126855_180000_Linux-x86-64.zip"
    ],
```

```
    "otherPatchFileNames":[
        "p28730253_180000_Linux-x86-64.zip",
        "p27539475_1813000DBRU_Linux-x86-64.zip",
        "p29213893_180000_Generic.zip",
        "p29374604_1813000DBRU_Linux-x86-64.zip",
        "p29782284_180000_Generic.zip",
        "p28125601_180000_Linux-x86-64.zip",
        "p28852325_180000_Linux-x86-64.zip",
        "p29997937_180000_Linux-x86-64.zip",
        "p31335037_180000_Linux-x86-64.zip",
        "p31335142_180000_Generic.zip"
    ]
    "installationParameters": {
        "unixGroupName": "dba",
        "unixGroupId": 12345,
        "unixUname": "oracle",
        "unixUid": 12345,
        "oracleHome": "/home/oracle/18.0.0.0.ru-2020-10.rur-2020-10.r1",
        "oracleBase": "/home/oracle/"
    }
}
```

Example Exemple de manifeste CEV pour Oracle Database 19c

Dans l'exemple suivant pour Oracle Database 19c, RDS Custom applique p32126828, puis p29213893, puis p29782284, et ainsi de suite. L'exemple définit de nouvelles valeurs pour l'utilisateur et le groupe UNIX/Linux, ainsi que pour le répertoire de base de base de données Oracle et la base Oracle.

```
{
    "mediaImportTemplateVersion": "2020-08-14",
    "databaseInstallationFileNames": [
        "V982063-01.zip"
    ],
    "opatchFileNames": [
        "p6880880_190000_Linux-x86-64.zip"
    ],
    "psuRuPatchFileNames": [
        "p32126828_190000_Linux-x86-64.zip"
    ],
    "otherPatchFileNames": [
        "p29213893_1910000DBRU_Generic.zip",
        "p29782284_1910000DBRU_Generic.zip",
```

```
        "p28730253_190000_Linux-x86-64.zip",
        "p29374604_1910000DBRU_Linux-x86-64.zip",
        "p28852325_190000_Linux-x86-64.zip",
        "p29997937_190000_Linux-x86-64.zip",
        "p31335037_190000_Linux-x86-64.zip",
        "p31335142_190000_Generic.zip"
    ],
    "installationParameters": {
        "unixGroupName": "dba",
        "unixGroupId": 12345,
        "unixUname": "oracle",
        "unixUid": 12345,
        "oracleHome": "/home/oracle/oracle.19.0.0.0.ru-2020-04.rur-2020-04.r1.EE.1",
        "oracleBase": "/home/oracle"
    }
}
```

Étape 6 (Facultative) : Valider le manifeste CEV

Vous pouvez également vérifier que le manifeste est un fichier JSON valide en exécutant le script Python `json.tool`. Par exemple, si vous changez vers le répertoire contenant un manifeste CEV nommé `manifest.json`, exécutez la commande suivante.

```
python -m json.tool < manifest.json
```

Étape 7 : Ajouter les autorisations IAM nécessaires

Assurez-vous que le principal IAM qui crée la CEV possède les politiques nécessaires décrites dans Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM.

Création d’un CEV

Vous pouvez créer un CEV en utilisant le AWS Management Console ou le AWS CLI. Spécifiez l'architecture multilocataire ou non multilocataire. Pour de plus amples informations, veuillez consulter Considérations relatives à l'architecture multilocataire.

En général, la création d'un CEV prend environ deux heures. Après avoir créé leCEV, vous pouvez l'utiliser pour créer ou mettre à niveau une instance de base de données RDS personnalisée. Pour plus d’informations, consultez Création d'une instance de base de données RDS Custom for Oracle et Mise à niveau d'une instance de base de données RDS personnalisée pour Oracle.

> **Note**
>
> Si votre instance de base de données utilise actuellement Oracle Linux 7.9, créez-en une nouvelle CEV qui utilise la dernière versionAMI, qui utilise Oracle Linux 8. Modifiez ensuite votre instance pour utiliser la nouvelleCEV.

Notez les exigences et limites suivantes pour créer un CEV :

- Le compartiment Amazon S3 contenant vos fichiers d'installation doit se trouver dans le même compartiment Région AWS que le vôtreCEV. Dans le cas contraire, le processus de création échoue.
- Le CEV nom doit être au format*major-engine-version*.*customized_string*, comme dans`19.cdb_cev1`.
- Le CEV nom doit contenir de 1 à 50 caractères alphanumériques, des traits de soulignement, des tirets ou des points.
- Le CEV nom ne peut pas contenir de points consécutifs, comme dans`19..cdb_cev1`.

Console

Pour créer une CEV

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).

   La page Versions personnalisées du moteur affiche tout CEVs ce qui existe actuellement. Si vous n'en avez pas crééCEVs, la page est vide.
3. Choisissez Create custom engine version (Créer une version de moteur personnalisée).
4. Dans Options de moteur, procédez comme suit :

   a. Pour Engine type (Type de moteur), choisissez Oracle.

   b. Pour les paramètres d'architecture, choisissez éventuellement Architecture multitenant pour créer un multitenant OracleCEV, qui utilise le moteur `custom-oracle-ee-cdb` de base de données ou. `custom-oracle-se2-cdb` Vous pouvez créer un RDS Custom pour Oracle

CDB avec un multitenant CEV uniquement. Si vous ne choisissez pas cette option, CEV il s'agit d'un non-CDB, qui utilise le moteur `custom-oracle-ee` ou`custom-oracle-se2`.

> **Note**
>
> L'architecture que vous choisissez est une caractéristique permanente de votreCEV. Vous ne pourrez pas modifier votre architecture CEV pour utiliser une autre architecture ultérieurement.

c. Choisissez l'une des options suivantes :

- Créer un nouveau CEV — Créez un CEV à partir de zéro. Dans ce cas, vous devez spécifier un JSON manifeste spécifiant les fichiers binaires de base de données.
- Créer CEV à partir de la source : dans Spécifiez CEV ce que vous souhaitez copier, choisissez un existant CEV à utiliser comme sourceCEV. Dans ce cas, vous pouvez spécifier une nouvelle Amazon Machine Image (AMI), mais vous ne pouvez pas spécifier de fichiers binaires de base de données différents.

d. Pour Version du moteur, choisissez la version majeure du moteur.

5. Dans Détails de la version, procédez comme suit :

a. Saisissez un nom valide dans Nom de version de moteur personnalisée. Par exemple, vous pouvez saisir le nom **`19.cdb_cev1`**.

b. (Facultatif) Entrez une description pour votreCEV.

6. Dans Média d'installation, procédez comme suit :

a. (Facultatif) Pour AMIID, laissez le champ vide pour utiliser le dernier service fourni ou AMI entrez un numéro AMI que vous avez utilisé précédemment pour créer un. CEV Pour obtenir un résultat valide AMIIDs, utilisez l'une des techniques suivantes :

- Dans la console, choisissez Versions personnalisées du moteur dans le volet de navigation de gauche, puis choisissez le nom d'unCEV. L'AMIID utilisé par le CEV apparaît dans l'onglet Configuration.
- Dans le AWS CLI, utilisez la `describe-db-engine-versions` commande. Recherchez la sortie pour `ImageID`.

b. Pour S3 location of manifest files (Emplacement S3 des fichiers manifestes), saisissez l'emplacement du compartiment Amazon S3 que vous avez spécifié dans Étape 3 : Charger

vos fichiers d'installation dans Amazon S3. Par exemple, saisissez **`s3://my-custom-installation-files/123456789012/cev1/`**.

> **Note**
>
> Le compartiment Région AWS dans lequel vous le créez CEV doit se trouver dans la même région que le compartiment S3.

c. (Créer un nouveau CEV uniquement) Pour le CEVmanifeste, entrez le JSON manifeste dans lequel vous l'avez créé Création du manifeste CEV.

7. Dans la section KMSclé, sélectionnez Entrer une clé ARN pour répertorier les AWS KMS clés disponibles. Sélectionnez ensuite votre KMS clé dans la liste.

   Une AWS KMS clé est requise pour RDS Custom. Pour de plus amples informations, veuillez consulter Étape 1 : Créer ou réutiliser une clé de chiffrement symétrique AWS KMS.

8. (Facultatif) Choisissez Ajouter une nouvelle balise pour créer une paire clé-valeur pour votre. CEV

9. Choisissez Create custom engine version (Créer une version de moteur personnalisée).

   Si le format du JSON manifeste n'est pas valide, la console affiche une erreur lors de la validation du CEV manifeste. Résolvez les problèmes et réessayez.

La page Custom engine versions (Versions de moteur personnalisées) s'affiche. Votre CEV est affiché avec le statut Création. Le processus de création d'un CEV prend environ deux heures.

AWS CLI

Pour créer un CEV en utilisant le AWS CLI, exécutez la commande create-custom-db-engine-version.

Les options suivantes sont requises :

- `--engine`— Spécifiez le type de moteur. Pour unCDB, spécifiez `custom-oracle-ee-cdb` soit`custom-oracle-se2-cdb`. Pour une valeur non-CDB, spécifiez l'un `custom-oracle-ee` ou l'autre`custom-oracle-se2`. Vous ne pouvez créer CDBs qu'à partir d'un CEV fichier créé avec `custom-oracle-ee-cdb` ou`custom-oracle-se2-cdb`. Vous pouvez créer non CDBs seulement à partir d'un CEV objet créé avec `custom-oracle-ee` ou`custom-oracle-se2`.
- `--engine-version` : spécifiez la version du moteur. Le format est*`major-engine-version`*. *`customized_string`*. Le CEV nom doit contenir de 1 à 50 caractères alphanumériques, des

traits de soulignement, des tirets ou des points. Le CEV nom ne peut pas contenir de points consécutifs, comme dans`19..cdb_cev1`.

- `--kms-key-id`— Spécifiez un AWS KMS key.
- `--manifest` : spécifiez *`manifest_json_string`* ou `--manifest file:`*`file_name`*. Les caractères de saut de ligne ne sont pas autorisés dans *`manifest_json_string`*. Assurez-vous d'éviter les guillemets («) dans le JSON code en les préfixant par une barre oblique inverse (\).

  L'exemple suivant illustre la *`manifest_json_string`* pour 19c à partir de Étape 5 : Préparer le manifeste CEV. L'exemple définit de nouvelles valeurs pour la base Oracle, le répertoire d'origine Oracle Home, ainsi que l'ID et le nom de l'utilisateur et du groupe UNIX /Linux. Si vous copiez cette chaîne, supprimez tous les caractères de nouvelle ligne avant de la coller dans votre commande.

  ```
  "{\"mediaImportTemplateVersion\": \"2020-08-14\",
  \"databaseInstallationFileNames\": [\"V982063-01.zip\"],
  \"opatchFileNames\": [\"p6880880_190000_Linux-x86-64.zip\"],
  \"psuRuPatchFileNames\": [\"p32126828_190000_Linux-x86-64.zip\"],
  \"otherPatchFileNames\": [\"p29213893_1910000DBRU_Generic.zip\",
  \"p29782284_1910000DBRU_Generic.zip\",\"p28730253_190000_Linux-
  x86-64.zip\",\"p29374604_1910000DBRU_Linux-x86-64.zip\",
  \"p28852325_190000_Linux-x86-64.zip\",\"p29997937_190000_Linux-x86-64.zip
  \",\"p31335037_190000_Linux-x86-64.zip\",\"p31335142_190000_Generic.zip
  \"]\"installationParameters\":{ \"unixGroupName\":\"dba\",
  \ \"unixUname\":\"oracle\", \ \"oracleHome\":\"/home/oracle/
  oracle.19.0.0.0.ru-2020-04.rur-2020-04.r1.EE.1\", \ \"oracleBase\":\"/
  home/oracle/\"}}"
  ```

- `--database-installation-files-s3-bucket-name` : spécifiez le même nom de compartiment que celui que vous avez indiqué dans Étape 3 : Charger vos fichiers d'installation dans Amazon S3. Le compartiment Région AWS dans lequel vous exécutez `create-custom-db-engine-version` doit appartenir à la même région que votre compartiment Amazon S3.

Vous pouvez également spécifier les options suivantes :

- `--description`— Spécifiez une description de votreCEV.
- `--database-installation-files-s3-prefix` : spécifiez le nom de dossier que celui que vous avez indiqué dans Étape 3 : Charger vos fichiers d'installation dans Amazon S3.

- `--image-id`— Spécifiez un AMI identifiant que vous souhaitez réutiliser. Pour trouver une IDs valeur valide, exécutez la `describe-db-engine-versions` commande, puis recherchez le résultat pour`ImageID`. Par défaut, RDS Custom for Oracle utilise la version la plus récente disponibleAMI.

L'exemple suivant crée un multitenant Oracle CEV nommé`19.cdb_cev1`. L'exemple réutilise une version existante AMI au lieu d'utiliser la dernière version disponibleAMI. Assurez-vous que le nom de votre moteur CEV commence par le numéro de version principal du moteur.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-custom-db-engine-version \
    --engine custom-oracle-se2-cdb \
    --engine-version 19.cdb_cev1 \
    --database-installation-files-s3-bucket-name us-east-1-123456789012-custom-
installation-files \
    --database-installation-files-s3-prefix 123456789012/cev1 \
    --kms-key-id my-kms-key \
    --description "test cev" \
    --manifest manifest_string \
    --image-id ami-012a345678901bcde
```

Dans Windows:

```
aws rds create-custom-db-engine-version ^
    --engine custom-oracle-se2-cdb ^
    --engine-version 19.cdb_cev1 ^
    --database-installation-files-s3-bucket-name us-east-1-123456789012-custom-
installation-files ^
    --database-installation-files-s3-prefix 123456789012/cev1 ^
    --kms-key-id my-kms-key ^
    --description "test cev" ^
    --manifest manifest_string ^
    --image-id ami-012a345678901bcde
```

Example

Obtenez des détails sur votre CEV en utilisant la `describe-db-engine-versions` commande.

```
aws rds describe-db-engine-versions \
    --engine custom-oracle-se2-cdb \
    --include-all
```

L'exemple de sortie partielle suivant affiche le moteur, les groupes de paramètres, le manifeste et d'autres informations.

```
{
    "DBEngineVersions": [
        {
            "Engine": "custom-oracle-se2-cdb",
            "EngineVersion": "19.cdb_cev1",
            "DBParameterGroupFamily": "custom-oracle-se2-cdb-19",
            "DBEngineDescription": "Containerized Database for Oracle Custom SE2",
            "DBEngineVersionDescription": "test cev",
            "Image": {
                "ImageId": "ami-012a345678901bcde",
                "Status": "active"
            },
            "ValidUpgradeTarget": [],
            "SupportsLogExportsToCloudwatchLogs": false,
            "SupportsReadReplica": true,
            "SupportedFeatureNames": [],
            "Status": "available",
            "SupportsParallelQuery": false,
            "SupportsGlobalDatabases": false,
            "MajorEngineVersion": "19",
            "DatabaseInstallationFilesS3BucketName": "us-east-1-123456789012-custom-
installation-files",
            "DatabaseInstallationFilesS3Prefix": "123456789012/cev1",
            "DBEngineVersionArn": "arn:aws:rds:us-east-1:123456789012:cev:custom-
oracle-se2-cdb/19.cdb_cev1/abcd12e3-4f5g-67h8-i9j0-k1234l56m789",
            "KMSKeyId": "arn:aws:kms:us-
east-1:732027699161:key/1ab2345c-6d78-9ef0-1gh2-3456i7j89k01",
            "CreateTime": "2023-03-07T19:47:58.131000+00:00",
            "TagList": [],
            "SupportsBabelfish": false,
...
```

Impossible de créer un CEV

Si le processus de création d'un CEV échoue, RDS Custom émet un problème `RDS-EVENT-0198` avec le message `Creation failed for custom engine version` *`major-engine-version.cev_name`* et inclut des informations sur l'échec. Par exemple, l'événement imprime les fichiers manquants.

Vous ne pouvez pas modifier un échecCEV. Vous ne pouvez que le supprimer, puis réessayer d'en créer un CEV après avoir corrigé les causes de l'échec. Pour plus d'informations sur la résolution des problèmes à l'origine de l'échec de CEV création, consultezRésolution des problèmes liés à la création d'une version de moteur personnalisée pour RDS Custom for Oracle.

Modification de l'état de la CEV

Vous pouvez modifier un CEV à l'aide du AWS Management Console ou du AWS CLI. Vous pouvez modifier la description de la CEV ou son état de disponibilité. Votre CEV possède l'une des valeurs d'état suivantes :

- `available` – Vous pouvez utiliser cette CEV pour créer une nouvelle instance de base de données RDS Custom ou mettre à niveau une instance de base de données. Il s'agit de l'état par défaut d'une nouvelle CEV.
- `inactive` – Vous ne pouvez pas créer ou mettre à niveau une instance RDS Custom avec cette CEV. Vous ne pouvez pas restaurer un instantané de base de données pour créer une nouvelle instance de base de données RDS Custom avec cette CEV.

Vous pouvez changer la CEV de n'importe quel état pris en charge vers n'importe quel autre état pris en charge. Vous pouvez modifier l'état pour empêcher l'utilisation accidentelle d'une CEV ou rendre une CEV abandonnée à nouveau éligible à l'utilisation. Par exemple, vous pouvez modifier l'état de votre CEV de `available`en `inactive`, et à nouveau de `inactive` en `available`.

Console

Pour modifier une CEV

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).
3. Choisissez une CEV dont vous souhaitez modifier la description ou l'état.

4. Pour Actions, choisissez Modifier.
5. Effectuez une ou plusieurs des modifications suivantes :
   - Pour Paramètres d'état de CEV (CEV status settings), choisissez un nouvel état de disponibilité.
   - Sur la page Version description (Description de la version), saisissez une nouvelle description.
6. Choisissez Modify CEV (Modifier la CEV).

   Si la CEV est en cours d'utilisation, la console affiche You can't modify the CEV status (Vous ne pouvez pas modifier l'état de la CEV). Résolvez les problèmes et réessayez.

La page Custom engine versions (Versions de moteur personnalisées) s'affiche.

AWS CLI

Pour modifier un CEV à l'aide de AWS CLI, exécutez la commande modify-custom-db-engine-version. Vous pouvez trouver les CEV à modifier en exécutant la describe-db-engine-versionscommande.

Les options suivantes sont requises :

- `--engine` *`engine-type`*, où le *`type de moteur`* est`custom-oracle-ee`,`custom-oracle-se2`, `custom-oracle-ee-cdb` ou `custom-oracle-se2-cdb`
- `--engine-version` *`cev`*, où *`cev`* représente le nom de la version de moteur personnalisée que vous souhaitez modifier
- `--status` *`status`*, où *`status`* représente l'état de disponibilité que vous souhaitez attribuer à la CEV

L'exemple suivant modifie une CEV nommée `19.my_cev1` de son état actuel vers `inactive`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-custom-db-engine-version \
    --engine custom-oracle-se2 \
    --engine-version 19.my_cev1 \
    --status inactive
```

Dans Windows :

```
aws rds modify-custom-db-engine-version ^
    --engine custom-oracle-se2 ^
    --engine-version 19.my_cev1 ^
    --status inactive
```

Affichage des CEV informations relatives à Amazon RDS Custom pour Oracle

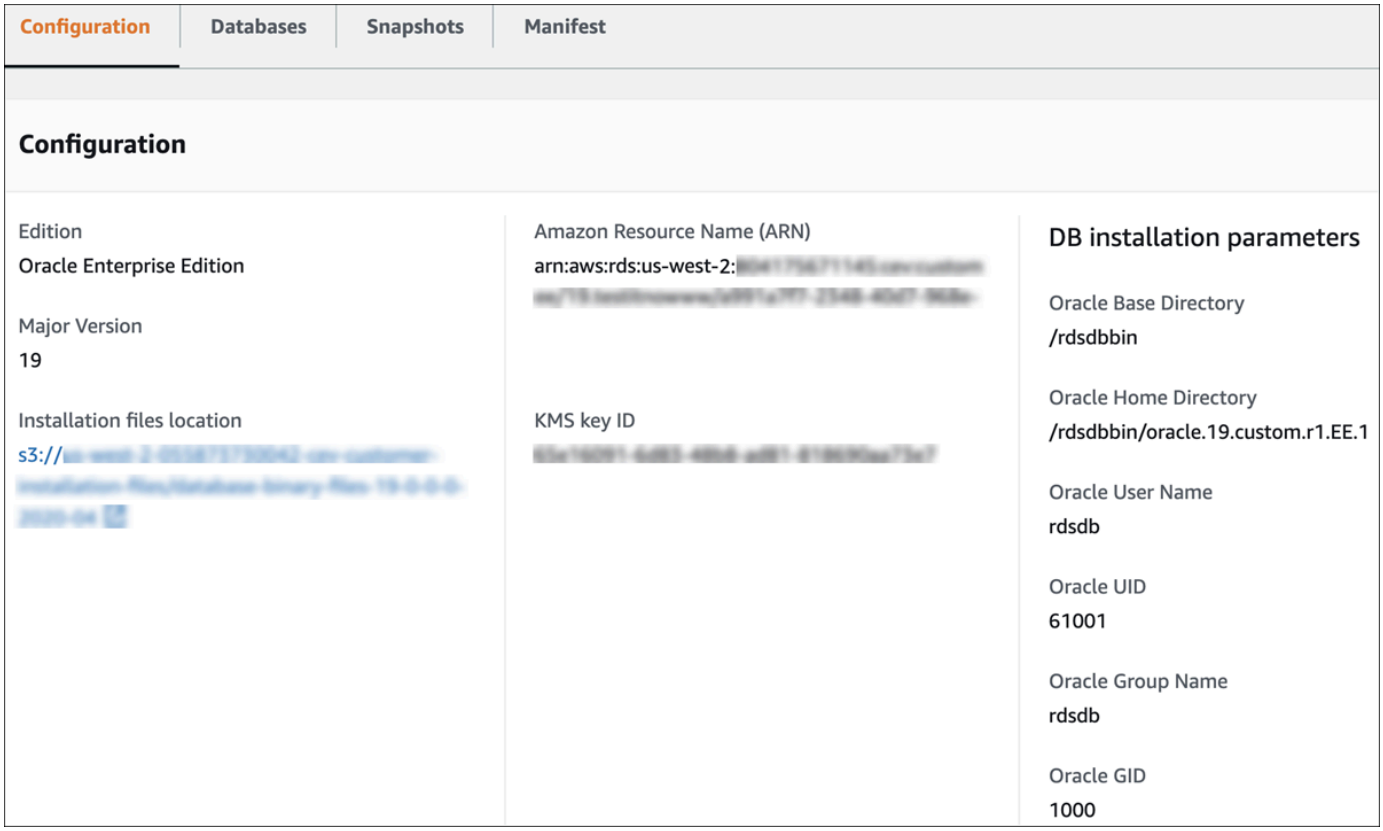
Vous pouvez consulter les détails de votre CEV manifeste et de la commande utilisée pour le créer en CEV utilisant le AWS Management Console ou le AWS CLI.

Console

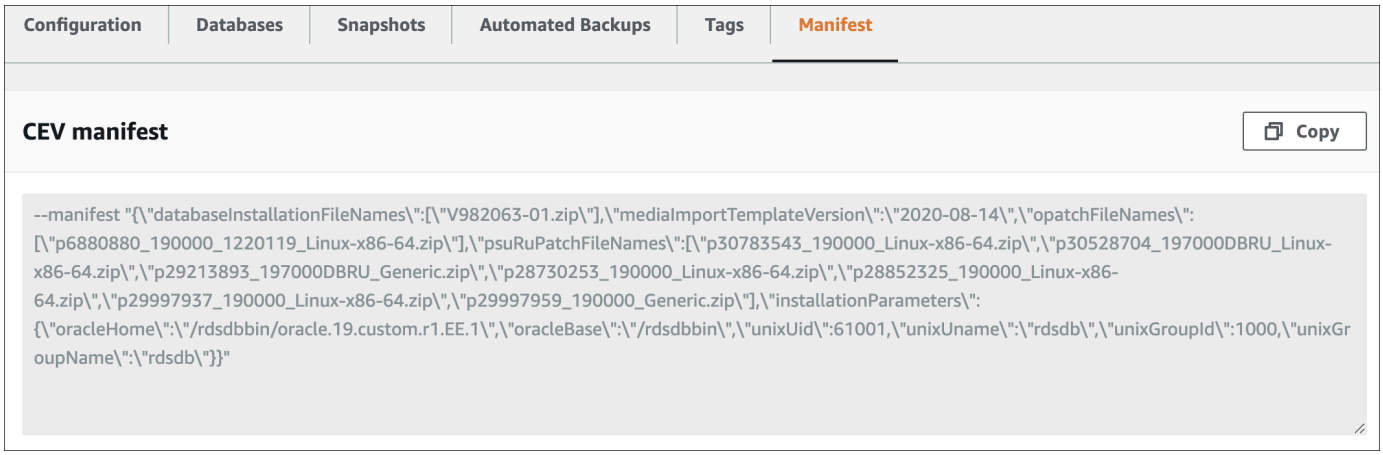
Pour afficher les CEV détails

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).

   La page Versions personnalisées du moteur affiche tout CEVs ce qui existe actuellement. Si vous n'en avez pas crééCEVs, la page est vide.
3. Choisissez le nom du CEV document que vous souhaitez consulter.
4. Choisissez Configuration pour afficher les paramètres d'installation spécifiés dans votre manifeste.



5. Choisissez Manifest (Manifeste) pour afficher les paramètres d'installation spécifiés dans l'option `--manifest` de la commande `create-custom-db-engine-version`. Vous pouvez copier ce texte, remplacer des valeurs selon vos besoins et les utiliser dans une nouvelle commande.



AWS CLI

Pour afficher les détails d'un CEV à l'aide de AWS CLI, exécutez la describe-db-engine-versionscommande.

Les options suivantes sont requises :

- `--engine` *`engine-type`*, où *`engine-type`* est `custom-oracle-ee` `custom-oracle-se2custom-oracle-ee-cdb`, ou `custom-oracle-se2-cdb`
- `--engine-version` *`major-engine-version.customized_string`*

L'exemple suivant crée un non- CDB CEV qui utilise Enterprise Edition. Le CEV nom `19.my_cev1` commence par le numéro de version du moteur principal, qui est obligatoire.

Example

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
    --engine custom-oracle-ee \
    --engine-version 19.my_cev1
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
    --engine custom-oracle-ee ^
    --engine-version 19.my_cev1
```

L'exemple de sortie partielle suivant affiche le moteur, les groupes de paramètres, le manifeste et d'autres informations.

```
"DBEngineVersions": [
    {
        "Engine": "custom-oracle-ee",
        "MajorEngineVersion": "19",
        "EngineVersion": "19.my_cev1",
        "DatabaseInstallationFilesS3BucketName": "us-east-1-123456789012-cev-customer-
installation-files",
        "DatabaseInstallationFilesS3Prefix": "123456789012/cev1",
        "CustomDBEngineVersionManifest": "{\n\"mediaImportTemplateVersion\":
 \"2020-08-14\",\n\"databaseInstallationFileNames\": [\n\"V982063-01.zip\"\n],
\n\"installationParameters\": {\n\"oracleBase\":\"/tmp\",\n\"oracleHome\":\"/
tmp/Oracle\"\n},\n\"opatchFileNames\": [\n\"p6880880_190000_Linux-x86-64.zip
\"\n],\n\"psuRuPatchFileNames\": [\n\"p32126828_190000_Linux-x86-64.zip
\"\n],\n\"otherPatchFileNames\": [\n\"p29213893_1910000DBRU_Generic.zip\",\n
```

```
\"p29782284_1910000DBRU_Generic.zip\",\n\"p28730253_190000_Linux-x86-64.zip\",\n
\"p29374604_1910000DBRU_Linux-x86-64.zip\",\n\"p28852325_190000_Linux-x86-64.zip\",
\n\"p29997937_190000_Linux-x86-64.zip\",\n\"p31335037_190000_Linux-x86-64.zip\",\n
\"p31335142_190000_Generic.zip\"\n]\n}\n",
        "DBParameterGroupFamily": "custom-oracle-ee-19",
        "DBEngineDescription": "Oracle Database server EE for RDS Custom",
        "DBEngineVersionArn": "arn:aws:rds:us-west-2:123456789012:cev:custom-oracle-
ee/19.my_cev1/0a123b45-6c78-901d-23e4-5678f901fg23",
        "DBEngineVersionDescription": "test",
        "KMSKeyId": "arn:aws:kms:us-east-1:123456789012:key/ab1c2de3-f4g5-6789-h012-
h3ijk4567l89",
        "CreateTime": "2022-11-18T09:17:07.693000+00:00",
        "ValidUpgradeTarget": [
        {
            "Engine": "custom-oracle-ee",
            "EngineVersion": "19.cev.2021-01.09",
            "Description": "test",
            "AutoUpgrade": false,
            "IsMajorVersionUpgrade": false
        }
]
```

Suppression d'une CEV

Vous pouvez supprimer un CEV en utilisant le AWS Management Console ou le AWS CLI. La suppression prend généralement quelques minutes.

Pour supprimer une CEV, elle ne peut être utilisée par aucun des éléments suivants :

- Une instance de base de données RDS Custom
- Un instantané d'une instance de base de données RDS Custom
- Une sauvegarde automatisée de votre instance de base de données RDS Custom

Console

Pour supprimer une CEV

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).

3. Choisissez une CEV dont vous souhaitez supprimer la description ou l'état.
4. Pour Actions, choisissez Supprimer.

   La boîte de dialogue Delete ***cev_name***? (Supprimer cev_name ?) s'affiche.
5. Entrez **`delete me`**, puis choisissez Delete (Supprimer).

   Dans Custom engine versions (Versions de moteur personnalisées), la bannière indique que votre CEV est en cours de suppression.

AWS CLI

Pour supprimer un CEV à l'aide de AWS CLI, exécutez la commande delete-custom-db-engine-version.

Les options suivantes sont requises :

- `--engine` *`engine-type`*, où le *`type de moteur`* est`custom-oracle-ee`,`custom-oracle-se2`, `custom-oracle-ee-cdb` ou `custom-oracle-se2-cdb`
- `--engine-version` *`cev`*, où *`cev`* représente le nom de la version de moteur personnalisée à supprimer

L'exemple suivant supprime une CEV nommée `19.my_cev1`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds delete-custom-db-engine-version \
    --engine custom-oracle-ee \
    --engine-version 19.my_cev1
```

Dans Windows :

```
aws rds delete-custom-db-engine-version ^
    --engine custom-oracle-ee ^
    --engine-version 19.my_cev1
```

Configuration d'une instance de base de données pour Amazon RDS Custom for Oracle

Vous pouvez créer une instance de base de données RDS Custom, puis vous y connecter à l'aide de Secure Shell (SSH) ou de AWS Systems Manager.

Pour plus d'informations sur la connexion et la connexion à une instance de base de données RDS Custom pour Oracle, consultez les rubriques suivantes.

- Connexion à votre instance de base de données RDS personnalisée à l'aide du gestionnaire de session
- Connexion à votre instance de base de données RDS Custom à l'aide de SSH
- Connexion à votre base de données RDS Custom for Oracle en tant que SYS

Création d'une instance de base de données RDS Custom for Oracle

Créez une instance de base de données Amazon RDS personnalisée pour Oracle à l'aide du AWS Management Console ou du AWS CLI. La procédure est similaire à la procédure de création d'une instance de base de données Amazon RDS. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

Si vous avez inclus des paramètres d'installation dans votre manifeste CEV, votre instance de base de données utilise la base Oracle, le répertoire de base de base de données Oracle, ainsi que l'ID et le nom de l'utilisateur et du groupe UNIX/Linux que vous avez spécifiés. Le fichier `oratab`, créé par Oracle Database lors de l'installation, pointe vers l'emplacement d'installation réel plutôt que vers un lien symbolique. Quand RDS Custom for Oracle exécute des commandes, il s'exécute en tant qu'utilisateur du système d'exploitation configuré plutôt qu'en tant qu'utilisateur par défaut `rdsdb`. Pour de plus amples informations, veuillez consulter Étape 5 : Préparer le manifeste CEV.

Avant d'essayer de créer une instance de base de données RDS Custom ou de vous y connecter, réalisez les tâches dans Configuration de votre environnement pour Amazon RDS Custom for Oracle.

console

Pour créer une instance de base de données RDS Custom for Oracle

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez Create database (Créer une base de données).
4. Dans Choose a database creation method (Choisir une méthode de création de base de données), sélectionnez Standard Create (Création standard).
5. Dans la section Options de moteur, procédez comme suit :

   a. Pour Engine type (Type de moteur), choisissez Oracle.

   b. Pour Database management type (Type de gestion de la base de données), choisissez Amazon RDS Custom.

   c. Pour Paramètres d'architecture, procédez de l'une des manières suivantes :

      - Sélectionnez Architecture à locataires multiples pour créer une base de données de conteneurs (CDB). Lors de sa création, votre CDB contient un conteneur initial de PDB et une PDB initiale.

        > **Note**
        >
        > Le paramètre Architecture à locataires multiples est pris en charge uniquement pour Oracle Database 19c.

      - Effacez Architecture à locataires multiples pour créer une base de données non-CDB. Un fichier non CDB ne peut pas contenir. PDBs

   d. Pour Edition, choisissez Oracle Enterprise Edition ou Oracle Standard Edition 2.

   e. Pour Version de moteur personnalisée, choisissez une version de moteur personnalisée (CEV) RDS Custom existante. Une CEV a le format suivant : *`major-engine-version.customized_string`*. Un exemple d'identificateur est `19.cdb_cev1`.

      Si vous avez choisi l'architecture multitenant à l'étape précédente, vous ne pouvez spécifier qu'un CEV utilisant le type de `custom-oracle-se2-cdb` moteur `custom-oracle-ee-cdb` or. La console filtre ceux CEVs qui ont été créés avec différents types de moteurs.
6. Dans Templates (Modèles), sélectionnez Production.
7. Dans la section Settings (Paramètres), procédez comme suit :

   a. Pour Identifiant d'instance de base de données, saisissez un nom unique pour votre instance de base de données.

b. Pour Identifiant principal, saisissez un nom d'utilisateur. Vous pouvez récupérer cette valeur à partir de la console ultérieurement.

   Lorsque vous vous connectez à une base de données non-CDB, l'utilisateur principal est l'utilisateur de la base de données non-CDB. Lorsque vous vous connectez à une CDB, l'utilisateur principal est l'utilisateur de la PDB. Pour vous connecter à la racine CDB, connectez-vous à l'hôte, démarrez un client SQL et créez un utilisateur administratif à l'aide de commandes SQL.

c. Effacez Générer automatiquement un mot de passe.

8. Choisissez une Classe d'instance de base de données.

   Pour connaître les classes prises en charge, consultez Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle.

9. Dans la section Storage (Stockage), procédez comme suit :

   a. Pour Type de stockage, choisissez un type de SSD : io1, gp2 ou gp3. Vous disposez des options supplémentaires suivantes :

      - Pour io1 ou gp3, choisissez un taux pour IOPS provisionnés. La valeur par défaut est 1 000 pour io1 et 12 000 pour gp3.
      - Pour gp3, choisissez un taux pour Débit de stockage. La valeur par défaut est 500 MiBps.

   b. Pour Stockage alloué, choisissez une taille de stockage. La valeur par défaut est 40 Gio.

10. Pour Connectivité, spécifiez votre Cloud privé virtuel (VPC), votre Groupe de sous-réseaux de base de données et votre Groupe de sécurité VPC (pare-feu).

11. Pour RDS Custom security (Sécurité RDS Custom), procédez comme suit :

    a. Pour IAM instance profile (Profil d'instance IAM), choisissez le profil d'instance de votre instance de base de données RDS Custom for Oracle.

       Le profil d'instance IAM doit commencer par `AWSRDSCustom`, par exemple *`AWSRDSCustomInstanceProfileForRdsCustomInstance`*.

    b. Pour le chiffrement, choisissez Enter a key ARN pour répertorier les AWS KMS clés disponibles. Choisissez ensuite votre clé dans la liste.

       Une AWS KMS clé est requise pour RDS Custom. Pour de plus amples informations, veuillez consulter Étape 1 : Créer ou réutiliser une clé de chiffrement symétrique AWS KMS.

12. Pour Options de base de données, procédez comme suit :

a. (Facultatif) Pour Identifiant système (SID), entrez une valeur pour le SID Oracle, qui est également le nom de votre CDB. Le SID est le nom de l'instance de base de données Oracle qui gère vos fichiers de base de données. Dans ce contexte, le terme « instance de base de données Oracle » fait exclusivement référence à la zone SGA (System Global Area) et aux processus d'arrière-plan d'Oracle. Si vous ne spécifiez pas de SID, la valeur par défaut est **`RDSCDB`**.

b. (Facultatif) Pour Nom de la base de données initiale, saisissez un nom. La valeur par défaut est **`ORCL`**. Dans une architecture à locataires multiples, le nom de la base de données initiale est le nom de la PDB.

> **Note**
>
> Le SID et le nom de la PDB doivent être différents.

c. Pour Groupe d'options, choisissez un groupe d'options ou acceptez le groupe par défaut.

> **Note**
>
> La seule option prise en charge pour RDS Custom pour Oracle est`Timezone`. Pour de plus amples informations, veuillez consulter Fuseau horaire Oracle.

d. Pour Période de rétention des sauvegardes, choisissez une valeur. Vous ne pouvez pas choisir 0 jour.

e. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données RDS Custom préférés. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données. Les paramètres suivants n'apparaissent pas dans la console et ne sont pas pris en charge :

- Processor features (Caractéristiques du processeur)
- Dimensionnement automatique du stockage
- Option Password and Kerberos authentication (Mot de passe et authentification Kerberos) dans Database authentication (Authentification de base de données) (seule l'authentification par mot de passe est prise en charge)
- Performance Insights
- Exportations des journaux

- Enable auto minor version upgrade (Activer la mise à niveau automatique de versions mineures)
- Deletion protection (Protection contre la suppression)

13. Choisissez Créer une base de données.

> **Important**
>
> Quand vous créez une instance de base de données RDS Custom for Oracle, vous pouvez recevoir l'erreur suivante : The service-linked role is in the process of being created (Le rôle lié à un service est en cours de création). Réessayez ultérieurement. Dans ce cas, attendez quelques minutes, puis réessayez de créer l'instance de base de données.

Le bouton View credential details (Afficher les détails des informations d'identification) apparaît sur la page Databases (Bases de données).

Pour afficher le nom d'utilisateur principal et le mot de passe pour l'instance de base de données RDS Custom, choisissez View credential details (Afficher les informations d'identification).

Pour vous connecter à l'instance de base de données en tant qu'utilisateur principal, utilisez l'identifiant et le mot de passe affichés.

> **Important**
>
> Vous ne pouvez pas afficher le mot de passe de l'utilisateur principal de nouveau dans la console. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier. Pour modifier le mot de passe de l'utilisateur principal une fois que l'instance de base de données RDS Custom est disponible, connectez-vous à la base de données et exécutez une commande `ALTER USER`. Vous ne pouvez pas réinitialiser le mot de passe à l'aide de l'option Modifier dans la console.

14. Choisissez Databases (Bases de données) pour afficher la liste des instances de base de données RDS Custom.
15. Choisissez l'instance de base de données RDS Custom que vous venez de créer.

Sur la console RDS, les détails de la nouvelle instance de base de données RDS Custom s'affichent.

- L'instance de base de données a le statut creating (création) jusqu'à ce que l'instance de base de données RDS Custom soit créée et prête à l'emploi. Lorsque l'état devient available (disponible), vous pouvez vous connecter à l'instance de base de données. En fonction du stockage et de la classe d'instance alloués, la mise à disposition de la nouvelle instance de base de données peut nécessiter plusieurs minutes.
- Role (Rôle) a la valeur Instance (RDS Custom).
- RDS Custom automation mode (Mode d'automatisation RDS Custom) a la valeur Full automation(Automatisation complète). Ce paramètre signifie que l'instance de base de données assure une surveillance et une récupération d'instance automatiques.

AWS CLI

Vous créez une instance de base de données personnalisée RDS à l'aide de la create-db-instance AWS CLI commande.

Les options suivantes sont requises :

- `--db-instance-identifier`
- `--db-instance-class` Pour obtenir la liste des classes d'instance de base de données prises en charge, veuillez consulter Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle).
- `--engine` *`engine-type`*, où *`engine-type`* se trouve `custom-oracle-eecustom-oracle-se2`,`custom-oracle-ee-cdb`, ou `custom-oracle-se2-cdb`
- `--engine-version` *`cev`* (où *`cev`* est le nom de la version de moteur personnalisée que vous avez spécifiée dans Création d'un CEV)
- `--kms-key-id` *`my-kms-key`*
- `--backup-retention-period` *`days`* (où *`days`* est une valeur supérieure à `0`)
- `--no-auto-minor-version-upgrade`
- `--custom-iam-instance-profile AWSRDSCustomInstanceProfile-`*`us-east-1`* (où *`region`* est la Région AWS où vous créez votre instance de base de données)

L'exemple suivant créé une instance de base de données RDS personnalisée nommée `my-cfo-cdb-instance`. La base de données est une CDB dont le nom n'est pas celui par défaut. *MYCDB* Le nom PDB autre que le nom par défaut est. *MYPDB* La période de rétention des sauvegardes est de trois jours.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --engine custom-oracle-ee-cdb \
    --db-instance-identifier my-cfo-cdb-instance \
    --engine-version 19.cdb_cev1 \
    --db-name MYPDB \
    --db-system-id MYCDB \
    --allocated-storage 250 \
    --db-instance-class db.m5.xlarge \
    --db-subnet-group mydbsubnetgroup \
    --master-username myuser \
    --master-user-password mypassword \
    --backup-retention-period 3 \
    --port 8200 \
    --kms-key-id my-kms-key \
    --no-auto-minor-version-upgrade \
    --custom-iam-instance-profile AWSRDSCustomInstanceProfile-us-east-1
```

Dans Windows:

```
aws rds create-db-instance ^
    --engine custom-oracle-ee-cdb ^
    --db-instance-identifier my-cfo-cdb-instance ^
    --engine-version 19.cdb_cev1 ^
    --db-name MYPDB ^
    --db-system-id MYCDB ^
    --allocated-storage 250 ^
    --db-instance-class db.m5.xlarge ^
    --db-subnet-group mydbsubnetgroup ^
    --master-username myuser ^
    --master-user-password mypassword ^
    --backup-retention-period 3 ^
    --port 8200 ^
    --kms-key-id my-kms-key ^
    --no-auto-minor-version-upgrade ^
```

```
    --custom-iam-instance-profile AWSRDSCustomInstanceProfile-us-east-1
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Obtenez des détails sur votre instance en utilisant la commande `describe-db-instances`.

Example

```
aws rds describe-db-instances --db-instance-identifier my-cfo-cdb-instance
```

La sortie partielle suivante affiche le moteur, les groupes de paramètres et d'autres informations.

```
        {
            "DBInstanceIdentifier": "my-cfo-cdb-instance",
            "DBInstanceClass": "db.m5.xlarge",
            "Engine": "custom-oracle-ee-cdb",
            "DBInstanceStatus": "available",
            "MasterUsername": "admin",
            "DBName": "MYPDB",
            "DBSystemID": "MYCDB",
            "Endpoint": {
                "Address": "my-cfo-cdb-instance.abcdefghijkl.us-
east-1.rds.amazonaws.com",
                "Port": 1521,
                "HostedZoneId": "A1B2CDEFGH34IJ"
            },
            "AllocatedStorage": 100,
            "InstanceCreateTime": "2023-04-12T18:52:16.353000+00:00",
            "PreferredBackupWindow": "08:46-09:16",
            "BackupRetentionPeriod": 7,
            "DBSecurityGroups": [],
            "VpcSecurityGroups": [
                {
                    "VpcSecurityGroupId": "sg-0a1bcd2e",
                    "Status": "active"
                }
            ],
            "DBParameterGroups": [
                {
```

```
                    "DBParameterGroupName": "default.custom-oracle-ee-cdb-19",
                    "ParameterApplyStatus": "in-sync"
                }
            ],
...
```

Considérations relatives à l'architecture multilocataire

Si vous créez une instance de base de données Amazon RDS Custom pour Oracle avec l'architecture mutualisée Oracle (`custom-oracle-ee-cdb`ou le type de `custom-oracle-se2-cdb` moteur), votre base de données est une base de données de conteneurs (CDB). Si vous ne spécifiez pas l'architecture mutualisée Oracle, votre base de données est une base de données non CDB traditionnelle qui utilise le type de moteur `custom-oracle-ee` or`custom-oracle-se2`. Une base de données non CDB ne peut pas contenir de bases de données enfichables (). PDBs Pour de plus amples informations, veuillez consulter Architecture de base de données pour Amazon RDS Custom for Oracle.

Lorsque vous créez une instance de CDB RDS Custom for Oracle, tenez compte des points suivants :

- Vous pouvez créer une base de données multilocataire uniquement à partir d'une version CEV Oracle Database 19c.
- Vous ne pouvez créer une instance CDB que si le CEV utilise le type de `custom-oracle-se2-cdb` moteur `custom-oracle-ee-cdb` or.
- Si vous créez une instance CDB à l'aide de l'édition Standard 2, la CDB peut en contenir un maximum de 3. PDBs
- Par défaut, votre CDB est nommée `RDSCDB`, qui est également le nom de l'identifiant système Oracle (SID Oracle). Vous pouvez choisir un autre nom.
- Votre CDB contient une seule PDB initiale. Le nom par défaut de la PDB est `ORCL`. Vous pouvez choisir un nom différent pour votre PDB initiale, mais le SID Oracle et le nom de la PDB ne peuvent pas être identiques.
- RDS Custom for Oracle ne fournit pas APIs de PDBs. Pour en créer d'autres PDBs, utilisez la commande Oracle SQL`CREATE PLUGGABLE DATABASE`. RDS Custom for Oracle ne limite pas le nombre de produits PDBs que vous pouvez créer. En général, vous êtes responsable de la création et de la gestion PDBs, comme dans le cas d'un déploiement sur site.
- Vous ne pouvez pas utiliser RDS APIs pour créer, modifier ou supprimer PDBs : vous devez utiliser les instructions Oracle SQL. Lorsque vous créez un PDB à l'aide d'Oracle SQL, nous

vous recommandons de prendre un instantané manuel par la suite au cas où vous auriez besoin d'effectuer une point-in-time restauration (PITR).

- Vous ne pouvez pas renommer un fichier existant à PDBs l'aide d'Amazon APIs RDS. Vous ne pouvez pas non plus renommer la CDB à l'aide de la commande `modify-db-instance`.
- Le mode ouvert pour la racine CDB est `READ WRITE` sur la base de données principale et `MOUNTED` sur une base de données de secours montée. RDS Custom for Oracle tente de tout ouvrir PDBs lors de l'ouverture du CDB. Si RDS Custom for Oracle ne peut pas tout ouvrir PDBs, il émet l'événement`tenant database shutdown`.

Rôle lié à un service RDS Custom

Un rôle lié à un service permet à Amazon RDS Custom d'accéder aux ressources de votre. Compte AWS RDS Custom est ainsi simplifié, étant donné que vous n'avez pas besoin d'ajouter manuellement les autorisations requises. RDS Custom définit les autorisations de ses rôles liés à un service ; sauf définition contraire, seul RDS Custom peut endosser ses rôles. Les autorisations définies comprennent la politique d'approbation et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Lorsque vous créez une instance de base de données RDS Custom, les rôles liés au service Amazon RDS et RDS Custom sont créés (s'ils n'existent pas déjà) et utilisés. Pour de plus amples informations, veuillez consulter Utilisation des rôles liés à un service pour Amazon RDS.

La première fois que vous créez une instance de base de données RDS Custom for Oracle, vous pouvez recevoir l'erreur suivante :The service-linked role is in the process of being created. (Le rôle lié à un service est en cours de création.) Réessayez ultérieurement. Dans ce cas, attendez quelques minutes, puis réessayez de créer l'instance de base de données.

Installation de composants logiciels supplémentaires sur votre instance de base de données RDS Custom for Oracle

Dans une instance de base de données nouvellement créée, votre environnement de base de données inclut des fichiers binaires Oracle, une base de données et un écouteur de base de données. Vous souhaiterez peut-être installer des logiciels supplémentaires sur le système d'exploitation hôte de l'instance de base de données. Par exemple, vous souhaiterez peut-être installer Oracle Application Express (APEX), l'agent Oracle Enterprise Manager (OEM) ou l'agent Guardium S-TAP. Pour obtenir des directives et des instructions de haut niveau, consultez le billet de AWS blog détaillé Installer des composants logiciels supplémentaires sur Amazon RDS Custom for Oracle.

Connexion à votre instance de base de données RDS personnalisée à l'aide du gestionnaire de session

Après avoir créé votre instance de base de données RDS personnalisée, vous pouvez vous y connecter à l'aide de AWS Systems Manager Session Manager. Il s'agit de la technique préférée lorsque votre instance de base de données n'est pas accessible au public.

Le gestionnaire de session vous permet d'accéder aux EC2 instances Amazon via un shell basé sur un navigateur ou via le. AWS CLI Pour plus d'informations, consultez AWS Systems Manager Session Manager.

Console

Connexion à votre instance de base de données à l'aide de Session Manager

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée à laquelle vous souhaitez vous connecter.
3. Choisissez Configuration.
4. Notez l'ID de ressource de l'instance de base de données. Par exemple, l'ID de la ressource peut être `db-ABCDEFGHIJKLMNOPQRS0123456`.
5. Ouvrez la EC2 console Amazon à l'adresse https://console.aws.amazon.com/ec2/.
6. Dans le panneau de navigation, choisissez Instances.
7. Recherchez le nom de votre EC2 instance, puis cliquez sur l'ID d'instance qui lui est associé. Par exemple, l'ID d'instance peut être `i-abcdefghijklm01234`.
8. Choisissez Se connecter.
9. Choisissez Session Manager.
10. Choisissez Se connecter.

    Une fenêtre s'ouvre pour votre session.

AWS CLI

Vous pouvez vous connecter à votre instance de base de données RDS personnalisée à l'aide du AWS CLI. Cette technique nécessite le plugin Session Manager pour la AWS CLI. Pour en savoir

plus sur l'installation du plugin, veuillez consulter Installez le plugin du gestionnaire de session pour la AWS CLI.

Pour trouver l'ID de ressource de base de données de votre instance de base de données RDS personnalisée, utilisez`aws rds describe-db-instances`.

```
aws rds describe-db-instances \
    --query 'DBInstances[*].[DBInstanceIdentifier,DbiResourceId]' \
    --output text
```

L'exemple de sortie suivant montre l'ID de ressource de votre instance RDS personnalisée. Le préfixe est `db-`.

```
db-ABCDEFGHIJKLMNOPQRS0123456
```

Pour trouver l'ID d'EC2instance de votre instance de base de données, utilisez`aws ec2 describe-instances`. L'exemple suivant utilise `db-ABCDEFGHIJKLMNOPQRS0123456` pour l'ID de la ressource

```
aws ec2 describe-instances \
    --filters "Name=tag:Name,Values=db-ABCDEFGHIJKLMNOPQRS0123456" \
    --output text \
    --query 'Reservations[*].Instances[*].InstanceId'
```

L'exemple de sortie suivant montre l'ID de l'EC2instance.

```
i-abcdefghijklm01234
```

Utilisez la `aws ssm start-session` commande en fournissant l'ID de l'EC2instance dans le `--target` paramètre.

```
aws ssm start-session --target "i-abcdefghijklm01234"
```

Un résultat réussi ressemble à ce qui suit.

```
Starting session with SessionId: yourid-abcdefghijklm1234
[ssm-user@ip-123-45-67-89 bin]$
```

Connexion à votre instance de base de données RDS Custom à l'aide de SSH

Le protocole SSH (Secure Shell Protocol) est un protocole réseau qui prend en charge les communications chiffrées sur un réseau non sécurisé. Après avoir créé votre instance de base de données RDS Custom, vous pouvez vous y connecter à l'aide d'un client SSH. Pour plus d'informations, consultez Connexion à votre instance Linux à l'aide de SSH.

Votre technique de connexion SSH dépend du caractère privé ou non de votre instance de base de données, ce qui signifie qu'elle n'accepte pas les connexions depuis l'Internet public. Dans ce cas, vous devez utiliser le tunneling SSH pour connecter l'utilitaire SSH à votre instance. Cette technique transporte les données via un flux de données dédié (tunnel) au sein d'une session SSH existante. Vous pouvez configurer le tunneling SSH à l'aide d' AWS Systems Manager.

> **Note**
>
> Différentes stratégies sont prises en charge pour accéder aux instances privées. Pour savoir comment connecter un client SSH à des instances privées à l'aide d'hôtes Bastion, consultez Hôtes bastion Linux sur AWS. Pour savoir comment configurer le réacheminement de port, consultez Réacheminement de port à l'aide d' AWS Systems Manager Session Manager (langue française non garantie).

Si votre instance de base de données se trouve dans un sous-réseau public et que le paramètre est accessible au public, aucun tunneling SSH n'est requis. Vous pouvez vous connecter via SSH comme vous le feriez avec une EC2 instance Amazon publique.

Pour connecter un client SSH à votre instance de base de données, procédez comme suit :

1. Étape 1 : Configurer votre instance de base de données pour autoriser les connexions SSH
2. Étape 2 : récupérer votre clé secrète SSH et votre identifiant d' EC2 instance
3. Étape 3 : Connectez-vous à votre EC2 instance à l'aide de l'utilitaire ssh

Étape 1 : Configurer votre instance de base de données pour autoriser les connexions SSH

Pour vérifier que votre instance de base de données accepte les connexions SSH, procédez comme suit :

- Assurez-vous que le groupe de sécurité de votre instance de base de données autorise les connexions entrantes sur le port 22 pour TCP.

Pour apprendre à configurer le groupe de sécurité de votre instance de base de données, consultez Contrôle d'accès par groupe de sécurité.

- Si vous ne prévoyez pas d'utiliser le tunneling SSH, assurez-vous que votre instance de base de données réside dans un sous-réseau public et qu'elle est accessible au public.

  Dans la console, le champ correspondant est Accessible publiquement dans l'onglet Connectivité et sécurité de la page de détails de la base de données. Pour vérifier vos paramètres dans l'interface de ligne de commande, exécutez la commande suivante :

  ```
  aws rds describe-db-instances \
  --query 'DBInstances[*].
  {DBInstanceIdentifier:DBInstanceIdentifier,PubliclyAccessible:PubliclyAccessible}' \
  --output table
  ```

  Pour modifier les paramètres d'accessibilité de votre instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Étape 2 : récupérer votre clé secrète SSH et votre identifiant d' EC2 instance

Pour vous connecter à l'instance de base de données avec SSH, vous devez disposer de la paire de clés SSH associée à l'instance. RDS Custom crée la paire de clés SSH en votre nom, en utilisant la convention de dénomination ou. `do-not-delete-rds-custom-ssh-privatekey-`*`resource_id`*`-`*`uuid` `rds-custom!oracle-do-not-delete-`*`resource_id`*`-`*`uuid`*`-ssh-privatekey` AWS Secrets Manager stocke votre clé privée SSH en tant que secret.

Récupérez votre clé secrète SSH en utilisant l'un AWS Management Console ou l'autre des AWS CLI. Si votre instance possède un DNS public et que vous n'avez pas l'intention d'utiliser le tunneling SSH, récupérez également le nom DNS. Vous spécifiez le nom DNS pour les connexions publiques.

console

Pour récupérer la clé secrète SSH

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données), puis l'instance de base de données RDS Custom que vous voulez arrêter.
3. Choisissez Configuration.

4. Notez la valeur Resource ID (ID de ressource). Par exemple, l'ID de la ressource de l'instance de base de données peut être `db-ABCDEFGHIJKLMNOPQRS0123456`.
5. Ouvrez la EC2 console Amazon à l'adresse https://console.aws.amazon.com/ec2/.
6. Dans le panneau de navigation, choisissez Instances.
7. Trouvez le nom de votre EC2 instance et choisissez l'ID d'instance qui lui est associé. Par exemple, l'ID d' EC2 instance peut être`i-abcdefghijklm01234`.
8. Dans Details (Détails), cherchez Key pair name (Nom de la paire de clés). Le nom de la paire inclut l'ID de ressource de l'instance de base de données. Par exemple, le nom de la paire peut être `do-not-delete-rds-custom-ssh-privatekey-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c` ou`rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey`.
9. Si votre EC2 instance est publique, notez le IPv4DNS public. Par exemple, l'adresse DNS (Domain Name System) publique peut être `ec2-12-345-678-901.us-east-2.compute.amazonaws.com`.
10. Ouvrez la AWS Secrets Manager console à l'adresse https://console.aws.amazon.com/secretsmanager/.
11. Choisissez le secret portant le même nom que votre paire de clés.
12. Choisissez Retrieve secret value (Récupérer la valeur d'un secret).
13. Copiez la clé privée SSH dans un fichier texte, puis enregistrez le fichier avec l'extension `.pem`. Par exemple, enregistrez le fichier sous le nom `/tmp/do-not-delete-rds-custom-ssh-privatekey-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem` ou`/tmp/rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey.pem`.

AWS CLI

Pour récupérer la clé privée SSH et l'enregistrer dans un fichier .pem, vous pouvez utiliser l' AWS CLI.

1. Trouvez l'ID de ressource de base de données de votre instance de base de données RDS Custom avec `aws rds describe-db-instances`.

```
aws rds describe-db-instances \
    --query 'DBInstances[*].[DBInstanceIdentifier,DbiResourceId]' \
    --output text
```

L'exemple de sortie suivant montre l'ID de ressource de votre instance RDS Custom. Le préfixe est `db-`.

```
db-ABCDEFGHIJKLMNOPQRS0123456
```

2. Trouvez l'ID d' EC2 instance de votre instance de base de données à l'aide de`aws ec2 describe-instances`. L'exemple suivant utilise `db-ABCDEFGHIJKLMNOPQRS0123456` pour l'ID de la ressource

```
aws ec2 describe-instances \
    --filters "Name=tag:Name,Values=db-ABCDEFGHIJKLMNOPQRS0123456" \
    --output text \
    --query 'Reservations[*].Instances[*].InstanceId'
```

L'exemple de sortie suivant montre l'ID de l' EC2 instance.

```
i-abcdefghijklm01234
```

3. Pour trouver le nom de la clé, spécifiez l'ID de l' EC2 instance. L'exemple suivant décrit l' EC2 instance`i-0bdc4219e66944afa`.

```
aws ec2 describe-instances \
    --instance-ids i-0bdc4219e66944afa \
    --output text \
    --query 'Reservations[*].Instances[*].KeyName'
```

L'exemple de sortie suivant montre le nom de la clé, qui utilise le format de dénomination `do-not-delete-rds-custom-ssh-privatekey-`*`resource_id`*`-`*`uuid`* ou`rds-custom!oracle-do-not-delete-`*`resource_id`*`-`*`uuid`*`-ssh-privatekey`.

```
do-not-delete-rds-custom-ssh-privatekey-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c
rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey
```

4. Enregistrez la clé privée dans un fichier .pem nommé d'après la clé avec `aws secretsmanager`.

L'exemple suivant enregistre la clé `do-not-delete-rds-custom-ssh-privatekey-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c` dans un fichier de votre `/tmp` répertoire.

```
aws secretsmanager get-secret-value \
    --secret-id do-not-delete-rds-custom-ssh-privatekey-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c \
    --query SecretString \
    --output text >/tmp/do-not-delete-rds-custom-ssh-privatekey-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem
```

L'exemple suivant enregistre la clé `rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey` dans un fichier de votre `/tmp` répertoire.

```
aws secretsmanager get-secret-value \
    --secret-id rds-custom!oracle-do-not-delete-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey \
    --query SecretString \
    --output text >/tmp/rds-custom!oracle-do-not-delete-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey.pem
```

Étape 3 : Connectez-vous à votre EC2 instance à l'aide de l'utilitaire ssh

Votre technique de connexion varie selon que vous vous connectez à une instance de base de données privée ou à une instance publique. Pour établir une connexion privée, vous devez configurer le tunneling SSH via AWS Systems Manager.

Pour vous connecter à une EC2 instance à l'aide de l'utilitaire ssh

1. Pour les connexions privées, modifiez votre fichier de configuration SSH pour obtenir des commandes proxy vers AWS Systems Manager Session Manager. Pour les connexions publiques, passez à l'étape 2.

   Ajoutez les lignes suivantes à `~/.ssh/config`. Ces lignes fournissent des commandes SSH proxy pour les hôtes dont le nom commence par `i-` ou `mi-`.

   ```
   Host i-* mi-*
       ProxyCommand sh -c "aws ssm start-session --target %h --document-name AWS-
   StartSSHSession --parameters 'portNumber=%p'"
   ```

2. Accédez au répertoire qui contient votre fichier .pem. Avec `chmod`, définissez les autorisations sur `400`.

L'exemple suivant modifie le `/tmp` répertoire et définit les autorisations pour le fichier .pem. `do-not-delete-rds-custom-ssh-privatekey-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem`

```
cd /tmp
chmod 400 do-not-delete-rds-custom-ssh-privatekey-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem
```

L'exemple suivant modifie le `/tmp` répertoire et définit les autorisations pour le fichier .pem. `rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-privatekey.pem`

```
cd /tmp
chmod 400 rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-
privatekey.pem
```

3. Exécutez l'utilitaire ssh en spécifiant le fichier .pem et le nom DNS public (pour les connexions publiques) ou l'ID de l' EC2 instance (pour les connexions privées). Connectez-vous en tant qu'utilisateur `ec2-user`.

   L'exemple suivant permet de se connecter à une instance publique à l'aide du nom DNS`ec2-12-345-678-901.us-east-2.compute.amazonaws.com`.

   ```
   # .pem file using naming prefix do-not-delete
   ssh -i \
     "do-not-delete-rds-custom-ssh-privatekey-db-
   ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem" \
     ec2-user@ec2-12-345-678-901.us-east-2.compute.amazonaws.com

   # .pem file using naming prefix rds-custom!oracle-do-not-delete
   ssh -i \
     "rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-
   privatekey.pem" \
     ec2-user@ec2-12-345-678-901.us-east-2.compute.amazonaws.com
   ```

   L'exemple suivant établit une connexion à une instance privée à l'aide de l'ID d' EC2 instance`i-0bdc4219e66944afa`.

   ```
   # .pem file using naming prefix do-not-delete
   ```

```
ssh -i \
  "do-not-delete-rds-custom-ssh-privatekey-db-
ABCDEFGHIJKLMNOPQRS0123456-0d726c.pem" \
  ec2-user@i-0bdc4219e66944afa

# .pem file using naming prefix rds-custom!oracle-do-not-delete
ssh -i \
  "rds-custom!oracle-do-not-delete-db-ABCDEFGHIJKLMNOPQRS0123456-0d726c-ssh-
privatekey.pem" \
  ec2-user@i-0bdc4219e66944afa
```

Connexion à votre base de données RDS Custom for Oracle en tant que SYS

Une fois que vous avez créé votre instance de base de données RDS Custom, vous pouvez vous connecter à votre base de données Oracle en tant qu'utilisateur SYS, ce qui vous donne des privilèges SYSDBA. Vous disposez des options de connexion suivantes :

- Obtenez le mot de passe SYS depuis Secrets Manager et spécifiez-le dans votre client SQL.
- Utilisez l'authentification du système d'exploitation pour vous connecter à votre base de données. Dans ce cas, il n'est pas nécessaire de disposer d'un mot de passe.

Recherche du mot de passe SYS pour votre base de données RDS Custom for Oracle

Vous pouvez vous connecter à votre base de données Oracle en tant que SYS ou SYSTEM, ou en spécifiant le nom d'utilisateur principal dans un appel d'API. Le mot de passe pour SYS et SYSTEM est stocké dans Secrets Manager.

Le secret utilise le format de dénomination `do-not-delete-rds-custom-`*`resource_id`*`-`*`uuid`* ou`rds-custom!oracle-do-not-delete-`*`resource_id`*`-`*`uuid`*. Vous pouvez trouver le mot de passe à l'aide de la AWS Management Console.

console

Pour trouver le mot de passe SYS de votre base de données dans Secrets Manager

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la console RDS, suivez les étapes ci-dessous :

   a. Dans le panneau de navigation, choisissez Databases (Bases de données).

b. Choisissez le nom de votre instance de base de données RDS Custom for Oracle.

c. Choisissez Configuration.

d. Copiez la valeur sous ID de ressource. Par exemple, votre identifiant de ressource peut être db- ABC12 CDE3 FGH4 I5 JKLMNO6 PQR7.

3. Ouvrez la console Secrets Manager à l'adresse https://console.aws.amazon.com/secretsmanager/.

4. Dans la console Secrets Manager, suivez les étapes ci-dessous :

a. Dans le volet de navigation de gauche, choisissez Secrets.

b. Filtrez les secrets en fonction de l'ID de ressource que vous avez copié à l'étape 2.d.

c. Choisissez le secret qui utilise le format de dénomination do-not-delete-rds-custom ***resource_id*** - ***uuid*** ou rds-custom ! oracle-do-not-delete***resource_id***-*uuid*. *resource_id*Il s'agit de l'ID de ressource que vous avez copié à l'étape 2.d.

Par exemple, si votre identifiant de ressource est db- ABC12 CDE3 FGH4 I5 JKLMNO6 PQR7 et que votre UUID est 1234ab, votre secret s'appelle do-not-delete-rds-custom-db - I5 -1234ab ou rds-custom ! ABC12 CDE3 FGH4 JKLMNO6 PQR7 oracle-do-not-delete-db-ABC12 CDE3 FGH4 I5-1234ab JKLMNO6 PQR7.

d. Dans Valeur du secret, choisissez Récupérer la valeur du secret.

e. Dans Clé/valeur, copiez la valeur du mot de passe.

5. Installez SQL*Plus sur votre instance de base de données et connectez-vous à votre base de données en tant que SYS. Pour de plus amples informations, veuillez consulter Étape 3 : Connecter votre client SQL à une instance de base de données Oracle.

Connexion à votre base de données RDS Custom for Oracle à l'aide de l'authentification du système d'exploitation

L'utilisateur `rdsdb` du système d'exploitation est propriétaire des fichiers binaires de la base de données Oracle. Vous pouvez passer à l'utilisateur `rdsdb` et vous connecter à votre base de données RDS Custom for Oracle sans mot de passe.

1. Connectez-vous à votre instance de base de données avec AWS Systems Manager. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données RDS personnalisée à l'aide du gestionnaire de session.

2. Basculez vers l'utilisateur `rdsdb`.

```
sudo su - rdsdb
```

3. Connectez-vous à votre base de données avec l'authentification du système d'exploitation. Vous pouvez utiliser `sqlplus / as sysdba` ou l'`sql`alias.

```
$ sqlplus / as sysdba

SQL*Plus: Release 21.0.0.0.0 - Production on Wed Apr 12 20:11:08 2023
Version 21.9.0.0.0

Copyright (c) 1982, 2020, Oracle.  All rights reserved.


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.10.0.0.0
```

Vous pouvez également utiliser l'`sql`alias :

```
$ sql
```

Gestion d'une instance de base de données Amazon RDS Custom pour Oracle

Amazon RDS Custom prend en charge un sous-ensemble des tâches de gestion habituelles pour les RDS instances de base de données Amazon. Vous trouverez ci-dessous des instructions relatives aux tâches de gestion RDS Custom for Oracle prises en charge à l'aide du AWS Management Console et du AWS CLI.

Rubriques

- Utilisation de bases de données de conteneurs (CDBs) dans RDS Custom for Oracle
- Utilisation de fonctions de haute disponibilité pour RDS Custom for Oracle
- Personnalisation de votre environnement RDS personnalisé
- Modification de votre instance de base de données RDS personnalisée pour Oracle
- Modification du jeu de caractères d'une instance de base de données RDS personnalisée pour Oracle
- Définition de la LANG valeur NLS _ dans RDS Custom for Oracle
- Prise en charge de Transparent Data Encryption
- Balisage de RDS Custom pour les ressources Oracle
- Suppression d'une instance de base de données RDS personnalisée pour Oracle

Utilisation de bases de données de conteneurs (CDBs) dans RDS Custom for Oracle

Vous pouvez créer votre instance de base de données RDS personnalisée pour Oracle avec l'architecture mutualisée Oracle (`custom-oracle-ee-cdb`ou le type de `custom-oracle-se2-cdb` moteur) ou avec la CDB non-architecture traditionnelle (`custom-oracle-ee`ou le type de `custom-oracle-se2` moteur). Lorsque vous créez une base de données conteneur (CDB), elle contient une base de données enfichable (PDB) et une PDB graine. Vous pouvez en créer d'autres PDBs manuellement à l'aide d'OracleSQL.

PDBet CDB noms

Lorsque vous créez une CDB instance RDS personnalisée pour Oracle, vous spécifiez un nom pour l'initialePDB. Par défaut, votre initiale PDB est nommée`ORCL`. Vous pouvez choisir un autre nom.

Par défaut, votre nom CDB est le vôtre`RDSCDB`. Vous pouvez choisir un autre nom. Le CDB nom est également le nom de votre identifiant système Oracle (SID), qui identifie de manière unique la

mémoire et les processus qui gèrent votreCDB. Pour plus d'informations sur OracleSID, consultez Oracle System Identifier (SID) dans Oracle Database Concepts.

Vous ne pouvez pas renommer un fichier existant à PDBs l'aide d'Amazon RDSAPIs. Vous ne pouvez pas non plus renommer le à l'CDBaide de la `modify-db-instance` commande.

PDBgestion

Dans le modèle de responsabilité partagée RDS Custom for Oracle, vous êtes responsable de la gestion PDBs et de la création de toute responsabilité supplémentairePDBs. RDSLa personnalisation ne limite pas le nombre dePDBs. Vous pouvez créer, modifier et supprimer manuellement en vous PDBs connectant à la CDB racine et en exécutant une SQL instruction. Créez PDBs sur un volume de EBS données Amazon pour empêcher l'instance de base de données de sortir du périmètre de support.

Pour modifier votre CDBs orPDBs, procédez comme suit :

1. Suspendez l'automatisation pour éviter toute interférence avec les actions RDS personnalisées.
2. Modifiez votre CDB orPDBs.
3. Sauvegardez toutes les modificationsPDBs.
4. Reprendre l'automatisation RDS personnalisée.

Restauration automatique de la CDB racine

RDSLa fonction Custom maintient la CDB racine ouverte de la même manière qu'une racine non CDB ouverte. Si l'état de la CDB racine change, l'automatisation de la surveillance et de la restauration tente de rétablir la CDB racine dans l'état souhaité. Vous recevez des notifications d'RDSévénements lorsque le root CDB est arrêté (`RDS-EVENT-0004`) ou redémarré (`RDS-EVENT-0006`), comme dans le cas d'une architecture hors CDB architecture. RDSCustom tente d'ouvrir tout PDBs en `READ WRITE` mode au démarrage de l'instance de base de données. Si certains ne PDBs peuvent pas être ouverts, RDS Custom publie l'événement suivant :`tenant database shutdown`.

Utilisation de fonctions de haute disponibilité pour RDS Custom for Oracle

RDS Custom for Oracle fournit une haute disponibilité intégrée grâce à des déploiements multi-AZ. Vous pouvez également utiliser Oracle Data Guard en tant qu'option gérée par le client en fonction de vos cas d'utilisation.

Déploiements multi-AZ (entièrement gérés)

Avec les déploiements multi-AZ pour RDS Custom for Oracle, Amazon RDS approvisionne et gère automatiquement une réplique de secours synchrone dans une autre zone de disponibilité (AZ). L'instance de base de données principale est répliquée de manière synchrone sur une réplique AZs de secours pour la redondance des données. Le déploiement multi-AZ est pris en charge à la fois dans l'édition Enterprise et dans l'édition Standard 2. Consultez the section called “Gestion d'un déploiement multi-AZ pour RDS Custom for Oracle” pour plus de détails.

Oracle Data Guard (géré par le client)

Vous pouvez également obtenir une haute disponibilité en configurant manuellement Oracle Data Guard pour répliquer les données entre les instances de base de données RDS Custom for Oracle. L'instance de base de données primaire synchronise automatiquement les données avec les instances de secours. Oracle Data Guard n'est pris en charge que dans l'édition Enterprise.

Vous pouvez configurer votre environnement à haute disponibilité des façons suivantes :

- Configurez les instances de secours AZs de manière différente pour résister aux défaillances AZ.
- Définition de vos bases de données de secours en mode monté ou en mode lecture seule. Le mode lecture seule nécessite une licence Oracle Active Data Guard.
- Basculement de la base de données primaire vers une base de données de secours sans perte de données.
- Migrez les données en configurant la haute disponibilité pour votre instance sur site, puis en basculant ou en basculant vers la base de données de secours RDS Custom for Oracle.

Pour savoir comment configurer Oracle Data Guard pour une haute disponibilité, consultez le AWS blog Créer une haute disponibilité pour RDS Custom for Oracle à l'aide de répliques de lecture. Vous pouvez effectuer les tâches suivantes :

- Utiliser un tunnel VPN pour chiffrer les données en transit pour vos instances à haute disponibilité. Le chiffrement en transit n'est pas configuré automatiquement par RDS Custom pour Oracle.
- Configurer Oracle Fast-Failover Observer (FSFO) pour surveiller vos instances à haute disponibilité.
- Autoriser l'observateur à effectuer un basculement automatique lorsque les conditions nécessaires sont remplies.

Personnalisation de votre environnement RDS personnalisé

RDSCustom for Oracle inclut des fonctionnalités intégrées qui vous permettent de personnaliser l'environnement de votre instance de base de données sans interrompre l'automatisation. Par exemple, vous pouvez l'utiliser RDS APIs pour personnaliser votre environnement comme suit :

- Créez et restaurez des instantanés de base de données pour créer un environnement de clonage.
- Créez des réplicas en lecture.
- Modifiez les paramètres de stockage.
- Modifiez le CEV pour appliquer les mises à jour de version

Pour certaines personnalisations, telles que la modification du jeu de caractères, vous ne pouvez pas utiliser le RDSAPIs. Dans ces cas, vous devez modifier l'environnement manuellement en accédant à votre EC2 instance Amazon en tant qu'utilisateur root ou en vous connectant à votre base de données Oracle en tant que`SYSDBA`.

Pour personnaliser votre instance manuellement, vous devez suspendre et reprendre l'automatisation RDS personnalisée. Cette pause garantit que vos personnalisations n'interfèrent pas avec l'automatisation RDS personnalisée. De cette façon, vous évitez de briser le périmètre de support, qui place l'instance dans l'état `unsupported-configuration` jusqu'à ce que vous résolviez les problèmes sous-jacents. La pause et la reprise sont les seules tâches d'automatisation prises en charge lorsque vous modifiez une instance de base de données RDS personnalisée pour Oracle.

Étapes générales pour personnaliser votre environnement RDS personnalisé

Pour personnaliser votre instance de base de données RDS personnalisée, procédez comme suit :

1. Suspendre l'automatisation RDS personnalisée pendant une période spécifiée à l'aide de la console ouCLI.
2. Identifiez votre EC2 instance Amazon sous-jacente.
3. Connectez-vous à votre EC2 instance Amazon sous-jacente à l'aide de SSH clés ou AWS Systems Manager.
4. Vérifiez vos paramètres de configuration actuels au niveau de la base de données ou de la couche du système d'exploitation.

   Vous pouvez valider vos modifications en comparant la configuration initiale à la configuration modifiée. Selon le type de personnalisation, utilisez les outils du système d'exploitation ou les requêtes de base de données.

5. Personnalisez votre instance de base de données RDS personnalisée pour Oracle selon vos besoins.
6. Redémarrez votre instance ou votre base de données, si nécessaire.

> **Note**
>
> Dans un Oracle sur siteCDB, vous pouvez conserver un mode ouvert spécifié à PDBs l'aide d'une commande intégrée ou après un déclenchement de démarrage. Ce mécanisme atteint PDBs un état spécifié lors du CDB redémarrage. Lorsque vous ouvrez votreCDB, RDS Custom Automation supprime tous les états conservés spécifiés par l'utilisateur et tente de tout ouvrir. PDBs Si RDS Custom ne peut pas tout ouvrirPDBs, l'événement suivant est émis :`The following PDBs failed to open: list-of-PDBs`.

7. Vérifiez vos nouveaux paramètres de configuration en les comparant aux paramètres précédents.
8. Reprenez l'automatisation RDS personnalisée de l'une des manières suivantes :
   - Relancez manuellement l'automatisation.
   - Attendez la fin de la période de pause. Dans ce cas, RDS Custom reprend automatiquement la surveillance et la restauration de l'instance.
9. Vérifiez le cadre d'automatisation RDS personnalisé

   Si vous avez correctement suivi les étapes précédentes, RDS Custom lance une sauvegarde automatique. Le statut de l'instance dans la console indique Disponible.

Pour connaître les meilleures pratiques et step-by-step les instructions, consultez les articles de AWS blog Apporter des modifications de configuration à une instance Amazon RDS Custom for Oracle : Part 1 et Recreate an Amazon RDS Custom for Oracle database : Part 2.

Suspension et reprise de votre RDS instance de base de données personnalisée

Vous pouvez suspendre et reprendre l'automatisation de votre instance de base de données à l'aide de la console ouCLI.

Console

Pour suspendre ou reprendre l'automatisation RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée que vous souhaitez modifier.
3. Sélectionnez Modifier. La page Modifier l'instance de base de données s'affiche.
4. Pour le mode d'automatisation RDS personnalisé, choisissez l'une des options suivantes :
   - La pause interrompt la surveillance et la restauration de l'instance pour l'instance de base de données RDS personnalisée. Saisissez la durée de pause souhaitée (en minutes) pour Automation mode duration (Durée du mode d'automatisation). La valeur minimale est de 60 minutes (par défaut). La valeur maximale est de 1 440 minutes.
   - L'option Full automation (Automatisation complète) relance l'automatisation.
5. Sélectionnez Continuer pour consulter le récapitulatif des modifications.

   Un message indique que RDS Custom appliquera les modifications immédiatement.
6. Si elles sont correctes, sélectionnez Modifier l'instance de base de données. Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

   Les détails de la modification apparaissent sur la RDS console. Si vous avez suspendu l'automatisation, le statut de votre instance de base de données RDS personnalisée indique que l'automatisation est suspendue.
7. (Facultatif) Dans le volet de navigation, choisissez Databases, puis votre instance de base de données RDS personnalisée.

   Dans le volet Résumé, le mode d'automatisation RDS personnalisé indique l'état de l'automatisation. Si l'automatisation est suspendue, la valeur est Suspendue. L'automatisation reprend en ***num*** minutes.

AWS CLI

Pour suspendre ou reprendre l'automatisation RDS personnalisée, utilisez la `modify-db-instance` AWS CLI commande. Identifiez l'instance de base de données à l'aide du paramètre requis `--db-instance-identifier`. Contrôlez le mode d'automatisation avec les paramètres suivants :

- `--automation-mode` spécifie l'état de pause de l'instance de base de données. Les valeurs valides sont `all-paused`, qui suspend l'automatisation, et `full`, qui relance l'opération.
- `--resume-full-automation-mode-minutes` spécifie la durée de la pause. La valeur par défaut est de 60 minutes.

> **Note**
>
> Que vous spécifiiez `--no-apply-immediately` ou non`--apply-immediately`, RDS Custom applique les modifications de manière asynchrone dès que possible.

Dans la réponse à la commande, `ResumeFullAutomationModeTime` indique l'heure de reprise sous forme d'UTChorodatage. Lorsque le mode d'automatisation est `all-paused`, vous pouvez utiliser `modify-db-instance` pour relancer le mode d'automatisation ou prolonger la période de pause. Aucune autre option `modify-db-instance` n'est prise en charge.

L'exemple suivant suspend pendant 90 minutes l'automatisation de l'instance `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode all-paused \
    --resume-full-automation-mode-minutes 90
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode all-paused ^
    --resume-full-automation-mode-minutes 90
```

L'exemple suivant prolonge la durée de pause de 30 minutes. Les 30 minutes sont ajoutées à la durée d'origine affichée dans `ResumeFullAutomationModeTime`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode all-paused \
    --resume-full-automation-mode-minutes 30
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode all-paused ^
    --resume-full-automation-mode-minutes 30
```

L'exemple suivant reprend l'automatisation complète pour `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode full \
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode full
```

Dans l'exemple de sortie partielle ci-dessous, la valeur `AutomationMode` en attente est `full`.

```
{
    "DBInstance": {
        "PubliclyAccessible": true,
        "MasterUsername": "admin",
        "MonitoringInterval": 0,
        "LicenseModel": "bring-your-own-license",
        "VpcSecurityGroups": [
            {
```

```
            "Status": "active",
            "VpcSecurityGroupId": "0123456789abcdefg"
        }
    ],
    "InstanceCreateTime": "2020-11-07T19:50:06.193Z",
    "CopyTagsToSnapshot": false,
    "OptionGroupMemberships": [
        {
            "Status": "in-sync",
            "OptionGroupName": "default:custom-oracle-ee-19"
        }
    ],
    "PendingModifiedValues": {
        "AutomationMode": "full"
    },
    "Engine": "custom-oracle-ee",
    "MultiAZ": false,
    "DBSecurityGroups": [],
    "DBParameterGroups": [
        {
            "DBParameterGroupName": "default.custom-oracle-ee-19",
            "ParameterApplyStatus": "in-sync"
        }
    ],
    ...
    "ReadReplicaDBInstanceIdentifiers": [],
    "AllocatedStorage": 250,
    "DBInstanceArn": "arn:aws:rds:us-west-2:012345678912:db:my-custom-instance",
    "BackupRetentionPeriod": 3,
    "DBName": "ORCL",
    "PreferredMaintenanceWindow": "fri:10:56-fri:11:26",
    "Endpoint": {
        "HostedZoneId": "ABCDEFGHIJKLMNO",
        "Port": 8200,
        "Address": "my-custom-instance.abcdefghijk.us-west-2.rds.amazonaws.com"
    },
    "DBInstanceStatus": "automation-paused",
    "IAMDatabaseAuthenticationEnabled": false,
    "AutomationMode": "all-paused",
    "EngineVersion": "19.my_cev1",
    "DeletionProtection": false,
    "AvailabilityZone": "us-west-2a",
    "DomainMemberships": [],
    "StorageType": "gp2",
```

```
        "DbiResourceId": "db-ABCDEFGHIJKLMNOPQRSTUVW",
        "ResumeFullAutomationModeTime": "2020-11-07T20:56:50.565Z",
        "KmsKeyId": "arn:aws:kms:us-west-2:012345678912:key/
aa111a11-111a-11a1-1a11-1111a11a1a1a",
        "StorageEncrypted": false,
        "AssociatedRoles": [],
        "DBInstanceClass": "db.m5.xlarge",
        "DbInstancePort": 0,
        "DBInstanceIdentifier": "my-custom-instance",
        "TagList": []
    }
```

Modification de votre instance de base de données RDS personnalisée pour Oracle

La modification d'une instance de base de données RDS personnalisée pour Oracle est similaire à la modification d'une RDS instance de base de données Amazon. Vous pouvez modifier des paramètres tels que les suivants :

- Classe d'instances de base de données
- Allocation et type de stockage
- Période de rétention des sauvegardes
- Deletion protection (Protection contre la suppression)
- Groupe d'options
- CEV(voirMise à niveau d'une instance de base de données RDS personnalisée pour Oracle)
- Port

Rubriques

- Exigences et limites liées à la modification du stockage de votre instance de base de données
- Exigences et limites liées à la modification de votre classe d'instances de base de données
- Comment RDS Custom crée votre instance de base de données lorsque vous modifiez la classe d'instance
- Modification de votre instance de base de données RDS personnalisée pour Oracle

Exigences et limites liées à la modification du stockage de votre instance de base de données

Tenez compte des exigences et limites suivantes lorsque vous modifiez le stockage d'une instance de base de données RDS personnalisée pour Oracle :

- Le stockage minimum alloué pour RDS Custom for Oracle est de 40 GiB, et le maximum est de 64 TiB.
- Comme pour AmazonRDS, vous ne pouvez pas diminuer le stockage alloué. Il s'agit d'une limitation des EBS volumes Amazon.
- Le dimensionnement automatique du stockage n'est pas pris en charge pour les instances de base de données RDS personnalisées.
- Tous les volumes de stockage que vous attachez manuellement à votre instance de base de données RDS personnalisée se trouvent en dehors du périmètre de support.

  Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.
- Le EBS stockage magnétique (standard) sur Amazon n'est pas pris en charge pour le RDS mode personnalisé. Vous ne pouvez choisir que les types de stockage io1, gp2 ou gp3SSD.

Pour plus d'informations sur le EBS stockage Amazon, consultezStockage d'instance de base de données Amazon RDS. Pour obtenir des informations générales sur la modification du stockage, consultez Utilisation du stockage pour les RDS instances de base de données Amazon.

Exigences et limites liées à la modification de votre classe d'instances de base de données

Tenez compte des exigences et limites suivantes lorsque vous modifiez la classe d'instance d'une instance de base de données RDS personnalisée pour Oracle :

- Votre instance de base de données doit être dans l'état `available`.
- Votre instance de base de données doit disposer d'un minimum de 100 Mio d'espace disponible sur le volume racine, le volume de données et le volume binaire.
- Vous ne pouvez attribuer qu'une seule adresse IP élastique (EIP) à votre instance de base de données RDS Custom for Oracle lorsque vous utilisez l'interface réseau élastique (ENI) par défaut. Si vous en attachez plusieurs ENIs à votre instance de base de données, l'opération de modification échoue.
- Toutes les balises RDS Custom for Oracle doivent être présentes.
- Si vous utilisez RDS Custom pour la réplication Oracle, tenez compte des exigences et limites suivantes :
  - Pour les instances de base de données principales et les réplicas en lecture, vous pouvez modifier la classe d'instances d'une seule instance de base de données à la fois.
  - Si votre instance de base de données RDS personnalisée pour Oracle possède une base de données principale ou de réplication sur site, veillez à mettre à jour manuellement les adresses

IP privées sur l'instance de base de données locale une fois la modification terminée. Cette action est nécessaire pour préserver les DataGuard fonctionnalités d'Oracle. RDSCustom for Oracle publie un événement lorsque la modification aboutit.

- Vous ne pouvez pas modifier votre classe d'instance de base de données RDS personnalisée pour Oracle lorsque les instances de base de données principales ou de répliques en lecture sont configurées FSFO (Fast-Start Failover).

Comment RDS Custom crée votre instance de base de données lorsque vous modifiez la classe d'instance

Lorsque vous modifiez votre classe d'instance, RDS Custom crée votre instance de base de données comme suit :

- Crée l'EC2instance Amazon.
- Crée le volume racine à partir du dernier instantané de base de données. RDSCustom for Oracle ne conserve pas les informations ajoutées au volume racine après le dernier instantané de base de données.
- Crée des CloudWatch alarmes Amazon.
- Crée une paire de EC2 SSH clés Amazon si vous avez supprimé la paire de clés d'origine. Dans le cas contraire, RDS Custom for Oracle conserve la paire de clés d'origine.
- Crée de nouvelles ressources à l'aide des identifications attachées à votre instance de base de données lorsque vous initiez la modification. RDSCustom ne transfère pas les balises vers les nouvelles ressources lorsqu'elles sont associées directement aux ressources sous-jacentes.
- Transfère les volumes binaires et de données avec les modifications les plus récentes vers la nouvelle instance de base de données.
- Transfère l'adresse IP élastique (EIP). Si l'instance de base de données est accessible au public, RDS Custom attache temporairement une adresse IP publique à la nouvelle instance de base de données avant de transférer leEIP. Si l'instance de base de données n'est pas accessible au public, RDS Custom ne crée pas d'adresses IP publiques.

Modification de votre instance de base de données RDS personnalisée pour Oracle

Vous pouvez modifier la classe ou le stockage de l'instance de base de données à l'aide de la console AWS CLI, ou RDSAPI.

Console

Pour modifier une instance de base de données RDS personnalisée pour Oracle

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de bases de données que vous souhaitez modifier.
4. Sélectionnez Modify.
5. (Facultatif) Dans Configuration de l'instance, choisissez une valeur pour la classe d'instance de base de données. Pour connaître les classes prises en charge, consultez Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle.
6. (Facultatif) Dans Stockage, apportez les modifications suivantes selon vos besoins :

   a. Saisissez une nouvelle valeur pour Stockage alloué. Elle doit être supérieure à la valeur actuelle et être comprise entre 40 Gio et 64 Tio.
   b. Modifiez la valeur du type de stockage en General Purpose SSD (gp2), General Purpose SSD (gp3) ou Provisioned IOPS (io1).
   c. Si vous utilisez Provisioned IOPS (io1) ou General Purpose SSD (gp3), vous pouvez modifier la valeur Provisioned. IOPS
7. (Facultatif) Dans Configuration supplémentaire, apportez les modifications suivantes selon vos besoins :

   - Pour Groupe d'options, choisissez un nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Utilisation de groupes d'options dans RDS Custom for Oracle.
8. Choisissez Continuer.
9. Sélectionnez Appliquer immédiatement ou Appliquer au cours de la prochaine fenêtre de maintenance planifiée.
10. Choisissez Modifier l'instance de base de données.

AWS CLI

Pour modifier le stockage d'une instance de base de données RDS personnalisée pour Oracle, utilisez la modify-db-instance AWS CLI commande. Définissez les paramètres suivants selon les besoins :

- `--db-instance-class` : une nouvelle classe d'instances. Pour connaître les classes prises en charge, consultez Prise en charge de la classe d'instance de base de données pour RDS Custom for Oracle.
- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets. Elle doit être supérieure à la valeur actuelle et être comprise entre 40 et 65 536 Gio.
- `--storage-type` : type de stockage : gp2, gp3 ou io1.
- `--iops`— Provisionné IOPS pour l'instance de base de données, si vous utilisez les types de stockage io1 ou gp3.
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer immédiatement les modifications apportées au stockage.

  Vous pouvez également utiliser `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

L'exemple suivant remplace la classe d'instance de base de données par my-cfo-instance db.m5.16xlarge. La commande modifie également la taille de stockage à 1 TiB, le type de stockage à io1, Provisioned à 3000 et le groupe d'options IOPS à cfo-ee-19-mt.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-cfo-instance \
    --db-instance-class db.m5.16xlarge \
    --storage-type io1 \
    --iops 3000 \
    --allocated-storage 1024 \
    --option-group cfo-ee-19-mt \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-cfo-instance ^
    --db-instance-class db.m5.16xlarge ^
    --storage-type io1 ^
    --iops 3000 ^
```

```
    --allocated-storage 1024 ^
    --option-group cfo-ee-19-mt ^
    --apply-immediately
```

Modification du jeu de caractères d'une instance de base de données RDS personnalisée pour Oracle

RDSPersonnaliser pour Oracle utilise par défaut le jeu de caractèresUS7ASCII. Vous souhaiterez peut-être spécifier différents jeux de caractères pour satisfaire aux exigences relatives à la langue ou aux caractères multioctets. Lorsque vous utilisez RDS Custom for Oracle, vous pouvez suspendre l'automatisation, puis modifier manuellement le jeu de caractères de votre base de données.

La modification du jeu de caractères d'une instance de base de données RDS personnalisée pour Oracle répond aux exigences suivantes :

- Vous ne pouvez modifier le caractère que sur une instance RDS personnalisée nouvellement provisionnée qui possède une base de données vide ou de départ ne contenant aucune donnée d'application. Pour tous les autres scénarios, modifiez le jeu de caractères à l'aide de DMU (Assistant de migration de base de données pour Unicode).
- Vous ne pouvez passer qu'à un jeu de caractères RDS compatible avec Oracle. Pour de plus amples informations, veuillez consulter Jeux de caractères de base de données pris en charge.

Pour modifier le jeu de caractères d'une instance de base de données RDS personnalisée pour Oracle

1. Suspendez l'automatisation RDS personnalisée. Pour de plus amples informations, veuillez consulter Suspension et reprise de votre RDS instance de base de données personnalisée.
2. Connectez-vous à votre base de données en tant qu'utilisateur avec privilèges SYSDBA.
3. Redémarrez la base de données en mode restreint, modifiez le jeu de caractères, puis redémarrez la base de données en mode normal.

   Exécutez le script suivant dans votre SQL client :

   ```
   SHUTDOWN IMMEDIATE;
   STARTUP RESTRICT;
   ALTER DATABASE CHARACTER SET INTERNAL_CONVERT AL32UTF8;
   SHUTDOWN IMMEDIATE;
   STARTUP;
   ```

```
SELECT VALUE FROM NLS_DATABASE_PARAMETERS WHERE PARAMETER = 'NLS_CHARACTERSET';
```

Vérifiez que la sortie affiche le jeu de caractères approprié :

```
VALUE
--------
AL32UTF8
```

4. Reprendre l'automatisation RDS personnalisée. Pour de plus amples informations, veuillez consulter Suspension et reprise de votre RDS instance de base de données personnalisée.

Définition de la LANG valeur NLS _ dans RDS Custom for Oracle

Un paramètre régional est un ensemble d'informations répondant aux exigences linguistiques et culturelles qui correspondent à une langue et à un pays donnés. Pour spécifier le comportement local du logiciel Oracle, définissez la variable d'environnement NLS_LANG sur votre hôte client. Cette variable définit la langue, le territoire et le jeu de caractères utilisés par l'application cliente dans une session de base de données.

Pour RDS Custom for Oracle, vous ne pouvez définir que la langue de la NLS_LANG variable : le territoire et le personnage utilisent les valeurs par défaut. La langue est utilisée pour les messages de la base de données Oracle, les classements, les noms des jours et des mois. Chaque langue prise en charge possède un nom unique, par exemple américain, français ou allemand. Si aucune langue n'est spécifiée, la valeur par défaut est americain.

Après avoir créé votre base de données RDS personnalisée pour Oracle, vous pouvez définir une langue autre que l'anglais NLS_LANG sur votre hôte client. Pour consulter la liste des langues prises en charge par Oracle Database, connectez-vous à votre base de données RDS Custom for Oracle et exécutez la requête suivante :

```
SELECT VALUE FROM V$NLS_VALID_VALUES WHERE PARAMETER='LANGUAGE' ORDER BY VALUE;
```

Vous pouvez définir NLS_LANG sur la ligne de commande de l'hôte. L'exemple suivant définit l'allemand comme langue pour votre application cliente à l'aide du shell Z sous Linux.

```
export NLS_LANG=German
```

Votre application lit la valeur NLS_LANG au démarrage, puis la communique à la base de données lorsqu'elle se connecte.

Pour plus d'informations, voir Choisir un paramètre régional avec la variable d'LANGenvironnement NLS _ dans le manuel Oracle Database Globalization Support Guide.

Prise en charge de Transparent Data Encryption

RDSCustom prend en charge le chiffrement transparent des données (TDE) pour RDS Custom pour les instances de base de données Oracle.

Cependant, vous ne pouvez pas activer TDE l'utilisation d'une option dans un groupe d'options personnalisé comme c'est le cas RDS pour Oracle. Vous l'allumez TDE manuellement. Pour de plus amples informations sur l'utilisation d'Oracle TDE, consultez la section Securing Stored Data Using Transparent Data Encryption (Sécurisation des données stockées à l'aide de TDE) dans la documentation Oracle.

Balisage de RDS Custom pour les ressources Oracle

Vous pouvez étiqueter les ressources RDS Custom comme s'il s'agissait de ressources Amazon RDS. Il existe toutefois quelques différences importantes :

- Ne créez pas et ne modifiez pas l'étiquette `AWSRDSCustom` requise pour l'automatisation de RDS Custom. En cas de non-respect de cette consigne, l'automatisation risque d'être interrompue.
- La `Name` balise est ajoutée aux ressources personnalisées RDS avec la valeur de `do-not-delete-rds-custom` préfixe ou. `rds-custom!oracle-do-not-delete` Toute valeur de clé transmise par le client est remplacée.
- Les étiquettes ajoutées aux instances de base de données RDS Custom lors de la création sont propagées à toutes les autres ressources RDS Custom associées.
- Les étiquettes ne sont pas propagées lorsque vous les ajoutez à des ressources RDS Custom après la création d'une instance de base de données.

Pour obtenir des informations générales sur le balisage des ressources, consultez Marquage d' Amazon RDS.

Suppression d'une instance de base de données RDS personnalisée pour Oracle

Pour supprimer une instance de base de données RDS personnalisée, procédez comme suit :

- Indiquez le nom de l'instance de base de données.
- Désactivez l'option permettant de créer un instantané de base de données final de l'instance de base de données.

- Activez ou désactivez l'option de rétention des sauvegardes automatisées.

Vous pouvez supprimer une instance de base de données RDS personnalisée à l'aide de la console ou duCLI. Le temps nécessaire à la suppression d'une instance de base de données peut varier en fonction de la période de rétention des sauvegardes (c'est-à-dire du nombre de sauvegardes à supprimer) et de la quantité de données supprimées.

Console

Pour supprimer une instance de base de données RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée que vous souhaitez supprimer. RDSLes instances de base de données personnalisées affichent le rôle Instance (RDSCustom).
3. Pour Actions, choisissez Supprimer.
4. Pour conserver les sauvegardes automatisées, choisissez Conserver les sauvegardes automatiques.
5. Saisissez **`delete me`** dans la zone.
6. Sélectionnez Delete (Supprimer).

AWS CLI

Vous supprimez une instance de base de données RDS personnalisée à l'aide de la delete-db-instance AWS CLI commande. Identifiez l'instance de base de données à l'aide du paramètre requis `--db-instance-identifier`. Les paramètres restants sont les mêmes que pour une RDS instance de base de données Amazon, avec les exceptions suivantes :

- `--skip-final-snapshot` est obligatoire.
- `--no-skip-final-snapshot` n'est pas pris en charge.
- `--final-db-snapshot-identifier` n'est pas pris en charge.

L'exemple suivant supprime l'instance de base de données RDS personnalisée nommée et conserve `my-custom-instance` les sauvegardes automatisées.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance \
    --db-instance-identifier my-custom-instance \
    --skip-final-snapshot \
    --no-delete-automated-backups
```

Dans Windows:

```
aws rds delete-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --skip-final-snapshot ^
    --no-delete-automated-backups
```

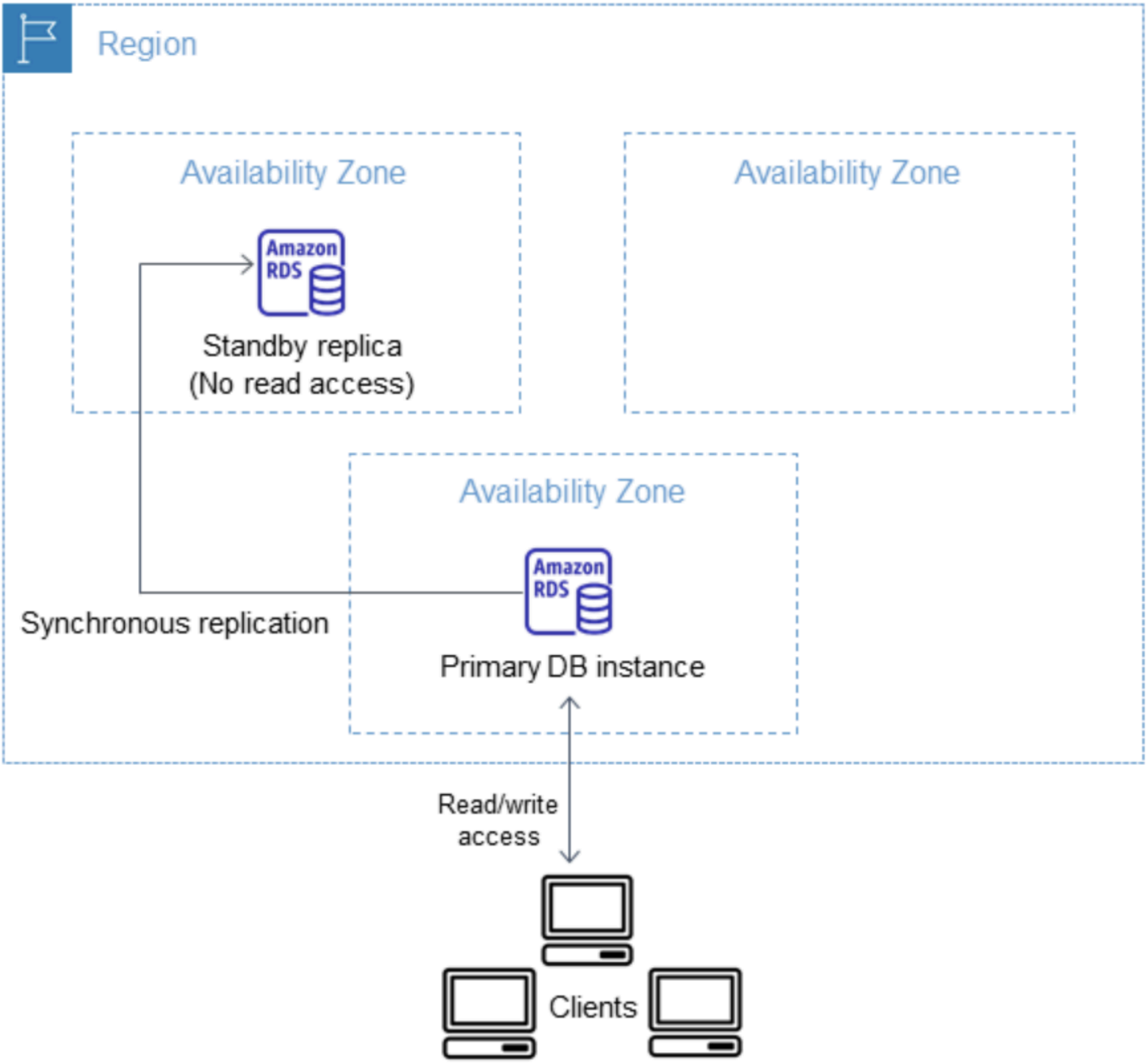
Gestion d'un déploiement multi-AZ pour RDS Custom for Oracle

Dans le cadre d'un déploiement d'instance de base de données multi-AZ pour RDS Custom for Oracle, Amazon RDS approvisionne et gère automatiquement une réplique de secours synchrone dans une autre zone de disponibilité (AZ). L'instance de base de données primaire est répliquée de manière synchrone dans les zones de disponibilité sur un réplica de secours afin d'assurer une redondance des données.

> **Important**
>
> Un déploiement multi-AZ pour RDS Custom for Oracle est différent d'un déploiement multi-AZ pour RDS for Oracle. Contrairement à Multi-AZ pour RDS pour Oracle, vous devez définir les prérequis pour RDS Custom pour Oracle avant de créer votre instance de base de données Multi-AZ, car RDS Custom s'exécute dans votre propre compte, qui nécessite des autorisations. Vous devez également créer l'instance Multi-AZ sur le CEV avec la dernière AMI fournie par le service qui prend en charge les déploiements Multi-AZ.
> Si vous ne remplissez pas les conditions préalables, votre instance de base de données multi-AZ risque de ne pas s'exécuter ou de revenir automatiquement à une instance de base de données mono-AZ. Pour plus d'informations sur les conditions préalables, consultez the section called “Prérequis”.

L'exécution d'une instance de base de données en haute disponibilité peut améliorer la disponibilité pendant la maintenance planifiée du système. En cas de maintenance planifiée de la base de données ou d'interruption de service imprévue, Amazon RDS bascule automatiquement vers l'instance de base de données de up-to-date secours. Cette fonctionnalité permet aux opérations de la base de données de reprendre rapidement sans intervention manuelle. Les instances principale et de secours utilisent le même point de terminaison, dont l'adresse réseau physique est transférée à la réplique de secours dans le cadre du processus de basculement. Vous n'avez pas à reconfigurer votre application lorsqu'un basculement se produit.

Le schéma suivant montre l'architecture multi-AZ de RDS Custom pour Oracle :



Vous pouvez créer un déploiement RDS Custom pour Oracle Multi-AZ en spécifiant Multi-AZ lors de la création d'une instance de base de données RDS Custom. Les instances de base de données RDS Custom pour Oracle créées avant le 30 juin 2025 ne peuvent pas être converties en déploiements multi-AZ en modifiant l'instance de base de données et en spécifiant l'option Multi-AZ. Vous devez créer une nouvelle instance de base de données RDS Custom pour Oracle en spécifiant l'option Multi-AZ. Les nouvelles instances de base de données créées à l'aide de l'AMI fournie par le service disponible après le 30 juin 2025 prennent en charge l'option Multi-AZ. Ensuite,

vous devez migrer votre base de données RDS Custom for Oracle existante vers la nouvelle instance de base de données. Pour en savoir plus, consultez the section called “Migration pour CEVs avant le 30.06.2025”.

La console Amazon RDS indique la zone de disponibilité de la réplique de secours (l'AZ secondaire). Vous pouvez également utiliser la commande d'interface de ligne de commande `describe-db-instances` ou l'opération d'API `DescribeDBInstances` pour rechercher la zone de disponibilité secondaire.

Les instances de base de données RDS Custom pour Oracle avec déploiement multi-AZ peuvent avoir une latence d'écriture et de validation accrue par rapport à un déploiement mono-AZ. Cette augmentation peut se produire en raison de la réplication de données synchrone entre les instances de base de données. La latence peut changer si votre déploiement bascule vers la réplique de secours, même si elle AWS est conçue avec une connectivité réseau à faible latence entre les zones de disponibilité.

Rubriques

- Disponibilité des régions et des versions
- Limites d'un déploiement multi-AZ avec RDS Custom pour Oracle
- Création d'une instance de base de données RDS Custom pour Oracle avec un déploiement multi-AZ
- Conditions préalables pour un déploiement multi-AZ avec RDS Custom pour Oracle
- Modification d'un déploiement RDS Custom for Oracle Single-AZ vers un déploiement multi-AZ
- Modification d'un déploiement RDS Custom for Oracle Multi-AZ vers un déploiement mono-AZ
- Processus de basculement pour un déploiement RDS Custom pour Oracle Multi-AZ
- Déploiement de RDS Custom pour Oracle avec AWS CloudFormation

Disponibilité des régions et des versions

Les déploiements multi-AZ pour RDS Custom for Oracle sont pris en charge pour les éditions de base de données Oracle suivantes :

- Oracle Database 12c version 1 (12.1)
- Oracle Database 12c version 2 (12.2)
- Oracle Database 19c

- Enterprise Edition (EE)
- Édition standard (2SE2)

> **Note**
>
> Les déploiements multi-AZ pour RDS Custom for Oracle ne sont pas pris en charge sur Oracle Database 18c.

Les déploiements multi-AZ pour RDS Custom for Oracle sont disponibles dans toutes les régions où RDS Custom for Oracle est disponible. Pour plus d'informations sur la disponibilité régionale des déploiements multi-AZ pour RDS Custom for Oracle, consultez. the section called “Régions et moteurs de base de données pris en charge pour RDS Custom pour Oracle”

Limites d'un déploiement multi-AZ avec RDS Custom pour Oracle

Les déploiements multi-AZ avec RDS Custom pour Oracle présentent les limites suivantes :

- Les déploiements multi-AZ entre régions ne sont pas pris en charge.
- Vous ne pouvez pas configurer l'instance de base de données de secours pour accepter l'activité de lecture de la base de données.
- Lorsque vous utilisez une version de moteur personnalisée (CEV) avec un déploiement multi-AZ, votre instance de base de données de secours utilisera également le même CEV. L'instance de base de données de secours ne peut pas utiliser un autre CEV.
- Vous ne pouvez pas créer de réplique en lecture sur une instance de déploiement multi-AZ et vous ne pouvez pas non plus modifier l'instance principale de la réplique en lecture pour un déploiement multi-AZ.

Création d'une instance de base de données RDS Custom pour Oracle avec un déploiement multi-AZ

Pour créer une instance de base de données RDS Custom pour Oracle avec un déploiement multi-AZ, suivez les étapes décrites the section called “Configuration de votre environnement RDS Custom for Oracle” pour configurer votre environnement avec les prérequis.

> **Important**
>
> Pour simplifier la configuration, nous vous recommandons d'utiliser le dernier fichier AWS CloudFormation modèle fourni dans les instructions de configuration réseau. Pour de plus amples informations, veuillez consulter the section called “Déployez avec AWS CloudFormation”.

Vous pouvez créer une instance RDS Custom pour Oracle avec un déploiement multi-AZ en choisissant l'option Multi-AZ lors de la création de l'instance de base de données dans la console Amazon RDS. Vous pouvez également spécifier le `--multi-az` paramètre dans la `create-db-instance` commande Amazon RDS du AWS CLI.

Conditions préalables pour un déploiement multi-AZ avec RDS Custom pour Oracle

> **Note**
>
> Si vous avez créé votre instance RDS Custom pour Oracle avant le 30 juin 2025, vous ne pouvez pas la modifier pour passer d'un déploiement mono-AZ à un déploiement multi-AZ. Votre version de moteur personnalisé (CEV) sous-jacente a été créée à partir d'une ancienne AMI fournie par un service qui ne prend pas en charge le déploiement multi-AZ.
> Vous devez migrer votre base de données vers une nouvelle instance à l'aide d'un CEV que vous créez avec une AMI fournie par un service après le 30 juin 2025.
> Consultez the section called “Migration pour CEVs avant le 30.06.2025”.

RDS Custom for Oracle nécessite des prérequis spécifiques lors de la conversion d'un déploiement mono-AZ vers un déploiement multi-AZ. Des prérequis incomplets entraîneront un échec de l'installation Multi-AZ. Utilisez la configuration manuelle ou le dernier fichier AWS CloudFormation modèle fourni dans les instructions de configuration réseau. Pour de plus amples informations, veuillez consulter the section called “Étape 3 : Extraire les CloudFormation modèles pour RDS Custom pour Oracle”.

Pour remplir les prérequis manuellement, suivez les étapes décrites dans la section the section called “Mono-AZ à Multi-AZ” et notez les points suivants :

- Assurez-vous que votre instance de base de données RDS Custom pour Oracle doit utiliser un CEV créé après le 30 juin 2025 avec la dernière AMI fournie par le service.

- Mettez à jour les règles entrantes et sortantes du groupe de sécurité Amazon RDS pour autoriser le port 1120.
- Créez de nouveaux points de terminaison Amazon SQS VPC qui permettent à l'instance de base de données Amazon RDS RDS Custom pour Oracle de communiquer avec Amazon SQS.
- Mettez à jour les autorisations Amazon SQS dans le rôle de profil d'instance.

> **Important**
>
> Ne redémarrez pas votre instance de base de données principale Multi-AZ manuellement en vous connectant à l'instance sans suspendre RDS Custom pour Oracle Automation.

Étapes de migration pour les instances de base de données CEVs créées avant le 30 juin 2025

Si vous avez créé votre instance de base de données RDS Custom pour Oracle avant le 30 juin 2025, vous ne pouvez pas la modifier directement pour un déploiement multi-AZ, car le CEV sous-jacent a été créé avec une ancienne AMI fournie par un service qui ne prend pas en charge le multi-AZ. Vous devez migrer votre base de données vers une nouvelle instance à l'aide d'un CEV que vous aurez créé après le 30 juin 2025. Procédez comme suit pour la migration.

1. Créez un CEV à partir de la source avec la dernière AMI fournie par le service (après le 30 juin 2025)
2. Lancez une nouvelle instance de base de données avec le nouveau CEV
3. Migrez votre base de données de l'instance de base de données existante qui ne prend pas en charge le déploiement multi-AZ vers l'instance nouvellement créée à l'aide de l'une des méthodes suivantes :
   - Migration physique des bases de données Oracle vers Amazon RDS Custom à l'aide de Data Guard
   - Migration physique des bases de données Oracle vers Amazon RDS Custom à l'aide de la duplication RMAN

Modification d'un déploiement RDS Custom for Oracle Single-AZ vers un déploiement multi-AZ

Vous pouvez modifier une instance RDS Custom for Oracle compatible multi-AZ existante pour passer d'un déploiement mono-AZ à un déploiement multi-AZ. Lorsque vous modifiez l'instance de base de données, Amazon RDS effectue plusieurs actions :

- Prend un instantané de l'instance de base de données primaire.
- Crée de nouveaux volumes pour le réplica en attente à partir de l'instantané. Ces volumes s'initialisent en arrière-plan, et les performances maximales du volume sont atteintes après l'initialisation complète des données.
- Active la réplication synchrone au niveau des blocs entre les instances de base de données principale et de secours.

> **Important**
>
> Nous vous recommandons d'éviter de modifier votre instance de base de données RDS Custom pour Oracle d'un déploiement mono-AZ à un déploiement multi-AZ sur une instance de base de données de production pendant les périodes de pointe.

AWS utilise un instantané pour créer l'instance de secours afin d'éviter les temps d'arrêt lors de la conversion du mode mono-AZ au mode multi-AZ, mais les performances peuvent être affectées pendant et après la conversion au mode multi-AZ. Cet impact peut être significatif pour les charges de travail sensibles à la latence d'écriture. Bien que cette fonctionnalité permette de restaurer rapidement de gros volumes à partir de snapshots, elle peut entraîner une augmentation de la latence des I/O opérations en raison de la réplication synchrone. Cette latence peut avoir un impact sur les performances de votre base de données.

Configuration des conditions requises pour modifier un déploiement mono-AZ en déploiement multi-AZ à l'aide de CloudFormation

Suivez the section called “Étape 3 : Extraire les CloudFormation modèles pour RDS Custom pour Oracle” pour configurer à nouveau votre profil VPC et IAM afin d'ajouter un point de terminaison VPC SQS et une autorisation SQS dans le profil IAM.

Configuration des conditions préalables pour modifier manuellement un déploiement mono-AZ en un déploiement multi-AZ

Si vous choisissez de configurer les conditions préalables manuellement, exécutez les tâches suivantes.

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/
2. Choisissez Point de terminaison. La page Créer un point de terminaison s'affiche.
3. Pour Catégorie de service, choisissez Services AWS .
4. Dans Services, recherchez`SQS`.
5. Dans VPC, choisissez le VPC sur lequel votre instance de base de données RDS Custom for Oracle est déployée.
6. Dans Sous-réseaux, choisissez les sous-réseaux sur lesquels votre instance de base de données RDS Custom for Oracle est déployée.
7. Dans Groupes de sécurité, choisissez le groupe de sécurité dans lequel votre instance de base de données RDS Custom for Oracle est déployée.
8. Pour Politique, choisissez Personnalisé.
9. Dans votre politique personnalisée, remplacez le `AWS partition` `Region`, `accountId`, et `IAM-Instance-role` par vos propres valeurs.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/AWSRDSCustom": "custom-oracle"
                }
            },
            "Action": [
                "SQS:SendMessage",
                "SQS:ReceiveMessage",
                "SQS:DeleteMessage",
                "SQS:GetQueueUrl"
            ],
```

```
            "Resource": "arn:aws:sqs:us-east-1:111122223333:do-not-delete-rds-
custom-*",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/{IAM-Instance-role}"
            }
        }
    ]
}
```

Mettez à jour le Profil d'instance avec l'autorisation d'accéder à Amazon SQS. Remplacez le `AWS partitionRegion`, et `accountId` par vos propres valeurs.

```
{
    "Sid": "13",
    "Effect": "Allow",
    "Action": [
        "SQS:SendMessage",
        "SQS:ReceiveMessage",
        "SQS:DeleteMessage",
        "SQS:GetQueueUrl"
    ],
    "Resource": [
        {
            "Fn::Sub": "arn:${AWS::Partition}:sqs:${AWS::Region}:${AWS::AccountId}:do-
not-delete-rds-custom-*"
        }
    ],
    "Condition": {
        "StringLike": {
            "aws:ResourceTag/AWSRDSCustom": "custom-oracle"
        }
    }
}
```

Mettez à jour les règles entrantes et sortantes du groupe de sécurité Amazon RDS pour autoriser le port 1120.

- Dans Groupes de sécurité, choisissez le groupe dans lequel votre instance de base de données RDS Custom for Oracle est déployée.

- Pour les règles entrantes, créez une règle TCP personnalisée pour autoriser le port `1120` depuis le groupe source.
- Pour les règles sortantes, créez une règle TCP personnalisée pour autoriser le port `1120` vers le groupe de destination.

Modifiez à l'aide de la console RDS ou de AWS CLI l'API RDS

Une fois les conditions requises remplies, vous pouvez modifier une instance de base de données personnalisée RDS pour Oracle d'un déploiement mono-AZ à un déploiement multi-AZ à l'aide de la console Amazon RDS AWS CLI ou de l'API Amazon RDS.

console

Pour modifier un déploiement RDS Custom for Oracle mono-AZ vers multi-AZ existant

1. Connectez-vous à la console de AWS gestion et ouvrez la console Amazon RDS à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la console Amazon RDS, choisissez Bases de données. Le volet Bases de données s'affiche.
3. Choisissez l'instance de base de données RDS Custom pour Oracle que vous souhaitez modifier.
4. Dans Actions, choisissez Convertir en déploiement multi-AZ.
5. Sur la page Confirmation, choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. Le choix de cette option n'entraîne pas d'interruption de service, mais il existe un impact possible sur les performances. Vous pouvez également choisir d'appliquer la mise à jour pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter the section called “Modifications de planification”.
6. Sur la page Confirmation, choisissez Convertir en Multi-AZ.

AWS CLI

Pour passer à un déploiement d'instance de base de données multi-AZ à l'aide de AWS CLI, appelez la modify-db-instancecommande et définissez l'`--multi-az`option. Spécifiez l'identifiant d'instance de base de données et les valeurs des autres options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter the section called “Paramètres disponibles”.

Example exemple

Le code suivant modifie `mycustomdbinstance` en incluant l'option `--multi-az`. Les modifications sont appliquées dans la prochaine fenêtre de maintenance à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`. Pour de plus amples informations, veuillez consulter the section called “Modifications de planification”.

Pour Linux, macOS ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mycustomdbinstance \
    --multi-az \
    [--no-apply-immediately | --apply-immediately]
```

Pour Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mycustomdbinstance ^
    --multi-az ^
    [--no-apply-immediately | --apply-immediately]
```

API RDS

Pour passer à un déploiement d'instance de base de données multi-AZ avec l'API Amazon RDS, appelez l'DBInstanceopération Modify et définissez le `MultiAZ` paramètre sur true.

Modification d'un déploiement RDS Custom for Oracle Multi-AZ vers un déploiement mono-AZ

Vous pouvez modifier une instance de base de données RDS Custom pour Oracle existante d'un déploiement multi-AZ à un déploiement mono-AZ.

console

Pour modifier une instance de base de données RDS Custom pour Oracle d'un déploiement multi-AZ à un déploiement mono-AZ

1. Connectez-vous à la console de AWS gestion et ouvrez la console Amazon RDS à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la console Amazon RDS, choisissez Bases de données. Le volet Bases de données s'affiche.

3. Choisissez l'instance de base de données RDS Custom pour Oracle que vous souhaitez modifier.
4. Pour Déploiement multi-AZ, choisissez Non.
5. Sur la page Confirmation, choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. Le choix de cette option n'entraîne pas d'interruption de service, mais il existe un impact possible sur les performances. Vous pouvez également choisir d'appliquer la mise à jour pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter the section called “Modifications de planification”.
6. Sur la page Confirmation, choisissez Modifier l'instance de base de données.

AWS CLI

Pour modifier un déploiement multi-AZ en un déploiement mono-AZ à l'aide de AWS CLI, appelez la modify-db-instancecommande et incluez l'`--no-multi-az`option. Spécifiez l'identifiant d'instance de base de données et les valeurs des autres options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter the section called “Paramètres disponibles”.

Example exemple

Le code suivant modifie `mycustomdbinstance` en incluant l'option `--no-multi-az`. Les modifications sont appliquées dans la prochaine fenêtre de maintenance à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`. Pour de plus amples informations, veuillez consulter the section called “Modifications de planification”.

Pour Linux, macOS ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mycustomdbinstance \
    --no-multi-az \
    [--no-apply-immediately | --apply-immediately]
```

Pour Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mycustomdbinstance ^
    --no-multi-az ^
    [--no-apply-immediately | --apply-immediately]
```

API RDS

Pour modifier un déploiement multi-AZ en un déploiement mono-AZ à l'aide de l'API Amazon RDS, appelez l'DBInstanceopération Modify et définissez le `MultiAZ` paramètre sur. `false`

Processus de basculement pour un déploiement RDS Custom pour Oracle Multi-AZ

Si une interruption prévue ou imprévue de votre instance de base de données est le résultat d'une anomalie de l'infrastructure, Amazon RDS bascule automatiquement sur le réplica de secours d'une autre zone de disponibilité si vous avez activé l'option Multi-AZ. La durée du basculement dépend de l'activité de la base de données et d'autres conditions au moment où l'instance de base de données primaire est devenue indisponible. Les temps de basculement oscillent généralement entre 60 et 120 secondes. Cependant, les transactions importantes ou les processus de récupération longs peuvent augmenter le temps de basculement. Lorsque le basculement est terminé, l'affichage de la nouvelle zone de disponibilité par la console Amazon RDS peut prendre plus de temps.

> **Note**
>
> Vous pouvez forcer un basculement manuellement lorsque vous arrêtez et redémarrez votre EC2 hôte principal alors que votre instance de base de données est disponible.

Étant donné qu'Amazon RDS gère automatiquement les basculements, vous pouvez reprendre les opérations de base de données aussi rapidement que possible sans intervention administrative. L'instance de base de données primaire bascule automatiquement vers le réplica de secours si l'une des conditions décrites dans le tableau suivant se produit : Vous pouvez consulter ces raisons de basculement dans le journal des événements Amazon RDS.

Raison du basculement	Description
Le système d'exploitation sous-jacent à l'instance de base de données RDS fait l'objet d'un correctif dans le cadre d'une opération hors connexion.	Un basculement a été déclenché pendant la fenêtre de maintenance d'un correctif du système d'exploit ation ou d'une mise à jour de sécurité. Pour de plus amples informations, veuillez consulter the section called “Entretien d'une instance de base de données”.

Raison du basculement	Description
L'hôte principal de l'instance RDS multi-AZ est non sain.	Le déploiement d'instance de base de données multi-AZ a détecté une instance de base de données primaire déficiente et a opéré un basculement.
L'hôte principal de l'instance RDS multi-AZ est inaccessible en raison d'une perte de connectivité réseau.	La surveillance RDS a détecté une défaillance de la capacité d'accessibilité du réseau à l'instance de base de données primaire et a déclenché un basculement.
L'instance RDS a été modifiée par le client.	Une modification d'instance de base de données RDS a déclenché un basculement. Pour de plus amples informations, veuillez consulter the section called “Modification de votre instance de base de données”.
Le volume de stockage sous-jacent à l'hôte principal de l'instance RDS multi-AZ a été défaillant.	Le déploiement d'instance de base de données multi-AZ a détecté un problème de stockage sur l'instance de base de données primaire et a opéré un basculement.
L'instance principale RDS multi-AZ est occupée et ne répond pas.	L'instance de base de données primaire ne répond pas. Nous vous recommandons de procéder comme suit : examinez l'événement et les CloudWatch journaux pour détecter toute utilisation excessive du processeur, de la mémoire ou de l'espace de swap. Pour plus d’informations, consultez the section called “Utilisation des notifications d'RDSévénements Amazon” et the section called “Création d'une règle qui se déclenche sur un évènement Amazon RDS”. Évaluez votre charge de travail pour déterminer si vous utilisez la classe d'instance de base de données appropriée. Pour de plus amples informations, veuillez consulter the section called “Classes d'instances de base de données”.

Pour déterminer si votre instance de base de données Multi-AZ a basculé, voici ce que vous pouvez faire :

- Configurez les abonnements aux événements de base de données de sorte qu'ils vous notifient par e-mail ou SMS qu'un basculement a été initié. Pour plus d'informations sur les événements, consultez the section called “Utilisation des notifications d'RDSévénements Amazon”.
- Examinez vos événements de base de données à l'aide de la console Amazon RDS ou des opérations d'API.
- Consultez l'état actuel de votre déploiement d'instance de base de données RDS Custom for Oracle Multi-AZ à l'aide de la console Amazon RDS, de la CLI ou des opérations d'API.

Paramètres de durée de vie (TTL) avec des applications utilisant un déploiement RDS Custom pour Oracle Multi-AZ

Le mécanisme de basculement modifie automatiquement l'enregistrement DNS de l'instance de base de données pour pointer vers l'instance de base de données en attente. Par conséquent, vous devez rétablir toutes les connexions existantes à votre instance de base de données. Assurez-vous que toute valeur de configuration du cache DNS time-to-live (TTL) est faible et vérifiez que votre application ne mettra pas le DNS en cache pendant une période prolongée. Une valeur de durée de vie élevée peut empêcher votre application de se reconnecter rapidement à l'instance de base de données après un basculement.

Déploiement de RDS Custom pour Oracle avec AWS CloudFormation

Automatisez votre déploiement de RDS Custom pour Oracle à l'aide du AWS CloudFormation modèle fourni. Remplissez les conditions préalables suivantes avant de déployer les ressources.

Prérequis

Téléchargez les fichiers Oracle requis

Vous avez besoin de fichiers d'installation Oracle spécifiques avant de pouvoir créer le AWS CloudFormation modèle. Téléchargez ces fichiers avant de les déployer.

1. Accédez à Oracle Database 19c (19.3)
2. Localisez et téléchargez le fichier `LINUX.X64_193000_db_home.zip`
3. Renommez le fichier en `V982063-01.zip`
4. Téléchargez les correctifs restants, en sélectionnant Plateforme ou Langue `Linux x86-64`

Dernier OPatch utilitaire

Patch 6880880

Patches PSU de janvier 2023

Correctifs PSU et RU de base de données

- Patch 34765931
- Patch 34786990

Correctifs supplémentaires requis

- Patch 35099667
- Patch 35099674
- Patch 28730253
- Patch 29213893
- Patch 35012866

Patches PSU d'avril 2023

Correctifs PSU et RU de base de données

- Patch 35042068
- Patch 35050341

Correctifs supplémentaires requis

- Patch 28730253
- Patch 29213893
- Patch 33125873
- Patch 35220732
- Patch 35239280

Configuration du compartiment Amazon S3

1. Créez un compartiment Amazon S3 dans votre Compte AWS compartiment ou choisissez un compartiment existant.
2. Créez une structure de dossiers dans le compartiment, comme dans l'exemple ci-dessous.

```
<bucket-name>/
### oracle_cev/
    ### V982063-01.zip
    ### p6880880_190000_Linux-x86-64.zip
    ### p34765931_190000_Linux-x86-64.zip
    ### p34786990_190000_Linux-x86-64.zip
    ### p35099667_190000_Linux-x86-64.zip
    ### p35099674_190000_Generic.zip
    ### p28730253_190000_Linux-x86-64.zip
    ### p29213893_1918000DBRU_Generic.zip
    ### p35012866_1918000DBRU_Linux-x86-64.zip
    ### p35042068_190000_Linux-x86-64.zip
    ### p35050341_190000_Linux-x86-64.zip
    ### p29213893_1919000DBRU_Generic.zip
    ### p33125873_1919000DBRU_Linux-x86-64.zip
    ### p35220732_190000_Linux-x86-64.zip
    ### p35239280_190000_Generic.zip
```

3. Chargez tous les fichiers Oracle que vous avez précédemment téléchargés dans les dossiers appropriés.

Déployez RDS Custom pour Oracle à l'aide de AWS CloudFormation

Étape 1 : préparer le AWS CloudFormation modèle

Avant de déployer RDS Custom pour Oracle, vous devez télécharger et configurer le AWS CloudFormation modèle qui crée les prérequis nécessaires.

Copiez et enregistrez le modèle

1. Accédez à Déploiement de RDS Custom pour Oracle avec une ou plusieurs zones de disponibilité
2. Copiez le modèle dans le format de votre choix (YAML ou JSON)
3. Enregistrez le fichier au format YAML ou JSON. Par exemple, `rds-custom-oracle-prereqs.yaml`

Lancez la pile dans la AWS console

1. Ouvrez la AWS console et accédez à AWS CloudFormation
2. Choisissez Créer une pile > Avec de nouvelles ressources (standard)
3. Sélectionnez Choisir un modèle existant
4. Sélectionnez Charger un fichier modèle > Choisir un fichier
5. Sélectionnez le fichier modèle que vous avez précédemment téléchargé
6. Conserver les valeurs des paramètres par défaut
7. Sélectionnez Next pour créer la pile

Alternative : utilisation de la AWS CLI

Au lieu d'utiliser la console, vous pouvez créer la pile en utilisant AWS CLI :

```
aws cloudformation create-stack \
  --stack-name rds-custom-oracle \
  --template-body file://rds-custom-oracle-prereqs.yaml \
  --capabilities CAPABILITY_NAMED_IAM
```

Étape 2 : Création des versions personnalisées du moteur (CEVs) et des instances Amazon RDS

Copiez et enregistrez le modèle

1. Accédez à Déploiement de RDS Custom pour Oracle avec une ou plusieurs zones de disponibilité
2. Copiez le modèle dans le format de votre choix (YAML ou JSON)
3. Mettez à jour les paramètres suivants dans le modèle si nécessaire :
   - `BucketName`
   - `CEVS3Prefix`
   - Mot de passe principal de la base de données (remplace **********)
4. Enregistrez le fichier au format YAML ou JSON

Étape 3 : déploiement à l'aide de la AWS console

1. Ouvrez la AWS console et accédez à AWS CloudFormation
2. Choisissez Créer une pile > Avec de nouvelles ressources (standard)

3. Sélectionnez Choisir un modèle existant
4. Sélectionnez Charger un fichier modèle > Choisir un fichier
5. Sélectionnez le fichier modèle que vous avez précédemment téléchargé
6. Laissez les paramètres comme valeurs par défaut
7. Renseignez les paramètres comme suit :

```
BucketName: rds-custom-id
CEVS3Prefix: oracle_cev
CEVCreation: Yes
```

8. Vérifiez la configuration et sélectionnez Suivant pour créer la pile

Facultatif : déploiement à l'aide de AWS CLI

```
aws cloudformation create-stack \
  --stack-name rds-custom-oracle \
  --template-body file://rds-custom-oracle.yaml \
  --parameters \
    ParameterKey=BucketName,ParameterValue=rds-custom-id \
    ParameterKey=CEVS3Prefix,ParameterValue=oracle_cev \
    ParameterKey=CEVCreation,ParameterValue=Yes \
  --capabilities CAPABILITY_NAMED_IAM
```

Ressources de déploiement créées

Le modèle crée les ressources suivantes :

- Amazon VPC avec sous-réseaux publics et privés
- Groupes de sécurité
- Points de terminaison Amazon VPC
- Rôles et politiques IAM
- AWS KMS clé de chiffrement
- Versions de moteur personnalisées (CEVs)
- Instances RDS Custom pour Oracle pour les configurations mono-AZ et multi-AZ

Surveillez la progression de votre déploiement

Après avoir créé la AWS CloudFormation pile, surveillez sa progression pour garantir un déploiement réussi. Le processus de déploiement inclut la création de versions de moteur personnalisées (CEVs) et d'instances RDS.

Pour suivre la progression du déploiement :

1. Ouvrez la AWS CloudFormation console.
2. Choisissez le nom de votre pile.
3. Cliquez sur l'onglet Événements pour voir la progression et identifier les éventuelles erreurs.

> **Note**
>
> La création d'un CEV nécessite généralement 2 à 3 heures. Une fois la création du CEV terminée avec succès, Amazon RDS commence automatiquement à créer l'instance Amazon RDS.

Après le déploiement

Une fois le processus de création de la pile terminé, effectuez les étapes de vérification et de configuration post-déploiement suivantes :

1. Sur la page de la console Amazon RDS, accédez à Versions du moteur personnalisées pour vérifier la création du CEV.
2. Vérifiez que les instances Amazon RDS sont créées et disponibles
3. Testez la connectivité aux instances Amazon RDS
4. Configurez des stratégies de surveillance et de sauvegarde selon les besoins

Nettoyage

Pour supprimer toutes les ressources, exécutez la AWS CLI commande suivante :

```
aws cloudformation delete-stack --stack-name rds-custom-oracle
```

Résolution des problèmes

Si vous rencontrez des problèmes lors du déploiement, utilisez les solutions suivantes pour résoudre les problèmes courants.

La création du CEV échoue

- Vérifiez que tous les correctifs requis sont chargés sur Amazon S3
- Vérifier les autorisations IAM
- Vérifiez que les versions des correctifs sont correctes ; consultez le the section called “Prérequis” pour la liste des correctifs requis.

La création d'une instance Amazon RDS échoue

- Vérifiez les VPC/subnet configurations
- Vérifier les règles du groupe de sécurité
- Confirmez que le CEV est disponible

Utilisation de répliques Oracle pour RDS Custom for Oracle

Vous pouvez créer des répliques Oracle pour RDS Custom pour les instances de base de données Oracle qui exécutent Oracle Enterprise Edition. Les bases de données de conteneurs (CDBs) et non-CDBs sont prises en charge. L'édition Standard 2 ne prend pas en charge Oracle Data Guard.

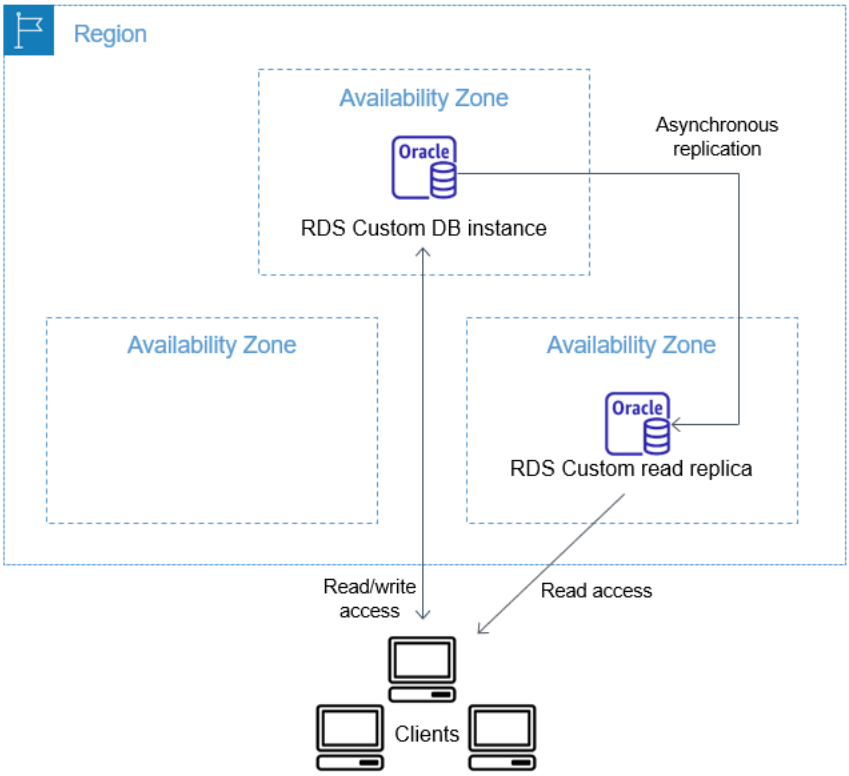
La création d'une réplique RDS personnalisée pour Oracle est similaire à la création d'une réplique RDS pour Oracle, mais avec des différences importantes. Pour obtenir des informations générales sur la création et la gestion de réplicas Oracle, consultez Utilisation des réplicas en lecture d'instance de base de données et Utilisation de réplicas en lecture pour Amazon RDS for Oracle.

Rubriques

- Présentation de la réplication RDS personnalisée pour Oracle
- Instructions et limites de la réplication RDS Custom for Oracle
- Promouvoir une réplique RDS personnalisée pour Oracle en une instance de base de données autonome
- Configuration d'un VPN tunnel entre les instances principales et répliques de RDS Custom for Oracle

Présentation de la réplication RDS personnalisée pour Oracle

L'architecture de RDS Custom for Oracle Replication est analogue à celle de RDS la réplication Oracle. Une instance de base de données principale se réplique de manière asynchrone vers un ou plusieurs réplicas Oracle.



Nombre maximum de réplicas

Comme RDS pour Oracle, vous pouvez créer jusqu'à cinq répliques Oracle gérées de votre instance de base de données principale RDS Custom for Oracle. Vous pouvez également créer vos propres réplicas Oracle configurées manuellement (externes). Les réplicas externes ne sont pas pris en compte dans le calcul de votre limite d'instance de base de données. Ils se situent également en dehors du périmètre de support RDS personnalisé. Pour plus d'informations sur le périmètre de support, consultez Périmètre de prise en charge RDS Custom.

Convention d'attribution de noms de réplica

Les noms des réplicas Oracle sont basés sur le nom unique de la base de données. Le format est *DB_UNIQUE_NAME_X*, avec des lettres ajoutées séquentiellement. Par exemple, si le nom unique de votre base de données est `ORCL`, les deux premiers réplicas sont nommés `ORCL_A` et `ORCL_B`. Les six premières lettres, A à F, sont réservées RDS à Custom. RDS Custom copie les paramètres de base de données de votre instance de base de données principale vers les répliques. Pour plus d'informations, consultez DB_ UNIQUE _ NAME dans la documentation Oracle.

Rétention des sauvegardes de réplicas

Par défaut, les répliques Oracle RDS personnalisées utilisent la même période de conservation des sauvegardes que votre instance de base de données principale. Vous pouvez modifier la

période de conservation des sauvegardes pour la porter à 1 à 35 jours. RDS Custom prend en charge la sauvegarde, la restauration et point-in-time la restauration (PITR). Pour plus d'informations sur la sauvegarde et la restauration d'instances de base de données RDS personnalisées, consultezSauvegarde et restauration d'une instance de base de données Amazon RDS Custom pour Oracle.

> **Note**
>
> Lors de la création d'une réplique Oracle, RDS Custom interrompt temporairement le nettoyage des fichiers de journalisation. RDSCustom s'assure ainsi de pouvoir appliquer ces journaux à la nouvelle réplique Oracle une fois qu'elle sera disponible.

Promotion de réplicas

Vous pouvez promouvoir les répliques Oracle gérées dans RDS Custom for Oracle à l'aide de la console, de la `promote-read-replica` AWS CLI commande ou `PromoteReadReplica`API. Si vous supprimez votre instance de base de données principale et que toutes les répliques sont saines, RDS Custom for Oracle transforme automatiquement vos répliques gérées en instances autonomes. Si un réplica a suspendu l'automatisation ou se trouve en dehors du périmètre de support, vous devez réparer le réplica pour que RDS Custom puisse le promouvoir automatiquement. Vous ne pouvez promouvoir que manuellement les réplicas Oracle externes.

Instructions et limites de la réplication RDS Custom for Oracle

Lorsque vous créez des réplicas RDS Custom for Oracle, toutes les options de réplica RDS Oracle ne sont pas prises en charge.

Rubriques

- Instructions générales de la réplication RDS Custom for Oracle
- Limites générales pour la réplication RDS Custom for Oracle
- Exigences et limites en matière de réseau de la réplication RDS Custom for Oracle
- Limites des réplicas externes pour RDS Custom for Oracle

Instructions générales de la réplication RDS Custom for Oracle

Lorsque vous utilisez RDS Custom for Oracle, suivez les instructions suivantes :

- Vous pouvez utiliser RDS Custom pour la réplication Oracle uniquement dans Oracle Enterprise Edition. L'édition Standard 2 n'est pas prise en charge.
- Nous vous recommandons vivement d'implémenter un tunnel VPN pour chiffrer les communications entre vos instances principales et de secours. Pour de plus amples informations, veuillez consulter Configuration d'un VPN tunnel entre les instances principales et répliques de RDS Custom for Oracle.
- Ne modifiez pas l'utilisateur `RDS_DATAGUARD`. Cet utilisateur est réservé pour l'automatisation de RDS Custom for Oracle. La modification de cet utilisateur peut entraîner des résultats indésirables, tels que l'impossibilité de créer des réplicas Oracle pour votre instance RDS Custom for Oracle DB.
- Ne modifiez pas le mot de passe de l'utilisateur de la réplication. Il est nécessaire pour administrer la configuration Oracle Data Guard sur l'hôte RDS Custom. Si vous modifiez le mot de passe, RDS Custom for Oracle risque de placer votre réplica Oracle en dehors du périmètre de support. Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.

  Le mot de passe est stocké et étiqueté avec l'ID de ressource de base de données. AWS Secrets Manager Chaque réplica Oracle possède son propre secret dans Secrets Manager. Le secret utilise l'un des formats de dénomination suivants.

  ```
  do-not-delete-rds-custom-db-DB_resource_id-uuid-dg
  rds-custom!oracle-do-not-delete-DB_resource_id-uuid-dg
  ```

- Ne modifiez pas `DB_UNIQUE_NAME` pour l'instance de base de données principale. La modification du nom entraîne le blocage de toute opération de restauration.
- Ne spécifiez pas la clause `STANDBYS=NONE` dans une commande `CREATE PLUGGABLE DATABASE` d'une CDB RDS Custom. Ainsi, en cas de basculement, votre CDB de secours contient tout. PDBs

Limites générales pour la réplication RDS Custom for Oracle

Voici les limites des réplicas pour RDS Custom for Oracle :

- Vous ne pouvez pas créer des réplicas RDS Custom for Oracle en mode lecture seule. Toutefois, vous pouvez modifier manuellement le mode des réplicas montés en lecture seule, et de lecture seule à monté. Pour plus d'informations, consultez la documentation de la AWS CLI commande create-db-instance-read-replica.
- Vous ne pouvez pas créer de réplicas RDS Custom for Oracle entre les régions.

- Vous ne pouvez pas modifier la valeur du paramètre `CommunicationTimeout` d'Oracle Data Guard. Ce paramètre est fixé à 15 secondes pour RDS Custom pour les instances de base de données Oracle.

Exigences et limites en matière de réseau de la réplication RDS Custom for Oracle

Assurez-vous que votre configuration réseau prend en charge RDS Custom pour les réplicas Oracle. Éléments à prendre en compte :

- Assurez-vous d'activer le port 1140 pour les communications entrantes et sortantes dans votre cloud privé virtuel (VPC) pour l'instance de base de données primaire et l'ensemble de ses réplicas. Il s'agit d'une obligation pour les communications Oracle Data Guard entre les réplicas en lecture.
- RDS Custom for Oracle valide le réseau tout en créant un réplica Oracle. Si l'instance de base de données principale et le nouveau réplica ne peuvent pas se connecter sur le réseau, RDS Custom for Oracle ne crée pas le réplica et le place dans l'état `INCOMPATIBLE_NETWORK`.
- Pour les répliques Oracle externes, telles que celles que vous créez sur Amazon EC2 ou sur site, utilisez un autre port et un autre écouteur pour la réplication Oracle Data Guard. La tentative d'utilisation du port 1140 peut entraîner des conflits avec l'automatisation de RDS Custom.
- Le fichier `/rdsdbdata/config/tnsnames.ora` contient des noms de service réseau mappés aux adresses du protocole d'écoute. Notez les exigences et recommandations suivantes :
  - Dans le fichier `tnsnames.ora`, les entrées dont le préfixe est `rds_custom_` sont réservées à RDS Custom lors de la gestion des opérations de réplica Oracle.

    N'utilisez pas ce préfixe lors de la création d'entrées manuelles dans le fichier `tnsnames.ora`.
  - Dans certains cas, il se peut que vous optiez pour un basculement manuel ou que vous utilisiez des technologies de basculement telles que FSFO (Fast-Start Failover). Dans ce cas, assurez-vous de synchroniser manuellement les entrées du fichier `tnsnames.ora` de l'instance de base de données primaire vers toutes les instances de secours. Cette recommandation s'applique à la fois aux réplicas Oracle gérés par RDS Custom et aux réplicas Oracle externes.

    L'automatisation de RDS Custom ne met à jour les entrées `tnsnames.ora` que sur l'instance de base de données principale. Veillez également à effectuer une synchronisation lorsque vous ajoutez ou supprimez un réplica Oracle.

Si vous ne synchronisez pas les fichiers `tnsnames.ora` et effectuez un basculement manuel, il se peut qu'Oracle Data Guard sur l'instance de base de données primaire ne soit pas en mesure de communiquer avec les réplicas Oracle.

Limites des réplicas externes pour RDS Custom for Oracle

Les réplicas externes de RDS Custom for Oracle, qui incluent les réplicas sur site, présentent les limitations suivantes :

- RDS Custom for Oracle ne détecte pas les changements de rôle des instances lors d'un basculement manuel, tel que FSFO, pour les réplicas Oracle externes.

  RDS Custom for Oracle détecte les modifications apportées aux réplicas gérés. Le changement de rôle est indiqué dans le journal des événements. Vous pouvez également voir le nouvel état à l'aide de la describe-db-instances AWS CLI commande.
- RDS Custom for Oracle ne détecte pas un retard de réplication important pour les réplicas Oracle externes.

  RDS Custom for Oracle détecte les retards pour les réplicas gérés. Un retard de réplication élevé génère l'événement `Replication has stopped`. Vous pouvez également voir l'état de la réplication à l'aide de la describe-db-instances AWS CLI commande, mais sa mise à jour peut être retardée.
- RDS Custom for Oracle ne promeut pas automatiquement les réplicas Oracle externes si vous supprimer votre instance de base de données principale.

  La fonction de promotion automatique n'est disponible que pour les réplicas Oracle gérés. Pour plus d'informations sur la promotion manuelle de réplicas Oracle, consultez le livre blanc Enabling high availability with Data Guard on Amazon RDS Custom for Oracle (Activation de la haute disponibilité avec Data Guard sur Amazon RDS Custom for Oracle).

Promouvoir une réplique RDS personnalisée pour Oracle en une instance de base de données autonome

Tout comme RDS pour Oracle, vous pouvez promouvoir une réplique RDS personnalisée pour Oracle en une instance de base de données autonome. Lorsque vous promouvez une réplique Oracle, RDS Custom for Oracle redémarre l'instance de base de données avant qu'elle ne soit

disponible. Pour obtenir plus d'informations sur la promotion des réplicas Oracle, consultez Promotion d'un réplica en lecture en instance de bases de données autonome.

Lorsque vous faites la promotion d'un réplica, tenez compte des directives suivantes :

- Ne lancez pas de basculement pendant que RDS Custom for Oracle fait la promotion de votre réplique. Sinon, le flux de travail de la promotion pourrait se bloquer.
- Ne changez pas d'instance de base de données principale pendant que RDS Custom for Oracle fait la promotion de votre réplique Oracle. Sinon, le flux de travail de la promotion pourrait se bloquer.
- N'arrêtez pas votre instance de base de données principale pendant que RDS Custom for Oracle fait la promotion de votre réplique Oracle. Sinon, le flux de travail de la promotion pourrait se bloquer.
- N'essayez pas de redémarrer la réplication avec votre instance de base de données nouvellement promue comme cible. Une fois que RDS Custom for Oracle a fait la promotion de votre réplique Oracle, celle-ci devient une instance de base de données autonome et n'a plus le rôle de réplique.

Notez les limites suivantes concernant la promotion de répliques RDS personnalisées pour Oracle :

- Vous ne pouvez pas promouvoir une réplique pendant que RDS Custom for Oracle la sauvegarde.
- Vous ne pouvez pas modifier la période de conservation des sauvegardes pour `0` lors de la promotion de votre réplica Oracle.
- Vous ne pouvez pas promouvoir votre réplica s'il n'est pas dans un état sain.

  Si vous émettez un problème `delete-db-instance` sur l'instance de base de données principale, RDS Custom for Oracle vérifie que chaque réplique Oracle gérée est saine et disponible pour la promotion. Un réplica peut être inéligible à la promotion parce que l'automatisation est en pause ou qu'il se trouve en dehors du périmètre de support. Dans ce cas, RDS Custom for Oracle publie un événement expliquant le problème afin que vous puissiez réparer votre réplique Oracle manuellement.

Les étapes suivantes montrent le processus général de promotion d'un réplica Oracle en instance de base de données :

1. Empêchez l'écriture de toute transaction dans l'instance de base de données principale.
2. Attendez que RDS Custom for Oracle applique toutes les mises à jour à votre réplique Oracle.

3. Promouvez votre réplique Oracle en choisissant l'option Promote sur la RDS console Amazon **promote-read-replica**, la AWS CLI commande ou l'RDSAPIopération **PromoteReadReplica**Amazon.

La promotion d'un réplica Oracle dure quelques minutes. Au cours du processus, RDS Custom for Oracle arrête la réplication et redémarre votre réplique. Une fois le redémarrage terminé, le réplica Oracle est disponible en tant qu'instance de base de données autonome. Pour plus d'informations sur la résolution des problèmes liés à la promotion des répliques, consultezDépannage de la promotion de réplica pour RDS Custom for Oracle.

Console

Pour promouvoir une réplique RDS personnalisée pour Oracle en instance de base de données autonome

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la RDS console Amazon, sélectionnez Databases.

   Le volet Bases de données s'affiche. Chaque réplica Oracle indique Replica (Réplica) dans la colonne Role (Rôle).
3. Choisissez la réplique RDS personnalisée pour Oracle que vous souhaitez promouvoir.
4. Pour Actions, choisissez Promote (Promouvoir).
5. Dans la page Promote Oracle replica (Promouvoir le réplica d'Oracle), saisissez la période de rétention des sauvegardes et la fenêtre de sauvegarde pour l'instance de base de données nouvellement promue. Vous ne pouvez pas fixer cette valeur à 0.
6. Lorsque les paramètres sont tels que vous les souhaitez, sélectionnez Promote Oracle replica (Promouvoir le réplica Oracle).

AWS CLI

Pour promouvoir votre réplique RDS personnalisée pour Oracle en une instance de base de données autonome, utilisez la AWS CLI `promote-read-replica`commande.

Example

Dans Linux, macOS, ou Unix:

```
aws rds promote-read-replica \
--db-instance-identifier my-custom-read-replica \
--backup-retention-period 2 \
--preferred-backup-window 23:00-24:00
```

Dans Windows:

```
aws rds promote-read-replica ^
--db-instance-identifier my-custom-read-replica ^
--backup-retention-period 2 ^
--preferred-backup-window 23:00-24:00
```

RDS API

Pour faire de votre réplique RDS Custom for Oracle une instance de base de données autonome, appelez l'RDSAPI`PromoteReadReplica`opération Amazon avec le paramètre `DBInstanceIdentifier` requis.

Configuration d'un VPN tunnel entre les instances principales et répliques de RDS Custom for Oracle

Un VPN tunnel est une connexion cryptée entre deux ou plusieurs appareils sur un réseau. Pour garantir le plus haut niveau de sécurité à vos instances Oracle Data Guard dans RDS Custom for Oracle, nous vous recommandons vivement d'implémenter un VPN tunnel pour chiffrer les communications entre vos instances principales et de secours. Le tunnel protège les données sensibles lorsqu'elles circulent sur le réseau entre les instances. Bien que cette configuration soit facultative, nous la recommandons comme meilleure pratique pour garantir la sécurité des données et la conformité réglementaire.

Assurez-vous de remplir les conditions préalables suivantes :

- Vous disposez d'un accès root aux hôtes principal et de secours.
- Vous disposez de l'expertise technique nécessaire pour exécuter la `ipsec` commande.

Pour configurer un VPN tunnel entre un serveur principal et un réplica dans RDS Custom for Oracle

1. Ajoutez les groupes de sécurité de l'instance principale et de l'instance de secours à la liste d'autorisation en respectant les règles suivantes :

```
ACTION FLOW SOURCE PROTO PORT

ALLOW ingress this-SG 50 (ESP) all (N/A)
ALLOW egress this-SG 50 (ESP) all (N/A)

ALLOW ingress this-SG 17 (UDP) 500 (IKE)
ALLOW egress this-SG 17 (UDP) 500 (IKE)
```

2. Basculez vers l'utilisateur racine.

```
$ sudo su - root
```

3. Exécutez les commandes suivantes sur l'instance principale et sur l'instance de secours pour initialiser la base de données Network Security Services (NSS) sous l'utilisateur`root`.

```
ipsec initnss --nssdir /etc/ipsec.d
```

4. Générez RSA les clés comme suit :

   a. Sur l'instance principale, générez les clés à l'aide de l'une des `ipsec` commandes suivantes, en fonction de la version de votre système d'exploitation.

   ```
   ipsec newhostkey --nssdir /etc/ipsec.d       ## for Oracle Linux Version 8
   ipsec newhostkey --output /etc/ipsec.secrets ## for Oracle Linux version 7.9
   ```

   b. Obtenez la clé publique dont vous avez besoin pour créer la configuration. Dans l'exemple suivant, l'instance principale est `left` due au fait que, dans le `ipsec` langage, elle `left` fait référence à l'appareil que vous êtes en train de configurer et `right` à l'appareil situé à l'autre bout du tunnel.

   ```
   ipsec showhostkey --left --ckaid ckaid-returned-in-last-statement
   ```

   c. Sur l'instance de secours, générez des clés pour l'instance de secours.

   ```
   ipsec newhostkey --nssdir /etc/ipsec.d       ## for Oracle Linux Version 8
   ipsec newhostkey --output /etc/ipsec.secrets ## for Oracle Linux version 7.9
   ```

   d. Obtenez la clé publique de l'instance de secours, dont vous avez besoin pour créer la configuration. Dans l'exemple suivant, l'instance de secours est `right` due au fait qu'elle fait référence au périphérique situé à l'autre bout du tunnel.

```
ipsec showhostkey --right --ckaid ckaid-returned-in-last-statement
```

5. Sur la base des RSA clés que vous avez obtenues, générez la configuration. La configuration est identique pour l'instance principale et pour l'instance de secours. Vous pouvez trouver l'adresse de l'instance principale et IPv4 l'adresse de l'instance IPv4 de secours dans la AWS console.

   Sur l'instance principale et sur l'instance de secours, enregistrez la configuration suivante dans le fichier`/etc/ipsec.d/custom-fb-tunnel.conf`.

   ```
   conn custom-db-tunnel
    type=transport
    auto=add
    authby=rsasig
    left=IPV4-for-primary
    leftrsasigkey=RSA-key-generated-on-primary
    right=IPV4-for-standby
    rightrsasigkey=RSA-key-generated-on-standby
   ```

6. Sur l'instance principale et sur l'instance de secours, démarrez le `ipsec` daemon sur les deux hôtes.

   ```
   ipsec setup start
   ```

7. Démarrez le tunnel sur l'instance principale ou sur l'instance de secours. La sortie doit ressembler à ce qui suit :

   ```
   [root@ip-172-31-6-81 ~]# ipsec auto --up custom-db-tunnel
   181 "custom-db-tunnel" #1: initiating IKEv2 connection
   181 "custom-db-tunnel" #1: sent IKE_SA_INIT request to 172.31.32.196:500
   182 "custom-db-tunnel" #1: sent IKE_AUTH request {cipher=AES_GCM_16_256 integ=n/a
    prf=HMAC_SHA2_512 group=DH19}
   003 "custom-db-tunnel" #1: initiator established IKE SA; authenticated peer '3584-
   bit PKCS#1 1.5 RSA with SHA1' signature using preloaded certificate '172.31.32.196'
   004 "custom-db-tunnel" #2: initiator established Child SA using #1; IPsec transport
    [172.31.6.81-172.31.6.81:0-65535 0] -> [172.31.32.196-172.31.32.196:0-65535 0]
    {ESP/ESN=>0xda9c4815 <0xb742ca42 xfrm=AES_GCM_16_256-NONE DPD=passive}
   [root@ip-172-31-6-81 ~]#
   ```

Sauvegarde et restauration d'une instance de base de données Amazon RDS Custom pour Oracle

Comme AmazonRDS, RDS Custom crée et enregistre des sauvegardes automatisées de votre instance de base de données RDS Custom for Oracle pendant la fenêtre de sauvegarde de votre instance de base de données. Vous pouvez également sauvegarder votre instance de base de données manuellement.

La procédure est identique à la prise d'un instantané d'une RDS instance de base de données Amazon. Le premier instantané d'une instance de base de données RDS personnalisée contient les données de l'instance de base de données complète. Les instantanés suivants sont progressifs.

Restaurez les instantanés de base de données à l'aide du AWS Management Console ou du AWS CLI.

Rubriques

- Création d'un instantané RDS personnalisé pour Oracle
- Restauration à partir d'un instantané de base de données RDS personnalisé pour Oracle
- Restauration d'une instance RDS Custom for Oracle à un moment donné
- Suppression d'un instantané RDS personnalisé pour Oracle
- Suppression RDS des sauvegardes automatisées personnalisées pour Oracle

Création d'un instantané RDS personnalisé pour Oracle

RDSCustom for Oracle crée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de l'instance de base de données et pas uniquement des bases de données individuelles. Lorsque votre instance de base de données contient une base de données conteneur (CDB), l'instantané de l'instance inclut la racine CDB et toutPDBs.

Lorsque vous créez un instantané RDS personnalisé pour Oracle, spécifiez l'instance de base de données RDS personnalisée à sauvegarder. Nommez votre instantané afin que vous puissiez restaurer ultérieurement à partir de ce dernier.

Lorsque vous créez un instantané, RDS Custom for Oracle crée un EBS instantané Amazon pour chaque volume attaché à l'instance de base de données. RDSCustom for Oracle utilise l'EBSinstantané du volume racine pour enregistrer une nouvelle Amazon Machine Image (AMI). Pour

que les instantanés soient faciles à associer à une instance de base de données spécifique, ils sont labelisés `DBSnapshotIdentifier`, `DbiResourceId` et `VolumeType`.

La création d'un instantané de base de données entraîne une brève suspension des I/O. Cette suspension peut durer de quelques secondes à quelques minutes, en fonction de la taille et de la classe de votre instance de base de données. Le temps de création d'instantanés varie en fonction de la taille de votre base de données. Étant donné que l'instantané inclut l'intégralité du volume de stockage, la taille de fichiers comme les fichiers temporaires a également une incidence sur le temps nécessaire à la création de l'instantané. Pour en savoir plus sur la création des instantanés, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Créez un instantané RDS personnalisé pour Oracle à l'aide de la console ou du AWS CLI.

Console

Pour créer un instantané RDS personnalisé

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Dans la liste des instances de base de données RDS personnalisées, choisissez l'instance pour laquelle vous souhaitez prendre un instantané.
4. Sous Actions, choisissez Take snapshot (Prendre un instantané).

   La fenêtre Capture d'un instantané DB apparaît.
5. Dans Snapshot name (Nom de l'instantané), saisissez le nom de l'instantané.
6. Choisissez Prendre un instantané.

AWS CLI

Vous créez un instantané d'une instance de base de données RDS personnalisée à l'aide de la create-db-snapshot AWS CLI commande.

Spécifiez les options suivantes :

- `--db-instance-identifier`— Identifie l'instance de base de données RDS personnalisée que vous allez sauvegarder

- `--db-snapshot-identifier`— Nomme votre instantané RDS personnalisé afin que vous puissiez le restaurer ultérieurement

Dans cet exemple, vous créez un instantané de base de données appelé *my-custom-snapshot* pour une instance de base de données RDS personnalisée appelée*my-custom-instance*.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-snapshot \
    --db-instance-identifier my-custom-instance \
    --db-snapshot-identifier my-custom-snapshot
```

Dans Windows:

```
aws rds create-db-snapshot ^
    --db-instance-identifier my-custom-instance ^
    --db-snapshot-identifier my-custom-snapshot
```

Restauration à partir d'un instantané de base de données RDS personnalisé pour Oracle

Lorsque vous restaurez une instance de base de données RDS personnalisée pour Oracle, vous fournissez le nom de l'instantané de base de données et le nom de la nouvelle instance. Vous ne pouvez pas effectuer de restauration à partir d'un instantané vers une instance de base de données RDS personnalisée existante. Une nouvelle instance de base de données RDS personnalisée pour Oracle est créée lors de la restauration.

Le processus de restauration diffère de la restauration sur Amazon des manières suivantes RDS :

- Avant de restaurer un instantané, RDS Custom for Oracle sauvegarde les fichiers de configuration existants. Ces fichiers sont disponibles sur l'instance restaurée dans le répertoire `/rdsdbdata/config/backup`. RDSCustom for Oracle restaure l'instantané de base de données avec les paramètres par défaut et remplace les fichiers de configuration de base de données précédents par les fichiers existants. Par conséquent, l'instance restaurée ne conserve pas les paramètres personnalisés ni les modifications apportées aux fichiers de configuration de la base de données.
- La base de données restaurée porte le même nom que dans l'instantané. Vous ne pouvez pas spécifier un autre nom. (Pour RDS Custom for Oracle, la valeur par défaut est`0RCL`.)

Console

Pour restaurer une instance de base de données RDS personnalisée à partir d'un instantané de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.
5. Sur la page Restaurer l'instance de base de données, pour l'identifiant de l'instance de base de données, entrez le nom de votre instance de base de données RDS personnalisée restaurée.
6. Choisissez Restore DB Instance (Restaurer une instance de base de données).

AWS CLI

Vous restaurez un instantané de base de données RDS personnalisé à l'aide de la commande restore-db-instance-from AWS CLI -db-snapshot.

Si l'instantané à partir duquel vous restaurez est destiné à une instance de base de données privée, assurez-vous de spécifier le `db-subnet-group-name` correct et `no-publicly-accessible`. Sinon, l'instance de base de données est accessible par défaut au public. Les options suivantes sont requises :

- `db-snapshot-identifier` – Identifie l'instantané à partir duquel restaurer
- `db-instance-identifier`— Spécifie le nom de l'instance de base de données RDS personnalisée à créer à partir du snapshot de base de données
- `custom-iam-instance-profile`— Spécifie le profil d'instance associé à l'EC2instance Amazon sous-jacente d'une instance de base de données RDS personnalisée.

Le code suivant restaure l'instantané nommé `my-custom-snapshot` pour `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
```

```
  --db-snapshot-identifier my-custom-snapshot \
  --db-instance-identifier my-custom-instance \
  --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance \
  --no-publicly-accessible
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
  --db-snapshot-identifier my-custom-snapshot ^
  --db-instance-identifier my-custom-instance ^
  --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance ^
  --no-publicly-accessible
```

Restauration d'une instance RDS Custom for Oracle à un moment donné

Vous pouvez restaurer une instance de base de données à un moment précis (PITR), en créant une nouvelle instance de base de données. Pour être prises en chargePITR, la rétention des sauvegardes de vos instances de base de données doit être définie sur une valeur différente de zéro.

L'heure de restauration la plus récente pour une instance de base de données RDS Custom for Oracle dépend de plusieurs facteurs, mais elle se situe généralement à moins de 5 minutes de l'heure actuelle. Pour connaître l'heure de restauration la plus récente pour une instance de base de données, utilisez la AWS CLI describe-db-instancescommande et examinez la valeur renvoyée dans le `LatestRestorableTime` champ correspondant à l'instance de base de données. Pour connaître l'heure de restauration la plus récente pour chaque instance de base de données dans la RDS console Amazon, choisissez Sauvegardes automatisées.

Vous pouvez procéder à une restauration à n'importe quel moment dans le passé au cours de la période de rétention des sauvegardes. Pour connaître l'heure de restauration la plus proche pour chaque instance de base de données, choisissez Sauvegardes automatisées dans la RDS console Amazon.

Pour obtenir des informations générales sur PITR, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Rubriques

- PITRconsidérations relatives à RDS Custom for Oracle

PITRconsidérations relatives à RDS Custom for Oracle

Dans RDS Custom for Oracle, il PITR présente des différences importantes par rapport à Amazon sur PITR les points suivants RDS :

- La base de données restaurée porte le même nom que dans l'instance de base de données source. Vous ne pouvez pas spécifier un autre nom. L'argument par défaut est `ORCL`.
- `AWSRDSCustomIamRolePolicy` nécessite de nouvelles autorisations. Pour de plus amples informations, veuillez consulter Étape 2 : ajouter une politique d'accès à AWSRDSCustom InstanceRoleForRdsCustomInstance.
- Toutes les instances de base de données RDS Custom for Oracle doivent avoir une valeur de rétention des sauvegardes définie sur une valeur différente de zéro.
- Si vous modifiez le système d'exploitation ou le fuseau horaire de l'instance de base de données, PITR cela risque de ne pas fonctionner. Pour plus d'informations sur les changements de fuseaux horaires, consultez Fuseau horaire Oracle.
- Si vous définissez l'automatisation RDS sur`ALL_PAUSED`, Custom suspend le téléchargement des fichiers de journalisation archivés, y compris les journaux créés avant la dernière date de restauration ()LRT. Nous vous recommandons de mettre l'automatisation en pause pendant une courte période.

  À titre d'exemple, supposons que vous vous LRT êtes produit il y a 10 minutes. Vous mettez l'automatisation en pause. Pendant la pause, RDS Custom ne télécharge pas les journaux de restauration archivés. Si votre instance de base de données tombe en panne, vous ne pouvez la récupérer qu'à une époque antérieure à LRT celle qui existait lorsque vous l'avez interrompue. Lorsque vous reprenez l'automatisation, RDS Custom reprend le téléchargement des journaux. Les LRT avancées. PITRLes règles normales s'appliquent.
- Dans RDS Custom, vous pouvez spécifier manuellement un nombre arbitraire d'heures pour conserver les journaux de restauration archivés avant que RDS Custom ne les supprime après le téléchargement. Spécifiez le nombre d'heures comme suit :

  1. Créez un fichier texte nommé `/opt/aws/rdscustomagent/config/redo_logs_custom_configuration.json`.
  2. Ajoutez un JSON objet au format suivant :`{"archivedLogRetentionHours" : "num_of_hours"}`. Le nombre doit être un nombre entier compris entre 1 et 840.
- Supposons que vous branchez un objet non- CDB dans une base de données de conteneurs (CDB) en tant que fichier, PDB puis que vous essayezPITR. L'opération ne réussit que si

vous avez précédemment sauvegardé lePDB. Après avoir créé ou modifié unPDB, nous vous recommandons de toujours le sauvegarder.

- Nous vous recommandons de ne pas personnaliser les paramètres d'initialisation de base de données. Par exemple, la modification des paramètres suivants affecte PITR :
  - `CONTROL_FILE_RECORD_KEEP_TIME` affecte les règles de chargement et de suppression des journaux.
  - `LOG_ARCHIVE_DEST_`*n* ne prend pas en charge plusieurs destinations.
  - `ARCHIVE_LAG_TARGET`affecte la date de restauration la plus récente. `ARCHIVE_LAG_TARGET`est défini sur `300` parce que l'objectif du point de récupération (RPO) est de 5 minutes. Pour atteindre cet objectif, RDS change le journal de journalisation en ligne toutes les 5 minutes et le stocke dans un compartiment Amazon S3. Si la fréquence du changement de journal entraîne un problème de performance pour votre base de données RDS Custom for Oracle, vous pouvez adapter votre instance de base de données et votre stockage à une instance offrant un débit plus élevéIOPS. Si cela est nécessaire pour votre plan de reprise, vous pouvez régler le paramètre d'`ARCHIVE_LAG_TARGET`initialisation sur une valeur comprise entre 60 et 7200.
- Si vous personnalisez les paramètres d'initialisation de la base de données, nous vous recommandons vivement de ne personnaliser que les éléments suivants :
  - `COMPATIBLE`
  - `MAX_STRING_SIZE`
  - `DB_FILES`
  - `UNDO_TABLESPACE`
  - `ENABLE_PLUGGABLE_DATABASE`
  - `CONTROL_FILES`
  - `AUDIT_TRAIL`
  - `AUDIT_TRAIL_DEST`

  Pour tous les autres paramètres d'initialisation, RDS Custom rétablit les valeurs par défaut. Si vous modifiez un paramètre qui ne figure pas dans la liste précédente, cela peut avoir un effet négatif PITR et entraîner des résultats imprévisibles. Par exemple, `CONTROL_FILE_RECORD_KEEP_TIME` affecte les règles de chargement et de suppression des journaux.

Vous pouvez restaurer une instance de base de données RDS personnalisée à un moment donné en utilisant le AWS Management Console, le AWS CLI, ou le RDSAPI.

Console

Pour restaurer une instance de base de données RDS personnalisée à une heure spécifiée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Choisissez l'instance de base de données RDS personnalisée que vous souhaitez restaurer.
4. Sous Actions, sélectionnez Restaurer à un moment donné.

   La fenêtre Restaurer à un instant dans le passé s'affiche.
5. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

   Si vous choisissez Custom (Personnalisé), saisissez la date et l'heure auxquelles vous souhaitez restaurer l'instance.

   Les heures sont affichées dans votre fuseau horaire local, qui est indiqué par un décalage par rapport au temps universel coordonné (UTC). Par exemple, UTC -5 correspond à l'heure normale de l'Est/heure avancée du centre.
6. Pour l'identifiant de l'instance de base de données, entrez le nom de l'instance de base de données RDS personnalisée restaurée cible. Le nom doit être unique.
7. Choisissez d'autres options selon vos besoins, comme la classe d'instance de base de données.
8. Choisissez Restaurer à un instant dans le passé.

AWS CLI

Vous restaurez une instance de base de données à une heure spécifiée en utilisant la point-in-time AWS CLI commande restore-db-instance-to- pour créer une nouvelle instance de base de données RDS personnalisée.

Utilisez l'une des options suivantes pour spécifier la sauvegarde à partir de laquelle effectuer la restauration :

- `--source-db-instance-identifier` *`mysourcedbinstance`*
- `--source-dbi-resource-id` *`dbinstanceresourceID`*
- `--source-db-instance-automated-backups-arn` *`backupARN`*

L'option `custom-iam-instance-profile` est obligatoire.

L'exemple suivant restaure `my-custom-db-instance` vers une nouvelle instance de base de données nommée `my-restored-custom-db-instance` au moment spécifié.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-identifier my-custom-db-instance\
    --target-db-instance-identifier my-restored-custom-db-instance \
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance \
    --restore-time 2022-10-14T23:45:00.000Z
```

Dans Windows:

```
aws rds restore-db-instance-to-point-in-time ^
    --source-db-instance-identifier my-custom-db-instance ^
    --target-db-instance-identifier my-restored-custom-db-instance ^
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance ^
    --restore-time 2022-10-14T23:45:00.000Z
```

Suppression d'un instantané RDS personnalisé pour Oracle

Vous pouvez supprimer des instantanés de base de données gérés par RDS Custom for Oracle lorsque vous n'en avez plus besoin. La procédure de suppression est la même pour les instances de base de données Amazon RDS et RDS Custom.

Les EBS instantanés Amazon des volumes binaire et racine restent dans votre compte plus longtemps car ils peuvent être liés à certaines instances exécutées sur votre compte ou à d'autres instantanés RDS personnalisés pour Oracle. Ces EBS instantanés sont automatiquement supprimés une fois qu'ils ne sont plus liés à aucune ressource RDS personnalisée pour Oracle existante (instances de base de données ou sauvegardes).

Console

Pour supprimer un instantané d'une instance de base de données RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à supprimer.
4. Pour Actions, choisissez Delete snapshot (Supprimer la pile).
5. Dans la page de confirmation, sélectionnez Supprimer.

AWS CLI

Pour supprimer un instantané RDS personnalisé, utilisez la AWS CLI commande delete-db-snapshot.

L'option suivante est requise :

- `--db-snapshot-identifier` – L'instantané à supprimer

L'exemple suivant supprime l'instantané de base de données `my-custom-snapshot`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-snapshot \
  --db-snapshot-identifier my-custom-snapshot
```

Dans Windows:

```
aws rds delete-db-snapshot ^
  --db-snapshot-identifier my-custom-snapshot
```

Suppression RDS des sauvegardes automatisées personnalisées pour Oracle

Vous pouvez supprimer les sauvegardes automatiques conservées pour RDS Custom for Oracle lorsqu'elles ne sont plus nécessaires. La procédure est identique à celle de suppression des RDS sauvegardes Amazon.

Console

Pour supprimer une sauvegarde automatisée conservée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Choisissez Retained (Conservées).
4. Choisissez la sauvegarde automatisée conservée que vous souhaitez supprimer.
5. Pour Actions, choisissez Supprimer.
6. Dans la page de confirmation, entrez **`delete me`** et choisissez Delete (Supprimer).

AWS CLI

Vous pouvez supprimer une sauvegarde automatique conservée à l'aide de la AWS CLI commande delete-db-instance-automated-backup.

L'option suivante est utilisée pour supprimer une sauvegarde automatisée conservée.

- `--dbi-resource-id`— L'identifiant de ressource pour l'instance de base de données RDS personnalisée source.

  Vous pouvez trouver l'identifiant de ressource pour l'instance de base de données source d'une sauvegarde automatique conservée à l'aide de la AWS CLI commande describe-db-instance-automated-backups.

L'exemple suivant supprime la sauvegarde automatisée conservée avec l'identifiant de ressource d'instance de base de données `custom-db-123ABCEXAMPLE`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance-automated-backup \
    --dbi-resource-id custom-db-123ABCEXAMPLE
```

Dans Windows:

```
aws rds delete-db-instance-automated-backup ^
    --dbi-resource-id custom-db-123ABCEXAMPLE
```

Utilisation de groupes d'options dans RDS Custom for Oracle

RDSCustom utilise des groupes d'options pour activer et configurer des fonctionnalités supplémentaires. Un groupe d'options spécifie les fonctionnalités, appelées options, disponibles pour une instance de base de données RDS personnalisée pour Oracle. Les options peuvent avoir des paramètres spécifiant le mode de fonctionnement de l'option. Lorsque vous associez une instance de base de données RDS personnalisée pour Oracle à un groupe d'options, les options et paramètres d'options spécifiés sont activés pour cette instance. Pour obtenir des informations générales sur les groupes d'options sur AmazonRDS, consultezUtilisation de groupes d'options.

Rubriques

- Vue d'ensemble des groupes d'options dans RDS Custom for Oracle
- Fuseau horaire Oracle

Vue d'ensemble des groupes d'options dans RDS Custom for Oracle

Pour activer ces options pour votre base de données Oracle, ajoutez-les à un groupe d'options, puis associez celui-ci à votre instance de base de données. Pour de plus amples informations, veuillez consulter Utilisation de groupes d'options.

Rubriques

- Résumé des options RDS personnalisées pour Oracle
- Étapes de base pour ajouter une option à une instance de base de données RDS personnalisée pour Oracle
- Création d'un groupe d'options pour dans RDS Custom for Oracle
- Associer un groupe d'options à une instance de base de données RDS personnalisée pour Oracle

Résumé des options RDS personnalisées pour Oracle

RDSCustom for Oracle prend en charge les options suivantes pour une instance de base de données.

Option	ID d'option	Description
Fuseau horaire Oracle	`Timezone`	Fuseau horaire utilisé par votre instance de base de

Option	ID d'option	Description
		données RDS personnalisée pour Oracle.

Étapes de base pour ajouter une option à une instance de base de données RDS personnalisée pour Oracle

La procédure générale pour ajouter une option à votre instance de base de données RDS personnalisée pour Oracle est la suivante :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à votre instance de base de données lorsque vous la créez ou la modifiez.

Création d'un groupe d'options pour dans RDS Custom for Oracle

Vous pouvez créer un nouveau groupe d'options qui tire ses paramètres du groupe d'options par défaut. Vous ajoutez ensuite une ou plusieurs options au nouveau groupe d'options. Ou si vous avez déjà un groupe d'options existant, vous pouvez également copier ce groupe avec toutes ses options dans un nouveau groupe d'options. Pour savoir comment copier un groupe d'options, consultezCopie d'un groupe d'options.

Les groupes d'options par défaut pour RDS Custom for Oracle sont les suivants :

- `default:custom-oracle-ee`
- `default:custom-oracle-se2`
- `default:custom-oracle-ee-cdb`
- `default:custom-oracle-se2-cdb`

Lorsque vous créez un groupe d'options, les paramètres sont dérivés du groupe d'options par défaut. Après avoir ajouté l'`TIME_ZONE`option, vous pouvez associer le groupe d'options à votre instance de base de données.

Console

Une manière de créer un groupe d'options consiste à utiliser la AWS Management Console.

Pour créer un nouveau groupe d'options à l'aide de la console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
   a. Dans Nom, saisissez un nom unique au sein de votre AWS compte pour le groupe d'options. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
   b. Pour Description, saisissez une brève description du groupe d'options. La description est utilisée à des fins d'affichage.
   c. Pour Engine, choisissez l'une des options RDS personnalisées pour les moteurs de base de données Oracle suivantes :
      - custom-oracle-ee
      - custom-oracle-se2
      - custom-oracle-ee-cdb
      - custom-oracle-se2 cdb
   d. Pour la version du moteur principal, choisissez une version du moteur principale prise en charge par RDS Custom for Oracle. Pour de plus amples informations, veuillez consulter Régions et moteurs de base de données pris en charge pour RDS Custom pour Oracle.
5. Pour continuer, choisissez Créer. Pour annuler l'opération à la place, choisissez Cancel (Annuler).

AWS CLI

Pour créer un groupe d'options, utilisez la AWS CLI `create-option-group`commande avec les paramètres obligatoires suivants.

- `--option-group-name`
- `--engine-name`

- `--major-engine-version`
- `--option-group-description`

Example

L'exemple suivant crée un groupe d'options nommé `testoptiongroup` qui est associé au moteur de base de données Oracle Enterprise Edition. La description est entre guillemets.

Dans Linux, macOS, ou Unix:

```
aws rds create-option-group \
    --option-group-name testoptiongroup \
    --engine-name custom-oracle-ee-cdb \
    --major-engine-version 19 \
    --option-group-description "Test option group for a Custom Oracle CDB"
```

Dans Windows:

```
aws rds create-option-group ^
    --option-group-name testoptiongroup ^
    --engine-name custom-oracle-ee-cdb ^
    --major-engine-version 19 ^
    --option-group-description "Test option group for a Custom Oracle CDB"
```

RDS API

Pour créer un groupe d'options, appelez l'RDSAPI`CreateOptionGroup`opération Amazon.

Associer un groupe d'options à une instance de base de données RDS personnalisée pour Oracle

Vous pouvez associer votre groupe d'options à une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous créez l'instance. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données RDS Custom for Oracle.
- Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification de votre instance de base de données RDS personnalisée pour Oracle.

Fuseau horaire Oracle

Pour modifier le fuseau horaire du système utilisé par votre instance de base de données RDS personnalisée pour Oracle, utilisez l'option de fuseau horaire. Par exemple, vous devrez peut-être modifier le fuseau horaire d'une instance de base de données afin qu'elle soit compatible avec un environnement sur site ou une application héritée. L'option de fuseau horaire change le fuseau horaire au niveau de l'hôte. La modification du fuseau horaire impacte toutes les valeurs et colonnes date, y compris `SYSDATE` et `SYSTIMESTAMP`.

Rubriques

- Paramètres des options de fuseau horaire dans RDS Custom for Oracle
- Fuseaux horaires disponibles dans RDS Custom for Oracle
- Considérations relatives à la définition du fuseau horaire dans RDS Custom for Oracle
- Limitations relatives au réglage du fuseau horaire dans RDS Custom for Oracle
- Ajouter l'option de fuseau horaire à un groupe d'options
- Suppression de l'option de fuseau horaire

Paramètres des options de fuseau horaire dans RDS Custom for Oracle

Amazon RDS prend en charge les paramètres suivants pour l'option de fuseau horaire.

Paramètre d'option	Valeurs valides	Description
`TIME_ZONE`	Un des fuseaux horaires disponibles. Pour obtenir la liste complète, consultez Fuseaux horaires disponibles dans RDS Custom for Oracle.	Nouveau fuseau horaire de votre instance de base de données.

Fuseaux horaires disponibles dans RDS Custom for Oracle

Vous pouvez utiliser les valeurs suivantes pour l'option de fuseau horaire.

disponibilité	Fuseau horaire
Afrique	Afrique/le Caire, Afrique/Casablanca, Afrique/Harare, Afrique/Lagos, Afrique/Luanda, Afrique/Monrovia, Afrique/Nairobi, Afrique/Tripoli, Afrique/Windhoek
Amérique	Amérique/Araguaina, Amérique/Argentine/Buenos_Aires, Amérique/Asuncion, Amérique/Bogota, Amérique/Caracas, Amérique/Chicago, Amérique/Chihuahua, Amérique/Cuiaba, Amérique/Denver, Amérique/Detroit, Amérique/Fortaleza, Amérique/Godthab, Amérique/Guatemala, Amérique/Halifax, Amérique/Lima, Amérique/Los_Angeles, Amérique/Manaus, Amérique/Matamoros, Amérique/Mexico_City, Amérique/Monterrey, Amérique/Montevideo, Amérique/New_York, Amérique/Phoenix, AmériquI/Oantiago, AmériquI/Oao_Paulo, Amérique/Tijuana, Amérique/Toronto
Asie	Asie/Amman, Asie/Achgabat, Asie/Bagdad, Asie/Bakou, Asie/Bangkok, Asie/Beyrouth, Asie/Calcutta, Asie/Damas, Asie/Dhaka, Asie/Hong_Kong, Asie/Irkoutsk, Asie/Jakarta, Asie/Jérusalem, Asie/Kaboul, Asie/Karachi, Asie/Katmandou, Asie/Kolkata, Asie/Krasnoïarsk, Asie/Magadan, Asie/Manille, Asie/Muscat, Asie/Novosibirsk, Asie/Rangoon, Asie/Riyad, AsiI/Oéoul, AsiI/Ohanghai, AsiI/Oingapour, Asie/Taipei, Asie/Téhéran, Asie/Tokyo, Asie/Oulan_Bator, Asie/Vladivostok, Asie/Iakoutsk, Asie/Yerevan
Atlantique	Atlantique/Açores, Atlantic/Cap_Vert
Australie	Australie/Adelaide, Australie/Brisbane, Australie/Darwin, Australie/Eucla, Australie/Hobart, Australie/Lord_Howe, Australie/Perth, AustraliI/Oydney
Brésil	Brésil/, Brésil/Est DeNoronha
Canada	Canada/Terre-Neuve, Canada/Saskatchewan
Etc	Etc/ -3 GMT
Europe	Europe/Amsterdam, Europe/Athènes, Europe/Berlin, Europe/Dublin, Europe/Helsinki, Europe/Kaliningrad, Europe/Londres, Europe/Madrid, Europe/Moscou, Europe/Paris, Europe/Prague, Europe/Rome, EuropI/Oarajevo

disponibilité	Fuseau horaire
Pacifique	Pacifique/Apia, Pacifique/Auckland, Pacifique/Chatham, Pacifique/Fidji, Pacifique/Guam, Pacifique/Honolulu, Pacifique/Kiritimati, Pacifique/Marquises, PacifiquI/Oamoa, Pacifique/Tongatapu, Pacifique/Wake
ETATS-UNIS	États-Unis/Alaska, États-Unis/Centre, États-Unis/Indiana Est, États-Unis/Est, États-Unis/Pacifique
UTC	UTC

Considérations relatives à la définition du fuseau horaire dans RDS Custom for Oracle

Si vous choisissez de définir le fuseau horaire de votre instance de base de données, tenez compte des points suivants :

- Lorsque vous ajoutez l'option de fuseau horaire, une brève interruption de service se produit pendant le redémarrage automatique de votre instance de base de données.
- Si vous définissez accidentellement le fuseau horaire de manière incorrecte, vous devez rétablir le paramètre de fuseau horaire précédent de votre instance de base de données. C'est pourquoi nous vous conseillons vivement d'utiliser l'une des stratégies suivantes avant d'ajouter l'option de fuseau horaire à votre instance :
  - Si votre instance de base de données RDS personnalisée pour Oracle utilise le groupe d'options par défaut, prenez un instantané de votre instance de base de données. Pour de plus amples informations, veuillez consulter Création d'un instantané RDS personnalisé pour Oracle.
  - Si votre instance de base de données utilise actuellement un groupe d'options autre que celui par défaut, prenez un instantané de votre instance de base de données, puis créez un nouveau groupe d'options avec l'option de fuseau horaire.
- Nous vous recommandons vivement de sauvegarder votre instance de base de données manuellement après avoir appliqué l'`Timezone`option.
- Nous vous recommandons vivement de tester l'option de fuseau horaire sur une instance de base de données de test avant de l'ajouter à une instance de base de données de production. L'ajout de l'option de fuseau horaire peut entraîner des problèmes avec les tables qui utilisent la date système pour ajouter des dates ou des heures. Nous vous recommandons d'analyser vos données et applications pour évaluer l'incidence du changement de fuseau horaire.

Limitations relatives au réglage du fuseau horaire dans RDS Custom for Oracle

Prenez en compte les limitations suivantes :

- Vous ne pouvez pas modifier votre fuseau horaire directement sur votre hôte sans le déplacer en dehors du périmètre de support. Pour modifier le fuseau horaire de votre base de données, vous devez créer un groupe d'options.
- Comme l'option de fuseau horaire est une option persistante (mais pas une option permanente), vous ne pouvez pas effectuer les opérations suivantes :
  - Supprimer l'option d'un groupe d'options après l'avoir ajoutée
  - Remplacer le paramètre de fuseau horaire de l'option par un autre fuseau horaire
- Vous ne pouvez pas associer plusieurs groupes d'options à votre instance de base de données RDS personnalisée pour Oracle.
- Vous ne pouvez pas définir le fuseau horaire d'une personne PDBs dans unCDB.

Ajouter l'option de fuseau horaire à un groupe d'options

Les groupes d'options par défaut pour RDS Custom for Oracle sont les suivants :

- `default:custom-oracle-ee`
- `default:custom-oracle-se2`
- `default:custom-oracle-ee-cdb`
- `default:custom-oracle-se2-cdb`

Lorsque vous créez un groupe d'options, les paramètres sont dérivés du groupe d'options par défaut. Pour obtenir des informations générales sur les groupes d'options sur AmazonRDS, consultezUtilisation de groupes d'options.

Console

Pour ajouter l'option de fuseau horaire à un groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Cochez la case pour le groupe d'options que vous souhaitez modifier, puis choisissez Ajouter une option.

4. Dans la fenêtre Ajouter une option, procédez comme suit :

   a. Choisissez le fuseau horaire.

   b. Dans Paramètres des options, choisissez un fuseau horaire.

   c. Pour activer l'option sur toutes les instances de base de données RDS personnalisées associées pour Oracle dès que vous l'ajoutez, pour Appliquer immédiatement, sélectionnez Oui. Si vous choisissez Non (valeur par défaut), l'option est activée pour chaque instance de base de données associée lors de la fenêtre de maintenance suivante.

   d.

   > **Important**
   >
   > Si vous ajoutez l'option de fuseau horaire à un groupe d'options existant qui est déjà attaché à une ou plusieurs instances de bases de données, une brève interruption de service a lieu pendant le redémarrage automatique de toutes les instances de bases de données.

5. Lorsque les paramètres vous conviennent, choisissez Ajouter une option.

6. Sauvegardez les instances de base de données RDS personnalisées pour Oracle dont les fuseaux horaires ont été mis à jour. Pour de plus amples informations, veuillez consulter Création d'un instantané RDS personnalisé pour Oracle.

AWS CLI

L'exemple suivant utilise la commande AWS CLI add-option-to-option-group pour ajouter l'`Timezone`option et le paramètre d'`TIME_ZONE`option à un groupe d'options appelé`testoptiongroup`. Le fuseau horaire par défaut est défini sur `America/Los_Angeles`.

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name "testoptiongroup" \
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=America/
Los_Angeles}]" \
    --apply-immediately
```

Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name "testoptiongroup" ^
```

```
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=America/
Los_Angeles}]" ^
    --apply-immediately
```

Suppression de l'option de fuseau horaire

L'option de fuseau horaire est une option persistante, mais pas une option permanente. Vous ne pouvez pas supprimer l'option d'un groupe d'options après l'avoir ajoutée. Pour dissocier l'ancien groupe d'options de votre instance de base de données :

1. Créez un nouveau groupe d'options avec une `Timezone` option mise à jour.
2. Associez le nouveau groupe d'options à votre instance de base de données lorsque vous modifiez l'instance.

Migration d'une base de données locale vers RDS Custom for Oracle

Avant de migrer une base de données Oracle locale vers RDS Custom for Oracle, tenez compte des facteurs suivants :

- La durée de temps d'arrêt que l'application peut se permettre
- La taille de la base de données source
- La connectivité réseau
- La nécessité d'un plan de secours
- La version de base de données Oracle source et cible et les types de système d'exploitation d'instance de base de données
- Outils de réplication disponibles, tels que AWS Database Migration Service GoldenGate, Oracle ou des outils de réplication tiers

En fonction de ces facteurs, vous pouvez choisir une migration physique, une migration logique ou une combinaison des deux. Si vous choisissez la migration physique, vous pouvez utiliser les techniques suivantes :

RMANduplication

La duplication active de la base de données ne nécessite pas de sauvegarde de votre base de données source. Elle duplique la base de données source active vers l'hôte de destination en copiant les fichiers de base de données via le réseau vers l'instance auxiliaire. La RMAN `DUPLICATE` commande copie les fichiers requis sous forme de copies d'images ou de jeux de sauvegarde. Pour découvrir cette technique, consultez le billet de AWS blog Migration physique des bases de données Oracle vers Amazon RDS Custom à l'aide de la RMAN duplication.

Oracle Data Guard

Cette technique consiste à sauvegarder une base de données sur site principale et à copier les sauvegardes vers un compartiment Amazon S3. Vous copiez ensuite les sauvegardes RDS sur votre instance de base de données de secours personnalisée pour Oracle. Après avoir effectué la configuration nécessaire, vous passez manuellement de votre base de données principale à votre base de données de secours RDS personnalisée pour Oracle. Pour découvrir cette technique, consultez le billet de AWS blog Migration physique des bases de données Oracle vers Amazon RDS Custom à l'aide de Data Guard.

Pour des informations générales sur l'importation logique de données dans RDS for Oracle, consultezImportation de données dans Oracle sur Amazon RDS.

Mise à niveau d'une instance de base de données pour Amazon RDS Custom for Oracle

Vous pouvez mettre à niveau une instance de base de données Amazon RDS Custom en la modifiant pour utiliser une nouvelle version de moteur personnalisée (CEV). Pour des informations générales sur les mises à niveau, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Rubriques

- Vue d'ensemble des mises à niveau dans RDS Custom for Oracle
- Exigences relatives aux mises à niveau RDS personnalisées pour Oracle
- Considérations relatives aux mises à niveau des bases de données RDS Custom for Oracle
- Considérations relatives aux mises à niveau RDS personnalisées pour le système d'exploitation Oracle
- Affichage des cibles de CEV mise à niveau valides pour les instances de base de données RDS Custom for Oracle
- Mise à niveau d'une instance de base de données RDS personnalisée pour Oracle
- Affichage des mises à niveau de base de données en attente pour RDS les instances de base de
- Résolution d'un échec de mise à niveau pour une instance de base de données RDS personnalisée pour Oracle

Vue d'ensemble des mises à niveau dans RDS Custom for Oracle

Avec RDS Custom for Oracle, vous pouvez appliquer des correctifs à votre base de données Oracle ou à votre système d'exploitation (OS) d'instance de base de données en créant une nouvelle instance, CEVs puis en modifiant votre instance pour qu'elle utilise la nouvelleCEV.

Rubriques

- CEVoptions de mise à niveau
- Appliquer des correctifs sans CEVs
- Étapes générales pour appliquer des correctifs à votre instance de base de données avec un CEV

CEVoptions de mise à niveau

Lorsque vous créez un CEV pour une mise à niveau, les options suivantes s'excluent mutuellement :

Base de données uniquement

Réutilisez l'Amazon Machine Image (AMI) actuellement utilisée par votre instance de base de données, mais spécifiez des binaires de base de données différents. RDSCustom alloue un nouveau volume binaire, puis l'attache à l'instance Amazon EC2 existante. RDSCustom remplace l'intégralité du volume de base de données par un nouveau volume utilisant la version de votre base de données cible.

Système d'exploitation uniquement

Réutilisez les fichiers binaires de base de données actuellement utilisés par votre instance de base de données, mais spécifiez-en un autreAMI. RDSCustom alloue une nouvelle EC2 instance Amazon, puis attache le volume binaire existant à la nouvelle instance. Le volume de base de données existant est conservé.

Si vous souhaitez mettre à niveau à la fois le système d'exploitation et la base de données, vous devez les mettre à niveau CEV deux fois. Vous pouvez mettre à niveau le système d'exploitation puis la base de données, ou inversement.

> **Warning**
>
> Lorsque vous appliquez un correctif à votre système d'exploitation, vous perdez les données de votre volume racine ainsi que toute personnalisation existante du système d'exploitation. Par conséquent, nous vous recommandons vivement de ne pas utiliser le volume racine pour les installations ou pour le stockage de données ou de fichiers permanents. Nous vous recommandons également de sauvegarder vos données avant la mise à niveau.

Appliquer des correctifs sans CEVs

Nous vous recommandons vivement de mettre à niveau votre instance de base de données RDS personnalisée pour Oracle à l'aide deCEVs. RDSCustom for Oracle Automation synchronise les métadonnées du correctif avec le binaire de base de données de votre instance de base de données.

Dans des circonstances particulières, RDS Custom prend en charge l'application d'un correctif de base de données « unique » directement à l'EC2instance Amazon sous-jacente à l'aide de l'OPatchutilitaire. Un cas d'utilisation valide peut être un correctif de base de données que vous souhaitez appliquer immédiatement, mais l'équipe RDS personnalisée met à niveau la

CEV fonctionnalité, ce qui entraîne un retard. Pour appliquer un correctif de base de données manuellement, procédez comme suit :

1. Suspendez l'automatisation RDS personnalisée.
2. Appliquez votre correctif aux fichiers binaires de base de données de l'EC2instance Amazon.
3. Reprendre l'automatisation RDS personnalisée.

L'inconvénient de la technique précédente est que vous devez appliquer le correctif de base de données manuellement à chaque instance que vous souhaitez mettre à niveau. En revanche, lorsque vous en créez une nouvelleCEV, vous pouvez créer ou mettre à niveau plusieurs instances de base de données en utilisant la mêmeCEV.

Étapes générales pour appliquer des correctifs à votre instance de base de données avec un CEV

Effectuez les étapes de base suivantes, que vous appliquiez des correctifs au système d'exploitation ou à votre base de données :

1. Créez un fichier CEV contenant l'un des éléments suivants, selon que vous appliquez des correctifs à la base de données ou au système d'exploitation :
   - La révision de mise à jour de base de données Oracle que vous souhaitez appliquer à votre instance de base de données
   - Un fichier différent AMI (soit le dernier disponible, soit celui que vous spécifiez) et un existant CEV à utiliser comme source

   Suivez les étapes de Création d’un CEV.
2. (Facultatif pour l'application de correctifs à la base de données) Vérifiez les mises à niveau de version du moteur disponibles en exécutant `describe-db-engine-versions`.
3. Lancez le processus d'application des correctifs en exécutant `modify-db-instance`.

   L'état de l'instance à laquelle le correctif est appliqué diffère comme suit :
   - Lors de RDS l'application de correctifs à la base de données, le statut de l'instance de base de données passe à Mise à niveau.
   - Pendant RDS l'application de correctifs au système d'exploitation, le statut de l'instance de base de données passe à Modifying.

   Lorsque l'instance de base de données a le statut Disponible, l'application des correctifs est terminée.

4. Vérifiez que votre instance de base de données utilise le nouveau CEV en exécutant`describe-db-instances`.

Exigences relatives aux mises à niveau RDS personnalisées pour Oracle

Lorsque vous mettez à niveau votre instance de base de données RDS personnalisée pour Oracle vers une cibleCEV, assurez-vous de répondre aux exigences suivantes :

- La cible CEV vers laquelle vous effectuez la mise à niveau doit exister.
- Vous devez mettre à niveau le système d'exploitation ou la base de données au cours d'une seule opération. La mise à niveau du système d'exploitation et de la base de données en un seul API appel n'est pas prise en charge.
- La cible CEV doit utiliser les paramètres d'installation qui figurent dans le manifeste du courantCEV. Par exemple, vous ne pouvez pas mettre à niveau une base de données qui utilise le répertoire d'origine Oracle Home par défaut vers un CEV autre répertoire Oracle Home par défaut.
- Pour les mises à niveau de base de données, la cible CEV doit utiliser une nouvelle version mineure de base de données, et non une nouvelle version majeure. Par exemple, vous ne pouvez pas passer d'une base de données Oracle 12c CEV à une base de données Oracle CEV 19c. En revanche, vous pouvez mettre à niveau la version 21.0.0.0.ru-2023-04.rur-2023-04.r1 vers la version 21.0.0.0.ru-2023-07.rur-2023-07.r1.
- Pour les mises à niveau du système d'exploitation, la cible CEV doit utiliser une version majeure différente AMI mais avoir la même version majeure.

Considérations relatives aux mises à niveau des bases de données RDS Custom for Oracle

Si vous envisagez de mettre à niveau votre base de données, tenez compte des points suivants :

- La version du système d'exploitation actuellement prise en charge est Oracle Linux 8. Pour continuer à recevoir les dernières mises à jour de sécurité et correctifs de la part de RDS Custom for Oracle, mettez à niveau vos instances de base de données vers Oracle Linux 8 en spécifiant un CEV basé sur ce système d'exploitation. Oracle Database 12c version 1 (12.1), Oracle Database version 2 (12.2) et Oracle Database 19c sont les seules versions compatibles avec Oracle Linux 8. Pour migrer vers la dernière AMI Oracle Linux 8, mettez à niveau votre système d'exploitation vers la dernière AMI. Pour de plus amples informations, veuillez consulter Mise à niveau d'une instance de base de données RDS personnalisée pour Oracle.

Le support d'Oracle Linux 7.9 a pris fin le 31 décembre 2024. Pour continuer à exécuter Oracle Linux 7 après la fin du support, achetez une licence Oracle Extended Support. Vous êtes responsable des mises à jour de sécurité et devez corriger manuellement vos instances RDS Custom for Oracle. Pour plus d'informations, voir Politique de support à vie : couverture des offres de services Open Source d'Oracle.

- Lorsque vous mettez à niveau les fichiers binaires de base de données dans votre instance de base de données principale, RDS Custom for Oracle met automatiquement à niveau vos réplicas en lecture. Cependant, lorsque vous mettez à niveau le système d'exploitation, vous devez mettre à niveau les répliques de lecture manuellement.
- Lorsque vous mettez à niveau une base de données de conteneurs (CDB) vers une nouvelle version de base de données, RDS Custom for Oracle vérifie que toutes PDBs sont ouvertes ou pourraient être ouvertes. Si ces conditions ne sont pas remplies, RDS Custom arrête la vérification et rétablit l'état d'origine de la base de données sans tenter de procéder à la mise à niveau. Si les conditions sont remplies, RDS Custom applique d'abord des correctifs à la racine de la base de données, puis applique des correctifs à toutes les autres PDBs (y compris`PDB$SEED`) en parallèle.

  Une fois le correctif terminé, RDS Custom tente de tout ouvrir. PDBs Si l'un d'entre eux PDBs ne s'ouvre pas, vous recevez l'événement suivant :`The following PDBs failed to open: list-of-PDBs`. Si RDS Custom ne parvient pas à appliquer un correctif à la racine de la CDB ou à une autre PDBs, l'instance est mise dans l'`PATCH_DB_FAILED`état.
- Vous souhaiterez peut-être effectuer simultanément une mise à niveau de version majeure de base de données et une conversion d'une base de données non-CDB en CDB. Dans ce cas, nous vous recommandons de procéder comme suit :

  1. Créez une nouvelle instance de base de données RDS Custom pour Oracle qui utilise l'architecture mutualisée Oracle.
  2. Connectez une base de données non-CDB à votre racine CDB, en la créant en tant que PDB. Veillez à ce que la base de données non-CDB ait la même version majeure que votre CDB.
  3. Convertissez votre PDB en exécutant le script `noncdb_to_pdb.sql` Oracle SQL.
  4. Validez votre instance de CDB.
  5. Mettez à niveau votre instance de CDB.

Considérations relatives aux mises à niveau RDS personnalisées pour le système d'exploitation Oracle

Lorsque vous planifiez une mise à niveau du système d'exploitation, tenez compte des points suivants :

- Vous ne pouvez pas fournir le vôtre AMI pour une utilisation dans un RDS Custom for OracleCEV. Vous pouvez spécifier soit la valeur par défautAMI, qui utilise Oracle Linux 8, soit une valeur AMI qui a déjà été utilisée par un RDS Custom for OracleCEV.

  > **Note**
  >
  > RDSCustom for Oracle publie une nouvelle valeur par défaut AMI lorsque des vulnérabilités et des risques courants sont découverts. Aucun horaire fixe n'est disponible ou garanti. RDS Custom for Oracle a tendance à publier une nouvelle valeur par défaut AMI tous les 30 jours.

- Lorsque vous mettez à niveau le système d'exploitation de votre instance de base de données principale, vous devez mettre à niveau manuellement les répliques de lecture associées.
- Réservez une capacité de EC2 calcul Amazon suffisante pour votre type d'instance dans votre AZ avant de commencer à appliquer des correctifs au système d'exploitation.

  Lorsque vous créez une réservation de capacité, vous spécifiez la zone de disponibilité, le nombre d'instances et les attributs d'instance (y compris le type d'instance). Par exemple, si votre instance de base de données utilise le type d'EC2instance sous-jacent r5.large, nous vous recommandons de réserver de la EC2 capacité pour r5.large dans votre AZ. Lors de l'application de correctifs au système d'exploitation, RDS Custom crée un nouvel hôte de type db.r5.large, qui peut rester bloqué si l'AZ ne dispose pas de la EC2 capacité requise pour ce type d'instance. Si vous réservez de EC2 la capacité, vous réduisez le risque de blocage des correctifs en raison de contraintes de capacité. Pour plus d'informations, consultez la section Réservations de capacité à la demande dans le guide de EC2 l'utilisateur Amazon.
- Sauvegardez votre instance de base de données avant de mettre à niveau son système d'exploitation. La mise à niveau supprime les données de votre volume racine et toutes les personnalisations existantes du système d'exploitation.
- Dans le modèle de responsabilité partagée, vous êtes responsable de la mise à jour de votre système d'exploitation. RDS Custom for Oracle n'impose pas les correctifs que vous devez

appliquer à votre système d'exploitation. Si votre RDS Custom for Oracle est fonctionnel, vous pouvez utiliser ce qui y est AMI associé CEV indéfiniment.

Affichage des cibles de CEV mise à niveau valides pour les instances de base de données RDS Custom for Oracle

Vous pouvez voir les versions existantes CEVs sur la page des versions personnalisées du moteur dans le AWS Management Console.

Vous pouvez également utiliser la describe-db-engine-versions AWS CLI commande pour déterminer la validité CEVs à utiliser lors de la mise à niveau de vos instances de base de données, comme illustré dans l'exemple suivant. Cet exemple suppose que vous avez créé une instance de base de données à l'aide de la version de moteur `19.my_cev1` et que les versions de mise à niveau `19.my_cev2` et `19.my_cev` existent.

```
aws rds describe-db-engine-versions --engine custom-oracle-ee --engine-version
 19.my_cev1
```

La sortie se présente comme suit : Le `ImageId` champ indique l'AMIID.

```
{
    "DBEngineVersions": [
        {
            "Engine": "custom-oracle-ee",
            "EngineVersion": "19.my_cev1",
            ...
            "Image": {
                "ImageId": "ami-2345",
                "Status": "active"
            },
            "DBEngineVersionArn": "arn:aws:rds:us-west-2:123456789012:cev:custom-
oracle-ee/19.my_cev1/12a34b5c-67d8-90e1-2f34-gh56ijk78lm9"
            "ValidUpgradeTarget": [
                {
                    "Engine": "custom-oracle-ee",
                    "EngineVersion": "19.my_cev2",
                    "Description": "19.my_cev2 description",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                {
```

```
                "Engine": "custom-oracle-ee",
                "EngineVersion": "19.my_cev3",
                "Description": "19.my_cev3 description",
                "AutoUpgrade": false,
                "IsMajorVersionUpgrade": false
            }
        ]
        ...
```

Mise à niveau d'une instance de base de données RDS personnalisée pour Oracle

Pour mettre à niveau votre instance de base de données RDS personnalisée pour Oracle, modifiez-la pour en utiliser une nouvelleCEV. Cela CEV peut contenir de nouveaux fichiers binaires de base de données ou un nouveauAMI. Par exemple, pour mettre à niveau votre instance de base de données Oracle Linux 7.9 vers Oracle Linux 8, spécifiez la dernière versionAMI, qui utilise Oracle Linux 8. Pour mettre à niveau la base de données et le système d'exploitation, vous devez effectuer deux mises à niveau distinctes.

> **Note**
>
> Si vous mettez à niveau la base de données, RDS Custom met automatiquement à niveau les répliques de lecture après la mise à niveau de l'instance de base de données principale. Si vous mettez à niveau le système d'exploitation, vous devez mettre à niveau les réplicas manuellement.

Avant de commencer, consultez Exigences relatives aux mises à niveau RDS personnalisées pour Oracle et Considérations relatives aux mises à niveau des bases de données RDS Custom for Oracle.

Console

Pour mettre à niveau une instance de base de données RDS personnalisée pour Oracle

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée pour Oracle que vous souhaitez mettre à niveau.
3. Sélectionnez Modifier. La page Modifier l'instance de base de données s'affiche.

4. Pour la version du moteur de base de données, choisissez-en une nouvelleCEV. Procédez comme suit :

   - Si vous appliquez des correctifs à la base de données, assurez-vous que les CEV fichiers binaires de base de données sont différents de ceux utilisés par votre instance de base de données et qu'ils ne spécifient aucun autre AMI que ceux AMI actuellement utilisés par votre instance de base de données.
   - Si vous appliquez des correctifs au système d'exploitation, assurez-vous qu'il CEV spécifie un fichier différent de AMI celui AMI actuellement utilisé par votre instance de base de données et qu'il ne spécifie pas de fichiers binaires de base de données différents.

   > **Warning**
   >
   > Lorsque vous appliquez un correctif à votre système d'exploitation, vous perdez les données de votre volume racine ainsi que toute personnalisation existante du système d'exploitation.

5. Sélectionnez Continuer pour consulter le récapitulatif des modifications.

   Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement.

6. Si vos modifications sont correctes, sélectionnez Modifier l'instance de base de données. Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

AWS CLI

Les exemples suivants montrent des scénarios de mise à niveau possibles. Les exemples supposent que vous avez créé une instance de base de données RDS personnalisée pour Oracle présentant les caractéristiques suivantes :

- Instance de base de données nommée `my-custom-instance`
- CEVnommé `19.my_cev1`
- Oracle Database 19c utilisant la non-architecture CDB
- Oracle Linux 8 utilisant AMI `ami-1234`

Le dernier service fourni estAMI. `ami-2345` Vous pouvez le trouver AMIs en exécutant la CLI commande`describe-db-engine-versions`.

Rubriques

- Mise à niveau du système d'exploitation
- Mise à niveau de la base de données

Mise à niveau du système d'exploitation

Dans cet exemple, vous souhaitez effectuer une mise `ami-1234` à `ami-2345` niveau vers le dernier service fourni. AMI Comme vous mettez à niveau le système d'exploitation, les fichiers binaires de base de données pour `ami-1234` et `ami-2345` doivent être identiques. Vous créez un nouveau CEV nom `19.my_cev2` en fonction de`19.my_cev1`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-custom-db-engine-version \
    --engine custom-oracle-ee \
    --engine-version 19.my_cev2 \
    --description "Non-CDB CEV based on ami-2345" \
    --kms-key-id key-name \
    --source-custom-db-engine-version-identifer arn:aws:rds:us-
west-2:123456789012:cev:custom-oracle-ee/19.my_cev1/12345678-ab12-1234-cde1-
abcde123456789 \
    --image-id ami-2345
```

Dans Windows:

```
aws rds create-custom-db-engine-version ^
    --engine custom-oracle-ee ^
    --engine-version 19.my_cev2 ^
    --description "Non-CDB CEV based on ami-2345" ^
    --kms-key-id key-name ^
    --source-custom-db-engine-version-identifer arn:aws:rds:us-
west-2:123456789012:cev:custom-oracle-ee/19.my_cev1/12345678-ab12-1234-cde1-
abcde123456789 ^
    --image-id ami-2345
```

Pour mettre à niveau une instance de base de données RDS personnalisée, utilisez la modify-db-instance AWS CLI commande avec les paramètres suivants :

- `--db-instance-identifier`— Spécifiez l'instance de base de données RDS personnalisée pour Oracle à mettre à niveau.
- `--engine-version`— Spécifiez celui CEV qui contient le nouveauAMI.
- `--no-apply-immediately` | `--apply-immediately` – Indiquez s'il faut effectuer la mise à niveau immédiatement ou attendre la fenêtre de maintenance planifiée.

Dans l'exemple suivant, l'instance `my-custom-instance` est mise à niveau vers la version `19.my_cev2`. Seul le système d'exploitation est mis à niveau.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --engine-version 19.my_cev2 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --engine-version 19.my_cev2 ^
    --apply-immediately
```

Mise à niveau de la base de données

Dans cet exemple, vous souhaitez appliquer le correctif Oracle p35042068 à votre instance de base RDS de données Oracle. Comme vous avez mis à niveau votre système d'exploitation dans Mise à niveau du système d'exploitation, votre instance de base de données utilise actuellement `19.my_cev2`, qui est basé sur `ami-2345`. Vous créez un nouveau CEV nom `19.my_cev3` qui utilise également`ami-2345`, mais vous spécifiez un nouveau JSON manifeste dans la variable d'`$MANIFEST`environnement. Ainsi, seuls les fichiers binaires de base de données sont différents dans votre nouvelle instance CEV et CEV ceux que votre instance utilise actuellement.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-custom-db-engine-version \
    --engine custom-oracle-ee \
    --engine-version 19.my_cev3 \
    --description "Non-CDB CEV with p35042068 based on ami-2345" \
    --kms-key-id key-name \
    --image-id ami-2345 \
    --manifest $MANIFEST
```

Dans Windows:

```
aws rds create-custom-db-engine-version ^
    --engine custom-oracle-ee ^
    --engine-version 19.my_cev3 ^
    --description "Non-CDB CEV with p35042068 based on ami-2345" ^
    --kms-key-id key-name ^
    --image-id ami-2345 ^
    --manifest $MANIFEST
```

Dans l'exemple suivant, l'instance `my-custom-instance` est mise à niveau vers la version de moteur `19.my_cev3`. Seule la base de données est mise à niveau.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --engine-version 19.my_cev3 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --engine-version 19.my_cev3 ^
    --apply-immediately
```

Affichage des mises à niveau de base de données en attente pour RDS les instances de base de

Vous pouvez voir les mises à niveau de base de données en attente pour vos instances de base de données Amazon RDS Custom à l'aide de la describe-pending-maintenance-actions AWS CLI commande describe-db-instancesor.

Notez toutefois que cette méthode ne fonctionne pas si vous avez utilisé l'option `--apply-immediately` ou si la mise à niveau est en cours.

La commande `describe-db-instances` suivante affiche les mises à niveau de base de données en attente pour `my-custom-instance`.

```
aws rds describe-db-instances --db-instance-identifier my-custom-instance
```

La sortie se présente comme suit :

```
{
    "DBInstances": [
        {
           "DBInstanceIdentifier": "my-custom-instance",
            "EngineVersion": "19.my_cev1",
            ...
            "PendingModifiedValues": {
                "EngineVersion": "19.my_cev3"
            ...
            }
        }
    ]
}
```

Résolution d'un échec de mise à niveau pour une instance de base de données RDS personnalisée pour Oracle

Si la mise à niveau d'une instance de base de données RDS personnalisée échoue, un RDS événement est généré et le statut de l'instance de base de données devient le mêmeupgrade-failed.

Vous pouvez voir cet état à l'aide de la describe-db-instances AWS CLI commande, comme illustré dans l'exemple suivant.

```
aws rds describe-db-instances --db-instance-identifier my-custom-instance
```

La sortie se présente comme suit :

```
{
    "DBInstances": [
        {
           "DBInstanceIdentifier": "my-custom-instance",
            "EngineVersion": "19.my_cev1",
            ...
            "PendingModifiedValues": {
                "EngineVersion": "19.my_cev3"
            ...
            }
            "DBInstanceStatus": "upgrade-failed"
        }
    ]
}
```

Après un échec de mise à niveau, toutes les actions de base de données sont bloquées, à l'exception de la modification de l'instance de base de données pour effectuer les tâches suivantes :

- Nouvelle tentative d'exécution de la même mise à niveau
- Suspension et reprise de l'automatisation personnalisée RDS
- Point-in-time récupération (PITR)
- Suppression d'une instance de base de données

> **Note**
>
> Si l'automatisation a été suspendue pour l'instance de base de données RDS personnalisée, vous ne pouvez pas réessayer la mise à niveau tant que vous n'avez pas repris l'automatisation.
> Les mêmes actions s'appliquent en cas d'échec de mise à niveau d'un réplica en RDS lecture géré que pour le réplica principal.

Pour de plus amples informations, veuillez consulter Dépannage des mises à niveau de RDS Custom for Oracle.

Résolution des problèmes de base de données pour Amazon RDS Custom for Oracle

Le modèle de responsabilité partagée de RDS Custom fournit un accès au niveau du shell du système d'exploitation et un accès administrateur de base de données. RDS Custom exécute les ressources de votre compte, contrairement à Amazon RDS qui exécute les ressources d'un compte système. Un meilleur accès s'accompagne de responsabilités plus importantes. Dans les sections suivantes, vous apprendrez à résoudre les problèmes liés aux instances de base de données Amazon RDS Custom.

> **Note**
>
> Cette section explique comment résoudre les problèmes liés à RDS Custom for Oracle. Pour la résolution des problèmes liés à RDS Custom for SQL Server, consultez Résolution des problèmes de base de données pour Amazon RDS Custom for SQL Server.

Rubriques

- Affichage des événements RDS Custom
- Abonnement aux événements personnalisés RDS
- Résolution des problèmes liés à la création d'une version de moteur personnalisée pour RDS Custom for Oracle
- Correction des configurations non prises en charge dans RDS Custom for Oracle
- Dépannage des mises à niveau de RDS Custom for Oracle
- Dépannage de la promotion de réplica pour RDS Custom for Oracle

Affichage des événements RDS Custom

La procédure d'affichage est la même pour les instances de base de données RDS Custom et Amazon RDS. Pour de plus amples informations, veuillez consulter Consulter les RDS événements Amazon.

Pour afficher la notification d'événement personnalisée RDS à l'aide de AWS CLI, utilisez la `describe-events` commande. RDS Custom s'accompagne de plusieurs nouveaux événements. Les catégories d'événements sont les mêmes que pour Amazon RDS. Pour obtenir la liste des événements, consultez Catégories d'événements Amazon RDS et messages d'événements .

L'exemple suivant récupère les détails des événements qui se sont produits pour l'instance de base de données RDS Custom spécifiée.

```
aws rds describe-events \
    --source-identifier my-custom-instance \
    --source-type db-instance
```

Abonnement aux événements personnalisés RDS

La procédure d'abonnement à des événements est la même pour les instances de base de données RDS Custom et Amazon RDS. Pour de plus amples informations, veuillez consulter S'abonner aux notifications d'RDSévénements Amazon.

Pour vous abonner à la notification d'événements RDS Custom à l'aide de l'interface de ligne de commande, utilisez la commande `create-event-subscription`. Incluez les paramètres requis suivants :

- `--subscription-name`
- `--sns-topic-arn`

L'exemple suivant montre comment créer un abonnement pour les événements de sauvegarde et de restauration d'une instance de base de données RDS Custom dans le compte AWS actuel. Les notifications sont envoyées à une rubrique Amazon Simple Notification Service (Amazon SNS) spécifiée par `--sns-topic-arn`.

```
aws rds create-event-subscription \
    --subscription-name my-instance-events \
    --source-type db-instance \
    --event-categories '["backup","recovery"]' \
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:interesting-events
```

Résolution des problèmes liés à la création d'une version de moteur personnalisée pour RDS Custom for Oracle

Lorsque la création d'une CEV échoue, RDS Custom émet `RDS-EVENT-0198` avec le message `Creation failed for custom engine version` *`major-engine-version.cev_name`* et ajoute des détails sur l'échec. Par exemple, l'événement imprime les fichiers manquants.

La création d'une CEV peut échouer en raison des problèmes suivants :

- Le compartiment Amazon S3 contenant vos fichiers d'installation ne se trouve pas dans la même AWS région que votre CEV.
- Lorsque vous demandez la création d'un CEV dans et Région AWS pour la première fois, RDS Custom crée un compartiment S3 pour stocker les ressources personnalisées RDS (telles que les artefacts CEV, les journaux et AWS CloudTrail les journaux de transactions).

  La création de la CEV échoue si RDS Custom ne parvient pas à créer le compartiment S3. Soit l'appelant ne dispose pas des autorisations S3, comme décrit dans la section Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM, soit le nombre de compartiments S3 a atteint la limite.
- L'appelant ne dispose pas des autorisations nécessaires pour obtenir des fichiers de votre compartiment S3 contenant les fichiers multimédias d'installation. Ces autorisations sont décrites dans la section Étape 7 : Ajouter les autorisations IAM nécessaires.
- Votre politique IAM est dotée d'une condition `aws:SourceIp`. Assurez-vous de suivre les recommandations de la section AWS refuse l'accès à AWS en fonction de l'adresse IP source dans le Guide de l'utilisateur AWS Identity and Access Management . Assurez-vous également que l'appelant dispose des autorisations S3 décrites dans Étape 5 : accordez les autorisations requises à votre utilisateur ou à votre rôle IAM.
- Les fichiers multimédias d'installation répertoriés dans le manifeste CEV ne se trouvent pas dans votre compartiment S3.
- RDS Custom ne connaît pas les totaux de contrôle SHA-256 des fichiers d'installation.

  Vérifiez que les totaux de contrôle SHA-256 des fichiers fournis correspondent à celui qui se trouve sur le site Web Oracle. Si les totaux de contrôle correspondent, contactezAWS Support et indiquez le nom de la CEV qui a échoué, le nom de fichier et le total de contrôle.
- La OPatch version n'est pas compatible avec vos fichiers correctifs. Vous pourriez obtenir le message suivant :`OPatch is lower than minimum required version. Check that the version meets the requirements for all patches, and try again`. Pour appliquer un correctif Oracle, vous devez utiliser une version compatible de l' OPatch utilitaire. Vous pouvez trouver la version requise de l'utilitaire Opatch dans le fichier readme du correctif. Téléchargez l' OPatch utilitaire le plus récent sur My Oracle Support et réessayez de créer votre CEV.
- Les correctifs spécifiés dans le manifeste CEV ne sont pas dans le bon ordre.

Vous pouvez afficher les événements RDS sur la console RDS (dans le volet de navigation, choisissez Events) ou à l'aide de la `describe-events` AWS CLI commande. La durée par défaut est de 60 minutes. Si aucun événement n'est renvoyé, spécifiez une durée plus importante, comme illustré dans l'exemple suivant.

```
aws rds describe-events --duration 360
```

Actuellement, le MediaImport service qui importe des fichiers depuis Amazon S3 à des fins de création CEVs n'est pas intégré à AWS CloudTrail. Par conséquent, si vous activez l'enregistrement des données pour Amazon RDS in CloudTrail, les appels au MediaImport service tels que l'`CreateCustomDbEngineVersion`événement ne sont pas enregistrés.

Vous pouvez toutefois voir des appels provenant de l'API Gateway qui accède à votre compartiment Amazon S3. Ces appels proviennent du MediaImport service de l'`CreateCustomDbEngineVersion`événement.

Correction des configurations non prises en charge dans RDS Custom for Oracle

Dans le modèle de responsabilité partagée, il vous incombe de corriger les problèmes de configuration qui redonnent à votre instance de base de données RDS Custom for Oracle le statut `unsupported-configuration`. Si le problème est lié à l' AWS infrastructure, utilisez la console ou le AWS CLI pour le résoudre. Si le problème est lié au système d'exploitation ou à la configuration de la base de données, connectez-vous à l'hôte pour le résoudre.

> **Note**
>
> Cette section explique comment corriger les configurations non prises en charge dans RDS Custom for Oracle. Pour obtenir des informations sur RDS Custom for SQL Server, consultez Correction des configurations non prises en charge dans RDS Custom for SQL Server.

Les tableaux suivants décrivent les notifications et les événements envoyés par le périmètre de support et indiquent comment y remédier. Ces notifications et le périmètre de prise en charge sont susceptibles d'être modifiés. Pour en savoir plus sur le périmètre de prise en charge, consultez Périmètre de prise en charge RDS Custom. Pour les descriptions des événements, consultez Catégories d'événements Amazon RDS et messages d'événements .

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-00000	Configuration manuelle non prise en charge	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes :. *reason*	Pour résoudre ce problème, créez un Support dossier.

AWS ressources (infrastructure)

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O1001	Volumes Amazon Elastic Block Store (Amazon EE	Les volumes EBS suivants ont été ajoutés à l' EC2 instance *ec2_id* : *volume_id* Pour résoudre le problème, détachez les volumes spécifiés de l'instance.	RDS Custom crée deux types de volumes EBS, outre le volume racine créé à partir de l'Amazon Machine Image (AMI), et les associe à l' EC2 instance : • Volume binaire dans lequel se trouvent les fichiers binaires du logiciel de base de données • Les volumes de données dans lesquels se trouvent les fichiers de base de données Lorsque vous créez votre instance de base de données, les configurations de stockage que vous spécifiez configurent les volumes de données.

ID de l'événement	Configuration	Message d'événement RDS	Action
			Le périmètre de prise en charge surveille ce qui suit : • Les volumes EBS initiaux créés avec l'instance de base de données sont toujours associés à l'instance. • Les volumes EBS initiaux ont toujours les mêmes configurations que celles qui ont été définies initialement : type de stockage, taille, IOPS provisionnés et débit de stockage. • Aucun volume EBS supplémentaire n'est attaché à l'instance de base de données. Utilisez la commande CLI suivante pour comparer le type de volume des détails du volume EBS et les détails de l'instance de base de données RDS Custom for Oracle : `aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep StorageType`

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O1002	Volumes Amazon Elastic Block Store (Amazon EE	Le volume EBS *volume_id* a été détaché de l' EC2 instance [*ec2_id*]. Vous ne pouvez pas détacher le volume d'origine de cette instance. Pour résoudre le problème, rattachez-le *volume_id* à. *ec2_id*	RDS Custom crée deux types de volumes EBS, outre le volume racine créé à partir de l'Amazon Machine Image (AMI), et les associe à l' EC2 instance : • Volume binaire dans lequel se trouvent les fichiers binaires du logiciel de base de données • Les volumes de données dans lesquels se trouvent les fichiers de base de données Lorsque vous créez votre instance de base de données, les configurations de stockage que vous spécifiez configurent les volumes de données. Le périmètre de prise en charge surveille ce qui suit : • Les volumes EBS initiaux créés avec l'instance de base de données sont toujours associés à l'instance. • Les volumes EBS initiaux ont toujours les mêmes configurations que celles qui ont été définies initialement : type de stockage, taille, IOPS provisionnés et débit de stockage. • Aucun volume EBS supplémentaire n'est attaché à l'instance de base de données. Utilisez la commande CLI suivante pour comparer le type de volume des détails du volume EBS et les détails de l'instance de base de données RDS Custom for Oracle : `aws rds describe-db-instances \`

ID de l'événement	Configuration	Message d'événement RDS	Action
			`--db-instance-identifier db-instance-name | grep StorageType`

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O1003	Volumes Amazon Elastic Block Store (Amazon EE	Le volume EBS d'origine *volume_id* attaché à l' EC2 instance *ec2_id* a été modifié comme suit : taille [*X*] à [*Y*], type [*N*] à [*M*] ou IOPS [*J*] à [*K*]. Pour résoudre le problème, annulez la modification.	RDS Custom crée deux types de volumes EBS, outre le volume racine créé à partir de l'Amazon Machine Image (AMI), et les associe à l' EC2 instance : • Volume binaire dans lequel se trouvent les fichiers binaires du logiciel de base de données • Les volumes de données dans lesquels se trouvent les fichiers de base de données Lorsque vous créez votre instance de base de données, les configurations de stockage que vous spécifiez configurent les volumes de données. Le périmètre de prise en charge surveille ce qui suit : • Les volumes EBS initiaux créés avec l'instance de base de données sont toujours associés à l'instance. • Les volumes EBS initiaux ont toujours les mêmes configurations que celles qui ont été définies initialement : type de stockage, taille, IOPS provisionnés et débit de stockage. • Aucun volume EBS supplémentaire n'est attaché à l'instance de base de données. Utilisez la commande CLI suivante pour comparer le type de volume des détails du volume EBS et les détails de l'instance de base de données RDS Custom for Oracle : `aws rds describe-db-instances \`

ID de l'événement	Configuration	Message d'événement RDS	Action
			`--db-instance-identifier db-instance-name | grep StorageType`
SP-O1004	État de l' EC2 instance Amazon	La restauration automatique a laissé l' EC2 instance [*ec2_id*] dans un état altéré. Pour résoudre le problème, consultez la section Dépannage des échecs de restauration d'instance.	Pour vérifier l'état d'une instance de base de données, utilisez la console ou exécutez la AWS CLI commande suivante : `aws rds describe-db-instances \ --db-instance-identifier *db-instance-name* |grep DBInstanceStatus`
SP-O1005	Attributs de EC2 l'instance Amazon	EC2 l'instance [*ec2_id*] a été modifiée comme suit : l'attribut [*att1*] est passé de [*val-old*] à [*val-new*], l'attribut [*att2*] est passé de [*val-old*] à [*val-new*]. Pour résoudre le problème, revenez à la valeur d'origine .	

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O1006	État de l' EC2 instance Amazon	EC2 l'instance [*ec2_id*] a été interrompue ou est introuvable. Pour résoudre le problème, supprimez l'instance de base de données personnalisée RDS.	Le périmètre de support surveille les notifications de changement d'état des EC2 instances. L' EC2 instance doit toujours être en cours d'exécution. Pour supprimer votre instance de base de données 1. Pour vérifier l'état d'une instance de base de données, utilisez la console ou exécutez la AWS CLI commande suivante : `aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep DBInstanceStatus` 2. Supprimez votre instance de base de données RDS Custom pour Oracle.
SP-O1007	État de l' EC2 instance Amazon	EC2 l'instance [*ec2_id*] a été arrêtée. Pour résoudre le problème, démarrez l'instance.	Le périmètre de support surveille les notifications de changement d'état des EC2 instances. L' EC2 instance doit toujours être en cours d'exécution. Pour redémarrer votre instance de base de données 1. Pour vérifier l'état d'une instance de base de données, utilisez la console ou exécutez la AWS CLI commande suivante : `aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep DBInstanceStatus` 2. Démarrez votre instance de base de données. 3. Remontez les volumes binaires et les volumes de données.

ID de l'événement	Configuration	Message d'événement RDS	Action

SP-10	Autorisation Amazon SQS	Les autorisations sont manquantes pour Amazon SQS. Vérifiez les autorisations pour le profil d'instance IAM, la politique de point de terminaison VPC et les connexions de service dépendantes, puis réessayez.	Vous pouvez résoudre ce problème en vous assurant que le profil IAM associé à l'hôte dispose des autorisations suivantes :
			`"SQS:SendMessage"`
			`"SQS:ReceiveMessage"`
			`"SQS:DeleteMessage"`
			`"SQS:GetQueueUrl"`

SP-10	Amazon Simple Queue Service (Amazon SQS)	La file d'attente SQS [%s] a été supprimée et n'a pas pu être récupérée. Pour résoudre ce problème, recréez la file d'attente.	Recréez la file d'attente Amazon SQS.

Système d'exploitation

ID de l'événement	Configuration	Message d'événement RDS	Action

SP-O2001	Statut de l'agent RDS Custom	L'agent RDS Custom n'est pas en cours d'exécution sur l' EC2 instance [*ec2_id*]. Assurez-vous que l'agent est exécuté sur [*ec2_id*].	Sur RDS Custom for Oracle, l'instance de base de données sort du périmètre de prise en charge si l'agent RDS Custom s'arrête. L'agent publie la `IamAlive` métrique sur Amazon CloudWatch toutes les 30 secondes. Une alarme est déclenchée si la métrique n'a pas été publiée depuis 30 secondes. Le périmètre de prise en charge surveille également l'état du processus d'agent RDS Custom sur l'hôte toutes les 30 minutes.
			
			Pour redémarrer l'agent RDS Custom
			
			1. Connectez-vous à votre hôte et assurez-vous que l'agent RDS Custom est en cours d'exécution.
			2. Exécutez la commande suivante pour connaître le statut de l'agent.
			`service rdscustomagent status`
			3. Utilisez la commande suivante pour démarrer l'agent.
			`service rdscustomagent start`
			
			Lorsque l'agent RDS Custom s'exécute à nouveau, la `IamAlive` métrique est publiée sur Amazon CloudWatch et l'alarme passe à l'`OK`état. Le périmètre de prise en charge est ainsi informé que l'agent est en cours d'exécution.

SP-O2002	AWS Systems	L'agent Systems Manager sur l'	L'agent SSM doit toujours être en cours d'exécution. L'agent RDS Custom est chargé de s'assurer

ID de l'événement	Configuration	Message d'événement RDS	Action
	Manager statut de l'agent (agent SSM)	EC2 instance [*ec2_id*] n'est pas joignable. Assurez-vous que vous avez correctement configuré le réseau, l'agent et les autorisations IAM.	que Systems Manager Agent est en cours d'exécution. Si l'agent SSM a été arrêté puis redémarré, l'agent personnalisé RDS publie une métrique sur. CloudWatch L'agent RDS Custom est doté d'une alarme sur la métrique configurée pour se déclencher si un redémarrage a eu lieu au cours des trois dernières minutes. Le périmètre de support surveille également l'état du processus de l'agent SSM sur l'hôte toutes les 30 minutes. Pour de plus amples informations, consultez la section Résolution des problèmes de SSM Agent.
SP-02003	AWS Systems Manager statut de l'agent (agent SSM)	L'agent Systems Manager sur l' EC2 instance [*ec2_id*] s'est écrasé à plusieurs reprises. Pour plus d'informations, consultez la documentation de dépannage de l'agent SSM.	Pour de plus amples informations, consultez la section Résolution des problèmes de SSM Agent.

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O 2004	Fuseau horaire du système d'exploitation	Le fuseau horaire de l' EC2 instance [*ec2_id*] a été modifié. Pour résoudre ce problème, rétablissez le réglage précédent de [] *previous-time-zone* pour le fuseau horaire. Utilisez ensuite un groupe d'options RDS pour modifier le fuseau horaire.	L'automatisation RDS a détecté que le fuseau horaire de l'hôte avait été modifié sans utiliser de groupe d'options. Ce changement au niveau de l'hôte peut provoquer des échecs d'automatisation RDS, de sorte que l' EC2 instance est placée dans l'état. `unsupported-configuration` Pour corriger le réglage du fuseau horaire 1. Connectez-vous à votre EC2 hôte et vérifiez le fuseau horaire du système d'exploitation comme suit : `timedatectl` 2. Mettez en pause l'automatisation de RDS Custom. Pour de plus amples informations, veuillez consulter Suspension et reprise de votre RDS instance de base de données personnalisée. 3. Arrêtez l'instance de base de données. 4. Annulez le changement de fuseau horaire sur le système d'exploitation. 5. Démarrez l'instance de base de données. 6. Relancez l'automatisation de RDS Custom. Votre instance de base de données devient disponible dans les 30 minutes. Pour éviter de vous déplacer hors du périmètre à l'avenir, modifiez votre fuseau horaire via un groupe d'options. Pour de plus amples informations, veuillez consulter Fuseau horaire Oracle.

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O 2005	Configurations sudo	Les configurations sudo sur l' EC2 instance [*ec2_id*] ne disposent pas des autorisations nécessaires. Pour résoudre ce problème, annulez les modifications récentes apportées aux configurations sudo.	Le périmètre de support vérifie que certains utilisateurs du système d'exploitation sont autorisés à exécuter certaines commandes sur l'hôte. Il surveille les `sudo` configurations et les compare à l'état pris en charge. Si les `sudo` configurations ne sont pas prises en charge, RDS Custom essaie de les remplacer et de revenir à l'état pris en charge précédent. Si la tentative aboutit, RDS Custom envoie la notification suivante : RDS Custom successfully overwrote your configuration. (RDS Custom a réussi à écraser votre configuration.) Si le remplacement échoue, votre instance de base de données reste dans l'état de configuration non pris en charge. Pour résoudre ce problème, annulez les modifications apportées au `sudoers.d/` fichier ou corrigez les autorisations. Pour étudier les modifications apportées aux **sudo** configurations 1. Connectez-vous à votre hôte. 2. Exécutez la commande suivante. `visudo -c -f /etc/sudoers.d/ individual_sudo_files` 3. Modifiez les `sudo` configurations si nécessaire.

ID de l'événement	Configuration	Message d'événement RDS	Action
			Une fois que le périmètre de support a déterminé que les `sudo` configurations sont prises en charge, votre instance de base de données RDS Custom for Oracle est disponible dans les 30 minutes.
SP-O 2006	Accessibilité du compartiment S3	L'automatisation personnalisée RDS ne peut pas télécharger de fichiers depuis le compartiment S3 sur l' EC2 instance [*ec2_id*]. Vérifiez votre configuration réseau et assurez-vous que l'instance autorise les connexions depuis et vers S3.	
SP-20	Version de la solution logicielle à haute disponibilité	La solution HA de votre instance est différente de la version attendue. Pour résoudre ce problème, créez un AWS Support dossier.	Créez un AWS Support dossier.

Database (Base de données)

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O3001	Cible de retard d'archivage de la base de données	Le paramètre ARCHIVE_LAG_TARGET sur l' EC2 instance [*ec2_id*] est en dehors de la plage recommandée. *value_range* Pour résoudre le problème, définissez le paramètre sur une valeur comprise dans value_range.	Le périmètre de support surveille le paramètre `ARCHIVE_LAG_TARGET` de base de données afin de vérifier que l'heure de restauration la plus récente de l'instance de base de données se situe dans des limites raisonnables. Pour modifier l'objectif de décalage pour les redo logs archivés 1. Connectez-vous à votre EC2 hôte 2. Connectez-vous à votre instance de base de données RDS Custom pour Oracle 3. Remplacez le `ARCHIVE_LAG_TARGET` paramètre par une valeur comprise entre 60 et 7200. Par exemple, utilisez l'instruction SQL suivante. `ALTER SYSTEM SET ARCHIVE_LAG_TARGET=300 SCOPE=BOTH;` Votre instance de base de données devient disponible dans les 30 minutes.
SP-O3002	Rôle Oracle Data Guard	Le rôle de base de données [*role_name*] n'est pas pris en charge pour Oracle Data Guard sur l' EC2 instance [*ec2_id*].	Le périmètre de support surveille le rôle de base de données actuel toutes les 15 secondes et envoie une CloudWatch notification si le rôle de base de données a changé. Le paramètre Oracle Data Guard `DATABASE_ROLE` doit être `PRIMARY` ou `PHYSICAL STANDBY`.

ID de l'événement	Configuration	Message d'événement RDS	Action

		Pour résoudre le problème, définissez le paramètre DATABASE_ROLE sur PRIMARY ou PHYSICAL STANDBY.	Pour restaurer votre rôle de base de données Oracle Data Guard à une valeur prise en charge
			1. Vérifiez le rôle d'Oracle Data Guard en exécutant l'instruction suivante :
			`SELECT DATABASE_ROLE FROM V$DATABASE;`
			2. Si votre instance de base de données est autonome, utilisez l'une des instructions suivantes pour lui redonner le PRIMARY rôle :
			`ALTER DATABASE COMMIT TO SWITCHOVER PRIMARY;`
			`ALTER DATABASE ACTIVATE STANDBY DATABASE;`
			Si votre instance de base de données est une réplique, utilisez l'instruction suivante pour lui redonner le PHYSICAL STANDBY rôle :
			`ALTER DATABASE CONVERT TO PHYSICAL STANDBY;`
			Une fois que le périmètre de prise en charge a déterminé que le rôle de base de données est pris en charge, votre instance de base de données RDS Custom for Oracle devient disponible dans les 15 secondes.

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O3003	État de la base de données	Le processus SMON de la base de données Oracle est dans un état zombie. Pour résoudre le problème, restaurez manuellement la base de données sur l' EC2instance [*ec2_id*], ouvrez-la, puis sauvegardez-la immédiatement. Pour obtenir de l'aide supplémentaire, contactez Support.	Le périmètre de prise en charge surveille l'état de l'instance de base de données. Il surveille également le nombre de redémarrages qui se sont produits au cours de la dernière heure et du jour précédent. Vous êtes averti lorsque l'instance se trouve dans un état où elle se trouve toujours, mais vous ne pouvez pas interagir avec elle. Pour que le périmètre de support évalue l'état de votre instance 1. Connectez-vous à votre hébergeur et déterminez l'état de la base de données. `ps -eo pid,state,command | grep smon` 2. Si nécessaire, redémarrez votre instance de base de données. Si le redémarrage échoue, passez à l'étape suivante. 3. Si nécessaire, redémarrez votre EC2 hébergeur. Après le redémarrage de votre instance de base de données, l'agent personnalisé RDS détecte que votre instance de base de données n'est plus dans un état de non-réponse. Il notifie alors le périmètre de prise en charge qu'il faut réévaluer le statut de votre instance de base de données.

ID de l'événement	Configuration	Message d'événement RDS	Action

SP-O3004	Mode journal de base de données	Le mode de journalisation de la base de données sur l' EC2 instance [*ec2_id*] a été remplacé par [*value_b*]. Pour résoudre le problème, réglez le mode journal sur [*value_a*].	Pour modifier le mode de journalisation de votre instance de base de données sur **ARCHIVELOG**
			1. Connectez-vous à votre EC2 hôte.
			2. Connectez-vous à votre base de données et exécutez l'instruction suivante :
			`SELECT LOG_MODE FROM V$DATABASE;`
			Vous pouvez également exécuter la commande suivante dans SQL*Plus :
			`ARCHIVE LOG LIST`
			3. Exécutez la commande SQL*Plus suivante pour lancer un arrêt cohérent.
			`SHUTDOWN IMMEDIATE`
			L'agent RDS Custom redémarre automatiquement votre instance de base de données et définit le mode journal sur. `ARCHIVELOG` Votre instance de base de données devient disponible dans les 30 minutes.

ID de l'événement	Configuration	Message d'événement RDS	Action
SP-O3005	Chemin d'accès Oracle Home	Le répertoire d'origine Oracle Home de l' EC2 instance [*ec2_id*] a été remplacé par*new_path*. Pour résoudre le problème, rétablissez le paramètre sur*old_path*.	

ID de l'événement	Configuration	Message d'événement RDS	Action

SP-O3006	Nom unique de la base de données	Le nom unique de la base de données sur l' EC2 instance [*ec2_id*] a été remplacé par*new_value* . Pour résoudre le problème, remplacez le nom par. *old_value*	Pour modifier le nom unique de base de données de votre instance de base de données
			1. Connectez-vous à votre EC2 hôte.
			2. Connectez-vous à la base de données et exécutez l'instruction suivante :
			`SELECT DB_UNIQUE_NAME FROM V$DATABASE;`
			3. Spécifiez le nom unique de la base de données d'origine à l'aide de la commande`ALTER SYSTEM SET DB_UNIQUE_NAME` .
			4. Exécutez l'instruction SQL suivante pour lancer un arrêt cohérent.
			`SHUTDOWN IMMEDIATE;`
			L'agent RDS Custom redémarre automatiquement votre instance de base de données et définit le mode journal sur. `ARCHIVELOG` Votre instance de base de données devient disponible dans les 30 minutes.

Dépannage des mises à niveau de RDS Custom for Oracle

Votre mise à niveau d'une instance de RDS Custom for Oracle peut échouer. Vous trouverez ci-dessous des techniques que vous pouvez utiliser lors des mises à niveau des bases de données RDS Custom pour les instances de base de données Oracle :

- Examinez tous les fichiers journaux de sortie de la mise à niveau dans l'annuaire `/tmp` sur votre instance de base de données. Les noms des journaux dépendent de la version du moteur de votre

base de données. Par exemple, vous pouvez voir des journaux contenant les chaînes `catupgrd` ou `catup`.

- Examinez le fichier `alert.log` situé dans l'annuaire `/rdsdbdata/log/trace`.
- Exécutez la commande `grep` suivante dans le répertoire `root` pour suivre le processus de mise à niveau du système d'exploitation. Cette commande indique l'emplacement d'écriture des fichiers journaux et détermine l'état du processus de mise à niveau.

```
ps -aux | grep upg
```

Voici un exemple de résultat.

```
root     18884  0.0  0.0 235428  8172 ?        S<   17:03   0:00 /usr/bin/
sudo -u rdsdb /rdsdbbin/scripts/oracle-control ORCL op_apply_upgrade_sh RDS-
UPGRADE/2.upgrade.sh
rdsdb    18886  0.0  0.0 153968 12164 ?        S<   17:03   0:00 /usr/bin/perl -T -
w /rdsdbbin/scripts/oracle-control ORCL op_apply_upgrade_sh RDS-UPGRADE/2.upgrade.sh
rdsdb    18887  0.0  0.0 113196  3032 ?        S<   17:03   0:00 /bin/sh /rdsdbbin/
oracle/rdbms/admin/RDS-UPGRADE/2.upgrade.sh
rdsdb    18900  0.0  0.0 113196  1812 ?        S<   17:03   0:00 /bin/sh /rdsdbbin/
oracle/rdbms/admin/RDS-UPGRADE/2.upgrade.sh
rdsdb    18901  0.1  0.0 167652 20620 ?        S<   17:03   0:07 /rdsdbbin/oracle/
perl/bin/perl catctl.pl -n 4 -d /rdsdbbin/oracle/rdbms/admin -l /tmp catupgrd.sql
root     29944  0.0  0.0 112724  2316 pts/0    S+   18:43   0:00 grep --color=auto
 upg
```

- Exécutez la requête SQL suivante pour vérifier l'état actuel des composants, et trouver la version de la base de données et les options installées sur l'instance de base de données.

```
SET LINESIZE 180
COLUMN COMP_ID FORMAT A15
COLUMN COMP_NAME FORMAT A40 TRUNC
COLUMN STATUS FORMAT A15 TRUNC
SELECT COMP_ID, COMP_NAME, VERSION, STATUS FROM DBA_REGISTRY ORDER BY 1;
```

La sortie se présente comme suit :

```
COMP_NAME                                STATUS               PROCEDURE
---------------------------------------- --------------------
 ------------------------------------------------
```

```
Oracle Database Catalog Views           VALID
 DBMS_REGISTRY_SYS.VALIDATE_CATALOG
Oracle Database Packages and Types      VALID
 DBMS_REGISTRY_SYS.VALIDATE_CATPROC
Oracle Text                             VALID               VALIDATE_CONTEXT
Oracle XML Database                     VALID               DBMS_REGXDB.VALIDATEXDB

4 rows selected.
```

- Exécutez la requête SQL suivante pour rechercher d'éventuels objets non valides susceptibles de perturber le processus de mise à niveau.

```
SET PAGES 1000 LINES 2000
COL OBJECT FOR A40
SELECT SUBSTR(OWNER,1,12) OWNER,
       SUBSTR(OBJECT_NAME,1,30) OBJECT,
       SUBSTR(OBJECT_TYPE,1,30) TYPE, STATUS,
       CREATED
FROM   DBA_OBJECTS
WHERE  STATUS <>'VALID'
AND    OWNER IN ('SYS','SYSTEM','RDSADMIN','XDB');
```

Dépannage de la promotion de réplica pour RDS Custom for Oracle

Vous pouvez promouvoir les répliques Oracle gérées dans RDS Custom for Oracle à l'aide de la console, de la `promote-read-replica` AWS CLI commande ou `PromoteReadReplica` de l'API. Si vous supprimez votre instance de base de données principale et que tous les réplicas sont sains, RDS Custom for Oracle transforme automatiquement vos réplicas gérée en instances autonomes. Si un réplica a mis en pause l'automatisation ou se trouve en dehors du périmètre de support, vous devez réparer le réplica avant que RDS Custom puisse le promouvoir automatiquement. Pour de plus amples informations, veuillez consulter Promouvoir une réplique RDS personnalisée pour Oracle en une instance de base de données autonome.

Le flux de travail de promotion des réplicas peut se bloquer dans la situation suivante :

- L'instance de base de données principale est dans l'état `STORAGE_FULL`.
- La base de données principale ne peut pas archiver tous ses journaux de rétablissement en ligne.
- Il existe un écart entre les fichiers journaux de reprise archivés sur votre réplica Oracle et la base de données principale.

Pour répondre au flux de travail bloqué

1. Synchronisez l'écart du journal de reprise sur votre réplica d’instance de base de données Oracle.
2. Forcez la promotion de votre réplica en lecture vers le dernier journal de reprise appliqué. Exécutez les commandes suivantes dans SQL*Plus :

```
ALTER DATABASE ACTIVATE STANDBY DATABASE;
SHUTDOWN IMMEDIATE
STARTUP
```

3. Contactez-leur Support et demandez-leur de passer votre instance de base de données au `available` statut.

Problèmes connus relatifs à Amazon RDS Custom pour Oracle

Lorsque vous travaillez avec RDS Custom for Oracle, notez les problèmes suivants concernant les instances de base de données :

- Le redimensionnement des volumes root ou dbbin n'est pas pris en charge.

  > **Warning**
  >
  > Nous vous recommandons vivement de ne pas redimensionner les volumes root ou dbbin manuellement. Nous vous recommandons de stocker toutes les configurations dans le volume de données, qui persiste après l'application des correctifs, et de redimensionner le volume en utilisant uniquement le stockage de la RDS balance. API

- Certaines RDS APIs peuvent être bloquées lorsqu'une instance de base de données se trouve sur une ancienne AMI instance, par exemple une instance AMI qui utilise Oracle Linux 7. Pour résoudre ce problème, appliquez les correctifs les plus récents à votre instance de base de données à l'AMIaide des correctifs du système d'exploitation. Pour de plus amples informations, veuillez consulter CEVoptions de mise à niveau.
- Avant d'effectuer RDS des opérations, assurez-vous que votre quota de calcul et de stockage Compte AWS est suffisant.
- Si la base de données est en cours de création et que vous vous connectez activement à la base de données ou à l'EC2hôte Amazon et que vous exécutez des commandes, la création de la base de données risque de ne pas être terminée.
- Le multiplexage des fichiers de contrôle n'est actuellement pas pris en charge en raison d'un problème de réplication en lecture. Avant de créer une réplique en lecture, assurez-vous de ne spécifier qu'un seul nom de fichier dans le paramètre d'`CONTROL_FILES`initialisation de la base de données source.
- Vous ne pouvez pas changer le mode de base de données de `PHYSICAL STANDBY` (monté ou lecture seule) à `SNAPSHOT STANDBY` (conversion en lecture/écriture).
- Si un Compte AWS fait partie d'une AWS organisation dotée d'une politique de contrôle des services (SCP) et SCP contient une clé de condition, la création d'une instance de base de données RDS personnalisée pour Oracle peut échouer avec l'erreur suivante :

```
You can't create the DB instance because of incompatible resources.
The IAM instance profile role [AWSRDSCustomInstanceRole1-us-east-1] is missing the
 following permissions:
```

```
EFFECT [Allow] on ACTION(S) [ssm:DescribeAssociation, ssm:DescribeDocument,
 ssm:GetConnectionStatus,
  ssm:GetDeployablePatchSnapshotForInstance, ssmmessages:OpenControlChannel,
 ssm:GetParameters,
  ssm:ListInstanceAssociations, ssm:PutConfigurePackageResult,
 ssmmessages:CreateControlChannel,
  ssm:GetParameter, ssm:UpdateAssociationStatus, ssm:GetManifest,
 ssmmessages:CreateDataChannel,
  ssm:PutInventory, ssm:UpdateInstanceInformation, ssm:DescribeInstanceInformation,
  ssmmessages:OpenDataChannel, ssm:GetDocument, ssm:ListAssociations,
 ssm:PutComplianceItems,
  ssm:UpdateInstanceAssociationStatus] for RESOURCE(S) [], EFFECT [Allow] on
 ACTION(S) [ec2messages:DeleteMessage,
  ec2messages:FailMessage, ec2messages:GetEndpoint, ec2messages:AcknowledgeMessage,
 ec2messages:GetMessages,
  ec2messages:SendReply] for RESOURCE(S) [], EFFECT [Allow] on ACTION(S)
 [logs:CreateLogStream,
  logs:DescribeLogStreams, logs:PutRetentionPolicy, logs:PutLogEvents]
```

Pour résoudre ce problème, créez un ticket avec Support.

Problèmes connus liés aux comptes utilisateur de base de données

Notez les problèmes suivants :

- Ne supprimez pas les comptes utilisateur de base de données qui commencent par une chaîne`RDS`, telle que `RDSADMIN` et`RDS_DATAGUARD`. RDS Custom for Oracle utilise le `RDS` compte à des fins d'automatisation. Si vous supprimez ce compte utilisateur, RDS Custom fait passer l'instance à l'état de configuration non pris en charge.
- Vous ne pouvez pas modifier le nom d'utilisateur principal de votre instance de base de données RDS personnalisée pour Oracle à l'aide du `ModifyDBInstance`API.
- RDSCustom for Oracle fait pivoter les informations d'identification des comptes utilisateur sur toutes les instances de base de données. Pour de plus amples informations, veuillez consulter Rotation des informations d'identification RDS Custom for Oracle pour les programmes de conformité. Si vous utilisez une configuration principale/de secours sur site, la rotation des informations d'identification peut affecter les ressources suivantes :
  - Système de secours créé manuellement RDS personnalisé pour les instances Oracle

Pour résoudre ce problème, supprimez les bases de données de secours manuelles, puis créez une réplique de lecture Oracle à l'aide d'un API appel. Gérez les secrets manuellement pour les bases de données de secours manuelles afin qu'ils correspondent à l'instance de base de données source.

- Répliques de lecture interrégionales créées manuellement

  Pour résoudre ce problème, conservez manuellement les secrets afin qu'ils correspondent à l'instance de base de données principale.

Problèmes connus liés aux fichiers de paramètres et de configuration

- Vous devez configurer le `crontab` fichier après le calcul à l'échelle, les mises à niveau du système d'exploitation et les autres opérations dans lesquelles RDS Custom remplace le volume racine. Nous vous recommandons vivement de conserver une sauvegarde de`crontab`.
- Respectez les consignes suivantes lorsque vous configurez le `listener.ora` fichier :
  - Assurez-vous que chaque entrée du fichier se trouve sur une seule ligne. Cette approche permet d'éviter les problèmes d'indentation lors de la création de l'instance.
  - Assurez-vous que `GLOBAL_DBNAME` c'est égal à la valeur de`SID_NAME`.
  - Assurez-vous que la valeur de `LISTENER` respecte la convention de dénominationL_*dbname*_001.
  - Assurez-vous que le `listener.ora` fichier maintient une connexion avec le nom de la base de données. RDS Custom utilise cette connexion pour vérifier le démarrage de la base de données. Si vous modifiez ce fichier de manière incorrecte, les opérations telles que le calcul à l'échelle ou l'application de correctifs risquent d'échouer.

    L'exemple suivant `listener.ora` montre une configuration correcte :

```
ADR_BASE_L_ORCL_001=/rdsdbdata/log/
USE_SID_AS_SERVICE_L_ORCL_001=ON
SID_LIST_L_ORCL_001=(SID_LIST = (SID_DESC = (SID_NAME = ORCL)(GLOBAL_DBNAME = ORCL)
 (ORACLE_HOME = /rdsdbbin/oracle.19.custom.r1.EE.1)))
SUBSCRIBE_FOR_NODE_DOWN_EVENT_L_ORCL_001=OFF
L_ORCL_001=(DESCRIPTION_LIST = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(PORT =
 XXXX)(HOST = x.x.x.x))) (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(PORT = XXXX)
(HOST = 127.0.0.1))))
```

- Les commentaires ne sont pas pris en charge dans un fichier de paramètres de serveur ou un fichier de paramètres d'initialisation.
- Vous devez déclarer les paramètres d'initialisation suivants dans le fichier de paramètres du serveur (`/rdsdbdata/config/oracle_pfile`) :
  - `MEMORY_MAX_TARGET`
  - `MEMORY_TARGET`
  - `PGA_AGGREGATE_TARGET`
  - `PROCESSES`
  - `SGA_TARGET`
  - `USE_LARGE_PAGES`

  Si les paramètres précédents ne sont pas déclarés dans`/rdsdbdata/config/oracle_pfile`, la création de répliques en lecture et le calcul à l'échelle peuvent échouer.
- Vous ne pouvez pas supprimer les liens symboliques pour les fichiers de configuration tels que le fichier de paramètres du serveur, les fichiers d'audit `listener.ora`,`tnsnames.ora`, ou`sqlnet.ora`. Vous ne pouvez pas non plus modifier la structure des répertoires de ces fichiers. RDS L'automatisation personnalisée s'attend à ce que ces fichiers existent dans une structure de répertoire spécifique.

  Pour créer un fichier de paramètres de serveur à partir d'un fichier de paramètres d'initialisation, utilisez la syntaxe suivante.

```
CREATE SPFILE='/rdsdbdata/admin/$ORACLE_SID/pfile/spfile$ORACLE_SID.ora'
    FROM PFILE='/rdsdbdata/config/oracle_pfile';
```

Utilisation de RDS Custom for SQL Server

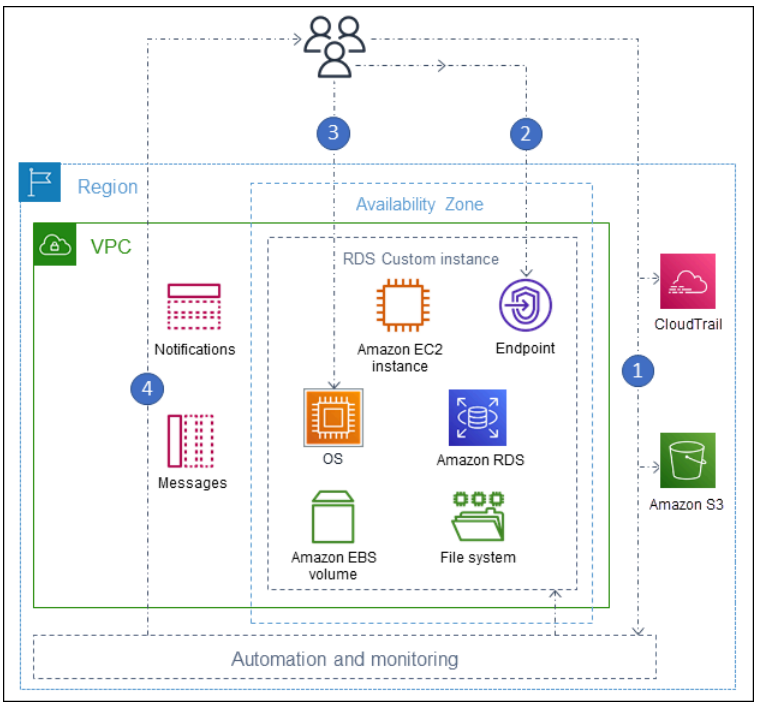
Vous trouverez ci-dessous des instructions pour créer, gérer et maintenir vos instances de base de données RDS personnalisées pour le SQL serveur.

Rubriques

- RDSFlux de travail personnalisé pour SQL le serveur
- Conditions requises et limitations d'Amazon RDS Custom for SQL Server
- Configuration de votre environnement pour Amazon RDS Custom for SQL Server
- Apportez votre propre contenu multimédia avec RDS Custom for SQL Server
- Utilisation de versions de moteur personnalisées pour RDS Custom for SQL Server
- Création et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server
- Gestion d'une instance de base de données Amazon RDS Custom for SQL Server
- Utilisation de Microsoft Active Directory avec RDS Custom for SQL Server
- Gestion d'un déploiement multi-AZ pour RDS Custom for Server SQL
- Sauvegarde et restauration d'une instance de base de données Amazon RDS Custom for SQL Server
- Copier un instantané de base de données Amazon RDS personnalisé pour SQL Server
- Migration d'une base de données sur site vers Amazon RDS Custom for SQL Server
- Mise à niveau d'une instance de base de données Amazon RDS Custom pour SQL Server
- Résolution des problèmes de base de données pour Amazon RDS Custom for SQL Server

RDSFlux de travail personnalisé pour SQL le serveur

Le schéma suivant montre le flux de travail typique de RDS Custom for SQL Server.



La procédure est la suivante :

1. Créez une instance de base de données RDS personnalisée pour le SQL serveur à partir d'une version du moteur proposée par RDS Custom.

   Pour de plus amples informations, veuillez consulter Création d'une instance de base de données RDS personnalisée pour le SQL serveur.
2. Connectez votre application au point de terminaison de l'instance de base de données RDS personnalisée.

   Pour plus d'informations, consultez Connexion à votre instance de base de données RDS personnalisée à l'aide de AWS Systems Manager et Connexion à votre instance de base de données RDS personnalisée à l'aide de RDP.
3. (Facultatif) Accédez à l'hôte pour personnaliser votre logiciel.
4. Surveillez les notifications et les messages générés par l'automatisation RDS personnalisée.

Création d'une instance de base de données pour RDS Custom

Vous créez votre instance de base de données RDS personnalisée à l'aide de la `create-db-instance` commande. La procédure est similaire à la création d'une RDS instance Amazon. Cependant, certains des paramètres sont différents. Pour de plus amples informations, veuillez

consulter Création et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server.

Connexion de la base de données

Comme une RDS instance de base de données Amazon, votre instance de base de données RDS personnalisée pour SQL serveur réside dans unVPC. Votre application se connecte à l'instance RDS personnalisée à l'aide d'un client tel que SQL Server Management Suite (SSMS), comme dans RDS SQL Server.

RDSPersonnalisation personnalisée

Vous pouvez accéder à l'hôte RDS personnalisé pour installer ou personnaliser le logiciel. Pour éviter les conflits entre vos modifications et l'automatisation RDS personnalisée, vous pouvez suspendre l'automatisation pendant une période spécifiée. Pendant cette période, RDS Custom n'effectue pas de surveillance ni de restauration d'instance. À la fin de la période, RDS Custom reprend l'automatisation complète. Pour de plus amples informations, veuillez consulter Suspension et reprise de l'automatisation personnalisée RDS.

Conditions requises et limitations d'Amazon RDS Custom for SQL Server

Retrouvez ci-dessous un résumé des conditions requises et limitations d'Amazon RDS Custom for SQL Server pour une référence rapide. Les conditions requises et les limitations apparaissent également dans les sections pertinentes.

Rubriques

- Disponibilité des régions et des versions
- Exigences générales pour RDS Custom for SQL Server
- Support de classe d'instance de base de données pour RDS Custom for SQL Server
- Limitations pour RDS Custom for SQL Server
- Définition de jeux de caractères et de classements pour les instances de base de données RDS Custom for SQL Server
- Fuseau horaire local pour les instances de base de données RDS Custom for SQL Server
- Utilisation d'une clé principale de service avec RDS Custom for SQL Server
- Prise en charge de la capture des données modifiées (CDC) avec RDS Custom pour SQL Server

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions d'Amazon RDS avec Amazon RDS Custom for SQL Server, consultez Régions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server.

Exigences générales pour RDS Custom for SQL Server

Assurez-vous de respecter ces conditions requises pour Amazon RDS Custom for SQL Server :

- Utilisez les classes d'instance présentées dans Support de classe d'instance de base de données pour RDS Custom for SQL Server. Les seuls types de stockage pris en charge sont les disques SSD (Solid State Drive) des types gp2, gp3, io1 et io2 Block Express. La limite de stockage maximale pour io1, gp2 et gp3 est de 16 TiB, tandis que io2 prend en charge 64 TiB.
- Assurez-vous de disposer d'une AWS KMS clé de chiffrement symétrique pour créer une instance de base de données personnalisée RDS. Pour de plus amples informations, veuillez consulter Vérifiez que vous disposez d'une clé de chiffrement AWS KMS symétrique.

- Assurez-vous de créer un rôle AWS Identity and Access Management (IAM) et un profil d'instance. Pour plus d'informations, consultez Création manuelle de votre profil d'instance et de votre rôle IAM et Création automatique de profils d'instance à l'aide du AWS Management Console.
- Assurez-vous de fournir une configuration réseau que RDS Custom peut utiliser pour accéder à d'autres Services AWS. Pour voir les conditions requises spécifiques, consultez Étape 2 : Configuration du réseau, du profil d'instance et du chiffrement.
- Le nombre combiné d'instances de base de données RDS Custom et Amazon RDS ne dépasse pas votre limite de quota. Par exemple, si votre quota est de 40 instances de base de données, vous pouvez avoir 20 instances de base de données RDS Custom for SQL Server et 20 instances de base de données Amazon RDS.
- RDS Custom crée automatiquement un AWS CloudTrail parcours dont le nom commence `do-not-delete-rds-custom-` par. Le périmètre de support RDS Custom s'appuie sur les événements survenus CloudTrail pour déterminer si vos actions affectent l'automatisation de RDS Custom. RDS Custom crée le journal de suivi lorsque vous créez votre première instance de base de données. Pour utiliser une solution déjà existante CloudTrail, contactez le AWS Support. Pour de plus amples informations, veuillez consulter AWS CloudTrail.

Support de classe d'instance de base de données pour RDS Custom for SQL Server

Vérifiez si la classe d'instance de base de données est prise en charge dans votre région à l'aide de la commande describe-orderable-db-instance-options.

RDSCustom for SQL Server prend en charge les classes d'instances de base de données indiquées dans le tableau suivant :

SQLÉdition serveur	RDSSupport personnalisé
Enterprise Edition	db.r5.xlarge–db.r5.24xlarge db.r5b.xlarge—db.r5b.24xlarge db.m5.xlarge–db.m5.24xlarge db.r6i.xlarge—db.r6i.32xlarge db.m6i.xlarge—db.m6i.32xlarge

SQLÉdition serveur	RDSSupport personnalisé
	db.x2iedn.xlarge–db.x2iedn.32xlarge
Standard Editior	db.r5.large–db.r5.24xlarge db.r5b.large–db.r5b.8xlarge db.m5.large–db.m5.24xlarge db.r6i.large — db.r6i.8xlarge db.m6i.large — db.m6i.8xlarge db.x2iedn.xlarge—db.x2iedn.8xlarge
Edition Développeur	db.r5.xlarge–db.r5.24xlarge db.r5b.xlarge—db.r5b.24xlarge db.m5.xlarge–db.m5.24xlarge db.r6i.xlarge—db.r6i.32xlarge db.m6i.xlarge—db.m6i.32xlarge db.x2iedn.xlarge–db.x2iedn.32xlarge
Web Edition	db.r5.large–db.r5.4xlarge db.m5.large–db.m5.4xlarge db.r6i.large–db.r6i.4xlarge db.m6i.large — db.m6i.4xlarge db.r5b.large–db.r5b.4xlarge

Les recommandations suivantes s'appliquent aux types de classe db.x2iedn :

- Lors de sa création, le stockage local est un périphérique brut et non alloué. Avant d'utiliser une instance de base de données avec cette classe d'instance, vous devez monter et formater le stockage local. Ensuite, configurez-le `tempdb` pour garantir des performances optimales. Pour plus d'informations, consultez Optimiser les performances tempdb dans Amazon RDS Custom for SQL Server à l'aide du stockage d'instance local.
- Le stockage local revient à son état brut et non alloué lorsque vous exécutez des opérations d'instance de base de données telles que le calcul à l'échelle, le remplacement d'instance, la restauration de snapshots ou la point-in-time restauration ()PITR. Dans ces situations, vous devez remonter, reformater et reconfigurer le lecteur et `tempdb` rétablir ses fonctionnalités.
- Pour les instances Multi-AZ, nous vous recommandons d'effectuer la configuration sur une instance de base de données de secours. Ainsi, en cas de basculement, le système continue de fonctionner sans problème car la configuration est déjà en place sur l'instance de secours.

Limitations pour RDS Custom for SQL Server

Les limites suivantes s'appliquent à l'utilisation de RDS Custom for SQL Server :

- Vous ne pouvez pas créer de réplicas en lecture dans les instances de base de données Amazon RDS pour RDS Custom for SQL Server. Toutefois, vous pouvez configurer la haute disponibilité automatiquement avec un déploiement multi-AZ. Pour de plus amples informations, veuillez consulter Gestion d'un déploiement multi-AZ pour RDS Custom for Server SQL.
- Vous ne pouvez pas modifier l'identifiant d'instance de base de données d'une instance de base de données RDS Custom for SQL Server existante.
- Pour une instance de base de données RDS Custom pour SQL Server qui n'a pas été créée avec une version de moteur personnalisée (CEV), la persistance des modifications apportées au système d'exploitation Microsoft Windows n'est pas garantie. Par exemple, vous perdez ces modifications lorsque vous lancez un instantané ou une opération de point-in-time restauration. Si l'instance de base de données RDS Custom for SQL Server a été créée avec une version CEV, ces modifications sont conservées.
- Toutes les options ne sont pas prises en charge. Par exemple, lorsque vous créez une instance de base de données RDS Custom for SQL Server, vous ne pouvez pas effectuer les opérations suivantes :
  - Modifier le nombre de cœurs d'UC et de threads par cœur sur la classe d'instance de base de données.
  - Activer la scalabilité automatique du stockage.

  - Spécifier votre propre groupe de paramètres de base de données, groupe d'options ou jeu de caractères.
  - Activer l'option Performance Insights.
  - Activer la mise à niveau automatique des versions mineures
- Le stockage maximal de l'instance de base de données est de 64 TiO.
- Vous ne pouvez pas utiliser le proxy RDS avec RDS Custom pour SQL Server.
- Vous ne pouvez pas utiliser l'`describe-reserved-db-instances`API pour les instances de base de données RDS Custom pour SQL Server.

Définition de jeux de caractères et de classements pour les instances de base de données RDS Custom for SQL Server

Présentation

Avec les instances de base de données RDS Custom for SQL Server, vous pouvez configurer le jeu de caractères et les paramètres de classement qui déterminent la manière dont les données sont stockées et triées. Les jeux de caractères définissent les caractères autorisés, tandis que les classements définissent les règles de tri et de comparaison des données. Il n'est pas important de définir les jeux de caractères et les classements appropriés pour les applications qui fonctionnent avec des données multilingues ou qui ont des exigences de tri spécifiques. Par exemple, vous devrez peut-être gérer les caractères accentués et définir des règles de tri spécifiques à la langue, ou préserver l'intégrité des données dans différentes régions. Les sections suivantes fournissent des informations sur le jeu de caractères et la prise en charge du classement pour vos instances de base de données RDS Custom for SQL Server.

RDSCustom for SQL Server prend en charge un large éventail de classements de serveur, à la fois en codage traditionnel et en codage UTF -8, pour les SQL paramètres régionaux _latin, japonais, allemand et arabe. Le classement de serveur par défaut est `SQL_Latin1_General_CP1_CI_AS`, mais vous pouvez sélectionner un autre classement pris en charge et l'utiliser. Vous pouvez sélectionner un classement en suivant la même procédure que celle utilisée RDS pour le SQL serveur. Pour de plus amples informations, veuillez consulter Gestion des classements et des jeux de caractères pour Amazon RDS pour Microsoft SQL Server.

Considérations

Les exigences et limitations suivantes s'appliquent lorsque vous travaillez avec des classements de serveur sur RDS Custom for SQL Server :

- Vous pouvez définir le classement du serveur lorsque vous créez une instance de base de données RDS personnalisée pour le SQL serveur. Vous ne pouvez plus modifier le classement au niveau du serveur une fois que l'instance de base de données est créée.
- Vous ne pouvez pas modifier le classement au niveau du serveur lors d'une restauration à partir d'un instantané de base de données ou lors d'une restauration instantanée (PITR).
- Lorsque vous créez une instance de base de données à partir d'un RDS Custom for SQL ServerCEV, l'instance de base de données n'hérite pas du classement du serveur duCEV. Au lieu de cela, c'est le classement de serveur par défaut de `SQL_Latin1_General_CP1_CI_AS` qui est utilisé. Si vous avez configuré un classement de serveur autre que celui par défaut sur un RDS Custom for SQL Server CEV et que vous souhaitez utiliser ce même classement de serveur sur une nouvelle instance de base de données, veillez à sélectionner le même classement lorsque vous créez l'instance de base de données à partir duCEV.

  > **Note**
  >
  > Si le classement que vous sélectionnez lors de la création de l'instance de base de données est différent du classement deCEV, les bases de données SQL du système Microsoft Server sur la nouvelle instance de base de données RDS Custom for SQL Server seront reconstruites pour utiliser le classement mis à jour. Le processus de reconstruction est uniquement effectué sur la nouvelle instance de base de données RDS Custom for SQL Server et n'a aucun impact sur CEV celle-ci. Les modifications précédentes que vous avez apportées aux bases de données système sur le ne CEV seront pas conservées sur la nouvelle instance de base de données RDS personnalisée pour le SQL serveur une fois les bases de données système reconstruites. Certaines modifications incluent des objets définis par l'utilisateur dans la `master` base de données, des tâches planifiées dans la `msdb` base de données ou des modifications des paramètres de base de données par défaut dans la `model` base de données de votreCEV. Vous pouvez recréer manuellement vos modifications une fois que la nouvelle instance de base de données RDS Custom for SQL Server est créée.

- Lorsque vous créez une instance de base de données à partir d'une RDS version personnalisée SQL du moteur Custom for Server (CEV) et que vous sélectionnez un classement différent de celui duCEV, assurez-vous que l'image dorée (AMI) utilisée pour la CEV création répond aux exigences suivantes afin que les bases de données SQL du système Microsoft Server sur la nouvelle instance de base de données puissent être reconstruites :

- Pour SQL Server 2022, assurez-vous que le `setup.exe` fichier se trouve dans le chemin suivant : `C:\Program Files\Microsoft SQL Server\160\Setup Bootstrap \SQL2022\setup.exe`
- Pour SQL Server 2019, assurez-vous que le `setup.exe` fichier se trouve dans le chemin suivant : `C:\Program Files\Microsoft SQL Server\150\Setup Bootstrap \SQL2019\setup.exe`
- Des copies des modèles de données et de journaux pour les bases de données `master`, `model` et `msdb` doivent exister à leurs emplacements par défaut. Pour en savoir plus, consultez Reconstruire les bases de données système dans la documentation publique de Microsoft.
- Assurez-vous que le moteur de base de données de votre SQL serveur utilise `NT Service \MSSQLSERVER` ou `NT AUTHORITY\NETWORK SERVICE` comme compte de service. Aucun autre compte ne disposera des autorisations nécessaires sur le lecteur `C:\` lors de la configuration d'un classement de serveur autre que celui par défaut pour l'instance de base de données.

- Si le classement des serveurs sélectionné pour une nouvelle instance de base de données est le même que celui configuré sur votre instance de base de donnéesCEV, les bases de données SQL du système Microsoft Server sur la nouvelle instance de base de données RDS Custom for SQL Server ne sont pas soumises au processus de reconstruction. Toutes les modifications que vous avez apportées précédemment aux bases de données du système CEV seront automatiquement conservées dans la nouvelle instance de base de données RDS Custom for SQL Server.

Collations prises en charge

Vous pouvez attribuer à votre classement l'une des valeurs répertoriées dans le tableau suivant.

Classement (Collation)	Description
Arabic_100_BIN	Arabic-100, binary sort
Arabic_100_BIN2	Arabic-100, binary code point comparison sort
Arabic_100_CI_AI	Arabic-100, case-insensitive, accent-insensitive , kanatype-insensitive, width-insensitive
Arabic_100_CI_AI_KS	Arabic-100, case-insensitive, accent-insensitive , kanatype-sensitive, width-insensitive

Classement (Collation)	Description
Arabic_100_CI_AI_KS_SC	Arabic-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Arabic_100_CI_AI_KS_SC_UTF8	Arabic-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CI_AI_KS_WS	Arabic-100, case-insensitive, accent-insensitive , kanatype-sensitive, width-sensitive
Arabic_100_CI_AI_KS_WS_SC	Arabic-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Arabic_100_CI_AI_KS_WS_SC_UTF8	Arabic-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Arabic_100_CI_AI_SC	Arabic-100, case-insensitive, accent-insensitive , kanatype-insensitive, width-insensitive, supplementary characters
Arabic_100_CI_AI_SC_UTF8	Arabic-100, case-insensitive, accent-insensitive , kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CI_AI_WS	Arabic-100, case-insensitive, accent-insensitive , kanatype-insensitive, width-sensitive
Arabic_100_CI_AI_WS_SC	Arabic-100, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Arabic_100_CI_AI_WS_SC_UTF8	Arabic-100, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
Arabic_100_CI_AS	Arabic-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Arabic_100_CI_AS_KS	Arabic-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Arabic_100_CI_AS_KS_SC	Arabic-100, case-insensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters
Arabic_100_CI_AS_KS_SC_UTF8	Arabic-100, case-insensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CI_AS_KS_WS	Arabic-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Arabic_100_CI_AS_KS_WS_SC	Arabic-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplemen tary characters
Arabic_100_CI_AS_KS_WS_SC_UTF8	Arabic-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplemen tary characters, UTF8
Arabic_100_CI_AS_SC	Arabic-100, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive, supplementary characters
Arabic_100_CI_AS_SC_UTF8	Arabic-100, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CI_AS_WS	Arabic-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive

Classement (Collation)	Description
Arabic_100_CI_AS_WS_SC	Arabic-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Arabic_100_CI_AS_WS_SC_UTF8	Arabic-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Arabic_100_CS_AI	Arabic-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Arabic_100_CS_AI_KS	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Arabic_100_CS_AI_KS_SC	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
Arabic_100_CS_AI_KS_SC_UTF8	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CS_AI_KS_WS	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Arabic_100_CS_AI_KS_WS_SC	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
Arabic_100_CS_AI_KS_WS_SC_UTF8	Arabic-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Arabic_100_CS_AI_SC	Arabic-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters

Classement (Collation)	Description
Arabic_100_CS_AI_SC_UTF8	Arabic-100, case-sensitive, accent-insensitive , kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CS_AI_WS	Arabic-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Arabic_100_CS_AI_WS_SC	Arabic-100, case-sensitive, accent-insensitive , kanatype-insensitive, width-sensitive, supplementary characters
Arabic_100_CS_AI_WS_SC_UTF8	Arabic-100, case-sensitive, accent-insensitive , kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Arabic_100_CS_AS	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Arabic_100_CS_AS_KS	Arabic-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Arabic_100_CS_AS_KS_SC	Arabic-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters
Arabic_100_CS_AS_KS_SC_UTF8	Arabic-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CS_AS_KS_WS	Arabic-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Arabic_100_CS_AS_KS_WS_SC	Arabic-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplemen tary characters

Classement (Collation)	Description
Arabic_100_CS_AS_KS_WS_SC_UTF8	Arabic-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Arabic_100_CS_AS_SC	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Arabic_100_CS_AS_SC_UTF8	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Arabic_100_CS_AS_WS	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Arabic_100_CS_AS_WS_SC	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Arabic_100_CS_AS_WS_SC_UTF8	Arabic-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Arabic_BIN	Arabic, binary sort
Arabic_BIN2	Arabic, binary code point comparison sort
Arabic_CI_AI	Arabic, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Arabic_CI_AI_KS	Arabic, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Arabic_CI_AI_KS_WS	Arabic, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Arabic_CI_AI_WS	Arabic, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive

Classement (Collation)	Description
Arabic_CI_AS	Arabic, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Arabic_CI_AS_KS	Arabic, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Arabic_CI_AS_KS_WS	Arabic, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Arabic_CI_AS_WS	Arabic, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Arabic_CS_AI	Arabic, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Arabic_CS_AI_KS	Arabic, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Arabic_CS_AI_KS_WS	Arabic, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Arabic_CS_AI_WS	Arabic, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Arabic_CS_AS	Arabic, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Arabic_CS_AS_KS	Arabic, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Arabic_CS_AS_KS_WS	Arabic, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Arabic_CS_AS_WS	Arabic, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Chinese_PRC_BIN2	Chinese-PRC, binary code point comparison sort

Classement (Collation)	Description
Chinese_PRC_CI_AS	Chinese-PRC, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
Chinese_Taiwan_Stroke_CI_AS	Chinese-Taiwan-Stroke, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Danish_Norwegian_CI_AS	Danish-Norwegian, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
Finnish_Swedish_CI_AS	Finnish-Swedish, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
French_CI_AS	French, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
German_PhoneBook_100_BIN	German-PhoneBook-100, binary sort
German_PhoneBook_100_BIN2	German-PhoneBook-100, binary code point comparison sort
German_PhoneBook_100_CI_AI	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
German_PhoneBook_100_CI_AI_KS	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_100_CI_AI_KS_SC	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CI_AI_KS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8

Classement (Collation)	Description
German_PhoneBook_100_CI_AI_KS_WS	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_100_CI_AI_KS_WS_SC	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CI_AI_KS_WS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AI_SC	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CI_AI_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AI_WS	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive
German_PhoneBook_100_CI_AI_WS_SC	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CI_AI_WS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AS	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

Classement (Collation)	Description
German_PhoneBook_100_CI_AS_KS	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_100_CI_AS_KS_SC	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CI_AS_KS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AS_KS_WS	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_100_CI_AS_KS_WS_SC	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CI_AS_KS_WS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AS_SC	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CI_AS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
German_PhoneBook_100_CI_AS_WS	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive

Classement (Collation)	Description
German_PhoneBook_100_CI_AS_WS_SC	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CI_AS_WS_SC_UTF8	German-PhoneBook-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_100_CS_AI	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive
German_PhoneBook_100_CS_AI_KS	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_100_CS_AI_KS_SC	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CS_AI_KS_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
German_PhoneBook_100_CS_AI_KS_WS	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_100_CS_AI_KS_WS_SC	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CS_AI_KS_WS_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
German_PhoneBook_100_CS_AI_SC	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CS_AI_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
German_PhoneBook_100_CS_AI_WS	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive
German_PhoneBook_100_CS_AI_WS_SC	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CS_AI_WS_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_100_CS_AS	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
German_PhoneBook_100_CS_AS_KS	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_100_CS_AS_KS_SC	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
German_PhoneBook_100_CS_AS_KS_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8

Classement (Collation)	Description
German_PhoneBook_100_CS_AS_KS_WS	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_100_CS_AS_KS_WS_SC	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
German_PhoneBook_100_CS_AS_KS_WS_SC_UTF8	German-PhoneBook-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
German_PhoneBook_BIN	German-PhoneBook, binary sort
German_PhoneBook_BIN2	German-PhoneBook, binary code point comparison sort
German_PhoneBook_CI_AI	German-PhoneBook, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
German_PhoneBook_CI_AI_KS	German-PhoneBook, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_CI_AI_KS_WS	German-PhoneBook, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_CI_AI_WS	German-PhoneBook, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive
German_PhoneBook_CI_AS	German-PhoneBook, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

Classement (Collation)	Description
German_PhoneBook_CI_AS_KS	German-PhoneBook, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_CI_AS_KS_WS	German-PhoneBook, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_CI_AS_WS	German-PhoneBook, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
German_PhoneBook_CS_AI	German-PhoneBook, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive
German_PhoneBook_CS_AI_KS	German-PhoneBook, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive
German_PhoneBook_CS_AI_KS_WS	German-PhoneBook, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive
German_PhoneBook_CS_AI_WS	German-PhoneBook, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive
German_PhoneBook_CS_AS	German-PhoneBook, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
German_PhoneBook_CS_AS_KS	German-PhoneBook, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive
German_PhoneBook_CS_AS_KS_WS	German-PhoneBook, case-sensitive, accent-se nsitive, kanatype-sensitive, width-sensitive
German_PhoneBook_CS_AS_WS	German-PhoneBook, case-sensitive, accent-se nsitive, kanatype-insensitive, width-sensitive
Hebrew_BIN	Hebrew, binary sort
Hebrew_CI_AS	Hebrew, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

Classement (Collation)	Description
Japanese_90_BIN	Japanese-90, binary sort
Japanese_90_BIN2	Japanese-90, binary code point comparison sort
Japanese_90_CI_AI	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-ins ensitive
Japanese_90_CI_AI_KS	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive
Japanese_90_CI_AI_KS_SC	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_90_CI_AI_KS_SC_UTF8	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CI_AI_KS_WS	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive
Japanese_90_CI_AI_KS_WS_SC	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_90_CI_AI_KS_WS_SC_UTF8	Japanese-90, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CI_AI_SC	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters

Classement (Collation)	Description
Japanese_90_CI_AI_SC_UTF8	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, UTF8
Japanese_90_CI_AI_WS	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive
Japanese_90_CI_AI_WS_SC	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_90_CI_AI_WS_SC_UTF8	Japanese-90, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CI_AS	Japanese-90, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
Japanese_90_CI_AS_KS	Japanese-90, case-insensitive, accent-se nsitive, kanatype-sensitive, width-insensitive
Japanese_90_CI_AS_KS_SC	Japanese-90, case-insensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_90_CI_AS_KS_SC_UTF8	Japanese-90, case-insensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CI_AS_KS_WS	Japanese-90, case-insensitive, accent-se nsitive, kanatype-sensitive, width-sensitive
Japanese_90_CI_AS_KS_WS_SC	Japanese-90, case-insensitive, accent-se nsitive, kanatype-sensitive, width-sensitive, supplementary characters

Classement (Collation)	Description
Japanese_90_CI_AS_KS_WS_SC_UTF8	Japanese-90, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CI_AS_SC	Japanese-90, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_90_CI_AS_SC_UTF8	Japanese-90, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CI_AS_WS	Japanese-90, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_90_CI_AS_WS_SC	Japanese-90, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_90_CI_AS_WS_SC_UTF8	Japanese-90, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CS_AI	Japanese-90, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Japanese_90_CS_AI_KS	Japanese-90, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Japanese_90_CS_AI_KS_SC	Japanese-90, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_90_CS_AI_KS_SC_UTF8	Japanese-90, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_90_CS_AI_KS_WS	Japanese-90, case-sensitive, accent-insensitive , kanatype-sensitive, width-sensitive
Japanese_90_CS_AI_KS_WS_SC	Japanese-90, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_90_CS_AI_KS_WS_SC_UTF8	Japanese-90, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CS_AI_SC	Japanese-90, case-sensitive, accent-insensitive , kanatype-insensitive, width-insensitive, supplementary characters
Japanese_90_CS_AI_SC_UTF8	Japanese-90, case-sensitive, accent-insensitive , kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CS_AI_WS	Japanese-90, case-sensitive, accent-insensitive , kanatype-insensitive, width-sensitive
Japanese_90_CS_AI_WS_SC	Japanese-90, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_90_CS_AI_WS_SC_UTF8	Japanese-90, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CS_AS	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_90_CS_AS_KS	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive

Classement (Collation)	Description
Japanese_90_CS_AS_KS_SC	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_90_CS_AS_KS_SC_UTF8	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CS_AS_KS_WS	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_90_CS_AS_KS_WS_SC	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_90_CS_AS_KS_WS_SC_UTF8	Japanese-90, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_90_CS_AS_SC	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_90_CS_AS_SC_UTF8	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_90_CS_AS_WS	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_90_CS_AS_WS_SC	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_90_CS_AS_WS_SC_UTF8	Japanese-90, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_BIN	Japanese, binary sort
Japanese_BIN2	Japanese, binary code point comparison sort
Japanese_Bushu_Kakusu_100_BIN	Japanese-Bushu-Kakusu-100, binary sort
Japanese_Bushu_Kakusu_100_BIN2	Japanese-Bushu-Kakusu-100, binary code point comparison sort
Japanese_Bushu_Kakusu_100_CI_AI	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CI_AI_KS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CI_AI_KS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AI_KS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CI_AI_KS_WS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CI_AI_KS_WS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AI_KS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_100_CI_AI_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AI_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CI_AI_WS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CI_AI_WS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AI_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CI_AS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CI_AS_KS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CI_AS_KS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AS_KS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_100_CI_AS_KS_WS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CI_AS_KS_WS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AS_KS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CI_AS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CI_AS_WS	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CI_AS_WS_SC	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CI_AS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AI	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive

Classement (Collation)	Description
Japanese_Bushu_Kakusu_100_CS_AI_KS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CS_AI_KS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AI_KS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AI_KS_WS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CS_AI_KS_WS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AI_KS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AI_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AI_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AI_WS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive

Classement (Collation)	Description
Japanese_Bushu_Kakusu_100_CS_AI_WS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AI_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CS_AS_KS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_Bushu_Kakusu_100_CS_AS_KS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AS_KS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AS_KS_WS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CS_AS_KS_WS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AS_KS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_100_CS_AS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_100_CS_AS_WS	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_Bushu_Kakusu_100_CS_AS_WS_SC	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_Bushu_Kakusu_100_CS_AS_WS_SC_UTF8	Japanese-Bushu-Kakusu-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_Bushu_Kakusu_140_BIN	Japanese-Bushu-Kakusu-140, binary sort
Japanese_Bushu_Kakusu_140_BIN2	Japanese-Bushu-Kakusu-140, binary code point comparison sort
Japanese_Bushu_Kakusu_140_CI_AI	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AI_KS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CI_AI_KS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_KS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AI_KS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_KS_WS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AI_KS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplement ary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_KS_WS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AI_KS_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CI_AI_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AI_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_WS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AI_WS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AI_WS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AI_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CI_AS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AS_KS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AS_KS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AS_KS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AS_KS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AS_KS_WS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AS_KS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CI_AS_KS_WS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AS_KS_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CI_AS_WS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CI_AS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CI_AS_WS_VSS	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CI_AS_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AI_KS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AI_KS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI_KS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AI_KS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CS_AI_KS_WS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AI_KS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI_KS_WS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AI_KS_WS_VSS_UTF8	Japanese-Bushu-Kaku su-140, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AI_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CS_AI_WS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AI_WS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AI_WS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AI_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AS_KS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AS_KS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CS_AS_KS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AS_KS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS_KS_WS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AS_KS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS_KS_WS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AS_KS_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_Bushu_Kakusu_140_CS_AS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS_WS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_Bushu_Kakusu_140_CS_AS_WS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_Bushu_Kakusu_140_CS_AS_WS_VSS	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_Bushu_Kakusu_140_CS_AS_WS_VSS_UTF8	Japanese-Bushu-Kakusu-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_CI_AI	Japanese, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Japanese_CI_AI_KS	Japanese, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive

Classement (Collation)	Description
Japanese_CI_AI_KS_WS	Japanese, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Japanese_CI_AI_WS	Japanese, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Japanese_CI_AS	Japanese, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_CI_AS_KS	Japanese, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_CI_AS_KS_WS	Japanese, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_CI_AS_WS	Japanese, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_CS_AI	Japanese, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Japanese_CS_AI_KS	Japanese, case-sensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Japanese_CS_AI_KS_WS	Japanese, case-sensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Japanese_CS_AI_WS	Japanese, case-sensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Japanese_CS_AS	Japanese, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_CS_AS_KS	Japanese, case-sensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_CS_AS_KS_WS	Japanese, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive

Classement (Collation)	Description
Japanese_CS_AS_WS	Japanese, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_Unicode_BIN	Japanese-Unicode, binary sort
Japanese_Unicode_BIN2	Japanese-Unicode, binary code point comparison sort
Japanese_Unicode_CI_AI	Japanese-Unicode, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Japanese_Unicode_CI_AI_KS	Japanese-Unicode, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive
Japanese_Unicode_CI_AI_KS_WS	Japanese-Unicode, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive
Japanese_Unicode_CI_AI_WS	Japanese-Unicode, case-insensitive, accent-insensitive, kanatype-insensitive, width-sensitive
Japanese_Unicode_CI_AS	Japanese-Unicode, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Japanese_Unicode_CI_AS_KS	Japanese-Unicode, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_Unicode_CI_AS_KS_WS	Japanese-Unicode, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_Unicode_CI_AS_WS	Japanese-Unicode, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_Unicode_CS_AI	Japanese-Unicode, case-sensitive, accent-insensitive, kanatype-insensitive, width-insensitive

Classement (Collation)	Description
Japanese_Unicode_CS_AI_KS	Japanese-Unicode, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive
Japanese_Unicode_CS_AI_KS_WS	Japanese-Unicode, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive
Japanese_Unicode_CS_AI_WS	Japanese-Unicode, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive
Japanese_Unicode_CS_AS	Japanese-Unicode, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
Japanese_Unicode_CS_AS_KS	Japanese-Unicode, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive
Japanese_Unicode_CS_AS_KS_WS	Japanese-Unicode, case-sensitive, accent-se nsitive, kanatype-sensitive, width-sensitive
Japanese_Unicode_CS_AS_WS	Japanese-Unicode, case-sensitive, accent-se nsitive, kanatype-insensitive, width-sensitive
Japanese_XJIS_100_BIN	Japanese-XJIS-100, binary sort
Japanese_XJIS_100_BIN2	Japanese-XJIS-100, binary code point comparison sort
Japanese_XJIS_100_CI_AI	Japanese-XJIS-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive
Japanese_XJIS_100_CI_AI_KS	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive
Japanese_XJIS_100_CI_AI_KS_SC	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters

Classement (Collation)	Description
Japanese_XJIS_100_CI_AI_KS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AI_KS_WS	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-sensitive, width-sensitive
Japanese_XJIS_100_CI_AI_KS_WS_SC	Japanese-XJIS-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CI_AI_KS_WS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AI_SC	Japanese-XJIS-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters
Japanese_XJIS_100_CI_AI_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AI_WS	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive
Japanese_XJIS_100_CI_AI_WS_SC	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CI_AI_WS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AS	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive

Classement (Collation)	Description
Japanese_XJIS_100_CI_AS_KS	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Japanese_XJIS_100_CI_AS_KS_SC	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_XJIS_100_CI_AS_KS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AS_KS_WS	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive
Japanese_XJIS_100_CI_AS_KS_WS_SC	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CI_AS_KS_WS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AS_SC	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters
Japanese_XJIS_100_CI_AS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, UTF8
Japanese_XJIS_100_CI_AS_WS	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_XJIS_100_CI_AS_WS_SC	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters

Classement (Collation)	Description
Japanese_XJIS_100_CI_AS_WS_SC_UTF8	Japanese-XJIS-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AI	Japanese-XJIS-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive
Japanese_XJIS_100_CS_AI_KS	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive
Japanese_XJIS_100_CS_AI_KS_SC	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_XJIS_100_CS_AI_KS_SC_UTF8	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AI_KS_WS	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive
Japanese_XJIS_100_CS_AI_KS_WS_SC	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CS_AI_KS_WS_SC_UTF 8	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AI_SC	Japanese-XJIS-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters
Japanese_XJIS_100_CS_AI_SC_UTF8	Japanese-XJIS-100, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_XJIS_100_CS_AI_WS	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive
Japanese_XJIS_100_CS_AI_WS_SC	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CS_AI_WS_SC_UTF8	Japanese-XJIS-100, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AS	Japanese-XJIS-100, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive
Japanese_XJIS_100_CS_AS_KS	Japanese-XJIS-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive
Japanese_XJIS_100_CS_AS_KS_SC	Japanese-XJIS-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters
Japanese_XJIS_100_CS_AS_KS_SC_UTF8	Japanese-XJIS-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AS_KS_WS	Japanese-XJIS-100, case-sensitive, accent-se nsitive, kanatype-sensitive, width-sensitive
Japanese_XJIS_100_CS_AS_KS_WS_SC	Japanese-XJIS-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters
Japanese_XJIS_100_CS_AS_KS_WS_SC_UTF 8	Japanese-XJIS-100, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, UTF8

Classement (Collation)	Description
Japanese_XJIS_100_CS_AS_SC	Japanese-XJIS-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters
Japanese_XJIS_100_CS_AS_SC_UTF8	Japanese-XJIS-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8
Japanese_XJIS_100_CS_AS_WS	Japanese-XJIS-100, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive
Japanese_XJIS_140_BIN	Japanese-XJIS-140, binary sort
Japanese_XJIS_140_BIN2	Japanese-XJIS-140, binary code point comparison sort
Japanese_XJIS_140_CI_AI	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AI_KS	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AI_KS_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AI_KS_VSS	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive

Classement (Collation)	Description
Japanese_XJIS_140_CI_AI_KS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AI_KS_WS	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AI_KS_WS_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AI_KS_WS_VSS	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AI_KS_WS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AI_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AI_VSS	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive

Classement (Collation)	Description
Japanese_XJIS_140_CI_AI_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AI_WS	Japanese-XJIS-140, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AI_WS_UTF8	Japanese-XJIS-140, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AI_WS_VSS	Japanese-XJIS-140, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AI_WS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AS_KS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive

Classement (Collation)	Description
Japanese_XJIS_140_CI_AS_KS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AS_KS_VSS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AS_KS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AS_KS_WS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AS_KS_WS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AS_KS_WS_VSS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AS_KS_WS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CI_AS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AS_VSS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CI_AS_WS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CI_AS_WS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CI_AS_WS_VSS	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CI_AS_WS_VSS_UTF8	Japanese-XJIS-140, case-insensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CS_AI	Japanese-XJIS-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AI_KS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AI_KS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AI_KS_VSS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AI_KS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CS_AI_KS_WS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AI_KS_WS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CS_AI_KS_WS_VSS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AI_KS_WS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CS_AI_UTF8	Japanese-XJIS-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AI_VSS	Japanese-XJIS-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AI_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CS_AI_WS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AI_WS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CS_AI_WS_VSS	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AI_WS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-in sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CS_AS	Japanese-XJIS-140, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AS_KS	Japanese-XJIS-140, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AS_KS_UTF8	Japanese-XJIS-140, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AS_KS_VSS	Japanese-XJIS-140, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AS_KS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-se nsitive, kanatype-sensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CS_AS_KS_WS	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AS_KS_WS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AS_KS_WS_VSS	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AS_KS_WS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-sensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Japanese_XJIS_140_CS_AS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AS_VSS	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, variation selector sensitive, UTF8

Classement (Collation)	Description
Japanese_XJIS_140_CS_AS_WS	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive
Japanese_XJIS_140_CS_AS_WS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector insensitive, UTF8
Japanese_XJIS_140_CS_AS_WS_VSS	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive
Japanese_XJIS_140_CS_AS_WS_VSS_UTF8	Japanese-XJIS-140, case-sensitive, accent-sensitive, kanatype-insensitive, width-sensitive, supplementary characters, variation selector sensitive, UTF8
Korean_Wansung_CI_AS	Korean-Wansung, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Latin1_General_100_BIN	Latin1-General-100, binary sort
Latin1_General_100_BIN2	Latin1-General-100, binary code point comparison sort
Latin1_General_100_BIN2_UTF8	Latin1-General-100, binary code point comparison sort, UTF8
Latin1_General_100_CI_AS	Latin1-General-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Latin1_General_100_CI_AS_SC_UTF8	Latin1-General-100, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive, supplementary characters, UTF8

Classement (Collation)	Description
Latin1_General_BIN	Latin1-General, binary sort
Latin1_General_BIN2	Latin1-General, binary code point comparison sort
Latin1_General_CI_AI	Latin1-General, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive
Latin1_General_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Latin1_General_CI_AS_KS	Latin1-General, case-insensitive, accent-sensitive, kanatype-sensitive, width-insensitive
Latin1_General_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive
Modern_Spanish_CI_AS	Modern-Spanish, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive
SQL_1xCompat_CP850_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 49 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_CP1_CI_AI	Latin1-General, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 54 on Code Page 1252 for non-Unicode Data
SQL_Latin1_General_CP1_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 52 on Code Page 1252 for non-Unicode Data

Classement (Collation)	Description
SQL_Latin1_General_CP1_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 51 on Code Page 1252 for non-Unicode Data
SQL_Latin1_General_CP1250_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 82 on Code Page 1250 for non-Unicode Data
SQL_Latin1_General_CP1250_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 81 on Code Page 1250 for non-Unicode Data
SQL_Latin1_General_CP1251_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 106 on Code Page 1251 for non-Unicode Data
SQL_Latin1_General_CP1251_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 105 on Code Page 1251 for non-Unicode Data
SQL_Latin1_General_CP1253_CI_AI	Latin1-General, case-insensitive, accent-insensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 124 on Code Page 1253 for non-Unicode Data
SQL_Latin1_General_CP1253_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 114 on Code Page 1253 for non-Unicode Data

Classement (Collation)	Description
SQL_Latin1_General_CP1253_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 113 on Code Page 1253 for non-Unicode Data
SQL_Latin1_General_CP1254_CI_AS	Turkish, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 130 on Code Page 1254 for non-Unicode Data
SQL_Latin1_General_CP1254_CS_AS	Turkish, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 129 on Code Page 1254 for non-Unicode Data
SQL_Latin1_General_CP1255_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 138 on Code Page 1255 for non-Unicode Data
SQL_Latin1_General_CP1255_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 137 on Code Page 1255 for non-Unicode Data
SQL_Latin1_General_CP1256_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 146 on Code Page 1256 for non-Unicode Data
SQL_Latin1_General_CP1256_CS_AS	Latin1-General, case-sensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 145 on Code Page 1256 for non-Unicode Data

Classement (Collation)	Description
SQL_Latin1_General_CP1257_CI_AS	Latin1-General, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 154 on Code Page 1257 for non-Unicode Data
SQL_Latin1_General_CP1257_CS_AS	Latin1-General, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 153 on Code Page 1257 for non-Unicode Data
SQL_Latin1_General_CP437_BIN	Latin1-General, binary sort for Unicode Data, SQL Server Sort Order 30 on Code Page 437 for non-Unicode Data
SQL_Latin1_General_CP437_BIN2	Latin1-General, binary code point comparison sort for Unicode Data, SQL Server Sort Order 30 on Code Page 437 for non-Unicode Data
SQL_Latin1_General_CP437_CI_AI	Latin1-General, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive for Unicode Data, SQL Server Sort Order 34 on Code Page 437 for non-Unicode Data
SQL_Latin1_General_CP437_CI_AS	Latin1-General, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 32 on Code Page 437 for non-Unicode Data
SQL_Latin1_General_CP437_CS_AS	Latin1-General, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 31 on Code Page 437 for non-Unicode Data

Classement (Collation)	Description
SQL_Latin1_General_CP850_BIN	Latin1-General, binary sort for Unicode Data, SQL Server Sort Order 40 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_CP850_BIN2	Latin1-General, binary code point comparison sort for Unicode Data, SQL Server Sort Order 40 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_CP850_CI_AI	Latin1-General, case-insensitive, accent-insensitive, kanatype-insensitive, width-ins ensitive for Unicode Data, SQL Server Sort Order 44 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_CP850_CI_AS	Latin1-General, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 42 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_CP850_CS_AS	Latin1-General, case-sensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 41 on Code Page 850 for non-Unicode Data
SQL_Latin1_General_Pref_CP1_CI_AS	Latin1-General, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 53 on Code Page 1252 for non-Unicode Data
SQL_Latin1_General_Pref_CP437_CI_AS	Latin1-General, case-insensitive, accent-se nsitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 33 on Code Page 437 for non-Unicode Data

Classement (Collation)	Description
SQL_Latin1_General_Pref_CP850_CI_AS	Latin1-General, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive for Unicode Data, SQL Server Sort Order 43 on Code Page 850 for non-Unicode Data
Thai_CI_AS	Thai, case-insensitive, accent-sensitive, kanatype-insensitive, width-insensitive

Fuseau horaire local pour les instances de base de données RDS Custom for SQL Server

Le fuseau horaire d'une instance de base de données RDS Custom for SQL Server est défini par défaut. La valeur par défaut actuelle est le temps universel coordonné (UTC). Vous pouvez définir le fuseau horaire de votre instance de base de données à un fuseau horaire local, correspondant à celui de vos applications.

Vous définissez le fuseau horaire lorsque vous créez votre instance de base de données. Vous pouvez créer votre instance de base de données à l'AWS Management Consoleaide de l'reateDBInstanceaction Amazon RDS API C ou de la AWS CLI create-db-instancecommande.

Si votre instance de base de données fait partie d'un déploiement multi-AZ, lorsque vous basculez, votre fuseau horaire demeure le fuseau horaire local que vous avez défini.

Lorsque vous demandez une point-in-time restauration, vous spécifiez l'heure à laquelle la restauration doit être effectuée. L'heure est affichée dans votre fuseau horaire local. Pour plus d'informations, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Ci-après les restrictions affectant la définition du fuseau horaire local sur votre instance de base de données :

- Vous pouvez configurer le fuseau horaire d'une instance de base de données lors de sa création, mais vous ne pouvez pas modifier le fuseau horaire d'une instance de base de données RDS Custom for SQL Server existante.

- Si le fuseau horaire est modifié pour une instance de base de données RDS Custom for SQL Server existante, RDS Custom change le statut de l'instance de base de `unsupported-configuration` données en et envoie des notifications d'événements.
- Vous ne pouvez pas restaurer un instantané à partir d'une instance de base de données dans un fuseau horaire dans une instance de base de données d'un autre fuseau horaire.
- Nous vous recommandons vivement de ne pas restaurer de fichier de sauvegarde d'un fuseau horaire dans un autre fuseau horaire. Si vous restaurez un fichier de sauvegarde d'un fuseau horaire dans un autre fuseau horaire, vous devez auditer vos requêtes et vos applications afin de déterminer les effets du changement de fuseau horaire. Pour plus d'informations, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Fuseaux horaires pris en charge

Vous pouvez définir votre fuseau horaire local avec l'une des valeurs du tableau suivant.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Afghanistan	(UTC+ 04H30)	Kaboul	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Alaska	(UTC— 09H00)	Alaska	
Heure normale Aléoutiennes	(UTC— 10H00)	Îles Aléoutiennes	
Heure normale de l'Altaï	(UTC+ 07h00)	Barnaul, Gorno-Altaysk	
Heure normale arabe	(UTC+ 03:00)	Koweït, Riyad	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale des Emirats Arabe Unis	(UTC+ 04:00)	Abou Dhabi, Mascate	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale Arabie saoudite	(UTC+ 03:00)	Bagdad	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Argentine	(UTC— 03H00)	Ville de Buenos Aires	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Astrakhan	(UTC+ 04:00)	Astrakhan, Oulianovsk	
Heure normale de l'Atlantique	(UTC— 04H00)	Heure de l'Atlanti que (Canada)	
AUSHeure normale du Centre	(UTC+ 09H30)	Darwin	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Australie centrale	(UTC+ 08h45)	Eucla	
AUSHeure normale de l'Est	(UTC+ 10 H)	Canberra, Melbourne, Sydney	
Heure normale d'Azerbaïdjan	(UTC+ 04:00)	Bakou	
Heure normale des Açores	(UTC— 01H00)	Açores	
Heure normale de Bahia	(UTC— 03H00)	Salvador	
Heure normale du Banglades h	(UTC+ 06:00)	Dacca	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de Biélorussie	(UTC+ 03:00)	Minsk	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Bougainville	(UTC+ 11 h)	Île de Bougainville	
Heure normale du Canada central	(UTC— 06H00)	Saskatchewan	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Cap-Vert	(UTC— 01H00)	Cap-Vert	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Caucase	(UTC+ 04:00)	Erevan	
Heure normale de l'Australie centrale	(UTC+ 09H30)	Adélaïde	
Heure normale de l'Amérique centrale	(UTC— 06H00)	Amérique centrale	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Asie centrale	(UTC+ 06:00)	Astana	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Brésil central	(UTC— 04H00)	Cuiabá	
Heure normale de l'Europe centrale	(UTC+ 01:00)	Belgrade, Bratislava, Budapest, Ljubljana, Prague	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de l'Europe centrale	(UTC+ 01:00)	Sarajevo, Skopje, Varsovie, Zagreb	
Heure normale du Pacifique central	(UTC+ 11 h)	Îles Salomon, Nouvelle-Calédonie	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Centre	(UTC— 06H00)	Heure centrale (États-Unis et Canada)	
Heure normale du Centre (Mexique)	(UTC— 06H00)	Guadalajara, Mexico, Monterrey	
Heure normale des îles Chatham	(UTC+ 12h45)	Îles Chatham	
Heure normale de Chine	(UTC+ 08h00)	Pékin, Chongqing, Hong Kong, Urumqi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Cuba	(UTC— 05H00)	La Havane	
Heure normale de la ligne de changement de date	(UTC— 12H00)	Ligne de changement de date internationale Ouest	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique de l'Est	(UTC+ 03:00)	Nairobi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Australie de l'Est	(UTC+ 10 H)	Brisbane	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Europe de l'Est	(UTC+ 02:00)	Chi#inău	
Heure normale d'Amérique du Sud est	(UTC— 03H00)	Brasilia	
Heure normale de l'île de Pâques	(UTC— 06H00)	Île de Pâques	
Heure normale de l'Est	(UTC— 05H00)	Heure de l'Est (États-Unis et Canada)	
Heure normale de l'Est (Mexique)	(UTC— 05H00)	Chetumal	
Heure normale de l'Égypte	(UTC+ 02:00)	Le Caire	
Heure normale d'Iekater inbourg	(UTC+ 05:00)	Iekaterinbourg	
Heure normale des Fidji	(UTC+ 12H00)	Fidji	
FLEHeure standard	(UTC+ 02:00)	Helsinki, Kiev, Riga, Sofia, Tallinn, Vilnius	
Heure normale de Géorgie	(UTC+ 04:00)	Tbilisi	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
GMTHeure standard	(UTC)	Dublin, Edimbourg, Lisbonne, Londres	Ce fuseau horaire n'est pas le même que l'heure moyenne de Greenwich (GMT). Ce fuseau horaire respecte l'heure d'été.
Heure normale du Groenland	(UTC— 03H00)	Groenland	
Heure normale de Greenwich	(UTC)	Monrovia, Reykjavik	Ce fuseau horaire ne respecte pas l'heure d'été.
GTBHeure standard	(UTC+ 02:00)	Athènes, Bucarest	
Heure normale d'Haïti	(UTC— 05H00)	Haïti	
Heure normale de Hawaï	(UTC— 10H00)	Hawaii	
Heure normale d'Inde	(UTC+ 05H30)	Chennai, Calcutta, Mumbai, New Delhi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Iran	(UTC+ 03H30)	Téhéran	
Heure normale d'Israël	(UTC+ 02:00)	Jérusalem	
Heure normale de Jordanie	(UTC+ 02:00)	Amman	
Heure normale de Kaliningrad	(UTC+ 02:00)	Kaliningrad	
Heure normale du Kamtchatka	(UTC+ 12H00)	Petropavlovsk-Kamchatsky – Ancienne	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de Corée	(UTC+ 09h00)	Séoul	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Libye	(UTC+ 02:00)	Tripoli	
Heure normale des îles de la Ligne	(UTC+ 14h00)	Île Christmas	
Heure normale de l'île Lord Howe	(UTC+ 10 H 30)	Île Lord Howe	
Heure normale de Magadan	(UTC+ 11 h)	Magadan	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale en Magallanes	(UTC— 03H00)	Punta Arenas	
Heure normale des Marquises	(UTC— 9 H 30)	Îles Marquises	
Heure normale de Maurice	(UTC+ 04:00)	Port Louis	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Moyen-Orient	(UTC+ 02:00)	Beyrouth	
Heure normale de Montevideo	(UTC— 03H00)	Montevideo	
Heure normale du Maroc	(UTC+ 01:00)	Casablanca	
Heure normale des Rocheuses	(UTC— 07h00)	Heure des Rocheuses (États-Unis et Canada)	
Heure normale des Rocheuses (Mexique)	(UTC— 07h00)	Chihuahua, La Paz, Mazatlán	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Myanmar	(UTC+ 06H30)	Yangon (Rangoun)	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Asie centrale nord	(UTC+ 07h00)	Novossibirsk	
Heure normale de Namibie	(UTC+ 02:00)	Windhoek	
Heure normale du Népal	(UTC+ 05:45)	Katmandou	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Nouvelle-Zélande	(UTC+ 12H00)	Auckland, Wellington	
Heure normale de Terre-Neuve	(UTC— 03H30)	Terre-Neuve	
Heure normale de l'île Norfolk	(UTC+ 11 h)	Île Norfolk	
Heure normale de l'Asie du Nord-Est	(UTC+ 08h00)	Irkoutsk	
Heure normale de l'Asie du Nord	(UTC+ 07h00)	Krasnoïarsk	
Heure normale de la Corée du Nord	(UTC+ 09h00)	Pyongyang	
Heure normale d'Omsk	(UTC+ 06:00)	Omsk	
Heure normale du Pacifique	(UTC— 03H00)	Santiago	
Heure normale du Pacifique	(UTC— 08H00)	Heure du Pacifique (États-Unis et Canada)	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Pacifique (Mexique)	(UTC— 08H00)	Basse-Californie	
Heure normale du Pakistan	(UTC+ 05:00)	Islamabad, Karachi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Paraguay	(UTC— 04H00)	Asunción	
Heure normale Romance	(UTC+ 01:00)	Bruxelles, Copenhague, Madrid, Paris	
Fuseau horaire 10 Russie	(UTC+ 11 h)	Chokurdakh	
Fuseau horaire 11 Russie	(UTC+ 12H00)	Anadyr, Petropavl ovsk-Kamchatsky	
Fuseau horaire 3 Russie	(UTC+ 04:00)	Izhevsk, Samara	
Heure normale de Russie	(UTC+ 03:00)	Moscou, Saint-Pét ersbourg, Volgograd	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Est AS	(UTC— 03H00)	Cayenne, Fortaleza	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Pacifique	(UTC— 05H00)	Bogota, Lima, Quito, Rio Branco	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Ouest AS	(UTC— 04H00)	Georgetown, La Paz, Manaus, San Juan	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale Saint-Pierre	(UTC— 03H00)	Saint-Pierre-et-Miquelon	
Heure normale de Sakhaline	(UTC+ 11 h)	Sakhaline	
Heure normale des Samoa	(UTC+ 13h00)	Samoa	
Heure normale de Sao Tomé	(UTC+ 01:00)	Sao Tomé	
Heure normale de Saratov	(UTC+ 04:00)	Saratov	
Heure normale de l'Asie du Sud-Est	(UTC+ 07h00)	Bangkok, Hanoï, Djakarta	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Singapour	(UTC+ 08h00)	Kuala Lumpur, Singapour	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique du Sud	(UTC+ 02:00)	Harare, Pretoria	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Sri Lanka	(UTC+ 05H30)	Sri Jayawardenepura	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Soudan	(UTC+ 02:00)	Khartoum	
Heure normale de Syrie	(UTC+ 02:00)	Damas	
Heure normale de Taipei	(UTC+ 08h00)	Taipei	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Tasmanie	(UTC+ 10 H)	Hobart	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Tocantins	(UTC— 03H00)	Araguaina	
Heure normale de Tokyo	(UTC+ 09h00)	Osaka, Sapporo, Tokyo	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Tomsk	(UTC+ 07h00)	Tomsk	
Heure normale des Tonga	(UTC+ 13h00)	Nuku'alofa	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de la Transbaïkalie	(UTC+ 09h00)	Tchita	
Heure normale de Turquie	(UTC+ 03:00)	Istanbul	
Heure normale des îles Turques-et-Caïques	(UTC— 05H00)	Turques-et-Caïques	
Heure normale d'Oulan-Bator	(UTC+ 08h00)	Oulan-Bator	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Est	(UTC— 05H00)	Indiana (Est)	
Heure normale des Rocheuses	(UTC— 07h00)	Arizona	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC	UTC	Temps universel coordonné	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
UTC—02	(UTC— 02H00)	Temps universel coordonné–02	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC—08	(UTC— 08H00)	Temps universel coordonné–08	
UTC—09	(UTC— 09H00)	Temps universel coordonné–09	
UTC—11	(UTC— 11 H 00)	Temps universel coordonné–11	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC+12	(UTC+ 12H00)	Temps universel coordonné+12	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC+13	(UTC+ 13h00)	Temps universel coordonné+13	
Heure normale du Venezuela	(UTC— 04H00)	Caracas	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Vladivostok	(UTC+ 10 H)	Vladivostok	
Heure normale de Volgograd	(UTC+ 04:00)	Volgograd	
Heure normale d'Australie de l'Ouest	(UTC+ 08h00)	Perth	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique centrale ouest	(UTC+ 01:00)	Afrique centrale de l'Ouest	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Europe de l'ouest	(UTC+ 01:00)	Amsterdam, Berlin, Berne, Rome, Stockholm, Vienne	
Heure normale de Mongolie de l'Ouest	(UTC+ 07h00)	Hovd	
Heure normale de l'Asie de l'Est	(UTC+ 05:00)	Achgabat, Tachkent	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Cisjordanie	(UTC+ 02:00)	Gaza, Hébron	
Heure normale du Pacifique Ouest	(UTC+ 10 H)	Guam, Port Moresby	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Iakoutsk	(UTC+ 09h00)	Iakoutsk	

Utilisation d'une clé principale de service avec RDS Custom for SQL Server

RDSCustom for SQL Server prend en charge l'utilisation d'une clé principale de service (SMK). RDSCustom conserve les mêmes informations SMK pendant toute la durée de vie de votre instance de base de données RDS Custom for SQL Server. En les conservantSMK, votre instance de base de données peut utiliser des objets chiffrés avec leSMK, tels que les mots de passe et les informations d'identification des serveurs liés. Si vous utilisez un déploiement multi-AZ, RDS Custom synchronise et gère également les instances de base SMK de données principales et secondaires.

Rubriques

- Disponibilité des régions et des versions
- Fonctionnalités prises en charge
- En utilisant TDE
- Configuration des fonctionnalités
- Exigences et limitations

Disponibilité des régions et des versions

L'utilisation d'un SMK est prise en charge dans toutes les régions où RDS Custom for SQL Server est disponible, pour toutes les versions de SQL serveur disponibles sur RDS Custom. Pour plus d'informations sur la version et la disponibilité par région d'Amazon RDS avec RDS Custom for SQL Server, consultezRégions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server.

Fonctionnalités prises en charge

Lors de l'utilisation d'un SMK with RDS Custom for SQL Server, les fonctionnalités suivantes sont prises en charge :

- Chiffrement transparent des données (TDE)
- Chiffrement au niveau des colonnes
- Messagerie de base de données
- Serveurs liés
- SQLServices d'intégration de serveurs (SSIS)

En utilisant TDE

An SMK permet de configurer le chiffrement transparent des données (TDE), qui chiffre les données avant leur écriture dans le stockage, et déchiffre automatiquement les données lorsqu'elles sont lues depuis le stockage. Contrairement RDS au SQL serveur, la configuration TDE sur une instance de base de données RDS personnalisée pour SQL serveur ne nécessite pas l'utilisation de groupes d'options. Au lieu de cela, une fois que vous avez créé un certificat et une clé de chiffrement de base de données, vous pouvez exécuter la commande suivante pour l'activer TDE au niveau de la base de données :

```
ALTER DATABASE [myDatabase] SET ENCRYPTION ON;
```

Pour plus d'informations sur l'utilisation de TDE with RDS for SQL Server, consultezSupport pour le chiffrement transparent des données sur le SQL serveur.

Pour obtenir des informations détaillées sur TDE Microsoft SQL Server, consultez la section Chiffrement transparent des données dans la documentation Microsoft.

Configuration des fonctionnalités

Pour obtenir des instructions détaillées sur la configuration des fonctionnalités qui utilisent un SMK with RDS Custom for SQL Server, vous pouvez consulter les articles suivants sur le blog de la RDS base de données Amazon :

- Serveurs liés : configuration des serveurs liés sur RDS Custom for SQL Server.
- SSIS: Migrez SSIS les packages vers RDS Custom for SQL Server.
- TDE: Sécurisez vos données à TDE l'aide de RDS Custom for SQL Server.

Exigences et limitations

Lorsque vous utilisez SMK une instance de base de données RDS personnalisée pour SQL serveur, gardez à l'esprit les exigences et limites suivantes :

- Si vous le régénérez SMK sur votre instance de base de données, vous devez immédiatement effectuer un instantané de base de données manuel. Nous recommandons d'éviter de les régénérer SMK si possible.
- Vous devez conserver des copies de sauvegarde des certificats de serveur et des mots de passe de la clé principale de la base de données. Si vous ne les sauvegardez pas, cela peut entraîner une perte de données.
- Si vous configurezSSIS, vous devez utiliser un SSM document pour joindre l'instance de base de données RDS Custom for SQL Server au domaine en cas de calcul à l'échelle ou de remplacement d'un hôte.
- Lorsque TDE le chiffrement par colonne est activé, les sauvegardes de base de données sont automatiquement chiffrées. Lorsque vous effectuez une restauration instantanée ou une restauration instantanée, l'instance de base SMK de données source est restaurée afin de déchiffrer les données pour la restauration, et une nouvelle SMK instance est générée pour rechiffrer les données de l'instance restaurée.

Pour plus d'informations sur les clés principales de service dans Microsoft SQL Server, consultez la section Clés de chiffrement des SQL serveurs et des bases de données dans la documentation Microsoft.

Prise en charge de la capture des données modifiées (CDC) avec RDS Custom pour SQL Server

Présentation

RDS Custom for SQL Server fournit un support natif pour la capture des données de modification (CDC), vous permettant de suivre et de capturer les modifications de données dans vos tables SQL Server. Le CDC stocke des métadonnées détaillées sur ces modifications pour une récupération et une analyse ultérieures. Pour des informations détaillées sur les fonctionnalités du CDC, voir Modifier la capture des données dans la documentation Microsoft.

Le fonctionnement du CDC dans SQL Server nécessite la mise en correspondance des valeurs entre le serveur local (qui a `server_id` = 0) `sys.servers` et les `SERVERPROPERTY('ServerName')` identifiants. RDS Custom for SQL Server maintient automatiquement cette synchronisation tout au long du cycle de vie de l'instance afin de garantir le fonctionnement continu du CDC même si les hôtes sont remplacés pendant les opérations de maintenance ou de restauration.

Important

Après le basculement d'une instance Multi-AZ, la `SERVERPROPERTY('Servername')` fonction reflète automatiquement les modifications du network/computer nom. Toutefois, la `@@SERVERNAME` fonction conserve l'ancien nom du serveur jusqu'au redémarrage du `MSSQLSERVER` service. L'interrogation de @ @SERVERNAME renvoie le nom du serveur précédent après un basculement. Pour obtenir le nom exact du serveur après un basculement, utilisez la requête SQL suivante :

```
SELECT name FROM sys.servers WHERE server_id=0
```

Cette requête fournit le plus d'informations sur le nom up-to-date du serveur sans nécessiter de redémarrage du service.

Disponibilité des régions et des versions

La fonctionnalité CDC est prise en charge partout Régions AWS où RDS Custom for SQL Server est disponible, pour toutes les versions de SQL Server prises en charge par RDS Custom. Pour plus d'informations sur les versions prises en charge et la disponibilité régionale de RDS Custom pour SQL Server, consultezRégions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server.

Exigences et limitations

Lorsque vous implémentez CDC sur RDS Custom pour SQL Server, tenez compte des principales considérations suivantes :

- Si vous configurez manuellement le serveur *@@SERVERNAME* and/or local `sys.servers` pour utiliser des fonctionnalités telles que MS Replication, si la valeur du serveur local (qui possède`server_id = 0`) in `sys.servers` est définie sur un format correspondant `*.rds.amazonaws.com` ou `*.awsrds.*.com` si RDS Custom pour SQL Server n'essaie pas de le modifier pour qu'il `SERVERPROPERTY('ServerName')` corresponde.
- RDS ne peut pas modifier le serveur local (qui possède`server_id = 0`) un nouveau nom d'hôte lorsque les connexions `sys.servers` à distance ou les serveurs liés utilisent activement l'ancien nom d'hôte. Cette limitation s'applique dans deux scénarios :
  - Lorsqu'un serveur lié établit une connexion avec le serveur local à l'aide d'une connexion à distance associée à l'ancien nom d'hôte
  - Lorsqu'une instance RDS Custom for SQL Server agit en tant qu'éditeur ou distributeur et a lié les connexions associées à l'ancien nom d'hôte à ses instances d'abonnés.

Résolution des problèmes

Pour identifier les connexions à distance ou les connexions liées associées à l'ancien nom de serveur, utilisez les requêtes suivantes. Validez les résultats et supprimez ces connexions pour garantir le bon fonctionnement du CDC.

```
SELECT * FROM sys.remote_logins WHERE server_id=0
```

or

```
select sss.srvname,ssp.name,srl.remote_name  from sys.server_principals ssp
inner join sys.remote_logins srl on srl.local_principal_id=ssp.principal_id
inner join sys.sysservers sss  on srl.server_id = sss.srvid
where sss.srvname = @@SERVERNAME
```

Configuration de votre environnement pour Amazon RDS Custom for SQL Server

Avant de créer et de t degérer une instance de base de données Amazon RDS Custom for SQL Server, vous devez effectuer les tâches suivantes.

Table des matières

- Conditions préalables à la configuration de RDS Custom for SQL Server
  - Création automatique de profils d'instance à l'aide du AWS Management Console
- Étape 1 : accordez les autorisations requises à votre principal IAM
- Étape 2 : Configuration du réseau, du profil d'instance et du chiffrement
  - Configuration avec AWS CloudFormation
    - Paramètres requis par CloudFormation
    - Télécharger le fichier AWS CloudFormation modèle
    - Configuration des ressources à l'aide de CloudFormation
  - Configuration manuelle
    - Vérifiez que vous disposez d'une clé de chiffrement AWS KMS symétrique
    - Création manuelle de votre profil d'instance et de votre rôle IAM
      - Création du rôle AWSRDSCustom SQLServer InstanceRole IAM
      - Ajoutez une politique d'accès à AWSRDSCustom SQLServer InstanceRole
      - Création de votre profil d'instance RDS Custom for SQL Server
      - Ajoutez AWSRDSCustom SQLServer InstanceRole à votre profil d'instance RDS Custom pour SQL Server
    - Configuration manuelle de votre VPC
      - Configurez vos groupes de sécurité VPC
      - Configurer les points de terminaison pour les personnes dépendantes Services AWS
      - Configuration du service des métadonnées d'instance
- Restriction entre instances

> **Note**
>
> Pour un step-by-step didacticiel sur la configuration des prérequis et le lancement d'Amazon RDS Custom pour SQL Server, consultez Commencer à utiliser Amazon RDS Custom pour SQL Server à l'aide d'un CloudFormation modèle (Configuration réseau) et Découvrez les prérequis requis requis pour créer une instance Amazon RDS Custom pour SQL Server.

Conditions préalables à la configuration de RDS Custom for SQL Server

Avant de créer une instance de base de données RDS Custom for SQL Server, assurez-vous que votre environnement satisfait aux exigences décrites dans cette rubrique. Vous pouvez également utiliser le CloudFormation modèle pour configurer les prérequis au sein de votre Compte AWS. Pour de plus amples informations, consultez Configuration avec AWS CloudFormation.

RDS Custom pour SQL Server nécessite que vous configuriez les conditions préalables suivantes :

- Configurez les autorisations AWS Identity and Access Management (IAM) requises pour la création d'instances. Il s'agit de l'utilisateur ou du rôle AWS Identity and Access Management (IAM) nécessaire pour envoyer une `create-db-instance` demande à RDS.
- Configurez les ressources requises par RDS Custom pour l'instance de base de données SQL Server :
  - Configurez la AWS KMS clé requise pour le chiffrement de l'instance personnalisée RDS. RDS Custom nécessite une clé gérée par le client au moment de la création de l'instance pour le chiffrement. L'ARN, l'ID, l'ARN de l'alias ou le nom de l'alias de la clé KMS sont transmis en tant que `kms-key-id` paramètre dans la demande de création de l'instance de base de données personnalisée RDS.
  - Configurez les autorisations requises dans l'instance de base de données RDS Custom pour SQL Server. RDS Custom attache un profil d'instance à l'instance de base de données lors de sa création et l'utilise pour l'automatisation au sein de l'instance de base de données. Le nom du profil d'instance est défini `custom-iam-instance-profile` dans la demande de création personnalisée RDS. Vous pouvez créer un profil d'instance à partir du AWS Management Console ou créer manuellement votre profil d'instance. Pour plus d'informations, consultez Création automatique de profils d'instance à l'aide du AWS Management Console et Création manuelle de votre profil d'instance et de votre rôle IAM.
  - Configurez la configuration réseau conformément aux exigences de RDS Custom pour SQL Server. Les instances personnalisées RDS résident dans les sous-réseaux (configurés avec

le groupe de sous-réseaux de base de données) que vous fournissez lors de la création de l'instance. Ces sous-réseaux doivent permettre aux instances personnalisées RDS de communiquer avec les services requis pour l'automatisation RDS.

> **Note**
>
> Pour les exigences mentionnées ci-dessus, assurez-vous qu'aucune politique de contrôle des services (SCPs) ne restreint les autorisations au niveau du compte.
> Si le compte que vous utilisez fait partie d'une AWS organisation, il est possible que des politiques de contrôle des services (SCPs) restreignent les autorisations au niveau du compte. Assurez-vous de SCPs ne pas restreindre les autorisations sur les utilisateurs et les rôles que vous créez à l'aide des procédures suivantes.
> Pour plus d'informations SCPs, voir Politiques de contrôle des services (SCPs) dans le Guide de AWS Organizations l'utilisateur. Utilisez la AWS CLI commande describe-organization pour vérifier si votre compte fait partie d'une AWS organisation.
> Pour plus d'informations sur AWS les Organizations, voir What is AWS Organizations dans le guide de AWS Organizations l'utilisateur.

Pour connaître les exigences générales applicables à RDS Custom for SQL Server, consultez Exigences générales pour RDS Custom for SQL Server.

Création automatique de profils d'instance à l'aide du AWS Management Console

RDS Custom vous oblige à créer et à configurer un profil d'instance pour lancer toute instance de base de données RDS Custom pour SQL Server. Utilisez le AWS Management Console pour créer et associer un nouveau profil d'instance en une seule étape. Cette option est disponible dans la section Sécurité personnalisée RDS des pages Créer une base de données, Restaurer un instantané et Restaurer vers un point précis de la console. Choisissez Créer un nouveau profil d'instance pour fournir un suffixe de nom de profil d'instance. AWS Management Console crée un nouveau profil d'instance doté des autorisations requises pour les tâches d'automatisation personnalisées RDS. Pour créer automatiquement de nouveaux profils d'instance, votre AWS Management Console utilisateur connecté doit disposer des autorisations`iam:CreateInstanceProfile`, `iam:AddRoleToInstanceProfile`iam:CreateRole, et. `iam:AttachRolePolicy`

> **Note**
>
> Cette option n'est disponible que dans le AWS Management Console. Si vous utilisez la CLI ou le SDK, utilisez le CloudFormation modèle RDS Custom fourni ou créez manuellement un profil d'instance. Pour de plus amples informations, veuillez consulter Création manuelle de votre profil d'instance et de votre rôle IAM.

Étape 1 : accordez les autorisations requises à votre principal IAM

Assurez-vous que vous disposez d'un accès suffisant pour créer une instance personnalisée RDS. Le rôle IAM ou l'utilisateur IAM (appelé principal IAM) chargé de créer une instance de base de données RDS Custom pour SQL Server à l'aide de la console ou de la CLI doit disposer de l'une des politiques suivantes pour une création d'instance de base de données réussie :

- La stratégie `AdministratorAccess`
- La politique `AmazonRDSFullAccess` avec les autorisations supplémentaires suivantes :

```
iam:SimulatePrincipalPolicy
cloudtrail:CreateTrail
cloudtrail:StartLogging
s3:CreateBucket
s3:PutBucketPolicy
s3:PutBucketObjectLockConfiguration
s3:PutBucketVersioning
kms:CreateGrant
kms:DescribeKey
kms:Decrypt
kms:ReEncryptFrom
kms:ReEncryptTo
kms:GenerateDataKeyWithoutPlaintext
kms:GenerateDataKey
ec2:DescribeImages
ec2:RunInstances
ec2:CreateTags
```

RDS Custom utilise ces autorisations lors de la création de l'instance. Ces autorisations configurent les ressources de votre compte qui sont requises pour les opérations RDS Custom.

Pour plus d'informations sur l'autorisation `kms:CreateGrant`, consultez AWS KMS key gestion.

L'exemple de politique JSON suivant accorde les autorisations requises.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ValidateIamRole",
            "Effect": "Allow",
            "Action": "iam:SimulatePrincipalPolicy",
            "Resource": "*"
        },
        {
            "Sid": "CreateCloudTrail",
            "Effect": "Allow",
            "Action": [
                "cloudtrail:CreateTrail",
                "cloudtrail:StartLogging"
            ],
            "Resource": "arn:aws:cloudtrail:*:*:trail/do-not-delete-rds-custom-*"
        },
        {
            "Sid": "CreateS3Bucket",
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:PutBucketPolicy",
                "s3:PutBucketObjectLockConfiguration",
                "s3:PutBucketVersioning"
            ],
            "Resource": "arn:aws:s3:::do-not-delete-rds-custom-*"
        },
        {
            "Sid": "CreateKmsGrant",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
```

```
}
```

Le principal IAM a besoin des autorisations supplémentaires suivantes pour travailler avec des versions de moteur personnalisées (CEVs) :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ConfigureKmsKeyEncryptionPermission",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant",
                "kms:DescribeKey",
                "kms:Decrypt",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:GenerateDataKey"
            ],
            "Resource": "arn:aws:kms:us-east-1:111122223333:key/key_id"
        },
        {
            "Sid": "CreateEc2Instance",
            "Effect": "Allow",
            "Action": [
                "ec2:DescribeImages",
                "ec2:RunInstances",
                "ec2:CreateTags"
            ],
            "Resource": "*"
        }
    ]
}
```

*account_id*Remplacez-le par l'ID du compte que vous utilisez pour créer votre instance. *region*Remplacez-le par celui Région AWS que vous utilisez pour créer votre instance. Remplacez *key_id* par votre ID de clé gérée par le client. Vous pouvez ajouter plusieurs clés selon vos besoins.

Pour plus d'informations sur les autorisations au niveau des ressources requises pour lancer une EC2 instance, consultez Launch instances () RunInstances.

En outre, le principal IAM exige l'autorisation `iam:PassRole` sur le rôle IAM. Elle doit être attachée au profil d'instance transmis dans le paramètre `custom-iam-instance-profile` de la demande de création de l'instance de base de données RDS Custom. Le profil d'instance et son rôle attaché sont créés ultérieurement dans Étape 2 : Configuration du réseau, du profil d'instance et du chiffrement.

> **Note**
>
> Assurez-vous que les autorisations répertoriées précédemment ne sont pas limitées par les politiques de contrôle des services (SCPs), les limites d'autorisation ou les politiques de session associées au principal IAM.

Étape 2 : Configuration du réseau, du profil d'instance et du chiffrement

Vous pouvez configurer votre rôle de profil d'instance IAM, votre cloud privé virtuel (VPC) AWS KMS et votre clé de chiffrement symétrique en utilisant l'un des processus suivants :

- Configuration avec AWS CloudFormation (recommandé)
- Configuration manuelle

> **Note**
>
> Si votre compte en fait partie AWS Organizations, assurez-vous que les autorisations requises par le rôle de profil d'instance ne sont pas limitées par les politiques de contrôle des services (SCPs).
> Les configurations réseau décrites dans cette rubrique fonctionnent mieux avec les instances de base de données qui ne sont pas accessibles au public. Vous ne pouvez pas vous connecter directement à de telles instances de base de données depuis l'extérieur du VPC.

Configuration avec AWS CloudFormation

Pour simplifier la configuration, vous pouvez utiliser un fichier AWS CloudFormation modèle pour créer une CloudFormation pile. Un CloudFormation modèle crée toutes les ressources réseau, les profils d'instance et les ressources de chiffrement conformément aux exigences de RDS Custom.

Pour savoir comment créer des piles, consultez la section Création d'une pile sur la AWS CloudFormation console dans le Guide de l'AWS CloudFormation utilisateur.

Pour un didacticiel expliquant comment lancer Amazon RDS Custom pour SQL Server à l'aide d'un AWS CloudFormation modèle, consultez Commencer à utiliser Amazon RDS Custom pour SQL Server à l'aide d'un AWS CloudFormation modèle sur le blog de AWS base de données.

Rubriques

- Paramètres requis par CloudFormation
- Télécharger le fichier AWS CloudFormation modèle
- Configuration des ressources à l'aide de CloudFormation

Paramètres requis par CloudFormation

Les paramètres suivants sont requis pour configurer les ressources prérequises RDS Custom avec CloudFormation :

Groupe de paramètres	Nom du paramètre	Valeur par défaut	Description
Configuration de disponibilité	Sélectionnez une configuration de disponibilité pour la configuration des prérequis	Multi-AZ	Spécifiez si vous souhaitez configurer les prérequis dans une configuration mono-AZ ou multi-AZ pour les instances personnalisées RDS. Vous devez utiliser la configuration multi-AZ si vous avez besoin d'au moins une instance de base

Groupe de paramètres	Nom du paramètre	Valeur par défaut	Description
			de données multi-AZ dans cette configuration
Configuration réseau	IPv4 Bloc CIDR pour VPC	10.0.0.0/16	Spécifiez un bloc IPv4 CIDR (ou une plage d'adresses IP) pour votre VPC. Ce VPC est configuré pour créer et utiliser une instance de base de données personnalisée RDS.
	IPv4 Bloc CIDR pour 1 des 2 sous-réseaux privés	10.0.128.0/20	Spécifiez un bloc IPv4 CIDR (ou plage d'adresses IP) pour votre premier sous-réseau privé. Il s'agit de l'un des deux sous-réseaux dans lesquels l'instance de base de données personnalisée RDS peut être créée. Il s'agit d'un sous-réseau privé sans accès à Internet.

Groupe de paramètres	Nom du paramètre	Valeur par défaut	Description
	IPv4 Bloc CIDR pour 2 des 2 sous-réseaux privés	10.0.144.0/20	Spécifiez un bloc IPv4 CIDR (ou plage d'adresses IP) pour votre deuxième sous-réseau privé. Il s'agit de l'un des deux sous-réseaux dans lesquels l'instance de base de données personnalisée RDS peut être créée. Il s'agit d'un sous-réseau privé sans accès à Internet.
	IPv4 Bloc CIDR pour sous-réseau public	10,0.0.0/20	Spécifiez un bloc IPv4 CIDR (ou plage d'adresses IP) pour votre sous-réseau public. Il s'agit de l'un des sous-réseaux dans lesquels l'EC2 instance peut se connecter. L'instance de base de données personnalisée RDS peut être créée. Il s'agit d'un sous-réseau public avec accès à Internet.

Groupe de paramètres	Nom du paramètre	Valeur par défaut	Description
Configuration de l'accès RDP	IPv4 Bloc CIDR de votre source	-	Spécifiez un bloc IPv4 CIDR (ou plage d'adresses IP) de votre source. Il s'agit de la plage d'adresses IP à partir de laquelle vous établissez une connexion RDP à une EC2 instance du sous-réseau public. Si ce n'est pas le cas, la connexion RDP à l' EC2 instance n'est pas configurée.
	Configuration de l'accès RDP à l'instance RDS Custom pour SQL Server	Non	Spécifiez s'il faut activer la connexion RDP entre l' EC2 instance et l'instance RDS Custom for SQL Server. Par défaut, la connexion RDP entre l' EC2 instance et l'instance de base de données n'est pas configurée.

Ressources créées par CloudFormation

La création réussie de la CloudFormation pile à l'aide des paramètres par défaut crée les ressources suivantes dans votre Compte AWS :

- Clé KMS de chiffrement symétrique pour le chiffrement des données gérées par RDS Custom.

- Le profil d'instance est associé à un rôle IAM `AmazonRDSCustomInstanceProfileRolePolicy` pour fournir les autorisations requises par RDS Custom. Pour plus d'informations, consultez Amazon RDSCustom ServiceRolePolicy dans le AWS Managed Policy Reference Guide.
- VPC avec la plage CIDR spécifiée comme paramètre. CloudFormation La valeur par défaut est `10.0.0.0/16`.
- Deux sous-réseaux privés avec la plage CIDR spécifiée dans les paramètres, et deux zones de disponibilité différentes dans la Région AWS. Les valeurs par défaut du sous-réseau CIDRs sont `10.0.128.0/20` et`10.0.144.0/20`.
- Un sous-réseau public dont la plage d'adresses CIDR est spécifiée dans les paramètres. La valeur par défaut du CIDR du sous-réseau est 10.0.0.0/20. L' EC2 instance réside dans ce sous-réseau et peut être utilisée pour se connecter à l'instance personnalisée RDS.
- Jeu d'options DHCP pour le VPC avec résolution de nom de domaine sur un serveur Amazon Domain Name System (DNS).
- Table de routage à associer à deux sous-réseaux privés sans accès à Internet.
- Table de routage à associer au sous-réseau public et ayant accès à Internet.
- Passerelle Internet associée au VPC pour permettre l'accès Internet au sous-réseau public.
- Liste de contrôle d'accès réseau (ACL) à associer à deux sous-réseaux privés et accès restreint au protocole HTTPS et au port de base de données au sein du VPC.
- Groupe de sécurité VPC à associer à l'instance RDS Custom. L'accès est limité pour le HTTPS sortant aux Service AWS points de terminaison requis par RDS Custom et au port de base de données entrant du groupe de sécurité de EC2 l'instance.
- Groupe de sécurité VPC à associer à l' EC2 instance dans le sous-réseau public. L'accès est restreint pour le port de base de données sortant au groupe de sécurité d'instance personnalisé RDS.
- Groupe de sécurité VPC à associer aux points de terminaison VPC créés pour les points de terminaison requis par Service AWS RDS Custom.
- Groupe de sous-réseaux de base de données dans lequel les instances RDS Custom sont créées. Deux sous-réseaux privés créés par ce modèle sont ajoutés au groupe de sous-réseaux de base de données.
- Points de terminaison VPC pour chacun des points de Service AWS terminaison requis par RDS Custom.

La définition de la configuration de disponibilité sur multi-az créera les ressources suivantes en plus de la liste ci-dessus :

- Règles d'ACL réseau permettant la communication entre des sous-réseaux privés.
- Accès entrant et sortant au port Multi-AZ au sein du groupe de sécurité VPC associé à l'instance personnalisée RDS.
- Points de terminaison VPC vers les points AWS de terminaison de service requis pour les communications multi-AZ.

En outre, la configuration de l'accès RDP crée les ressources suivantes :

- Configuration de l'accès RDP au sous-réseau public à partir de votre adresse IP source :
  - Règles ACL réseau qui autorisent la connexion RDP entre votre adresse IP source et le sous-réseau public.
  - Accès d'entrée au port RDP depuis votre adresse IP source vers le groupe de sécurité VPC associé à l'instance. EC2
- Configuration de l'accès RDP depuis une EC2 instance dans un sous-réseau public vers une instance personnalisée RDS dans des sous-réseaux privés :
  - Règles d'ACL réseau autorisant la connexion RDP entre un sous-réseau public et un sous-réseau privé.
  - Accès entrant au port RDP depuis le groupe de sécurité VPC associé à l'instance EC2 vers le groupe de sécurité VPC associé à l'instance personnalisée RDS.

Utilisez les procédures suivantes pour créer la CloudFormation pile pour RDS Custom pour SQL Server.

Télécharger le fichier AWS CloudFormation modèle

Pour télécharger le fichier de modèle

1. Ouvrez le menu contextuel (clic droit) du lien custom-sqlserver-onboard.zip et choisissez Enregistrer le lien sous.
2. Enregistrez et extrayez le fichier de votre ordinateur.

Configuration des ressources à l'aide de CloudFormation

Pour configurer les ressources à l'aide de CloudFormation

1. Ouvrez la CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Pour lancer l'assistant de création de piles, choisissez Create Stack (Créer une pile).

   La page Create stack (Créer une pile) s'affiche.
3. Pour Prerequisite - Prepare template (Prérequis – Préparer le modèle), choisissez Template is ready (Le modèle est prêt).
4. Pour Specify template (Spécifier un modèle), procédez comme suit :

   a. Pour Source du modèle, choisissez Charger un fichier de modèle.

   b. Pour Choisir un fichier, accédez au bon fichier et sélectionnez-le.
5. Choisissez Suivant.

   La page Specify stack details (Spécifier les détails de la pile) s'affiche.
6. Dans le champ Nom de la pile, saisissez **`rds-custom-sqlserver`**.
7. Pour Parameters (Paramètres), procédez comme suit :

   a. Pour conserver les options par défaut, choisissez Next (Suivant).

   b. Pour modifier les options, choisissez la configuration de disponibilité, la configuration réseau et la configuration d'accès RDP appropriées, puis choisissez Next.

      Lisez attentivement la description de chaque paramètre avant de modifier les paramètres.

   > **Note**
   >
   > Si vous choisissez de créer au moins une instance multi-AZ dans cette CloudFormation pile, assurez-vous que le paramètre de CloudFormation pile Sélectionner une configuration de disponibilité pour la configuration des prérequis est défini sur. `Multi-AZ` Si vous créez la CloudFormation pile en tant que mono-AZ, mettez-la à jour en configuration multi-AZ avant de créer la première instance multi-AZ. CloudFormation
8. Sur la page Configurer les options de pile, choisissez Suivant.
9. Sur la page Review (Vérifier) rds-custom-sqlserver, procédez comme suit :

a. Sous Capacités, cochez la case Je sais qu' AWS CloudFormation peut créer des ressources IAM avec des noms personnalisés.

b. Sélectionnez Créer la pile.

> **Note**
>
> Ne mettez pas à jour les ressources créées à partir de cette AWS CloudFormation pile directement à partir des pages de ressources. Cela vous empêche d'appliquer de futures mises à jour à ces ressources à l'aide d'un AWS CloudFormation modèle.

CloudFormation crée les ressources requises par RDS Custom for SQL Server. Si la création de la pile échoue, consultez l'onglet Events (Événements) pour voir quelle création de ressource a échoué et le motif de l'échec.

L'onglet Sorties de cette CloudFormation pile de la console doit contenir des informations sur toutes les ressources à transmettre en tant que paramètres pour créer une instance de base de données RDS Custom for SQL Server. Assurez-vous d'utiliser le groupe de sécurité VPC et le groupe de sous-réseaux de base de données créés par CloudFormation pour les instances de base de données personnalisées RDS. Par défaut, RDS tente d'attacher le groupe de sécurité VPC par défaut, qui peut ne pas avoir l'accès dont vous avez besoin.

Si vous aviez CloudFormation l'habitude de créer des ressources, vous pouvez ignorerConfiguration manuelle.

Mettre à jour la CloudFormation pile

Vous pouvez également mettre à jour une partie de la configuration de la CloudFormation pile après sa création. Les configurations qui peuvent être mises à jour sont les suivantes :

- Configuration de disponibilité pour RDS Custom pour SQL Server
  - Sélectionnez une configuration de disponibilité pour la configuration des prérequis : mettez à jour ce paramètre pour passer de la configuration mono-AZ à la configuration multi-AZ. Si vous utilisez cette CloudFormation pile pour au moins une instance multi-AZ, vous devez mettre à jour la pile pour choisir une configuration multi-AZ.
- Configuration d'accès RDP pour RDS personnalisée pour SQL Server

- IPv4 Bloc d'adresse CIDR de votre source : vous pouvez mettre à jour le bloc d'adresse IPv4 CIDR (ou plage d'adresses IP) de votre source en mettant à jour ce paramètre. La définition de ce paramètre sur vide supprime la configuration d'accès RDP de votre bloc CIDR source vers le sous-réseau public.
- Configurer l'accès RDP à RDS Custom pour SQL Server : activez ou désactivez la connexion RDP entre l'instance et l' EC2 instance RDS Custom pour SQL Server.

Supprimer la CloudFormation pile

Vous pouvez supprimer la CloudFormation pile après avoir supprimé toutes les instances personnalisées RDS qui utilisent les ressources de la pile. RDS Custom ne suit pas la CloudFormation pile, il ne bloque donc pas la suppression de la pile lorsque certaines instances de base de données utilisent des ressources de pile. Assurez-vous qu'aucune instance de base de données personnalisée RDS n'utilise les ressources de la pile lors de la suppression de la pile.

> **Note**
>
> Lorsque vous supprimez une CloudFormation pile, toutes les ressources créées par la pile sont supprimées à l'exception de la clé KMS. La clé KMS passe dans un état en attente de suppression et est supprimée au bout de 30 jours. Pour conserver la clé KMS, effectuez une CancelKeyDeletionopération pendant la période de grâce de 30 jours.

Configuration manuelle

Si vous choisissez de configurer les ressources manuellement, exécutez les tâches suivantes.

> **Note**
>
> Pour simplifier la configuration, vous pouvez utiliser le fichier AWS CloudFormation modèle pour créer une CloudFormation pile plutôt qu'une configuration manuelle. Pour de plus amples informations, veuillez consulter Configuration avec AWS CloudFormation.
> Vous pouvez également utiliser le AWS CLI pour compléter cette section. Si tel est le cas, téléchargez et installez la dernière CLI.

Rubriques

- Vérifiez que vous disposez d'une clé de chiffrement AWS KMS symétrique
- Création manuelle de votre profil d'instance et de votre rôle IAM
- Configuration manuelle de votre VPC

Vérifiez que vous disposez d'une clé de chiffrement AWS KMS symétrique

Un chiffrement symétrique AWS KMS key est requis pour RDS Custom. Lorsque vous créez une instance de base de données RDS Custom pour SQL Server, assurez-vous de fournir l'identifiant de clé KMS en tant que paramètre`kms-key-id`. Pour de plus amples informations, veuillez consulter Création et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server.

Vous avez les options suivantes :

- Si vous possédez déjà une clé KMS gérée par le client Compte AWS, vous pouvez l'utiliser avec RDS Custom. Aucune action supplémentaire n'est nécessaire.
- Si vous avez déjà créé une clé KMS de chiffrement symétrique gérée par le client pour un moteur RDS Custom différent, vous pouvez réutiliser la même clé KMS. Aucune action supplémentaire n'est nécessaire.
- Si votre compte ne contient pas encore de clés de chiffrement KMS symétriques gérées par le client, créez-en une en suivant les instructions de la section Creating keys (Création de clés) dans le Guide du développeur AWS Key Management Service .
- Si vous créez une instance de base de données personnalisée CEV ou RDS et que votre clé KMS se trouve dans une autre Compte AWS, assurez-vous d'utiliser le. AWS CLI Vous ne pouvez pas utiliser la AWS console avec des clés KMS entre comptes.

> **Important**
>
> RDS Custom ne prend pas en charge les clés KMS AWS gérées.

Assurez-vous que votre clé de chiffrement symétrique autorise l'accès au rôle (IAM) `kms:Decrypt` et aux `kms:GenerateDataKey` opérations du rôle AWS Identity and Access Management (IAM) dans votre profil d'instance IAM. Si votre compte contient une nouvelle clé de chiffrement symétrique, aucune modification n'est requise. Sinon, veillez à ce que la stratégie de votre clé de chiffrement symétrique puisse fournir l'accès à ces opérations.

Pour de plus amples informations, veuillez consulter Étape 4 : Configuration personnalisée d'IAM pour RDS pour Oracle.

Création manuelle de votre profil d'instance et de votre rôle IAM

Vous pouvez créer manuellement un profil d'instance et l'utiliser pour lancer des instances personnalisées RDS. Si vous envisagez de créer l'instance dans le AWS Management Console, ignorez cette section. Vous AWS Management Console permet de créer et d'associer un profil d'instance à vos instances de base de données personnalisées RDS. Pour de plus amples informations, veuillez consulter Création automatique de profils d'instance à l'aide du AWS Management Console.

Lorsque vous créez manuellement un profil d'instance, transmettez le nom du profil d'instance en tant que `custom-iam-instance-profile` paramètre à votre commande `create-db-instance` CLI. RDS Custom utilise le rôle associé à ce profil d'instance pour exécuter l'automatisation afin de gérer l'instance.

Pour créer le profil d'instance IAM et les rôles IAM pour RDS Custom for SQL Server

1. Créez le rôle IAM nommé `AWSRDSCustomSQLServerInstanceRole` avec une politique de confiance qui permet à Amazon d' EC2assumer ce rôle.
2. Ajoutez la politique AWS gérée `AmazonRDSCustomInstanceProfileRolePolicy` à`AWSRDSCustomSQLServerInstanceRole`.
3. Créez un profil d'instance IAM pour RDS Custom for SQL Server nommé `AWSRDSCustomSQLServerInstanceProfile`.
4. Ajoutez `AWSRDSCustomSQLServerInstanceRole` au profil d'instance.

Création du rôle AWSRDSCustom SQLServer InstanceRole IAM

L'exemple suivant crée le rôle `AWSRDSCustomSQLServerInstanceRole`. La politique de confiance permet à Amazon d' EC2assumer le rôle.

```
aws iam create-role \
    --role-name AWSRDSCustomSQLServerInstanceRole \
    --assume-role-policy-document '{
        "Version": "2012-10-17",
          "Statement": [
            {
```

```
            "Action": "sts:AssumeRole",
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            }
          }
        ]
      }'
```

Ajoutez une politique d'accès à AWSRDSCustom SQLServer InstanceRole

Pour fournir les autorisations requises, attachez la politique AWS gérée `AmazonRDSCustomInstanceProfileRolePolicy` à`AWSRDSCustomSQLServerInstanceRole`. `AmazonRDSCustomInstanceProfileRolePolicy`permet aux instances personnalisées RDS d'envoyer et de recevoir des messages et d'effectuer diverses actions d'automatisation.

> **Note**
>
> Assurez-vous que les autorisations définies dans la politique d'accès ne sont pas limitées par le rôle de profil d'instance SCPs ou ne sont pas limitées par des limites d'autorisation associées à celui-ci.

L'exemple suivant associe une politique AWS gérée `AmazonRDSCustomInstanceProfileRolePolicy` au `AWSRDSCustomSQLServerInstanceRole` rôle.

```
aws iam attach-role-policy \
    --role-name AWSRDSCustomSQLServerInstanceRole \
    --policy-arn arn:aws:iam::aws:policy/AmazonRDSCustomInstanceProfileRolePolicy
```

Création de votre profil d'instance RDS Custom for SQL Server

Un profil d'instance est un conteneur qui inclut un rôle IAM unique. RDS Custom utilise le profil d'instance pour transmettre le rôle à l'instance.

Si vous utilisez le AWS Management Console pour créer un rôle pour Amazon EC2, la console crée automatiquement un profil d'instance et lui donne le même nom que le rôle lors de sa création. Créez votre profil d'instance comme suit, en le nommant `AWSRDSCustomSQLServerInstanceProfile`.

```
aws iam create-instance-profile \
    --instance-profile-name AWSRDSCustomSQLServerInstanceProfile
```

Ajoutez AWSRDSCustom SQLServer InstanceRole à votre profil d'instance RDS Custom pour SQL Server

Ajoutez le `AWSRDSCustomInstanceRoleForRdsCustomInstance` rôle au `AWSRDSCustomSQLServerInstanceProfile` profil créé précédemment.

```
aws iam add-role-to-instance-profile \
    --instance-profile-name AWSRDSCustomSQLServerInstanceProfile \
    --role-name AWSRDSCustomSQLServerInstanceRole
```

Configuration manuelle de votre VPC

Votre instance de base de données personnalisée RDS se trouve dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC, tout comme une instance Amazon ou une EC2 instance Amazon RDS. Vous fournissez et configurez votre propre VPC. Vous disposez ainsi d'un contrôle total sur la configuration réseau de votre instance.

RDS Custom envoie la communication de votre instance de base de données vers d'autres Services AWS. Assurez-vous que les services suivants sont accessibles depuis le sous-réseau dans lequel vous créez vos instances de base de données personnalisées RDS :

- Amazon CloudWatch (`com.amazonaws.`*`region`*`.monitoring`)
- Amazon CloudWatch Logs (`com.amazonaws.`*`region`*`.logs`)
- CloudWatch Événements Amazon (`com.amazonaws.`*`region`*`.events`)
- Amazon EC2 (`com.amazonaws.`*`region`*`.ec2etcom.amazonaws.`*`region`*`.ec2messages`)
- Amazon EventBridge (`com.amazonaws.`*`region`*`.events`)
- Amazon S3 (`com.amazonaws.`*`region`*`.s3`)
- AWS Secrets Manager (`com.amazonaws.`*`region`*`.secretsmanager`)
- AWS Systems Manager (`com.amazonaws.`*`region`*`.ssmetcom.amazonaws.`*`region`*`.ssmmessages`)

Si vous créez des déploiements multi-AZ

- Amazon Simple Queue Service (`com.amazonaws.`*`region`*`.sqs`)

Si RDS Custom ne parvient pas à communiquer avec les services nécessaires, il publie les événements suivants :

```
Database instance in incompatible-network. SSM Agent connection not available. Amazon
 RDS can't connect to the dependent AWS services.
```

```
Database instance in incompatible-network. Amazon RDS can't connect to dependent AWS
 services. Make sure port 443 (HTTPS) allows outbound connections, and try again.
 "Failed to connect to the following services: s3 events"
```

Pour éviter les `incompatible-network` erreurs, assurez-vous que les composants VPC impliqués dans la communication entre votre instance de base de données personnalisée RDS Services AWS répondent aux exigences suivantes :

- L'instance de base de données peut établir des connexions sortantes sur le port 443 vers d'autres Services AWS.
- Le VPC autorise les réponses entrantes aux demandes provenant de votre instance de base de données RDS Custom.
- RDS Custom peut correctement résoudre les noms de domaine des points de terminaison pour chaque Service AWS.

Si vous avez déjà configuré un VPC pour un moteur de base de données RDS Custom différent, vous pouvez réutiliser ce VPC et ignorer ce processus.

Rubriques

- Configurez vos groupes de sécurité VPC
- Configurer les points de terminaison pour les personnes dépendantes Services AWS
- Configuration du service des métadonnées d'instance

Configurez vos groupes de sécurité VPC

Un groupe de sécurité agit comme un pare-feu virtuel pour une instance de VPC, et contrôle le trafic entrant et sortant. Une instance de base de données personnalisée RDS possède un groupe de sécurité attaché à son interface réseau qui protège l'instance. Assurez-vous que votre groupe de

sécurité autorise le trafic entre RDS Custom et d'autres Services AWS via HTTPS. Vous transmettez ce groupe de sécurité comme `vpc-security-group-ids` paramètre dans la demande de création d'instance.

Pour configurer votre groupe de sécurité pour RDS Custom

1. Connectez-vous à la console Amazon VPC AWS Management Console et ouvrez-la à https://console.aws.amazon.com l'adresse /vpc.
2. Autorisez RDS Custom à utiliser le groupe de sécurité par défaut ou créez votre propre groupe de sécurité.

   Pour obtenir des instructions complètes, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.
3. Assurez-vous que votre groupe de sécurité autorise les connexions sortantes sur le port 443. RDS Custom a besoin de ce port pour communiquer avec des Services AWS dépendants.
4. Si vous disposez d'un VPC privé et utilisez les points de terminaison du VPC, assurez-vous que le groupe de sécurité associé à l'instance de base de données autorise les connexions sortantes sur le port 443 vers les points de terminaison du VPC. Assurez-vous également que le groupe de sécurité associé au point de terminaison du VPC autorise les connexions entrantes sur le port 443 à partir de l'instance de base de données.

   Si les connexions entrantes ne sont pas autorisées, l'instance personnalisée RDS ne peut pas se connecter aux points de EC2 terminaison Amazon AWS Systems Manager et Amazon. Pour en savoir plus, consultez Création d'un point de terminaison de cloud privé virtuel dans le Guide de l'utilisateur AWS Systems Manager .
5. Pour les instances multi-AZ RDS Custom for SQL Server, assurez-vous que le groupe de sécurité associé à l'instance de base de données autorise les connexions entrantes et sortantes sur le port 1120 avec ce groupe de sécurité lui-même. Cela est nécessaire pour la connexion à un hôte homologue sur une instance de base de données RDS personnalisée multi-AZ pour SQL Server.

Pour plus d'informations sur les groupes de sécurité, consultez Groupes de sécurité pour votre VPC dans le Guide du développeur Amazon VPC.

Configurer les points de terminaison pour les personnes dépendantes Services AWS

Nous vous recommandons d'ajouter des points de terminaison pour chaque service à votre VPC en suivant les instructions ci-dessous. Cependant, vous pouvez utiliser n'importe quelle solution

permettant à votre VPC de communiquer avec les points de terminaison de AWS service. Vous pouvez, par exemple, utiliser la traduction d'adresses réseau (NAT) ou AWS Direct Connect.

Pour configurer les points de terminaison Services AWS avec lesquels RDS Custom fonctionne

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans la barre de navigation, utilisez le sélecteur Région pour choisir la Région AWS.
3. Dans le volet de navigation, choisissez Points de terminaison. Dans le volet principal, choisissez Create Endpoint (Créer un point de terminaison).
4. Pour Service category (Catégorie de service), choisissez Services AWS.
5. Pour Nom du service, choisissez le point de terminaison affiché dans le tableau.
6. Pour VPC, choisissez votre VPC.
7. Pour Subnets (Sous-réseaux), choisissez un sous-réseau pour chaque zone de disponibilité à inclure.

   Le point de terminaison VPC peut couvrir plusieurs zones de disponibilité. AWS crée une interface réseau élastique pour le point de terminaison VPC dans chaque sous-réseau que vous choisissez. Chaque interface réseau possède un nom d'hôte DNS et une adresse IP privée.
8. Pour Groupe de sécurité, sélectionnez ou créez un groupe de sécurité.

   Vous pouvez utiliser des groupes de sécurité pour contrôler l'accès à votre point de terminaison, comme si vous utilisiez un pare-feu. Assurez-vous que le groupe de sécurité autorise les connexions entrantes sur le port 443 à partir des instances de base de données. Pour plus d'informations sur les groupes de sécurité, consultez Groupes de sécurité pour votre VPC dans le Guide de l'utilisateur Amazon VPC.
9. Vous pouvez éventuellement attacher une politique au point de terminaison du VPC. Les politiques relatives aux terminaux peuvent contrôler l'accès Service AWS au terminal auquel vous vous connectez. La politique par défaut permet à toutes les demandes de passer par le point de terminaison. Si vous utilisez une politique personnalisée, assurez-vous que les demandes issues de l'instance de base de données sont autorisées dans la politique.
10. Choisissez Créer un point de terminaison.

Le tableau suivant explique comment trouver la liste des points de terminaison dont votre VPC a besoin pour les communications sortantes.

Service	Format du point de terminaison	Notes et liens
AWS Systems Manager	Utilisez les formats de points de terminaison suivants : • `ssm.`*`region`*`.amazonaws.com` • `ssmmessages.`*`region`*`.amazonaws.com`	Pour obtenir la liste de tous les points de terminaison dans chaque région, consultez Points de terminaison et quotas AWS Systems Manager dans le Référence générale d'Amazon Web Services.
AWS Secrets Manager	Utilisez le format du point de terminaison `secretsmanager.`*`region`*`.amazonaws.com` .	Pour obtenir la liste de tous les points de terminaison dans chaque région, consultez Points de terminaison et quotas AWS Secrets Manager dans le Référence générale d'Amazon Web Services.
Amazon CloudWatch	Utilisez les formats de points de terminaison suivants : • Pour CloudWatch les métriques, utilisez `monitoring.`*`region`*`.amazonaws.com` • Pour les CloudWatch événements, utilisez `events.`*`region`*`.amazonaws.com` • Pour CloudWatch Logs, utilisez `logs.`*`region`*`.amazonaws.com`	Pour obtenir la liste des points de terminaison dans chaque région, consultez : • CloudWatch Points de terminaison et quotas Amazon dans le Référence générale d'Amazon Web Services • Points de terminaison et quotas Amazon CloudWatch Logs dans le Référence générale d'Amazon Web Services • Points de terminaison et quotas Amazon CloudWatch Events dans le Référence générale d'Amazon Web Services
Amazon EC2	Utilisez les formats de points de terminaison suivants :	Pour obtenir la liste des points de terminaison dans chaque

Service	Format du point de terminaison	Notes et liens
	• `ec2.region.amazonaws.com` • `ec2messages. region.amazonaws.com`	région, consultez Amazon Elastic Compute Cloud endpoints and quotas (Points de terminaison et quotas Amazon Elastic Compute Cloud) dans la Référence générale d'Amazon Web Services.
Amazon S3	Utilisez le format du point de terminaison `s3.region.amazonaws.com` .	Pour obtenir la liste des points de terminaison dans chaque région, consultez Amazon Simple Storage Service endpoints and quotas (Points de terminaison et quotas Amazon Simple Storage Service) dans la Référence générale d'Amazon Web Services. Pour en savoir plus sur les points de terminaison de passerelle pour Amazon S3, consultez Points de terminaison pour Amazon S3 dans le Guide du développeur Amazon VPC. Pour savoir comment créer un point d'accès, veuillez consulter Creating an Amazon S3 access point dans le Guide du développeur Amazon VPC. Pour savoir comment créer des points de terminaison de passerelle pour Amazon S3, consultez la section Gateway VPC endpoints (Points de terminaison VPC de la passerelle).

Service	Format du point de terminaison	Notes et liens
Amazon Simple Queue Service	Utiliser le format du point de terminaison `sqs.region.amazonaws.com`	Pour obtenir la liste des points de terminaison dans chaque région, consultez la section Points de terminaison et quotas Amazon Simple Queue Service.

Configuration du service des métadonnées d'instance

Assurez-vous que votre instance peut effectuer les opérations suivantes :

- Accédez au service de métadonnées d'instance à l'aide du service de métadonnées d'instance version 2 (IMDSv2).
- Autoriser les communications sortantes via le port 80 (HTTP) vers l'adresse IP de la liaison IMDS.
- Demandez les métadonnées de l'instance `http://169.254.169.254` à partir du IMDSv2 lien.

Pour plus d'informations, consultez la section Utilisation IMDSv2 dans le guide de EC2 l'utilisateur Amazon.

Restriction entre instances

Lorsque vous créez un profil d'instance en suivant les étapes ci-dessus, il utilise la politique AWS gérée `AmazonRDSCustomInstanceProfileRolePolicy` pour fournir les autorisations requises à RDS Custom, ce qui permet la gestion des instances et l'automatisation de la surveillance. La politique gérée garantit que les autorisations n'autorisent l'accès qu'aux ressources dont RDS Custom a besoin pour exécuter l'automatisation. Nous recommandons d'utiliser la politique gérée pour prendre en charge les nouvelles fonctionnalités et répondre aux exigences de sécurité qui sont automatiquement appliquées aux profils d'instance existants sans intervention manuelle. Pour plus d'informations, consultez la politique AWS gérée : Amazon RDSCusto InstanceProfileRolePolicym.

La politique `AmazonRDSCustomInstanceProfileRolePolicy` gérée limite l'accès entre comptes au profil d'instance, mais elle peut autoriser l'accès à certaines ressources gérées RDS Custom sur plusieurs instances RDS Custom au sein d'un même compte. En fonction de vos besoins, vous pouvez utiliser des limites d'autorisation pour restreindre davantage l'accès entre instances. Les limites d'autorisation définissent les autorisations maximales que les politiques basées sur l'identité

peuvent accorder à une entité, mais elles n'accordent pas d'autorisations en elles-mêmes. Pour plus d'informations, consultez la section Évaluation des autorisations effectives avec des limites.

Par exemple, la politique de limite suivante limite le rôle du profil d'instance à l'accès à une AWS KMS clé spécifique et limite l'accès aux ressources gérées RDS Custom entre les instances qui utilisent des clés différentes AWS KMS .

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DenyOtherKmsKeyAccess",
            "Effect": "Deny",
            "Action": "kms:*",
            "NotResource": "arn:aws:kms:us-east-1:111122223333:key/KMS_key_ID"
        }
    ]
}
```

> **Note**
>
> Assurez-vous que la limite des autorisations ne bloque aucune autorisation accordée à `AmazonRDSCustomInstanceProfileRolePolicy` RDS Custom.

Apportez votre propre contenu multimédia avec RDS Custom for SQL Server

RDSCustom for SQL Server prend en charge deux modèles de licence : License Included (LI) et Bring Your Own Media (BYOM).

AvecBYOM, vous pouvez effectuer les opérations suivantes :

1. Fournissez et installez vos propres fichiers binaires Microsoft SQL Server avec les mises à jour cumulatives (CU) prises en charge AWS EC2 sous WindowsAMI.
2. Enregistrez-le AMI sous forme d'image dorée, qui est un modèle que vous pouvez utiliser pour créer une version personnalisée du moteur (CEV).
3. Créez un CEV à partir de votre image dorée.
4. Créez de nouvelles instances de base de données RDS personnalisées pour le SQL serveur à l'aide de votreCEV.

Amazon gère RDS ensuite vos instances de base de données pour vous.

> **Note**
>
> Si vous disposez également d'une instance de base de données RDS personnalisée avec licence incluse (LI), vous ne pouvez pas utiliser le logiciel du SQL serveur de cette instance de base de données avecBYOM. SQL Vous devez apporter vos propres fichiers binaires de SQL serveur surBYOM.

Conditions requises BYOM pour RDS Custom for SQL Server

Les mêmes exigences générales pour les versions de moteur personnalisées avec RDS Custom for SQL Server s'appliquent également àBYOM. Pour de plus amples informations, veuillez consulter Exigences relatives à RDS Custom pour SQL Server CEVs.

Lors de l'utilisationBYOM, assurez-vous que vous répondez aux exigences supplémentaires suivantes :

- Utilisez l'une des éditions prises en charge suivantes : SQL Server 2022 ou 2019 Enterprise, Standard ou Developer Edition.

- Accordez le privilège de rôle SQL serveur Server sysadmin (SA) à. `NT AUTHORITY\SYSTEM`
- Conservez le système d'exploitation EC2 Windows Server configuré au `UTC` fil du temps.

  Les instances Amazon EC2 Windows sous-jacentes sont définies sur le UTC fuseau horaire par défaut. Pour plus d'informations sur l'affichage et la modification de l'heure d'une instance Windows, consultez la section Définir l'heure d'une instance Windows dans le guide de EC2 l'utilisateur Amazon.
- Ouvrez le TCP port 1433 et le UDP port 1434 pour autoriser SSM les connexions.

Limites de BYOM for RDS Custom for SQL Server

Les mêmes limites générales pour RDS Custom for SQL Server s'appliquent également àBYOM. Pour de plus amples informations, veuillez consulter Conditions requises et limitations d'Amazon RDS Custom for SQL Server.

AvecBYOM, les restrictions supplémentaires suivantes s'appliquent :

- Seule l'instance de SQL serveur par défaut (MSSQLSERVER) est prise en charge. Les instances SQL de serveurs nommés ne sont pas prises en charge. RDSCustom for SQL Server détecte et surveille uniquement l'instance de SQL serveur par défaut.
- Une seule installation du SQL serveur est prise en charge sur chacun d'entre euxAMI. Les installations multiples de différentes versions de SQL serveur ne sont pas prises en charge.
- SQLL'édition Web du serveur n'est pas prise en charge avecBYOM.
- Les versions d'évaluation des éditions SQL Server ne sont pas prises en charge avecBYOM. Lorsque vous installez SQL Server, ne cochez pas la case permettant d'utiliser une version d'évaluation.
- La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour plus d’informations, consultez Disponibilité régionale pour RDS Custom pour SQL Server CEVs et Support de version pour RDS Custom pour SQL Server CEVs.

Création d'une instance de base de données RDS personnalisée pour le SQL serveur avec BYOM

Pour préparer et créer une instance de base de données RDS personnalisée pour le SQL serveur avecBYOM, consultezPréparation d'une version CEV à l'aide du modèle Bring Your Own Media (BYOM).

Utilisation de versions de moteur personnalisées pour RDS Custom for SQL Server

Une version de moteur personnalisée (CEV) pour RDS Custom for SQL Server est une Amazon Machine Image (AMI) qui inclut Microsoft SQL Server.

Les étapes de base du flux de travail CEV sont les suivantes :

1. Choisissez une image AMI AWS EC2 Windows à utiliser comme image de base pour une version CEV. Vous avez la possibilité d'utiliser Microsoft SQL Server préinstallé ou le modèle Bring Your Own Media (BYOM) pour installer SQL Server vous-même.
2. Installez d'autres logiciels sur le système d'exploitation et personnalisez la configuration du système d'exploitation et de SQL Server pour répondre aux besoins de votre entreprise.
3. Enregistrez l'image AMI en tant qu'image finale.
4. Créez une version de moteur personnalisée (CEV) à partir de votre image finale.
5. Créez de nouvelles instances de base de données RDS Custom for SQL Server à l'aide de votre version CEV.

Amazon RDS gère ensuite ces instances de base de données pour vous.

Une version CEV vous permet de conserver votre configuration de base préférée du système d'exploitation et de la base de données. L'utilisation d'une version CEV garantit que la configuration de l'hôte, telle que toute installation d'agent tiers ou toute autre personnalisation du système d'exploitation, est conservée sur les instances de base de données RDS Custom for SQL Server. Une version CEV vous permet de déployer rapidement des flottes d'instances de base de données RDS Custom for SQL Server avec la même configuration.

Rubriques

- Préparation à la création d'une version CEV pour RDS Custom pour SQL Server
- Création d'un CEV for RDS Custom for SQL Server
- Modifier un CEV for RDS Custom for SQL Server
- Affichage des détails de version CEV pour Amazon RDS Custom for SQL Server
- Suppression d'une version CEV pour RDS Custom for SQL Server

Préparation à la création d'une version CEV pour RDS Custom pour SQL Server

Vous pouvez créer une version CEV à l'aide d'une Amazon Machine Image (AMI) contenant Microsoft SQL Server, License Included (LI), préinstallé, ou à l'aide d'une image AMI sur laquelle vous installez votre propre support d'installation SQL Server (BYOM).

Préparation d'un CEV

Utilisez les procédures suivantes pour créer un CEV à l'aide de Bring Your Own Media (BYOM) ou de Microsoft SQL Server (LI) préinstallé.

Préparation d'une version CEV à l'aide du modèle Bring Your Own Media (BYOM)

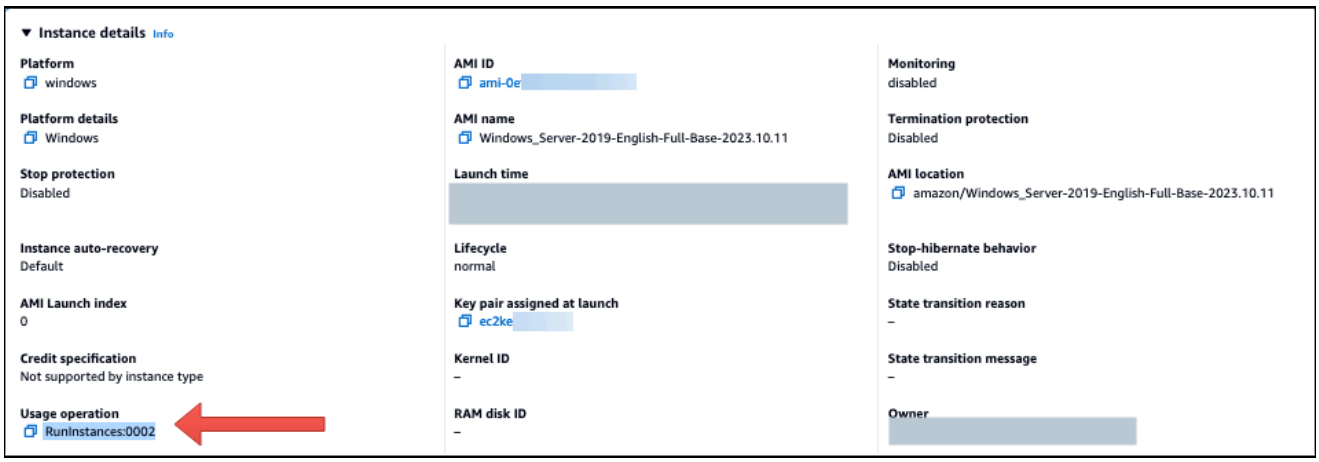
Les étapes suivantes utilisent une AMI avec Windows Server 2019 Base à titre d'exemple.

Pour créer une version CEV à l'aide du modèle BYOM

1. Sur la EC2 console Amazon, choisissez Launch Instance.
2. Dans Nom, entrez le nom de l'instance.
3. Sous Démarrage rapide, sélectionnez Windows.
4. Choisissez Microsoft Windows Server 2019 Base.
5. Choisissez le type d'instance, la paire de clés, les paramètres réseau et de stockage appropriés, puis lancez l'instance.
6. Après avoir lancé ou créé l' EC2 instance, assurez-vous que l'AMI Windows appropriée a été sélectionnée à l'étape 4 :

    a. Sélectionnez l' EC2 instance dans la EC2 console Amazon.

    b. Dans la section Détails, vérifiez l'opération Usage et assurez-vous qu'elle est définie sur : 0002RunInstances.



7. Connectez-vous à l' EC2 instance et copiez-y le support d'installation de SQL Server.

   > **Note**
   >
   > Si vous créez un CEV à l'aide de l'édition SQL Server Developer, vous devrez peut-être obtenir le support d'installation à l'aide de votre abonnement Microsoft Visual Studio.

8. Installez SQL Server. Veillez à effectuer les opérations suivantes :

   a. Révision Conditions requises BYOM pour RDS Custom for SQL Server etSupport de version pour RDS Custom pour SQL Server CEVs.

   b. Définissez le répertoire racine de l'instance sur le répertoire par défaut `C:\Program Files\Microsoft SQL Server\`. Ne modifiez pas ce répertoire.

   c. Définissez le nom du compte du moteur de base de données SQL Server sur `NT Service\MSSQLSERVER` ou `NT AUTHORITY\NETWORK SERVICE`.

   d. Définissez le mode de démarrage de SQL Server sur Manuel.

   e. Choisissez le mode d'authentification SQL Server Mixte.

   f. Conservez les paramètres actuels pour les répertoires de données et les emplacements TempDB par défaut.

9. Accordez le privilège de rôle de serveur SQL Server sysadmin (SA) à `NT AUTHORITY\SYSTEM` :

```
USE [master]
GO
EXEC master..sp_addsrvrolemember @loginame = N'NT AUTHORITY\SYSTEM' , @rolename =
 N'sysadmin'
GO
```

10. Installez des logiciels supplémentaires ou personnalisez la configuration du système d'exploitation et de la base de données pour répondre à vos exigences.

11. Exécutez Sysprep sur l'instance. EC2 Pour plus d'informations, consultez Créer une EC2 AMI Amazon à l'aide de Windows Sysprep.

12. Enregistrez l'image AMI qui contient la version de SQL Server que vous avez installée, d'autres logiciels et des personnalisations. Cela constituera votre image finale.

13. Créez une nouvelle version CEV en fournissant l'ID AMI de l'image que vous avez créée. Pour obtenir des instructions complètes, consultez Création d'un CEV for RDS Custom for SQL Server.

14. Créez une nouvelle instance de base de données RDS Custom pour SQL Server à l'aide de la version CEV. Pour obtenir des instructions complètes, consultez Créez une instance de base de données RDS personnalisée pour le SQL serveur à partir d'un CEV.

Préparation d'une version CEV utilisant SQL Server (LI) préinstallé

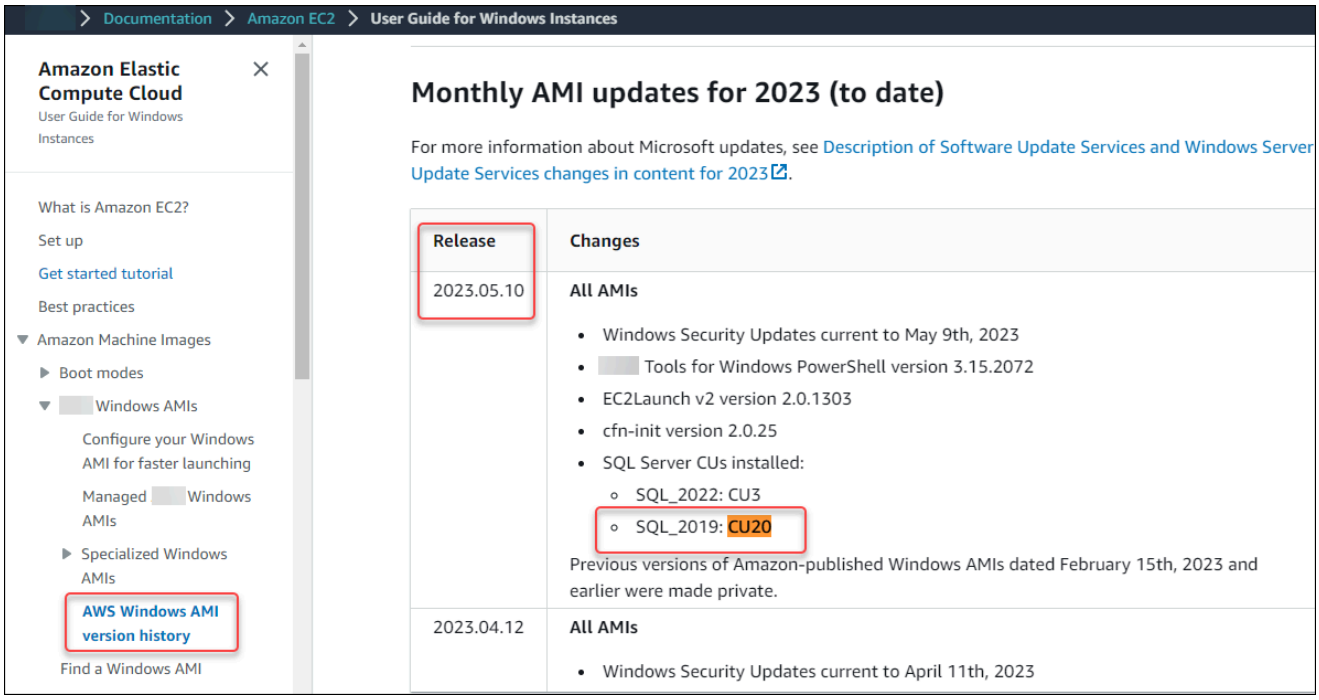
Les étapes suivantes pour créer un CEV à l'aide de Microsoft SQL Server (LI) préinstallé utilisent une AMI portant le numéro de version CU20 de SQL Server `2023.05.10` comme exemple. Lorsque vous créez une version CEV, choisissez une image AMI avec le numéro de version le plus récent. Cela garantit que vous utilisez une version prise en charge de Windows Server et de SQL Server avec la dernière mise à jour cumulative (CU).

Pour créer une version CEV utilisant Microsoft SQL Server (LI) préinstallé

1. Choisissez la dernière version disponible de AWS EC2 Windows Amazon Machine Image (AMI) avec licence incluse (LI) Microsoft Windows Server et SQL Server.

   a. Recherchez CU20 dans l'historique des versions de l'AMI Windows.

   b. Notez le numéro de version. Pour SQL Server 2019 CU2 0, le numéro de version est`2023.05.10`.



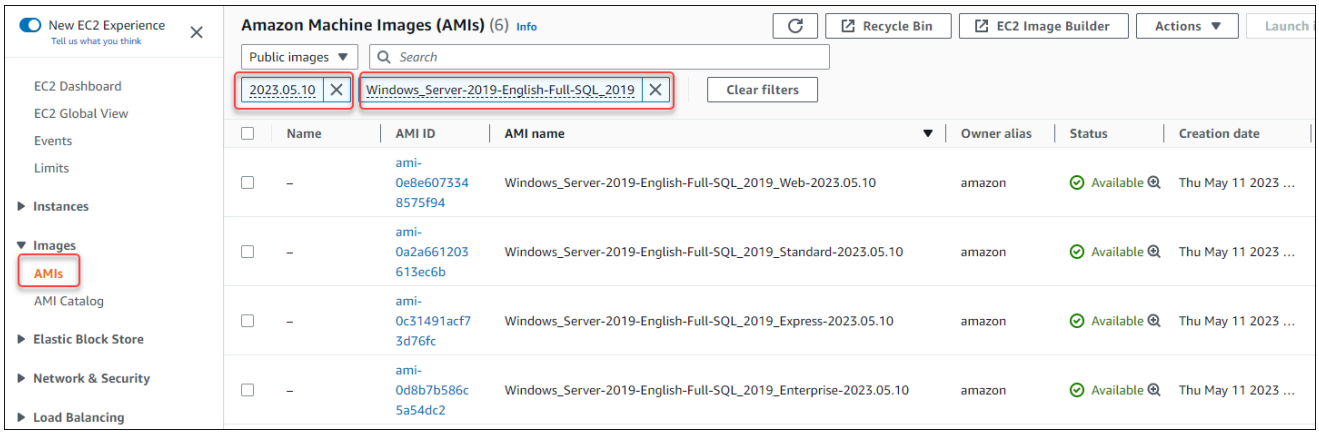
   c. Ouvrez la EC2 console Amazon à l'adresse https://console.aws.amazon.com/ec2/.

d. Dans le panneau de navigation de gauche de la EC2 console Amazon, choisissez Images, puis AMIs.

e. Choisissez Images publiques.

f. Entrez `2023.05.10` dans la zone de recherche. Une liste de AMIs apparaît.

g. Entrez `Windows_Server-2019-English-Full-SQL_2019` dans la zone de recherche pour filtrer les résultats. Les résultats suivants doivent apparaître.



h. Choisissez l'AMI avec l'édition de SQL Server que vous voulez utiliser.

2. Créez ou lancez une EC2 instance à partir de l'AMI de votre choix.

3. Connectez-vous à l' EC2 instance et installez des logiciels supplémentaires ou personnalisez la configuration du système d'exploitation et de la base de données en fonction de vos besoins.

4. Exécutez Sysprep sur l'instance. EC2 Pour plus d'informations sur la préparation d'une AMI à l'aide de Sysprep, consultez Créer une Amazon Machine Image (AMI) standardisée à l'aide de Sysprep.

5. Enregistrez l'image AMI qui contient la version de SQL Server que vous avez installée, d'autres logiciels et des personnalisations. Cela constituera votre image finale.

6. Créez une nouvelle version CEV en fournissant l'ID AMI de l'image que vous avez créée. Pour obtenir les étapes détaillées de la création d'une version CEV, consultez Création d'un CEV for RDS Custom for SQL Server.

7. Créez une nouvelle instance de base de données RDS Custom pour SQL Server à l'aide de la version CEV. Pour obtenir des instructions complètes, consultez Créez une instance de base de données RDS personnalisée pour le SQL serveur à partir d'un CEV.

Disponibilité régionale pour RDS Custom pour SQL Server CEVs

La prise en charge des versions de moteur personnalisées (CEV) pour RDS Custom pour SQL Server est disponible dans les versions suivantes : Régions AWS

- USA Est (Ohio)
- USA Est (Virginie du Nord)
- USA Ouest (Oregon)
- USA Ouest (Californie du Nord)
- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Osaka)
- Asia Pacific (Seoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Europe (Francfort)
- Europe (Irlande)
- Europe (Londres)
- Europe (Paris)
- Europe (Stockholm)
- Amérique du Sud (São Paulo)

Support de version pour RDS Custom pour SQL Server CEVs

La création de CEV pour RDS Custom for SQL Server est prise en charge pour les systèmes Windows suivants : AWS EC2 AMIs

- Pour CEVs utiliser un support préinstallé, AWS EC2 Windows AMIs avec licence incluse (LI), Microsoft Windows Server 2019 (OS) et SQL Server 2022 ou 2019
- Pour CEVs utiliser Bring your own media (BYOM), AWS EC2 Windows AMIs avec Microsoft Windows Server 2019 (OS)

La création d'une version CEV pour RDS Custom for SQL Server est prise en charge pour les éditions de système d'exploitation et de base de données suivantes :

- Pour CEVs utiliser un support préinstallé :
  - SQL Server 2022 Enterprise, Standard ou Web, avec, CU9, -GDR CU13, CU14 -GDR, CU15, CU16, CU17 et. CU18 CU19
  - SQL Server 2019 Enterprise, Standard ou Web, avec, CU8, CU17, CU2 0 CU18, CU24, -GDR CU26, CU28 -GDR, CU29 0 et. CU3 CU32
- Pour CEVs utiliser Bring your own media (BYOM) :
  - SQL Server 2022 Enterprise, Standard ou Developer CU9, avec CU13,, CU14 -GDR,, CU15 CU16, CU17 et. CU18 CU19
  - SQL Server 2019 Enterprise, Standard ou Developer, avec CU8, CU17, CU18, CU2 0, CU24, -GDR CU26, CU28 -GDR, CU29 0 et CU3. CU32
- Pour CEVs utiliser des supports préinstallés ou apporter votre propre contenu multimédia (BYOM), Windows Server 2019 est le seul système d'exploitation pris en charge.

Pour plus d'informations, consultez l'historique des versions de l'AMI AWS Windows.

Exigences relatives à RDS Custom pour SQL Server CEVs

Les exigences suivantes s'appliquent à la création d'une version CEV pour RDS Custom pour SQL Server :

- L'image AMI utilisée pour créer une version CEV doit être basée sur une configuration de système d'exploitation et de base de données prise en charge par RDS Custom for SQL Server. Pour plus d'informations sur les configurations prises en charge, consultez Conditions requises et limitations d'Amazon RDS Custom for SQL Server.
- La version CEV doit avoir un nom unique. Vous ne pouvez pas créer une version CEV qui porte le même nom qu'une version CEV existante.
- Vous devez nommer la version CEV en utilisant le modèle de dénomination de SQL Server version majeure + version mineure + chaîne personnalisée. La partie version majeure + version mineure doit correspondre à la version de SQL Server fournie avec l'AMI. Par exemple, vous pouvez nommer une AMI avec SQL Server 2019 CU17 15.00.4249.2.my_cevtest.
- Vous devez préparer une AMI à l'aide de Sysprep. Pour plus d'informations sur la préparation d'une AMI à l'aide de Sysprep, consultez Créer une Amazon Machine Image (AMI) standardisée à l'aide de Sysprep.

- Vous êtes responsable du maintien du cycle de vie de l'AMI. Une instance de base de données RDS Custom pour SQL Server créée à partir d'une version CEV ne stocke aucune copie de l'AMI. Il conserve un pointeur vers l'AMI que vous avez utilisée pour créer la version CEV. L'AMI doit exister pour qu'une instance de base de données RDS Custom for SQL Server reste opérationnelle.

Limitations de RDS Custom pour SQL Server CEVs

Les limites suivantes s'appliquent aux versions de moteur personnalisées avec RDS Custom for SQL Server :

- Vous ne pouvez pas supprimer une version CEV si des ressources, telles que des instances de base de données ou des instantanés de base de données, y sont associées.
- Pour créer une instance de base de données RDS Custom for SQL Server, une version CEV doit avoir un statut `pending-validation`, `available`, `failed` ou `validating`. Vous ne pouvez pas créer une instance de base de données RDS Custom for SQL Server à l'aide d'une version CEV si le statut de la version CEV est `incompatible-image-configuration`.
- Pour modifier une instance de base de données RDS Custom for SQL Server afin d'utiliser une nouvelle version CEV, la version CEV doit avoir un statut `available`.
- Vous ne pouvez pas créer une image AMI ni une version CEV à partir d'une instance de base de données RDS Custom for SQL Server.
- Vous ne pouvez pas modifier une version CEV existante pour utiliser une autre image AMI. Toutefois, vous pouvez modifier une instance de base de données RDS Custom for SQL Server afin d'utiliser une autre version CEV. Pour de plus amples informations, veuillez consulter Modification d'une instance de base de données RDS personnalisée pour le SQL serveur.
- Le chiffrement d'une AMI ou d'un CEV avec une clé KMS gérée par le client différente de la clé KMS fournie lors de la création de l'instance de base de données n'est pas pris en charge.
- La copie interrégionale de CEVs n'est pas prise en charge.
- La copie entre comptes CEVs n'est pas prise en charge.
- Vous ne pouvez pas restaurer ni récupérer une version CEV après l'avoir supprimée. Toutefois, vous pouvez créer une version CEV à partir de la même AMI.
- Une instance de base de données RDS Custom for SQL Server stocke vos fichiers de base de données SQL Server dans le lecteur D:\. L'AMI associée à une version CEV doit stocker les fichiers de base de données système Microsoft SQL Server dans le lecteur C:\.
- Une instance de base de données RDS Custom for SQL Server conserve vos modifications de configuration que vous avez apportées à SQL Server. Les modifications de configuration apportées

au système d'exploitation sur une instance de base de données RDS Custom for SQL Server en cours d'exécution, créée à partir d'une version CEV, ne sont pas conservées. Si vous devez apporter une modification permanente à la configuration du système d'exploitation et la conserver comme nouvelle configuration de base, créez une nouvelle version CEV et modifiez l'instance de base de données pour utiliser cette nouvelle version CEV.

> **Important**
>
> La modification d'une instance de base de données RDS Custom for SQL Server pour utiliser une nouvelle version CEV est une opération hors ligne. Vous pouvez effectuer la modification immédiatement ou la planifier au cours d'un créneau de maintenance hebdomadaire.

- Lorsque vous modifiez une version CEV, Amazon RDS ne transmet (push) ces modifications à aucune instance de base de données RDS Custom for SQL Server associée. Vous devez modifier chaque instance de base de données RDS Custom for SQL Server afin d'utiliser la version CEV nouvelle ou mise à jour. Pour de plus amples informations, veuillez consulter Modification d'une instance de base de données RDS personnalisée pour le SQL serveur.
-

> **Important**
>
> Si une AMI utilisée par une version CEV est supprimée, toutes les modifications pouvant nécessiter le remplacement de l'hôte, par exemple le calcul d'échelle, échoueront. L'instance de base de données RDS Custom for SQL Server sera alors placée en dehors du périmètre de prise en charge RDS. Nous vous recommandons d'éviter de supprimer toute AMI associée à une version CEV.

Création d'un CEV for RDS Custom for SQL Server

Vous pouvez créer une version personnalisée du moteur (CEV) à l'aide du AWS Management Console ou du AWS CLI. Vous pouvez ensuite utiliser le CEV pour créer une instance de base de données RDS personnalisée pour le SQL serveur.

Assurez-vous que l'Amazon Machine Image (AMI) se trouve dans le même AWS compte et dans la même région que le vôtreCEV. Dans le cas contraire, le processus de création d'un CEV échoue.

Pour de plus amples informations, veuillez consulter Création et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server.

> **Important**
>
> Les étapes pour créer un CEV sont les mêmes pour AMIs les créations avec un SQL serveur préinstallé et pour celles créées à l'aide de bring your own media (BYOM).

Console

Pour créer un CEV

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).

   La page Versions personnalisées du moteur affiche tout CEVs ce qui existe actuellement. Si vous n'en avez pas crééCEVs, le tableau est vide.
3. Choisissez Create custom engine version (Créer une version de moteur personnalisée).
4. Pour le type de moteur, choisissez Microsoft SQL Server.
5. Pour Edition, choisissez l'édition du moteur de base de données que vous souhaitez utiliser.
6. Pour la version majeure, choisissez la version principale du moteur installée sur votreAMI.
7. Dans Version details (Détails de la version), saisissez un nom valide dans Custom engine version name (Nom de version de moteur personnalisée).

   Le format de nom est *major-engine-version*.*minor-engine-version*.*customized_string*. Vous pouvez utiliser de 1 à 50 caractères alphanumériques, des traits de soulignement, des tirets et des points. Par exemple, vous pouvez saisir le nom **15.00.4249.2.my_cevtest**.

   Entrez éventuellement une description pour votreCEV.
8. Pour le support d'installation, recherchez ou entrez l'AMIID à CEV partir duquel vous souhaitez le créer.
9. Dans la section Tags, ajoutez des balises pour identifier leCEV.
10. Choisissez Create custom engine version (Créer une version de moteur personnalisée).

La page Custom engine versions (Versions de moteur personnalisées) s'affiche. Votre statut CEV est affiché avec le statut en attente de validation

AWS CLI

Pour créer un CEV en utilisant le AWS CLI, exécutez la commande create-custom-db-engine-version.

Les options suivantes sont requises :

- `--engine`
- `--engine-version`
- `--image-id`

Vous pouvez également spécifier les options suivantes :

- `--description`
- `--region`
- `--tags`

L'exemple suivant crée un CEV nom`15.00.4249.2.my_cevtest`. Assurez-vous que le nom de votre moteur CEV commence par le numéro de version principal du moteur.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-custom-db-engine-version \
    --engine custom-sqlserver-ee \
    --engine-version 15.00.4249.2.my_cevtest \
    --image-id ami-0r93cx31t5r596482 \
    --description "Custom SQL Server EE 15.00.4249.2 cev test"
```

La sortie partielle suivante affiche le moteur, les groupes de paramètres et d'autres informations.

```
"DBEngineVersions": [
    {
    "Engine": "custom-sqlserver-ee",
    "MajorEngineVersion": "15.00",
    "EngineVersion": "15.00.4249.2.my_cevtest",
```

```
    "DBEngineDescription": "Microsoft SQL Server Enterprise Edition for RDS Custom for
 SQL Server",
    "DBEngineVersionArn": "arn:aws:rds:us-east-1:<my-account-id>:cev:custom-sqlserver-
ee/15.00.4249.2.my_cevtest/a1234a1-123c-12rd-bre1-1234567890",
    "DBEngineVersionDescription": "Custom SQL Server EE 15.00.4249.2 cev test",

    "Image": [
        "ImageId": "ami-0r93cx31t5r596482",
        "Status": "pending-validation"
     ],
    "CreateTime": "2022-11-20T19:30:01.831000+00:00",
    "SupportsLogExportsToCloudwatchLogs": false,
    "SupportsReadReplica": false,
    "Status": "pending-validation",
    "SupportsParallelQuery": false,
    "SupportsGlobalDatabases": false,
    "TagList": []
    }
]
```

Si le processus de création d'un CEV échoue, RDS Custom for SQL Server émet un problème `RDS-EVENT-0198` avec le message`Creation failed for custom engine version` *`major-engine-version.cev_name`*. Le message comprend des détails sur l'échec. Par exemple, l'événement imprime les fichiers manquants. Pour trouver des idées de résolution des problèmes de CEV création, consultezRésolution des erreurs de version CEV pour RDS Custom for SQL Server.

Créez une instance de base de données RDS personnalisée pour le SQL serveur à partir d'un CEV

Une fois que vous avez créé unCEV, le CEVstatut s'affiche`pending-validation`. Vous pouvez désormais créer une nouvelle instance de base de données RDS personnalisée pour le SQL serveur à l'aide duCEV. Pour créer une nouvelle instance de base de données RDS personnalisée pour SQL serveur à partir d'unCEV, consultezCréation d'une instance de base de données RDS personnalisée pour le SQL serveur.

Cycle de vie d'un CEV

Le CEV cycle de vie inclut les statuts suivants.

CEVstatut	Description	Suggestions de dépannage	
`pending-validation`	A CEV a été créé et est	S'il n'existe aucune tâche, créez une nouvelle instance de base de données RDS	

CEVstatut	Description	Suggestions de dépannage	
	en attente de validation du code associéAMI. A CEV restera actif `pending-validation` jusqu'à ce qu'une instance de base de données RDS personnalisée pour le SQL serveur soit créée à partir de celui-ci.	personnalisée pour le SQL serveur à partir duCEV. Lors de la création de l'instance de base de données RDS personnalisée pour le SQL serveur, le système tente de valider l'instance associée AMI pour unCEV.	

CEVstatut	Description	Suggestions de dépannage	
`validating`	Une tâche de création pour l'instanc e de base de données RDS Custom for SQL Server basée sur une nouvelle instance CEV est en cours. Lors de la création de l'instance de base de données RDS personnal isée pour le SQL serveur, le système tente de valider l'association AMI d'unCEV.	Attendez que la tâche de création de l'instance de base de données RDS Custom for SQL Server existante soit terminée. Vous pouvez utiliser la RDS EVENTS console pour consulter les messages d'événements détaillés à des fins de résolution des problèmes.	

CEVstatut	Description	Suggestions de dépannage
`available`	Le CEV a été validé avec succès. A CEV entrera le `available` statut une fois qu'une instance de base de données RDS personnalisée pour le SQL serveur aura été créée avec succès à partir de celle-ci.	CEVCela ne nécessite aucune validation supplémentaire. Il peut être utilisé pour créer des instances de base de données RDS personnalisées supplémentaires pour le SQL serveur ou pour modifier des instances existantes.
`inactive`	Le CEV a été modifié pour passer à un état inactif.	Vous ne pouvez pas créer ou mettre à niveau une instance de base de données RDS personnalisée avec celaCEV. De plus, vous ne pouvez pas restaurer un instantané de base de données pour créer une nouvelle instance de base de données RDS personnalisée avec celui-ciCEV. Pour obtenir des informations sur la façon de modifier l'état en `ACTIVE`, consultez Modifier un CEV for RDS Custom for SQL Server.

CEVstatut	Description	Suggestions de dépannage	
`failed`	L'étape de création d'une instance de base de données a échoué pour cela CEV avant de pouvoir valider leAMI. Sinon, le sous-jacent AMI utilisé par le CEV n'est pas dans un état disponible.	Résolvez la cause racine pour laquelle le système n'a pas pu créer l'instance de base de données. Consultez le message d'erreur détaillé et essayez à nouveau de créer une nouvelle instance de base de données. Assurez-vous que le sous-jacent AMI utilisé par le CEV est dans un état disponible.	

CEVstatut	Description	Suggestions de dépannage
`incompatible-image-configuration`	Une erreur s'est produite lors de la validation duAMI.	Consultez les détails techniques de l'erreur. Vous ne pouvez pas essayer de le valider à CEV nouveau AMI avec ceci. Passez en revue les recommandations suivantes : • Assurez-vous que votre nom CEV est utilisé selon le modèle de dénomination requis : version majeure SQL du serveur, version mineure, chaîne personnalisée. • Assurez-vous que la version SQL du serveur indiquée dans le CEV nom correspond à la version fournie avec leAMI. • Veillez à ce que la version de build du système d'exploitation corresponde à la version de build minimale requise. • Veillez à ce que la version majeure du système d'exploitation corresponde à la version de build minimale requise. Créez-en un nouveau CEV en utilisant les informations correctes. Si nécessaire, créez une nouvelle EC2 instance à l'aide d'une instance compatible AMI et exécutez le processus Sysprep dessus.

Modifier un CEV for RDS Custom for SQL Server

Vous pouvez modifier un CEV en utilisant le AWS Management Console ou le AWS CLI. Vous pouvez modifier la CEV description ou son statut de disponibilité. Votre statut CEV possède l'une des valeurs suivantes :

- `available`— Vous pouvez l'utiliser CEV pour créer une nouvelle instance de base de données RDS personnalisée ou pour mettre à niveau une instance de base de données. Il s'agit du statut par défaut d'une personne nouvellement crééeCEV.
- `inactive`— Vous ne pouvez pas créer ou mettre à niveau une instance de base de données RDS personnalisée avec celaCEV. Vous ne pouvez pas restaurer un instantané de base de données pour créer une nouvelle instance de base de données RDS personnalisée avec celaCEV.

Vous pouvez modifier le CEV statut de `available` à `inactive` ou de `inactive` à`available`. Vous pouvez modifier le statut `INACTIVE` pour empêcher l'utilisation accidentelle d'un CEV ou pour qu'un produit abandonné puisse à nouveau CEV être réutilisé.

Console

Pour modifier un CEV

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).
3. Choisissez CEV la description ou le statut dont vous souhaitez modifier la description.
4. Pour Actions, choisissez Modifier.
5. Effectuez une ou plusieurs des modifications suivantes :
   - Pour les paramètres de CEV statut, choisissez un nouveau statut de disponibilité.
   - Sur la page Version description (Description de la version), saisissez une nouvelle description.
6. Sélectionnez ModifierCEV.

   Si le CEV est en cours d'utilisation, la console affiche Vous ne pouvez pas modifier le CEV statut. Résolvez les problèmes, puis réessayez.

La page Custom engine versions (Versions de moteur personnalisées) s'affiche.

AWS CLI

Pour modifier un à CEV l'aide de AWS CLI, exécutez la commande modify-custom-db-engine-version. Vous pouvez trouver CEVs un moyen de modifier en exécutant la describe-db-engine-versionscommande.

Les options suivantes sont requises :

- `--engine`
- `--engine-version` *`cev`*, où *`cev`* est le nom de la version personnalisée du moteur que vous souhaitez modifier
- `--status` *`status`*, où *`status`* est le statut de disponibilité que vous souhaitez attribuer au CEV

L'exemple suivant remplace le statut actuel `15.00.4249.2.my_cevtest` d'un CEV nom par`inactive`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-custom-db-engine-version \
    --engine custom-sqlserver-ee \
    --engine-version 15.00.4249.2.my_cevtest \
    --status inactive
```

Dans Windows:

```
aws rds modify-custom-db-engine-version ^
    --engine custom-sqlserver-ee ^
    --engine-version 15.00.4249.2.my_cevtest ^
    --status inactive
```

Pour modifier une instance de base de données RDS Custom for SQL Server afin d'en utiliser une nouvelleCEV, consultezModification d'une instance de base de données RDS personnalisée pour le SQL serveur afin d'utiliser une nouvelle CEV.

Modification d'une instance de base de données RDS personnalisée pour le SQL serveur afin d'utiliser une nouvelle CEV

Vous pouvez modifier une instance de base de données RDS Custom for SQL Server existante pour en utiliser une autreCEV. Les modifications que vous pouvez apporter incluent :

- Modification du CEV
- Modification de la classe d'instance

- Modification de la période de rétention des sauvegardes et de la fenêtre de sauvegarde
- Modification de la fenêtre de maintenance

Console

Pour modifier une instance de base de données RDS personnalisée pour le SQL serveur

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de bases de données que vous souhaitez modifier.
4. Sélectionnez Modify.
5. Effectuez les modifications suivantes selon les besoins :
   a. Pour la version du moteur de base de données, choisissez-en une autreCEV.
   b. Modifiez la valeur de Classe d'instance de base de données. Pour connaître les classes prises en charge, consultez Support de classe d'instance de base de données pour RDS Custom for SQL Server.
   c. Modifiez la valeur de Période de rétention des sauvegardes.
   d. Dans le champ Fenêtre de sauvegarde, définissez les valeurs pour Heure de début et Durée.
   e. Dans le champ Fenêtre de maintenance d'instance de bases de données, définissez les valeurs pour Jour de début, Heure de début et Durée.
6. Choisissez Continuer.
7. Sélectionnez Appliquer immédiatement ou Appliquer au cours de la prochaine fenêtre de maintenance planifiée.
8. Choisissez Modifier l'instance de base de données.

> **Note**
>
> Lors de la modification d'une instance de base de données de l'une CEV à l'autreCEV, par exemple lors de la mise à niveau d'une version mineure, les bases de données SQL du système du serveur, y compris leurs données et leurs configurations, sont conservées à partir de l'instance de base de données RDS Custom for SQL Server actuelle.

AWS CLI

Pour modifier une instance de base de données afin d'CEVen utiliser une autre à l'aide de AWS CLI, exécutez la modify-db-instancecommande.

Les options suivantes sont requises :

- `--db-instance-identifier`
- `--engine-version` *`cev`*, où *`cev`* est le nom de la version du moteur personnalisé que vous souhaitez remplacer par l'instance de base de données.

L'exemple suivant modifie une instance de base de données nommée `my-cev-db-instance` pour utiliser un CEV named `15.00.4249.2.my_cevtest_new` et applique la modification immédiatement.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-cev-db-instance \
    --engine-version 15.00.4249.2.my_cevtest_new \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-cev-db-instance ^
    --engine-version 15.00.4249.2.my_cevtest_new ^
    --apply-immediately
```

Affichage des détails de version CEV pour Amazon RDS Custom for SQL Server

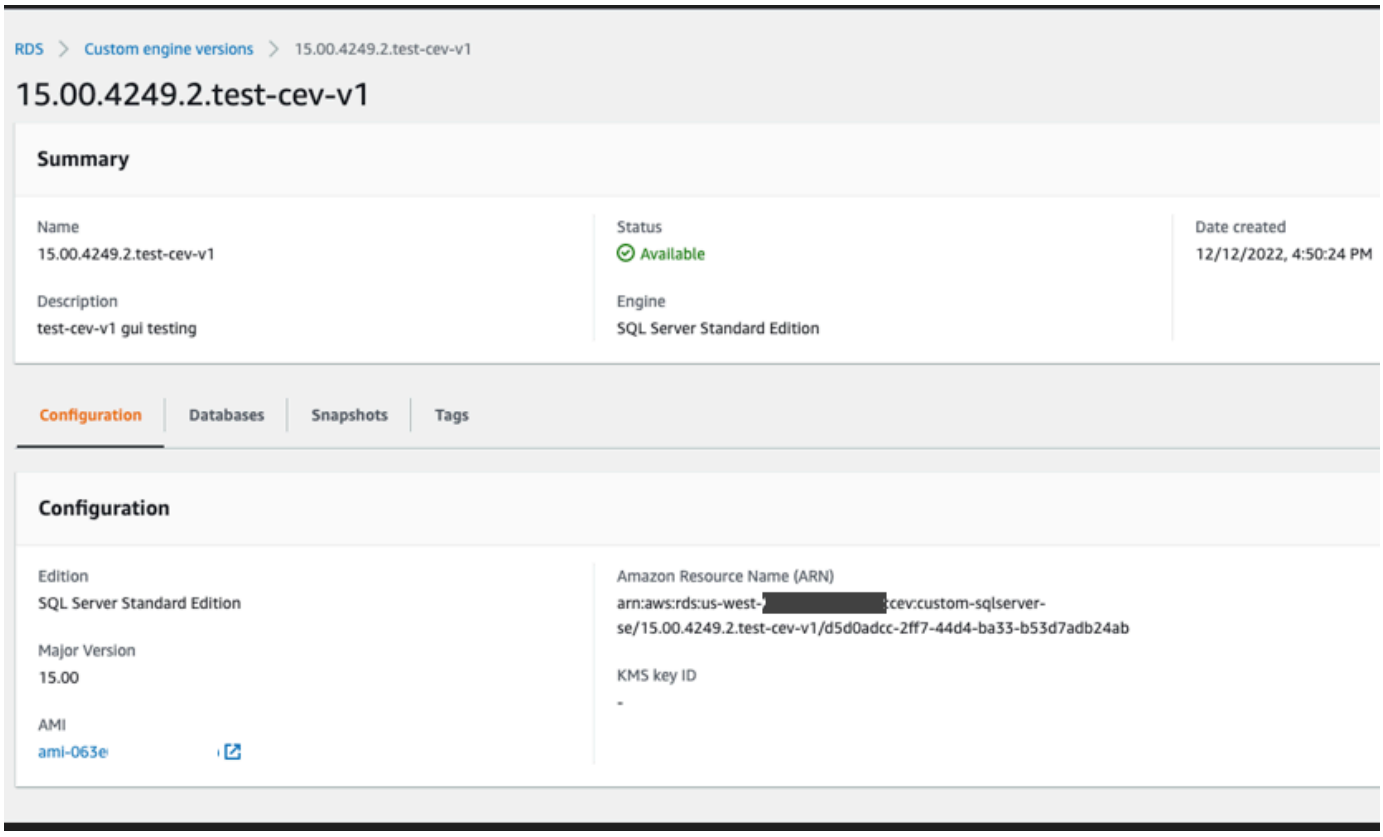
Vous pouvez consulter les détails de votre version CEV en utilisant la AWS Management Console ou AWS CLI.

Console

Pour afficher les détails de la version CEV

1. Connectez-vous à l'AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).

   La page Custom engine versions (Versions de moteur personnalisées) affiche toutes les CEV qui existent actuellement. Si vous n'avez pas créé de CEV, la page est vide.
3. Choisissez le nom de la version CEV que vous souhaitez consulter.
4. Choisissez Configuration pour afficher les détails.



AWS CLI

Pour afficher les détails d'une version CEV à l'aide d'AWS CLI, exécutez la commande describe-db-engine-versions.

Vous pouvez également spécifier les options suivantes :

- `--include-all`, pour afficher toutes les versions CEV, quel que soit leur état de cycle de vie. Sans l'option `--include-all`, seules les versions CEV dans un état de cycle de vie `available` seront renvoyées.

```
aws rds describe-db-engine-versions --engine custom-sqlserver-ee --engine-version
 15.00.4249.2.my_cevtest --include-all
{
    "DBEngineVersions": [
        {
            "Engine": "custom-sqlserver-ee",
            "MajorEngineVersion": "15.00",
            "EngineVersion": "15.00.4249.2.my_cevtest",
            "DBParameterGroupFamily": "custom-sqlserver-ee-15.0",
            "DBEngineDescription": "Microsoft SQL Server Enterprise Edition for custom
 RDS",
            "DBEngineVersionArn": "arn:aws:rds:us-east-1:{my-account-id}:cev:custom-
sqlserver-ee/15.00.4249.2.my_cevtest/a1234a1-123c-12rd-bre1-1234567890",
            "DBEngineVersionDescription": "Custom SQL Server EE 15.00.4249.2 cev test",
            "Image": {
                "ImageId": "ami-0r93cx31t5r596482",
                "Status": "pending-validation"
            },
            "DBEngineMediaType": "AWS Provided",
            "CreateTime": "2022-11-20T19:30:01.831000+00:00",
            "ValidUpgradeTarget": [],
            "SupportsLogExportsToCloudwatchLogs": false,
            "SupportsReadReplica": false,
            "SupportedFeatureNames": [],
            "Status": "pending-validation",
            "SupportsParallelQuery": false,
            "SupportsGlobalDatabases": false,
            "TagList": [],
            "SupportsBabelfish": false
        }
    ]
}
```

Vous pouvez utiliser des filtres pour afficher les versions CEV ayant un certain statut de cycle de vie. Par exemple, pour afficher les CEV dont le statut de cycle de vie est `pending-validation`, `available` ou `failed` :

```
aws rds describe-db-engine-versions engine custom-sqlserver-ee
                region us-west-2 include-all query 'DBEngineVersions[?Status ==
 pending-validation ||
                Status == available || Status == failed]'
```

Suppression d'une version CEV pour RDS Custom for SQL Server

Vous pouvez supprimer une CEV à l'aide de la AWS Management Console ou de la AWS CLI. Cette tâche prend généralement quelques minutes.

Avant de supprimer une CEV, veillez à ce qu'elle ne soit pas utilisée par aucun des éléments suivants :

- Une instance de base de données RDS Custom
- Un instantané d'une instance de base de données RDS Custom
- Une sauvegarde automatisée de votre instance de base de données RDS Custom

Console

Pour supprimer une CEV

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Custom engine versions (Versions de moteur personnalisées).
3. Choisissez une CEV dont vous souhaitez supprimer la description ou l'état.
4. Pour Actions, choisissez Supprimer.

   La boîte de dialogue Delete ***cev_name***? (Supprimer cev_name ?) s'affiche.
5. Entrez **`delete me`**, puis choisissez Delete (Supprimer).

   Dans Custom engine versions (Versions de moteur personnalisées), la bannière indique que votre CEV est en cours de suppression.

AWS CLI

Pour supprimer un CEV à l'aide deAWS CLI, exécutez la commande delete-custom-db-engine-version.

Les options suivantes sont requises :

- `--engine custom-sqlserver-ee`
- `--engine-version` *`cev`*, où *`cev`* représente le nom de la version de moteur personnalisée à supprimer

L'exemple suivant supprime une CEV nommée `15.00.4249.2.my_cevtest`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds delete-custom-db-engine-version \
    --engine custom-sqlserver-ee \
    --engine-version 15.00.4249.2.my_cevtest
```

Dans Windows :

```
aws rds delete-custom-db-engine-version ^
    --engine custom-sqlserver-ee ^
    --engine-version 15.00.4249.2.my_cevtest
```

Création et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server

Vous pouvez créer une instance de base de données RDS personnalisée, puis vous y connecter à l'aide AWS Systems Manager du protocole Remote Desktop (RDP).

> **Important**
>
> Avant de créer ou de vous connecter à une instance de base de données RDS personnalisée pour le SQL serveur, assurez-vous d'effectuer les tâches dansConfiguration de votre environnement pour Amazon RDS Custom for SQL Server.
> Vous pouvez étiqueter les instances de base de données RDS personnalisées lorsque vous les créez, mais ne créez ni ne modifiez la `AWSRDSCustom` balise requise pour l'automatisation RDS personnalisée. Pour de plus amples informations, veuillez consulter Balisage de ressources RDS Custom for SQL Server.
> La première fois que vous créez une instance de base de données RDS personnalisée pour le SQL serveur, le message d'erreur suivant peut s'afficher : Le rôle lié au service est en cours de création. Réessayez ultérieurement. Dans ce cas, attendez quelques minutes, puis réessayez de créer l'instance de base de données.

Rubriques

- Création d'une instance de base de données RDS personnalisée pour le SQL serveur
- RDSRôle personnalisé lié à un service
- Connexion à votre instance de base de données RDS personnalisée à l'aide de AWS Systems Manager
- Connexion à votre instance de base de données RDS personnalisée à l'aide de RDP

Création d'une instance de base de données RDS personnalisée pour le SQL serveur

Créez une instance de base de données Amazon RDS Custom for SQL Server à l'aide du AWS Management Console ou du AWS CLI. La procédure est similaire à la procédure de création d'une RDS instance de base de données Amazon.

Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

Console

Pour créer une instance de base de données RDS personnalisée pour le SQL serveur

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez Create database (Créer une base de données).
4. Sélectionnez Création standard comme méthode de création de la base de données.
5. Pour les options du moteur, choisissez Microsoft SQL Server comme type de moteur.
6. Pour le type de gestion de base de données, choisissez Amazon RDS Custom.
7. Dans la section Edition (Édition), sélectionnez l'édition du moteur de base de données que vous voulez utiliser.
8. (Facultatif) Si vous avez l'intention de créer l'instance de base de données à partir d'unCEV, cochez la case Utiliser une version de moteur personnalisée (CEV). Sélectionnez votre CEV dans la liste déroulante.
9. Pour la version de base de données, conservez la version par défaut.
10. Pour Modèles, sélectionnez Production.
11. Dans Settings (Paramètres), saisissez un nouveau nom pour DB instance identifier (Identifiant d'instance de base de données).
12. Pour entrer votre mot de passe principal, procédez comme suit :

    a. Dans la section Settings (Paramètres), ouvrez Credential Settings (Paramètres des informations d'identification).

    b. Décochez la case Auto generate a password (Générer un mot de passe automatiquement).

    c. Modifiez la valeur du champ Master username (Identifiant principal) et saisissez le même mot de passe dans les champs Master password (Mot de passe principal) et Confirm password (Confirmer le mot de passe).

    Par défaut, la nouvelle instance de base de données RDS personnalisée utilise un mot de passe généré automatiquement pour l'utilisateur principal.
13. Dans DB instance size (Taille de l'instance de base de données), choisissez une valeur pour DB instance class (Classe d'instance de base de données).

Pour connaître les classes prises en charge, voir Support de classe d'instance de base de données pour RDS Custom for SQL Server.

14. Choisissez les paramètres Storage (Stockage).
15. Pour une sécurité RDS personnalisée, procédez comme suit :

    a. Pour le profil d'IAMinstance, vous avez deux options pour choisir le profil d'instance de votre instance de base de données RDS Custom for SQL Server.

        1. Choisissez Créer un nouveau profil d'instance et fournissez un suffixe de nom de profil d'instance. Pour de plus amples informations, veuillez consulter Création automatique de profils d'instance à l'aide du AWS Management Console.
        2. Choisissez un profil d'instance existant. Dans la liste déroulante, choisissez le profil d'instance qui commence par. `AWSRDSCustom`

    b. Pour le chiffrement, choisissez Entrer une clé ARN pour répertorier les AWS KMS clés disponibles. Choisissez ensuite votre clé dans la liste.

        Une AWS KMS clé est requise pour RDS Custom. Pour de plus amples informations, veuillez consulter Vérifiez que vous disposez d'une clé de chiffrement AWS KMS symétrique.

16. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données RDS personnalisés préférés. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données. Les paramètres suivants n'apparaissent pas dans la console et ne sont pas pris en charge :

    - Processor features (Caractéristiques du processeur)
    - Dimensionnement automatique du stockage
    - Disponibilité et durabilité
    - Option Password and Kerberos authentication (Mot de passe et authentification Kerberos) dans Database authentication (Authentification de base de données) (seule l'authentification par mot de passe est prise en charge)
    - Groupe Database options (Options de base de données) dans Additional configuration (Configuration supplémentaire)
    - Performance Insights
    - Exportations des journaux

- Enable auto minor version upgrade (Activer la mise à niveau automatique de versions mineures)
- Deletion protection (Protection contre la suppression)

Backup retention period (Période de rétention des sauvegardes) est pris en charge, mais vous ne pouvez pas choisir0 days (0 jours).

17. Choisissez Créer une base de données.

    Le bouton View credential details (Afficher les détails des informations d'identification) apparaît sur la page Databases (Bases de données).

    Pour afficher le nom d'utilisateur et le mot de passe principaux de l'instance de base de données RDS personnalisée, choisissez Afficher les détails des informations d'identification.

    Pour vous connecter à l'instance de base de données en tant qu'utilisateur principal, utilisez l'identifiant et le mot de passe affichés.

    > **Important**
    >
    > Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier. Pour modifier le mot de passe de l'utilisateur principal une fois que l'instance de base de données RDS personnalisée est disponible, modifiez l'instance de base de données. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Gestion d'une instance de base de données Amazon RDS Custom for SQL Server.

18. Choisissez Databases pour afficher la liste des instances de base de données RDS personnalisées.
19. Choisissez l'instance de base de données RDS personnalisée que vous venez de créer.

    Sur la RDS console, les détails de la nouvelle instance de base de données RDS personnalisée apparaissent :

    - L'instance de base de données a le statut de création jusqu'à ce que l'instance de base de données RDS personnalisée soit créée et prête à être utilisée. Lorsque l'état devient available (disponible), vous pouvez vous connecter à l'instance de base de données. En fonction du

stockage et de la classe d'instance alloués, la mise à disposition de la nouvelle instance de base de données peut nécessiter plusieurs minutes.

- Le rôle a la valeur Instance (RDSCustom).
- RDSLe mode d'automatisation personnalisé a la valeur Automatisation complète. Ce paramètre signifie que l'instance de base de données assure une surveillance et une récupération d'instance automatiques.

AWS CLI

Vous créez une instance de base de données RDS personnalisée à l'aide de la create-db-instance AWS CLI commande.

Les options suivantes sont requises :

- `--db-instance-identifier`
- `--db-instance-class` Pour obtenir la liste des classes d'instance de base de données prises en charge, veuillez consulter Support de classe d'instance de base de données pour RDS Custom for SQL Server).
- `--engine` (`custom-sqlserver-ee`, `custom-sqlserver-se`, ou `custom-sqlserver-web`)
- `--kms-key-id`
- `--custom-iam-instance-profile`

L'exemple suivant crée une instance de base de données RDS personnalisée pour le SQL serveur nomméemy`-custom-instance`. La période de rétention des sauvegardes est de 3 jours.

> **Note**
>
> Pour créer une instance de base de données à partir d'une version de moteur personnalisée (CEV), attribuez un CEV nom existant au `--engine-version` paramètre. Par exemple, `--engine-version 15.00.4249.2.my_cevtest`

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
```

```
    --engine custom-sqlserver-ee \
    --engine-version 15.00.4073.23.v1 \
    --db-instance-identifier my-custom-instance \
    --db-instance-class db.m5.xlarge \
    --allocated-storage 20 \
    --db-subnet-group mydbsubnetgroup \
    --master-username myuser \
    --master-user-password mypassword \
    --backup-retention-period 3 \
    --no-multi-az \
    --port 8200 \
    --kms-key-id mykmskey \
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance
```

Dans Windows:

```
aws rds create-db-instance ^
    --engine custom-sqlserver-ee ^
    --engine-version 15.00.4073.23.v1 ^
    --db-instance-identifier my-custom-instance ^
    --db-instance-class db.m5.xlarge ^
    --allocated-storage 20 ^
    --db-subnet-group mydbsubnetgroup ^
    --master-username myuser ^
    --master-user-password mypassword ^
    --backup-retention-period 3 ^
    --no-multi-az ^
    --port 8200 ^
    --kms-key-id mykmskey ^
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Obtenez des détails sur votre instance en utilisant la commande `describe-db-instances`.

```
aws rds describe-db-instances --db-instance-identifier my-custom-instance
```

La sortie partielle suivante affiche le moteur, les groupes de paramètres et d'autres informations.

```
{
    "DBInstances": [
        {
            "PendingModifiedValues": {},
            "Engine": "custom-sqlserver-ee",
            "MultiAZ": false,
            "DBSecurityGroups": [],
            "DBParameterGroups": [
                {
                    "DBParameterGroupName": "default.custom-sqlserver-ee-15",
                    "ParameterApplyStatus": "in-sync"
                }
            ],
            "AutomationMode": "full",
            "DBInstanceIdentifier": "my-custom-instance",
            "TagList": []
        }
    ]
}
```

RDSRôle personnalisé lié à un service

Un rôle lié à un service permet à Amazon RDS Custom d'accéder aux ressources de votre. Compte AWS Cela facilite l'utilisation de RDS Custom car vous n'avez pas à ajouter manuellement les autorisations nécessaires. RDSCustom définit les autorisations associées à ses rôles liés aux services et, sauf indication contraire, seul RDS Custom peut assumer ses rôles. Les autorisations définies incluent la politique de confiance et la politique d'autorisations, et cette politique d'autorisations ne peut être attachée à aucune autre IAM entité.

Lorsque vous créez une instance de base de données RDS personnalisée, les rôles Amazon RDS et RDS Custom liés à un service sont créés (s'ils n'existent pas déjà) et utilisés. Pour de plus amples informations, veuillez consulter Utilisation des rôles liés à un service pour Amazon RDS.

La première fois que vous créez une instance de base de données RDS personnalisée pour le SQL serveur, le message d'erreur suivant peut s'afficher : Le rôle lié au service est en cours de création. Réessayez ultérieurement. Dans ce cas, attendez quelques minutes, puis réessayez de créer l'instance de base de données.

Connexion à votre instance de base de données RDS personnalisée à l'aide de AWS Systems Manager

Après avoir créé votre instance de base de données RDS personnalisée, vous pouvez vous y connecter à l'aide du gestionnaire de AWS Systems Manager session. Session Manager est une fonctionnalité de Systems Manager que vous pouvez utiliser pour gérer les EC2 instances Amazon via un shell basé sur un navigateur ou via le. AWS CLI Pour plus d'informations, consultez AWS Systems Manager Session Manager.

Console

Connexion à votre instance de base de données à l'aide de Session Manager

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée à laquelle vous souhaitez vous connecter.
3. Choisissez Configuration.
4. Notez la valeur de l'ID de ressource pour votre instance de base de données. Par exemple, l'ID de la ressource peut être `db-ABCDEFGHIJKLMNOPQRS0123456`.
5. Ouvrez la EC2 console Amazon à l'adresse https://console.aws.amazon.com/ec2/.
6. Dans le panneau de navigation, choisissez Instances.
7. Recherchez le nom de votre EC2 instance, puis choisissez l'ID d'instance qui lui est associé. Par exemple, l'ID d'instance peut être `i-abcdefghijklm01234`.
8. Choisissez Se connecter.
9. Choisissez Session Manager.
10. Choisissez Se connecter.

    Une fenêtre s'ouvre pour votre session.

AWS CLI

Vous pouvez vous connecter à votre instance de base de données RDS personnalisée à l'aide du AWS CLI. Cette technique nécessite le plugin Session Manager pour la AWS CLI. Pour en savoir plus sur l'installation du plugin, veuillez consulter Installez le plugin du gestionnaire de session pour la AWS CLI.

Pour trouver l'ID de ressource de base de données de votre instance de base de données RDS personnalisée, utilisez`describe-db-instances`.

```
aws rds describe-db-instances \
    --query 'DBInstances[*].[DBInstanceIdentifier,DbiResourceId]' \
    --output text
```

L'exemple de sortie suivant montre l'ID de ressource de votre instance RDS personnalisée. Le préfixe est `db-`.

```
db-ABCDEFGHIJKLMNOPQRS0123456
```

Pour trouver l'ID d'EC2instance de votre instance de base de données, utilisez`aws ec2 describe-instances`. L'exemple suivant utilise `db-ABCDEFGHIJKLMNOPQRS0123456` pour l'ID de la ressource

```
aws ec2 describe-instances \
    --filters "Name=tag:Name,Values=db-ABCDEFGHIJKLMNOPQRS0123456" \
    --output text \
    --query 'Reservations[*].Instances[*].InstanceId'
```

L'exemple de sortie suivant montre l'ID de l'EC2instance.

```
i-abcdefghijklm01234
```

Utilisez la `aws ssm start-session` commande en fournissant l'ID de l'EC2instance dans le `--target` paramètre.

```
aws ssm start-session --target "i-abcdefghijklm01234"
```

Un résultat réussi ressemble à ce qui suit.

```
Starting session with SessionId: yourid-abcdefghijklm1234
[ssm-user@ip-123-45-67-89 bin]$
```

Connexion à votre instance de base de données RDS personnalisée à l'aide de RDP

Après avoir créé votre instance de base de données RDS personnalisée, vous pouvez vous connecter à cette instance à l'aide d'un RDP client. La procédure est la même que pour la connexion

à une EC2 instance Amazon. Pour plus d'informations, consultez Connexion à votre instance Windows.

Pour vous connecter à l'instance de base de données, vous devez disposer de la paire de clés associée à l'instance. RDSCustom crée la paire de clés pour vous. Le nom de la paire utilise le préfixe `do-not-delete-rds-custom-`*`DBInstanceIdentifier`*. AWS Secrets Manager stocke votre clé privée comme secret.

Accomplissez la tâche suivant les étapes suivantes :

1. Configurez votre instance de base de données pour autoriser RDP les connexions.
2. Récupérer votre clé secrète.
3. Connectez-vous à votre EC2 instance à l'aide de l'RDPutilitaire.

Configurez votre instance de base de données pour autoriser RDP les connexions

Pour autoriser RDP les connexions, configurez votre groupe VPC de sécurité et définissez une règle de pare-feu sur l'hôte.

Configurez votre groupe VPC de sécurité

Assurez-vous que le groupe VPC de sécurité associé à votre instance de base de données autorise les connexions entrantes sur le port 3389 pour le protocole de contrôle de transmission ()TCP. Pour savoir comment configurer votre groupe VPC de sécurité, consultezConfigurez vos groupes de sécurité VPC.

Définir la règle de pare-feu sur l'hôte

Pour autoriser les connexions entrantes sur le port 3389 pourTCP, définissez une règle de pare-feu sur l'hôte. Les exemples suivants illustrent la marche à suivre.

Nous vous recommandons d'utiliser la valeur `-Profile` spécifique : `Public`, `Private` ou `Domain`. L'utilisation de `Any` fait référence aux trois valeurs. Vous pouvez également spécifier une combinaison de valeurs séparées par une virgule. Pour plus d'informations sur la définition des règles de pare-feu, consultez Set- NetFirewallRule dans la documentation Microsoft.

Pour utiliser Systems Manager Session Manager afin de configurer une règle de pare-feu

1. Connectez-vous à Session Manager comme illustré dans Connexion à votre instance de base de données RDS personnalisée à l'aide de AWS Systems Manager.

2. Exécutez la commande suivante.

```
Set-NetFirewallRule -DisplayName "Remote Desktop - User Mode (TCP-In)" -Direction
 Inbound -LocalAddress Any -Profile Any
```

Pour utiliser les CLI commandes de Systems Manager afin de définir une règle de pare-feu

1. Utilisez la commande suivante pour ouvrir RDP sur l'hôte.

```
OPEN_RDP_COMMAND_ID=$(aws ssm send-command --region $AWS_REGION \
    --instance-ids $RDS_CUSTOM_INSTANCE_EC2_ID \
    --document-name "AWS-RunPowerShellScript" \
    --parameters '{"commands":["Set-NetFirewallRule -DisplayName \"Remote Desktop -
 User Mode (TCP-In)\" -Direction Inbound -LocalAddress Any -Profile Any"]}' \
    --comment "Open RDP port" | jq -r ".Command.CommandId")
```

2. Utilisez l'ID de commande renvoyé dans la sortie pour obtenir le statut de la commande précédente. Pour utiliser la requête suivante afin de renvoyer l'ID de commande, vérifiez que le plug-in jq est bien installé.

```
aws ssm list-commands \
    --region $AWS_REGION \
    --command-id $OPEN_RDP_COMMAND_ID
```

Récupérer votre clé secrète

Récupérez votre clé secrète en utilisant l'un AWS Management Console ou l'autre des AWS CLI.

Console

Pour récupérer la clé secrète

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée à laquelle vous souhaitez vous connecter.
3. Cliquez sur l’onglet Configuration.

4. Notez l'ID d'instance de base de données de votre instance de base de données, par exemple, *my-custom-instance*.
5. Ouvrez la EC2 console Amazon à l'adresse https://console.aws.amazon.com/ec2/.
6. Dans le panneau de navigation, choisissez Instances.
7. Recherchez le nom de votre EC2 instance, puis choisissez l'ID d'instance qui lui est associé.

   Dans cet exemple, l'ID d'instance est `i-abcdefghijklm01234`.
8. Dans Details (Détails), cherchez Key pair name (Nom de la paire de clés). Le nom de la paire inclut l'identificateur de base de données. Dans cet exemple, le nom de la paire est `do-not-delete-rds-custom-`*`my-custom-instance`*`-0d726c`.
9. Dans le résumé de l'instance, recherchez Public IPv4 DNS. Par exemple, le public DNS pourrait l'êtreec2-12-345-678-901.us-east-2.compute.amazonaws.com.
10. Ouvrez la AWS Secrets Manager console à l'adresse https://console.aws.amazon.com/secretsmanager/.
11. Choisissez le secret portant le même nom que votre paire de clés.
12. Choisissez Retrieve secret value (Récupérer la valeur d'un secret).

AWS CLI

Pour récupérer la clé privée

1. Obtenez la liste de vos instances de base de données RDS personnalisées en appelant la `aws rds describe-db-instances` commande.

   ```
   aws rds describe-db-instances \
       --query 'DBInstances[*].[DBInstanceIdentifier,DbiResourceId]' \
       --output text
   ```
2. Choisissez l'identifiant d'instance de base de données dans l'exemple de sortie, par exemple `do-not-delete-rds-custom-`*`my-custom-instance`*.
3. Trouvez l'ID d'EC2instance de votre instance de base de données en appelant la `aws ec2 describe-instances` commande. L'exemple suivant utilise le nom de l'EC2instance pour décrire l'instance de base de données.

   ```
   aws ec2 describe-instances \
       --filters "Name=tag:Name,Values=do-not-delete-rds-custom-my-custom-instance" \
       --output text \
   ```

```
    --query 'Reservations[*].Instances[*].InstanceId'
```

L'exemple de sortie suivant montre l'ID de l'EC2instance.

```
i-abcdefghijklm01234
```

4. Trouvez le nom de la clé en spécifiant l'ID de l'EC2instance, comme indiqué dans l'exemple suivant.

```
aws ec2 describe-instances \
    --instance-ids i-abcdefghijklm01234 \
    --output text \
    --query 'Reservations[*].Instances[*].KeyName'
```

L'exemple de sortie suivant montre le nom de la clé, qui utilise le préfixe `do-not-delete-rds-custom-`*`DBInstanceIdentifier`*.

```
do-not-delete-rds-custom-my-custom-instance-0d726c
```

Connectez-vous à votre EC2 instance à l'aide de l'RDPutilitaire

Suivez la procédure décrite dans Connect to your Windows instance (Connect to your Windows instance) RDP à l'aide du guide de EC2 l'utilisateur Amazon. Cette procédure suppose que vous ayez créé un fichier .pem qui contient votre clé privée.

Gestion d'une instance de base de données Amazon RDS Custom for SQL Server

Amazon RDS Custom for SQL Server prend en charge un sous-ensemble des tâches de gestion habituelles pour les RDS instances de base de données Amazon. Vous trouverez ci-dessous des instructions pour les tâches de gestion RDS personnalisées prises en charge par le SQL serveur à l'aide AWS Management Console des AWS CLI.

Rubriques

- Suspension et reprise de l'automatisation personnalisée RDS
- Modification d'une instance de base de données RDS personnalisée pour le SQL serveur
- Modification du stockage pour une instance de base de données RDS Custom for SQL Server
- Balisage de ressources RDS Custom for SQL Server
- Suppression d'une instance de base de données RDS personnalisée pour le SQL serveur
- Démarrage et arrêt d'une instance de base de données RDS personnalisée pour SQL serveur

Suspension et reprise de l'automatisation personnalisée RDS

RDSCustom assure automatiquement la surveillance et la restauration d'instance pour une instance de base de données RDS Custom for SQL Server. Si l'instance doit être personnalisée, procédez comme suit :

1. Suspendez l'automatisation RDS personnalisée pendant une période spécifiée. La pause garantit que vos personnalisations n'interfèrent pas avec l'automatisation RDS personnalisée.
2. Personnalisez l'instance de base de données RDS personnalisée pour le SQL serveur selon vos besoins.
3. Effectuez l’une des actions suivantes :
   - Relancez manuellement l'automatisation.
   - Attendez la fin de la période de pause. Dans ce cas, RDS Custom reprend automatiquement la surveillance et la restauration de l'instance.

> **Important**
>
> La suspension et la reprise de l'automatisation sont les seules tâches d'automatisation prises en charge lors de la modification d'une instance de base de données RDS Custom for SQL Server.

Console

Pour suspendre ou reprendre l'automatisation RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée que vous souhaitez modifier.
3. Sélectionnez Modifier. La page Modifier l'instance de base de données s'affiche.
4. Pour le mode d'automatisation RDS personnalisé, choisissez l'une des options suivantes :
   - La pause met en pause la surveillance et la restauration de l'instance de base de données RDS personnalisée. Saisissez la durée de pause souhaitée (en minutes) pour Automation mode duration (Durée du mode d'automatisation). La valeur minimale est de 60 minutes (par défaut). La valeur maximale est de 1 440 minutes.
   - L'option Full automation (Automatisation complète) relance l'automatisation.
5. Sélectionnez Continuer pour consulter le récapitulatif des modifications.

   Un message indique que RDS Custom appliquera les modifications immédiatement.
6. Si elles sont correctes, sélectionnez Modifier l'instance de base de données. Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

   Les détails de la modification apparaissent sur la RDS console. Si vous avez suspendu l'automatisation, le statut de votre instance de base de données RDS personnalisée indique que l'automatisation est suspendue.
7. (Facultatif) Dans le volet de navigation, choisissez Databases, puis votre instance de base de données RDS personnalisée.

Dans le volet Résumé, le mode d'automatisation RDS personnalisé indique l'état de l'automatisation. Si l'automatisation est suspendue, la valeur est Suspendue. L'automatisation reprend en ***num*** minutes.

AWS CLI

Pour suspendre ou reprendre l'automatisation RDS personnalisée, utilisez la `modify-db-instance` AWS CLI commande. Identifiez l'instance de base de données à l'aide du paramètre requis `--db-instance-identifier`. Contrôlez le mode d'automatisation avec les paramètres suivants :

- `--automation-mode` spécifie l'état de pause de l'instance de base de données. Les valeurs valides sont `all-paused`, qui suspend l'automatisation, et `full`, qui relance l'opération.
- `--resume-full-automation-mode-minutes` spécifie la durée de la pause. La valeur par défaut est de 60 minutes.

> **Note**
>
> Que vous spécifiiez `--no-apply-immediately` ou non`--apply-immediately`, RDS Custom applique les modifications de manière asynchrone dès que possible.

Dans la réponse à la commande, `ResumeFullAutomationModeTime` indique l'heure de reprise sous forme d'UTChorodatage. Lorsque le mode d'automatisation est `all-paused`, vous pouvez utiliser `modify-db-instance` pour relancer le mode d'automatisation ou prolonger la période de pause. Aucune autre option `modify-db-instance` n'est prise en charge.

L'exemple suivant suspend pendant 90 minutes l'automatisation de l'instance `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode all-paused \
    --resume-full-automation-mode-minutes 90
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode all-paused ^
    --resume-full-automation-mode-minutes 90
```

L'exemple suivant prolonge la durée de pause de 30 minutes. Les 30 minutes sont ajoutées à la durée d'origine affichée dans `ResumeFullAutomationModeTime`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode all-paused \
    --resume-full-automation-mode-minutes 30
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode all-paused ^
    --resume-full-automation-mode-minutes 30
```

L'exemple suivant reprend l'automatisation complète pour `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --automation-mode full \
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --automation-mode full
```

Dans l'exemple de sortie partielle ci-dessous, la valeur `AutomationMode` en attente est `full`.

```
{
    "DBInstance": {
        "PubliclyAccessible": true,
        "MasterUsername": "admin",
        "MonitoringInterval": 0,
        "LicenseModel": "bring-your-own-license",
        "VpcSecurityGroups": [
            {
                "Status": "active",
                "VpcSecurityGroupId": "0123456789abcdefg"
            }
        ],
        "InstanceCreateTime": "2020-11-07T19:50:06.193Z",
        "CopyTagsToSnapshot": false,
        "OptionGroupMemberships": [
            {
                "Status": "in-sync",
                "OptionGroupName": "default:custom-oracle-ee-19"
            }
        ],
        "PendingModifiedValues": {
            "AutomationMode": "full"
        },
        "Engine": "custom-oracle-ee",
        "MultiAZ": false,
        "DBSecurityGroups": [],
        "DBParameterGroups": [
            {
                "DBParameterGroupName": "default.custom-oracle-ee-19",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        ...
        "ReadReplicaDBInstanceIdentifiers": [],
        "AllocatedStorage": 250,
        "DBInstanceArn": "arn:aws:rds:us-west-2:012345678912:db:my-custom-instance",
        "BackupRetentionPeriod": 3,
        "DBName": "ORCL",
        "PreferredMaintenanceWindow": "fri:10:56-fri:11:26",
        "Endpoint": {
            "HostedZoneId": "ABCDEFGHIJKLMNO",
            "Port": 8200,
```

```
            "Address": "my-custom-instance.abcdefghijk.us-west-2.rds.amazonaws.com"
        },
        "DBInstanceStatus": "automation-paused",
        "IAMDatabaseAuthenticationEnabled": false,
        "AutomationMode": "all-paused",
        "EngineVersion": "19.my_cev1",
        "DeletionProtection": false,
        "AvailabilityZone": "us-west-2a",
        "DomainMemberships": [],
        "StorageType": "gp2",
        "DbiResourceId": "db-ABCDEFGHIJKLMNOPQRSTUVW",
        "ResumeFullAutomationModeTime": "2020-11-07T20:56:50.565Z",
        "KmsKeyId": "arn:aws:kms:us-west-2:012345678912:key/
aa111a11-111a-11a1-1a11-1111a11a1a1a",
        "StorageEncrypted": false,
        "AssociatedRoles": [],
        "DBInstanceClass": "db.m5.xlarge",
        "DbInstancePort": 0,
        "DBInstanceIdentifier": "my-custom-instance",
        "TagList": []
    }
```

Modification d'une instance de base de données RDS personnalisée pour le SQL serveur

La modification d'une instance de base de données RDS Custom for SQL Server est similaire à la modification d'AmazonRDS, mais les modifications que vous pouvez apporter sont limitées aux suivantes :

- Modification de la classe d'instance
- Modification de la période de rétention des sauvegardes et de la fenêtre de sauvegarde
- Modification de la fenêtre de maintenance
- Mise à niveau de la version du moteur de base de données lorsqu'une nouvelle version est disponible
- Modification du stockage alloué, du provisionnement et du IOPS type de stockage
- Autorisation et suppression de déploiements multi-AZ

Les limites suivantes s'appliquent à la modification d'une instance de base de données RDS personnalisée pour SQL serveur :

- Les options de base de données personnalisées et les groupes de paramètres ne sont pas pris en charge.
- Tous les volumes de stockage que vous attachez manuellement à votre instance de base de données RDS personnalisée se trouvent en dehors du périmètre de support.

  Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.

Console

Pour modifier une instance de base de données RDS personnalisée pour le SQL serveur

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de bases de données que vous souhaitez modifier.
4. Sélectionnez Modify.
5. Effectuez les modifications suivantes selon les besoins :

   a. Dans le champ Version du moteur de base de données, sélectionnez la nouvelle version.

   b. Modifiez la valeur de Classe d'instance de base de données. Pour connaître les classes prises en charge, consultez Support de classe d'instance de base de données pour RDS Custom for SQL Server.

   c. Modifiez la valeur de Période de rétention des sauvegardes.

   d. Dans le champ Fenêtre de sauvegarde, définissez les valeurs pour Heure de début et Durée.

   e. Dans le champ Fenêtre de maintenance d'instance de bases de données, définissez les valeurs pour Jour de début, Heure de début et Durée.
6. Choisissez Continuer.
7. Sélectionnez Appliquer immédiatement ou Appliquer au cours de la prochaine fenêtre de maintenance planifiée.
8. Choisissez Modifier l'instance de base de données.

AWS CLI

Pour modifier une instance de base de données RDS personnalisée pour le SQL serveur, utilisez la modify-db-instance AWS CLI commande. Définissez les paramètres suivants selon les besoins :

- `--db-instance-class` : pour connaître les classes prises en charge, consultez Support de classe d'instance de base de données pour RDS Custom for SQL Server.
- `--engine-version` – Numéro de version du moteur de base de données vers lequel vous effectuez la mise à niveau.
- `--backup-retention-period` – Durée de rétention des sauvegardes automatisées entre 0 à 35 jours.
- `--preferred-backup-window` – Intervalle de temps quotidien au cours duquel les sauvegardes automatisées sont créées.
- `--preferred-maintenance-window`— La plage horaire hebdomadaire (inUTC) pendant laquelle la maintenance du système peut avoir lieu.
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer immédiatement les modifications apportées au stockage.

  Vous pouvez également utiliser `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

Modification du stockage pour une instance de base de données RDS Custom for SQL Server

La modification du stockage pour une instance de base de données RDS Custom for SQL Server est similaire à la modification du stockage pour une instance de base de données Amazon RDS, mais vous ne pouvez effectuer que les opérations suivantes :

- Augmentez la taille du stockage alloué.
- Modifiez le type de stockage. Vous pouvez utiliser les types de stockage disponibles, comme le stockage Usage général ou IOPS provisionnés. Les IOPS provisionnées sont prises en charge pour les types de stockage gp3, io1 et io2 Block Express.
- Modifiez les IOPS provisionnées, si vous utilisez les types de volumes qui prennent en charge les IOPS provisionnées.

Les limites suivantes s'appliquent à la modification du stockage pour une instance de base de données RDS Custom for SQL Server :

- La taille de stockage minimale allouée pour RDS Custom for SQL Server est de 20 GiB. La limite de stockage maximale pour io1, gp2 et gp3 est de 16 TiB, tandis que io2 prend en charge 64 TiB.
- Comme pour Amazon RDS, vous ne pouvez pas réduire le stockage alloué. Il s'agit d'une limitation des volumes Amazon Elastic Block Store (Amazon EBS). Pour de plus amples informations, consultez Utilisation du stockage pour les RDS instances de base de données Amazon.
- La mise à l'échelle automatique du stockage n'est pas pris en charge pour les instances de base de données RDS Custom for SQL Server.
- Tous les volumes de stockage que vous attachez manuellement à votre instance de base de données RDS Custom ne sont pas pris en compte pour la mise à l'échelle du stockage. Seuls les volumes de données par défaut fournis par RDS, c'est-à-dire le lecteur D, sont pris en compte pour la mise à l'échelle du stockage.

  Pour de plus amples informations, veuillez consulter Périmètre de prise en charge RDS Custom.
- La mise à l'échelle du stockage ne provoque généralement aucune panne ou dégradation des performances de l'instance de base de données. Après la modification de la taille de stockage d'une instance de base de données, l'instance passe à l'état storage-optimization.
- L'optimisation du stockage peut prendre plusieurs heures. Vous ne pouvez pas apporter d'autres modifications au stockage avant six (6) heures ou avant la fin de l'optimisation du stockage sur l'instance, le délai le plus long prévalant. Pour de plus amples informations, consultez Utilisation du stockage pour les RDS instances de base de données Amazon.

Pour plus d'informations sur le stockage, consultez Stockage d'instance de base de données Amazon RDS.

Pour obtenir des informations générales sur la modification du stockage, consultez Utilisation du stockage pour les RDS instances de base de données Amazon.

> **Important**
>
> Ne modifiez pas le stockage de votre instance de base de données RDS Custom for SQL Server à l'aide de consoles Amazon EC2 ou Amazon EBS ou. APIs Modifiez directement le stockage en dehors de la console Amazon RDS ou APIs entraînez un `unsupported-configuration` état de votre base de données.

Lorsque vous apportez des modifications de stockage directes à l'aide d'Amazon EC2 ou d'Amazon EBS, Amazon RDS ne peut ni suivre ni gérer l'état de votre instance de base de données. Cela peut provoquer :

- Les mécanismes de basculement à haute disponibilité ne fonctionnent pas correctement
- Configurations de réplication de base de données à interrompre
- Fonctionnalités de redondance vouées à échouer

Modifiez le stockage uniquement via la console Amazon RDS ou APIs pour maintenir votre base de données dans un état compatible. Consultez Correction des configurations non prises en charge dans RDS Custom for SQL Server les étapes de restauration.

console

Pour modifier le stockage pour une instance de base de données RDS Custom for SQL Server

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de bases de données que vous souhaitez modifier.
4. Sélectionnez Modify.
5. Effectuez les modifications suivantes selon les besoins :
    a. Saisissez une nouvelle valeur pour Stockage alloué. Elle doit être supérieure à la valeur actuelle et être comprise entre 20 Gio et 16 Tio.
    b. Modifiez la valeur de Type de stockage. Vous pouvez choisir entre les types de stockage à usage général ou IOPS provisionné disponibles. Les IOPS provisionnées sont prises en charge pour les types de stockage gp3, io1 et io2 Block Express.
    c. Si vous spécifiez un type de stockage qui prend en charge les IOPS provisionnées, vous pouvez définir la valeur des IOPS provisionnées.
6. Choisissez Continuer.
7. Sélectionnez Appliquer immédiatement ou Appliquer au cours de la prochaine fenêtre de maintenance planifiée.
8. Choisissez Modifier l'instance de base de données.

AWS CLI

Pour modifier le stockage d'une instance de base de données RDS Custom pour SQL Server, utilisez la modify-db-instance AWS CLI commande. Définissez les paramètres suivants selon les besoins :

- `--allocated-storage` – Volume de stockage à allouer à l'instance de base de données, exprimé en gibioctets. Elle doit être supérieure à la valeur actuelle et être comprise entre 20 et 16 384 Gio.
- `--storage-type`— Le type de stockage, par exemple, gp2, gp3, io1 ou io2.
- `--iops` : IOPS provisionnés pour l'instance de base de données. Vous ne pouvez le spécifier que pour les types de stockage qui prennent en charge les IOPS provisionnées (gp3, io1 et io2).
- `--apply-immediately` : utilisez `--apply-immediately` pour appliquer immédiatement les modifications apportées au stockage.

  Vous pouvez également utiliser `--no-apply-immediately` (valeur par défaut) pour appliquer les modifications au cours de la prochaine fenêtre de maintenance.

L'exemple suivant modifie la taille de stockage my-custom-instance à 200 GiB, le type de stockage à io1 et les IOPS provisionnées à 3000.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --storage-type io1 \
    --iops 3000 \
    --allocated-storage 200 \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --storage-type io1 ^
    --iops 3000 ^
    --allocated-storage 200 ^
    --apply-immediately
```

Balisage de ressources RDS Custom for SQL Server

Vous pouvez étiqueter les ressources RDS Custom comme s'il s'agissait de ressources Amazon RDS. Il existe toutefois quelques différences importantes :

- Ne créez pas et ne modifiez pas l'étiquette `AWSRDSCustom` requise pour l'automatisation de RDS Custom. En cas de non-respect de cette consigne, l'automatisation risque d'être interrompue.
- La `Name` balise est ajoutée aux ressources personnalisées RDS avec le préfixe`do-not-delete-rds-custom`. Toute valeur de clé transmise par le client est remplacée.
- Les étiquettes ajoutées aux instances de base de données RDS Custom lors de la création sont propagées à toutes les autres ressources RDS Custom associées.
- Les étiquettes ne sont pas propagées lorsque vous les ajoutez à des ressources RDS Custom après la création d'une instance de base de données.

Pour obtenir des informations générales sur le balisage des ressources, consultez Marquage d' Amazon RDS.

Suppression d'une instance de base de données RDS personnalisée pour le SQL serveur

Pour supprimer une instance de base de données RDS personnalisée pour le SQL serveur, procédez comme suit :

- Indiquez le nom de l'instance de base de données.
- Choisissez ou désactivez l'option permettant de prendre un instantané de base de données final de l'instance de base de données.
- Activez ou désactivez l'option de rétention des sauvegardes automatisées.

Vous pouvez supprimer une instance de base de données RDS personnalisée pour SQL serveur à l'aide de la console ou duCLI. Le temps nécessaire à la suppression d'une instance de base de données peut varier en fonction de la période de conservation de la sauvegarde (c'est-à-dire du nombre de sauvegardes à supprimer), de la quantité de données supprimées et de la réalisation d'un instantané final.

> **Warning**
>
> La suppression d'une instance de base de données RDS Custom for SQL Server supprimera définitivement l'EC2instance et les EBS volumes Amazon associés. Vous ne devez à aucun moment mettre fin à ces ressources ou les supprimer, sinon la suppression et la création de l'instantané final risquent d'échouer.

> **Note**
>
> Il est impossible de créer un instantané de base de données final de votre instance de base de données si elle se trouve dans un des états suivants : `creating`, `failed`, `incompatible-create`, `incompatible-restore` ou `incompatible-network`. Pour de plus amples informations, veuillez consulter Affichage de l'état de l'instance de base de données dans un cluster Aurora.

> **Important**
>
> Lorsque vous choisissez de prendre un instantané final, nous vous recommandons d'éviter d'écrire des données dans votre instance de base de données pendant que la suppression de l'instance de base de données est en cours. Une fois la suppression de l'instance de base de données initiée, il n'est pas garanti que les modifications de données soient capturées par l'instantané final.

Console

Pour supprimer une instance de base de données RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données RDS personnalisée pour le SQL serveur que vous souhaitez supprimer. RDSLes instances de base de données personnalisées pour le SQL serveur affichent le rôle Instance (RDSpersonnalisé pour le SQL serveur).
3. Pour Actions, choisissez Supprimer.

4. Pour prendre un instantané final, choisissez Create final snapshot (Créer un instantané final) et donnez un nom pour Final snapshot name (Nom de l'instantané final).
5. Pour conserver les sauvegardes automatisées, choisissez Conserver les sauvegardes automatiques.
6. Saisissez **`delete me`** dans la zone.
7. Sélectionnez Delete (Supprimer).

AWS CLI

Vous supprimez une instance de base de données RDS Custom for SQL Server à l'aide de la delete-db-instance AWS CLI commande. Identifiez l'instance de base de données à l'aide du paramètre requis `--db-instance-identifier`. Les paramètres restants sont les mêmes que pour une RDS instance de base de données Amazon.

L'exemple suivant supprime l'instance de base de données RDS Custom for SQL Server nomméemy`-custom-instance`, prend un instantané final et conserve les sauvegardes automatisées.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance \
    --db-instance-identifier my-custom-instance \
    --no-skip-final-snapshot \
    --final-db-snapshot-identifier my-custom-instance-final-snapshot \
    --no-delete-automated-backups
```

Dans Windows:

```
aws rds delete-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --no-skip-final-snapshot ^
    --final-db-snapshot-identifier my-custom-instance-final-snapshot ^
    --no-delete-automated-backups
```

Pour prendre un instantané final, l'option `--final-db-snapshot-identifier` est obligatoire et doit être spécifiée.

Pour ignorer l'instantané final, spécifiez l'option `--skip-final-snapshot` au lieu des options `--no-skip-final-snapshot` et `--final-db-snapshot-identifier` dans la commande.

Pour supprimer des sauvegardes automatiques, spécifiez l'option `--delete-automated-backups` plutôt que l'option `--no-delete-automated-backups` dans la commande.

Démarrage et arrêt d'une instance de base de données RDS personnalisée pour SQL serveur

Vous pouvez démarrer et arrêter votre instance de base de données RDS Custom for SQL Server. Les mêmes exigences et limites générales applicables aux instances de base RDS de données de SQL serveur s'appliquent à l'arrêt et au démarrage de vos instances de base de données RDS personnalisées pour SQL serveur. Pour de plus amples informations, veuillez consulter Arrêt temporaire d'une instance de bases de données Amazon RDS.

Les considérations suivantes s'appliquent également au démarrage et à l'arrêt de votre instance de base de données RDS Custom for SQL Server :

- La modification d'un attribut d'EC2instance d'une instance de base de données RDS personnalisée pour SQL serveur alors que l'instance de base de données est activée `STOPPED` n'est pas prise en charge.
- Vous pouvez arrêter et démarrer une instance de base de données RDS personnalisée pour SQL serveur uniquement si elle est configurée pour une seule zone de disponibilité. Vous ne pouvez pas arrêter une instance de base de données RDS Custom for SQL Server dans une configuration multi-AZ.
- Un `SYSTEM` instantané est créé lorsque vous arrêtez une instance de base de données RDS Custom for SQL Server. Le snapshot sera automatiquement supprimé lorsque vous redémarrerez l'instance de base de données RDS Custom for SQL Server.
- Si vous supprimez votre EC2 instance alors que votre instance de base de données RDS personnalisée pour SQL serveur est arrêtée, le `C:` lecteur sera remplacé lorsque vous redémarrerez l'instance de base de données RDS personnalisée pour SQL serveur.
- Le `C:\` lecteur, le nom d'hôte et vos configurations personnalisées sont conservés lorsque vous arrêtez une instance de base de données RDS Custom for SQL Server, tant que vous ne modifiez pas le type d'instance.
- Les actions suivantes permettront à RDS Custom de placer l'instance de base de données en dehors du périmètre de support, et les heures d'instance de base de données vous seront toujours facturées :

- Démarrage de l'EC2instance sous-jacente alors qu'Amazon RDS est arrêté. Pour résoudre le problème, vous pouvez appeler `start-db-instance` Amazon RDS API ou arrêter le EC2 afin que l'instance RDS personnalisée revienne`STOPPED`.
- Arrêt de l'EC2instance sous-jacente lorsque l'instance de base de données RDS Custom for SQL Server est personnalisée`ACTIVE`.

Pour plus de détails sur l'arrêt et le démarrage des instances de base de données, consultez Arrêt temporaire d'une instance de bases de données Amazon RDS et Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée.

Utilisation de Microsoft Active Directory avec RDS Custom for SQL Server

RDSCustom for SQL Server permet de joindre vos instances à un Active Directory (AD) autogéré ou AWS Managed Microsoft AD. Cela s'applique quel que soit l'endroit où votre AD est hébergé, comme un centre de données sur site, Amazon EC2 ou tout autre fournisseur de services cloud.

Pour l'authentification des utilisateurs et des services, vous pouvez utiliser l'authentification Kerberos sur votre instance de base de données RDS Custom for SQL Server sans utiliser de domaines intermédiaires NTLM ni d'approbations forestières. Lorsqu'un utilisateur essaie de s'authentifier sur votre instance de base de données RDS personnalisée pour SQL serveur à l'aide d'un Active Directory auto-joint, les demandes d'authentification sont transmises à un AD autogéré ou AWS Managed Microsoft AD que vous spécifiez.

Dans les sections suivantes, vous trouverez des informations sur l'utilisation d'Active Directory autogéré et d'Active Directory AWS géré pour RDS Custom for SQL Server.

Rubriques

- Disponibilité des régions et des versions
- Configurer AD autogéré ou sur site
- Configurez Microsoft Active Directory à l'aide de AWS Directory Service
- Règles relatives aux ports de configuration réseau
- Validation du réseau
- Configuration de l'authentification Windows pour RDS Custom pour les instances de SQL Server
- Gestion d'une instance de base de données dans un domaine
- Présentation de l'appartenance au domaine
- Dépannage d’Active Directory

Disponibilité des régions et des versions

RDSCustom for SQL Server prend en charge à la fois l'AD autogéré et AWS Managed Microsoft AD l'utilisation NTLM de Kerberos dans toutes les régions où RDS Custom for SQL Server est pris en charge. Pour de plus amples informations, veuillez consulter Régions et moteurs de base de données pris en charge pour RDS Custom.

Configurer AD autogéré ou sur site

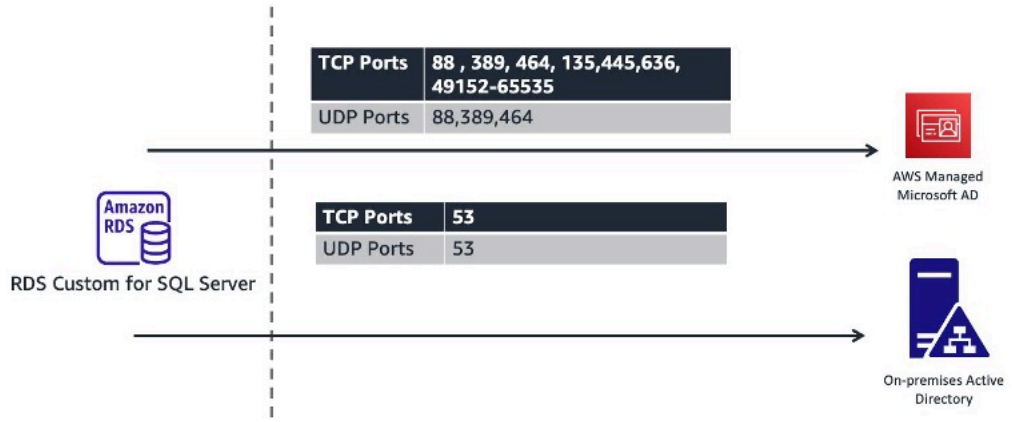
Pour associer votre Microsoft AD sur site ou autogéré à votre instance de base de données RDS personnalisée pour SQL serveur, votre domaine actif doit être configuré comme suit :

- Définissez les sous-réseaux VPC associés à votre instance de base de données RDS personnalisée pour SQL serveur dans votre AD autogéré ou sur site. Vérifiez qu'il n'y a aucun conflit entre les sous-réseaux de vos VPC sites AD et ceux de vos sites AD.
- Votre contrôleur de domaine AD a un niveau fonctionnel de domaine correspondant à Windows Server 2008 R2 ou supérieur.
- Votre nom de domaine AD ne peut pas être au format Single Label Domain (SLD). RDS Custom for SQL Server ne prend pas en charge SLD les domaines.
- Le nom de domaine complet (FQDN) de votre annonce ne peut pas dépasser 47 caractères.

Configuration de votre connectivité réseau

Configurez votre connectivité réseau AD autogérée ou sur site de la manière suivante :

- Configurez la connectivité entre Amazon sur VPC lequel votre instance RDS Custom for SQL Server est exécutée et votre AD. Utilisation AWS Direct Connect, AWS VPN AWS Transit Gateway, et VPC peering.
- Autorisez le trafic sur les ports de votre réseau et de vos groupes de sécurité RDS personnalisés pour le SQL serveur ACLs vers votre AD autogéré ou sur site. Pour de plus amples informations, veuillez consulter Règles relatives aux ports de configuration réseau.



Configuration de DNS la résolution

Définissez les exigences suivantes pour configurer la DNS résolution avec des AD autogérés ou sur site :

- Configurez DNS la résolution au sein de votre compte VPC pour résoudre le nom de domaine complet de votre Active Directory auto-hébergé (FQDN). Un exemple de FQDN `sicorp.example.local`. Pour configurer la DNS résolution, configurez le VPC DNS résolveur pour qu'il transmette les requêtes pour certains domaines avec un point de terminaison sortant et une règle de résolution Amazon Route 53. Pour plus d'informations, consultez Configurer un point de terminaison sortant du résolveur Route 53 pour résoudre DNS les enregistrements.
- Pour les charges de travail qui exploitent à la fois les ressources locales VPCs et les ressources locales, vous devez résoudre les DNS enregistrements hébergés sur site. Les ressources sur site peuvent avoir besoin de résoudre les noms hébergés sur AWS.

  Pour créer une configuration de cloud hybride, utilisez des points de terminaison de résolution et des règles de transfert conditionnel pour résoudre les DNS requêtes entre vos ressources sur site et personnalisées. VPC Pour plus d'informations, consultez la section Résolution DNS des requêtes entre VPCs et votre réseau dans le manuel Amazon Route 53 Developer Guide.

> **Important**
>
> La modification des paramètres du DNS résolveur de l'interface réseau sur le RDS Custom for SQL Server empêche les VPC points de terminaison DNS compatibles de fonctionner correctement. DNSdes VPC points de terminaison compatibles sont requis pour les instances situées dans des sous-réseaux privés sans accès à Internet.

Configurez Microsoft Active Directory à l'aide de AWS Directory Service

AWS Managed Microsoft AD crée un Microsoft Active Directory entièrement géré alimenté par Windows Server 2019 et fonctionnant aux niveaux fonctionnels de la forêt et du domaine R2 2012. AWS AWS Directory Service crée les contrôleurs de domaine dans différents sous-réseaux d'un AmazonVPC, ce qui rend votre répertoire hautement disponible même en cas de panne.

Pour créer un répertoire avec AWS Managed Microsoft AD, voir Getting started with AWS Managed Microsoft AD dans le Guide AWS Directory Service d'administration.

Configuration de votre connectivité réseau

Activer le VPC trafic croisé entre l'annuaire et l'instance de base de données

Pour localiser le répertoire et l'instance de base de données dans le même répertoireVPC, sautez cette étape et passez à l'étape suivanteRègles relatives aux ports de configuration réseau.

Pour localiser le répertoire et l'instance de base de données différemmentVPCs, configurez le VPC trafic croisé à l'aide du VPC peering ou AWS Transit Gateway. Pour plus d'informations sur l'utilisation du VPC peering, voir Qu'est-ce que le VPC peering ? dans le Amazon VPC Peering Guide et qu'est-ce que c'est ? AWS Transit Gateway dans les Amazon VPC Transit Gateways.

Activez le VPC trafic croisé à l'aide du VPC peering

1. Définissez des règles de VPC routage appropriées pour garantir que le trafic réseau peut circuler dans les deux sens.
2. Autorisez le groupe de sécurité de l'instance de base de données à recevoir le trafic entrant en provenance du groupe de sécurité de l'annuaire. Pour de plus amples informations, veuillez consulter Règles relatives aux ports de configuration réseau.
3. La liste de contrôle d'accès réseau (ACL) ne doit pas bloquer le trafic.

Si le répertoire Compte AWS appartient à une autre personne, vous devez le partager. Pour partager le répertoire Compte AWS dans lequel se trouve l'instance RDS Custom for SQL Server, suivez le didacticiel : Partage de votre nom de EC2 domaine AWS Managed Microsoft AD pour une connexion fluide dans le Guide d'AWS Directory Service administration.

Partage d'un répertoire entre Comptes AWS

1. Connectez-vous à la AWS Directory Service console à l'aide du compte de l'instance de base de données et vérifiez si le domaine possède le même `SHARED` statut avant de continuer.
2. Après vous être connecté à la AWS Directory Service console à l'aide du compte de l'instance de base de données, notez la valeur de l'ID du répertoire. Vous utilisez cet ID pour joindre l'instance de base de données au domaine.

Configurer DNS la résolution

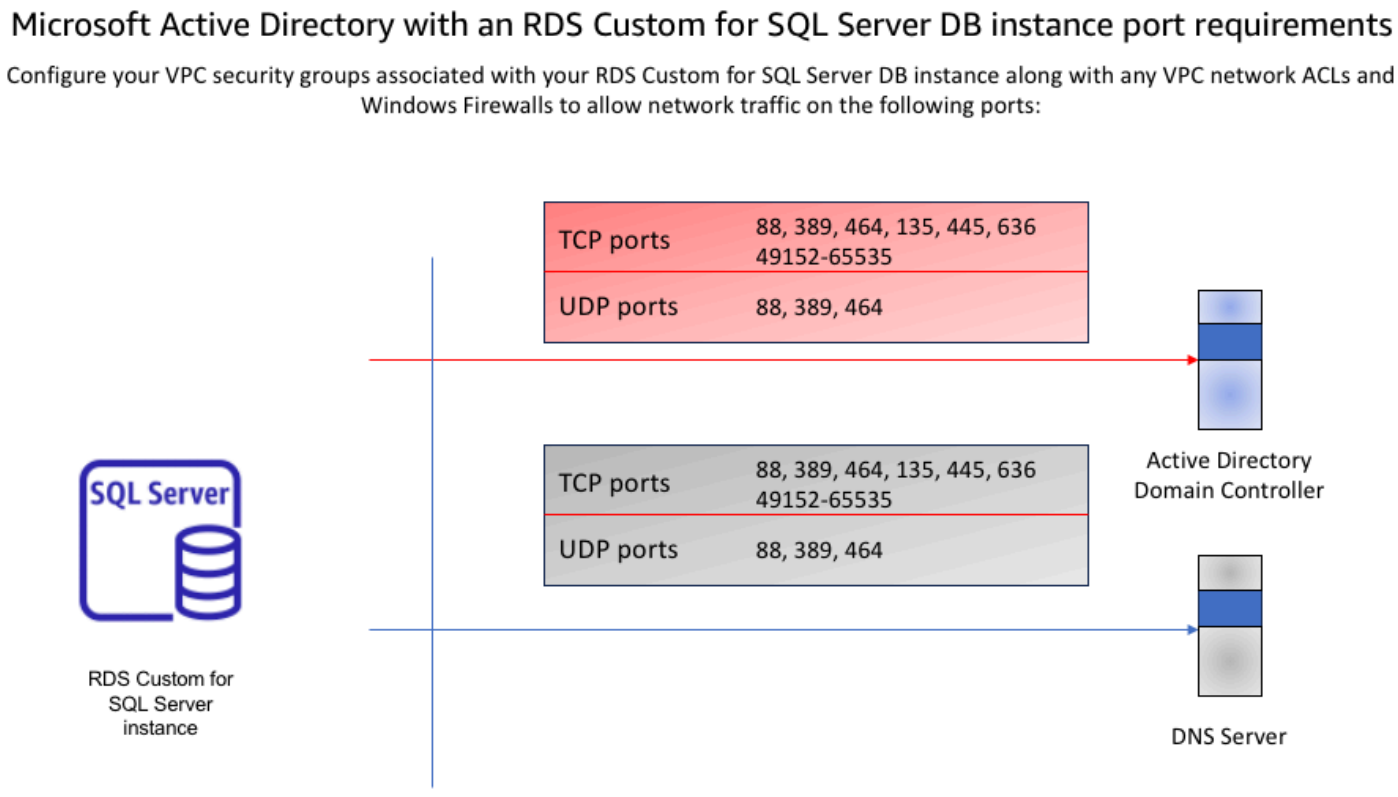
Lorsque vous créez un répertoire avec AWS Managed Microsoft AD, vous AWS Directory Service créez deux contrôleurs de domaine et ajoutez le DNS service en votre nom.

Si vous en avez une existante AWS Managed Microsoft AD ou si vous prévoyez d'en lancer une dans une instance de base de données VPC autre que votre instance de base de données RDS Custom for SQL Server, configurez le VPC DNS résolveur pour transférer les requêtes pour certains domaines avec une règle de résolution et de sortie Route 53, voir Configurer un point de terminaison sortant du résolveur Route 53 pour résoudre les enregistrements. DNS

Règles relatives aux ports de configuration réseau

Assurez-vous de respecter les configurations réseau suivantes :

- Connectivité configurée entre l'Amazon sur VPC lequel vous souhaitez créer l'instance de base de données RDS personnalisée pour le SQL serveur et votre Active Directory autogéré ou AWS Managed Microsoft AD. Pour un Active Directory autogéré, configurez la connectivité à l'aide de AWS Direct Connect AWS VPN, de VPC peering ou de AWS Transit Gateway. Pour configurer AWS Managed Microsoft AD la connectivité à l'aide du VPC peering.
- Assurez-vous que le groupe de sécurité et le VPC réseau du ou ACLs des sous-réseaux sur lesquels vous créez votre instance de base de données RDS Custom for SQL Server autorisent le trafic sur les ports et dans les directions indiquées dans le schéma suivant.



Le tableau suivant identifie le rôle de chaque port.

Protocole	Ports	Rôle
TCP/UDP	53	Système de noms de domaine (DNS)
TCP/UDP	88	Authentification Kerberos
TCP/UDP	464	Changement/définition de mot de passe
TCP/UDP	389	Protocole léger d'accès aux annuaires (LDAP)
TCP	135	Environnement informatique distribué/Endpoint Mapper (DCE/EPMAP)
TCP	445	Partage de SMB fichiers avec les services d'annuaire
TCP	636	Protocole léger d'accès aux répertoires viaTLS/SS L(LDAPS)
TCP	49152 - 65535	Ports éphémères pour RPC

- En général, les DNS serveurs de domaine sont situés dans les contrôleurs de domaine AD. Il n'est pas nécessaire de configurer le jeu d'VPCDHCPoptions pour utiliser cette fonctionnalité. Pour plus d'informations, consultez les ensembles d'DHCPoptions dans le guide de VPC l'utilisateur Amazon.

> **Important**
>
> Si vous utilisez le VPC réseauACLs, vous devez également autoriser le trafic sortant sur les ports dynamiques (49152-65535) depuis votre RDS instance de base de données Custom for Server. SQL Assurez-vous que ces règles de trafic sont également reflétées sur les pare-feux qui s'appliquent à chacun des contrôleurs de domaine AD, aux DNS serveurs et aux instances de base de données RDS Custom for SQL Server.

Alors que les groupes de VPC sécurité exigent que les ports soient ouverts uniquement dans le sens où le trafic réseau est initié, la plupart des pare-feux et VPC réseaux Windows ACLs exigent que les ports soient ouverts dans les deux sens.

Validation du réseau

Avant de rejoindre votre instance RDS Custom pour l'autogérer ou AWS Managed Microsoft AD pour vérifier les points suivants à partir d'une EC2 instance située dans le même VPC que celui où vous prévoyez de lancer l'instance RDS Custom pour SQL Server.

- Vérifiez si vous êtes en mesure de résoudre le nom de domaine complet (FQDN) en contrôleur IPs de domaine.

  ```
  nslookup corp.example.com
  ```

  La commande doit renvoyer un résultat similaire :

  ```
  Server:  ip-10-0-0-2.us-west-2.compute.internal
  Address:  25.0.0.2

  Non-authoritative answer:
  Name:    corp.example.com
  Addresses:  40.0.9.25 (DC1 IP)
              40.0.50.123 (DC2 IP)
  ```

- AWS Résolvez les services d'une EC2 instance du VPC sur laquelle vous lancez votre instance personnalisée RDS :

  ```
  $region='input-your-aws-region'
  $domainFQDN='input-your-domainFQDN'

  function Test-DomainPorts {
      param (
          [string]$Domain,
          [array]$Ports
      )

      foreach ($portInfo in $Ports) {
          try {
              $conn = New-Object System.Net.Sockets.TcpClient
  ```

```
            $connectionResult = $conn.BeginConnect($Domain, $portInfo.Port, $null,
 $null)
            $success = $connectionResult.AsyncWaitHandle.WaitOne(1000) # 1 second
 timeout
            if ($success) {
                $conn.EndConnect($connectionResult)
                $result = $true
            } else {
                $result = $false
            }
        }
        catch {
            $result = $false
        }
        finally {
            if ($null -ne $conn) {
                $conn.Close()
            }
        }
        Write-Host "$($portInfo.Description) port open: $result"
    }
}

# Check if ports can be reached
$ports = @(
    @{Port = 53;   Description = "DNS"},
    @{Port = 88;   Description = "Kerberos"},
    @{Port = 389;  Description = "LDAP"},
    @{Port = 445;  Description = "SMB"},
    @{Port = 5985; Description = "WinRM"},
    @{Port = 636;  Description = "LDAPS"},
    @{Port = 3268; Description = "Global Catalog"},
    @{Port = 3269; Description = "Global Catalog over SSL"},
    @{Port = 9389; Description = "AD DS"}
)

function Test-DomainReachability {
    param (
        [string]$DomainName
    )

    try {
        $dnsResults = Resolve-DnsName -Name $DomainName -ErrorAction Stop
```

```
        Write-Host "Domain $DomainName is successfully resolving to following IP
 addresses: $($dnsResults.IpAddress)"
        Write-Host ""
        return $true
    }
    catch {
        Write-Host ""
        Write-Host "Error Message: $($_.Exception.Message)"
        Write-Host "Domain $DomainName reachability check failed, please Configure
 DNS resolution"
        return $false
    }
}

$domain = (Get-WmiObject Win32_ComputerSystem).Domain
if ($domain -eq 'WORKGROUP') {
    Write-Host ""
    Write-Host "Host $env:computername is still part of WORKGROUP and not part of any
 domain"
    }
else {
    Write-Host ""
    Write-Host "Host $env:computername is joined to $domain domain"
    Write-Host ""
    }


$isReachable = Test-DomainReachability -DomainName $domainFQDN
if ($isReachable) {
    write-Host "Checking if domain $domainFQDN is reachable on required ports  "
    Test-DomainPorts -Domain $domainFQDN -Ports $ports
}
else {
    Write-Host "Port check skipped. Domain not reachable"
}



# Get network adapter configuration
$networkConfig = Get-WmiObject Win32_NetworkAdapterConfiguration |
                 Where-Object { $_.IPEnabled -eq $true } |
                 Select-Object -First 1

# Check DNS server settings
```

```
$dnsServers = $networkConfig.DNSServerSearchOrder

if ($dnsServers) {
    Write-Host "`nDNS Server settings:"
    foreach ($server in $dnsServers) {
        Write-Host "  - $server"
    }
} else {
    Write-Host "`nNo DNS servers configured or unable to retrieve DNS server
 information."
}

write-host ""

# Checks reachability to dependent services
$services = "s3", "ec2", "secretsmanager", "logs", "events", "monitoring", "ssm",
 "ec2messages", "ssmmessages"

function Get-TcpConnectionAsync {
    param (
        $ServicePrefix,
        $region
    )
    $endpoint = "${ServicePrefix}.${region}.amazonaws.com"
    $tcp = New-Object Net.Sockets.TcpClient
    $result = $false

    try {
        $connectTask = $tcp.ConnectAsync($endpoint, 443)
        $timedOut = $connectTask.Wait(3000)
        $result = $tcp.Connected
    }
    catch {
        $result = $false
    }
    return $result
}

foreach ($service in $services) {
    $validationResult = Get-TcpConnectionAsync -ServicePrefix $service -Region
 $region
    Write-Host "Reachability to $service is $validationResult"
}
```

La `TcpTestSucceeded` valeur doit être renvoyée `True` pour `s3ec2`,`secretsmanager`,`logs`,`events`,`monitoring`,`ssm`,`ec2messages`, et`ssmmessages`.

Configuration de l'authentification Windows pour RDS Custom pour les instances de SQL Server

Nous recommandons de créer une unité d'organisation dédiée et des informations d'identification de service étendues à cette unité d'organisation pour tout Compte AWS propriétaire d'une instance de base de données RDS Custom for SQL Server jointe à votre domaine AD. En consacrant une UO et des informations d'identification de service, vous évitez les conflits d'autorisations et respectez le principe du moindre privilège.

Les politiques de groupe au niveau d'Active Directory peuvent entrer en conflit avec AWS les automatisations et les autorisations. Nous vous recommandons de sélectionner des GPO qui s'appliquent uniquement à l'unité d'organisation que vous créez pour RDS Custom for SQL Server.

- Pour créer un utilisateur de domaine UO et AD dans votre AD autogéré ou sur site, vous pouvez connecter le contrôleur de domaine en tant qu'administrateur de domaine.
- Pour créer des utilisateurs et des groupes dans un AWS Directory Service annuaire, vous devez être connecté à une instance de gestion et vous devez également être connecté en tant qu'utilisateur autorisé à créer des utilisateurs et des groupes. Pour plus d'informations, consultez la section Gestion des utilisateurs et des groupes AWS Managed Microsoft AD dans le Guide d'AWS Directory Service administration.
- Pour gérer votre instance Active Directory depuis une instance Amazon EC2 Windows Server, vous devez installer les services de domaine Active Directory et les outils des services Active Directory Lightweight Directory sur l' EC2 instance. Pour plus d'informations, consultez la section Installation des outils d'administration Active Directory AWS Managed Microsoft AD dans le Guide d'AWS Directory Service administration.
- Nous vous recommandons d'installer ces outils sur une EC2 instance distincte à des fins d'administration, et non sur votre instance de base de données RDS Custom for SQL Server pour en faciliter l'administration.

Les exigences relatives à un compte de service de domaine AD sont les suivantes :

- Vous devez disposer d'un compte de service dans votre domaine AD avec des autorisations déléguées pour associer des ordinateurs au domaine. Un compte de service de domaine est un

compte utilisateur de votre AD qui dispose d'une autorisation déléguée pour effectuer certaines tâches.

- Déléguez les autorisations suivantes à votre compte de service de domaine dans l'unité organisationnelle à laquelle vous rejoignez votre instance RDS Custom for SQL Server :
  - Capacité validée d'écrire sur le nom d'hôte DNS
  - Capacité validée d'écrire dans le nom du principal de service
  - Création et suppression d'objets informatiques
- Pour AD autogéré et sur site, le compte de service de domaine doit être membre du groupe « Administrateurs AWS délégués du système de noms de domaine ».
- En AWS Managed Microsoft AD effet, le compte de service de domaine doit être membre du groupe DnsAdmins « ».

Il s'agit de l'ensemble minimal d'autorisations requis pour joindre des objets informatiques à votre AD autogéré et AWS Managed Microsoft AD. Pour plus d'informations, voir Erreur : l'accès est refusé lorsque des utilisateurs non administrateurs auxquels le contrôle a été délégué tentent de joindre des ordinateurs à un contrôleur de domaine dans la documentation Microsoft Windows Server.

> **Important**
>
> Ne déplacez pas les objets informatiques créés par RDS Custom for SQL Server dans l'unité organisationnelle (UO) après la création de votre instance de base de données. Le déplacement des objets associés peut entraîner une mauvaise configuration de votre instance de base de données RDS Custom for SQL Server. Si vous devez déplacer les objets informatiques créés par Amazon RDS, utilisez l'DBInstanceaction Modifier pour modifier les paramètres du domaine avec l'emplacement souhaité pour les objets informatiques.

Rubriques

- Étape 1 : créer une unité organisationnelle (UO) dans votre AD
- Étape 2 : créer un utilisateur de domaine AD
- Étape 3 : déléguer le contrôle à l'utilisateur AD en mode autogéré ou AWS Managed Microsoft AD
- Étape 4 : Création d'un secret
- Étape 5 : créer ou modifier une instance de base de données personnalisée RDS pour SQL Server
- Étape 6 : Création d'une connexion à SQL Server avec authentification Windows

- Étape 7 : Utilisation de l'authentification Kerberos ou NTLM

Étape 1 : créer une unité organisationnelle (UO) dans votre AD

Pour créer une unité organisationnelle dans votre AD, procédez comme suit :

Créez une UO dans votre AD

1. Connectez-vous à votre domaine AD en tant qu'administrateur de domaine.
2. Ouvrez Utilisateurs et ordinateurs Active Directory et sélectionnez le domaine où vous souhaitez créer votre unité d'organisation.
3. Cliquez avec le bouton droit sur le domaine et choisissez Nouveau, puis Unité organisationnelle.
4. Saisissez un nom pour l'unité d'organisation.

   Activez Protéger le conteneur contre toute suppression accidentelle.
5. Choisissez OK. Votre nouvelle unité d'organisation apparaît sous votre domaine.

En effet AWS Managed Microsoft AD, le nom de cette unité d'organisation est basé sur le nom NetBIOS que vous avez saisi lors de la création de votre répertoire. Cette unité d'organisation est détenue par AWS et contient tous vos AWS objets de répertoire associés, sur lesquels vous avez un contrôle total. Par défaut, deux enfants OUs existent sous cette UO, à savoir Computers et Users. La OUs nouveauté créée par RDS Custom est un enfant de l'unité d'organisation basée sur NetBIOS.

Étape 2 : créer un utilisateur de domaine AD

Les informations d'identification de l'utilisateur du domaine sont utilisées pour le secret dans Secrets Manager.

Créez un utilisateur de domaine AD dans votre AD

1. Ouvrez Utilisateurs et ordinateurs Active Directory et sélectionnez le domaine et l'unité d'organisation dans lesquels vous souhaitez créer l'utilisateur.
2. Cliquez avec le bouton droit sur l'objet Utilisateurs et choisissez Nouveau, puis Utilisateur.
3. Entrez le prénom, le nom de famille et le nom de connexion de l'utilisateur. Cliquez sur Next (Suivant).
4. Saisissez un mot de passe pour l'utilisateur. Ne sélectionnez pas L'utilisateur doit changer de mot de passe à la prochaine connexion ou le compte est désactivé. . Cliquez sur Next (Suivant).

5. Cliquez sur OK. Votre nouvel utilisateur apparaît sous votre domaine.

Étape 3 : déléguer le contrôle à l'utilisateur AD en mode autogéré ou AWS Managed Microsoft AD

Pour déléguer le contrôle à l'utilisateur du domaine AD dans votre domaine

1. Ouvrez le composant logiciel enfichable MMC Utilisateurs et ordinateurs Active Directory et sélectionnez votre domaine.
2. Cliquez avec le bouton droit sur l'unité d'organisation que vous avez créée précédemment et choisissez Déléguer le contrôle.
3. Dans l'assistant de contrôle des délégations, cliquez sur Suivant.
4. Dans la section Utilisateurs ou groupes, cliquez sur Ajouter.
5. Dans Sélectionner les utilisateurs, les ordinateurs ou les groupes, entrez l'utilisateur AD que vous avez créé et cliquez sur Vérifier les noms. Si votre vérification de l'utilisateur AD aboutit, cliquez sur OK.
6. Dans la section Utilisateurs ou groupes, confirmez que votre utilisateur AD a été ajouté et cliquez sur Suivant.
7. Dans la section Tâches à déléguer, choisissez Créer une tâche personnalisée à déléguer, puis cliquez sur Suivant.
8. Dans la section Type d'objet Active Directory :

   Choisissez ONly les objets suivants dans le dossier.

   Sélectionnez des objets informatiques

   Sélectionnez Créer les objets sélectionnés dans ce dossier

   Sélectionnez Supprimer les objets sélectionnés dans ce dossier et cliquez sur Suivant.
9. Dans la section Autorisations :

   Gardez l'option Général sélectionnée.

   Sélectionnez Écriture validée sur le nom d'hôte DNS.

   Sélectionnez Écriture validée sur le nom du principal de service et cliquez sur Suivant.
10. Après avoir terminé l'assistant de délégation de contrôle, confirmez vos paramètres et cliquez sur Terminer.

Étape 4 : Création d'un secret

Créez le secret dans la même Compte AWS région qui contient l'instance de base de données personnalisée RDS pour SQL Server que vous souhaitez inclure dans votre Active Directory. Stockez les informations d'identification de l'utilisateur du domaine AD créé dans[Étape 2 : créer un utilisateur de domaine AD](#).

Console

- Dans AWS Secrets Manager, choisissez Enregistrer un nouveau secret.
- Pour Secret type (Type de secret), choisissez Other type of secret (Autre type de secret).
- Pour les paires clé/valeur, ajoutez deux clés :
  - La première clé, `SELF_MANAGED_ACTIVE_DIRECTORY_USERNAME` et entrez le nom de votre utilisateur AD (sans le préfixe de domaine) pour la valeur.
  - Pour la deuxième clé, entrez `SELF_MANAGED_ACTIVE_DIRECTORY_PASSWORD` le mot de passe de votre utilisateur AD sur votre domaine.
- Pour la clé de chiffrement, entrez la même AWS KMS clé que celle que vous avez utilisée pour créer l'instance RDS Custom for SQL Server.
- Pour Nom du secret, choisissez le nom du secret commençant par `do-not-delete-rds-custom-` pour permettre à votre profil d'instance d'accéder à ce secret. SI vous souhaitez choisir un autre nom pour le secret, mettez-le à jour `RDSCustomInstanceProfile` pour accéder au nom du secret.
- (Facultatif) Pour Description, saisissez une description du nom du secret.
- Ajoutez les tags `Key="AWSRDSCustom",Value="custom-sqlserver"`
- Cliquez sur Enregistrer, puis sur Suivant.
- Pour Configurer les paramètres de rotation, conservez les valeurs par défaut et choisissez Suivant.
- Passez en revue les paramètres du secret et cliquez sur Stocker.
- Choisissez le nouveau secret et copiez la valeur de l'ARN secret. Nous l'utiliserons à l'étape suivante pour configurer Active Directory.

CLI

Exécutez la commande suivante dans votre CLI pour créer un secret :

```
# Linux based
```

```
aws secretsmanager create-secret \
--name do-not-delete-rds-custom-DomainUserCredentails \
--description "Active directory user credentials for managing RDS Custom" \
--secret-string "{\"SELF_MANAGED_ACTIVE_DIRECTORY_USERNAME\":\"tester\",
\"SELF_MANAGED_ACTIVE_DIRECTORY_PASSWORD\":\"xxxxxxxx\"}" \
--kms-key-id <RDSCustomKMSKey> \
--tags Key="AWSRDSCustom",Value="custom-sqlserver"

# Windows based
aws secretsmanager create-secret ^
--name do-not-delete-rds-custom-DomainUserCredentails ^
--description "Active directory user credentials for managing RDS Custom" ^
--secret-string "{\"SELF_MANAGED_ACTIVE_DIRECTORY_USERNAME\":\"tester\",
\"SELF_MANAGED_ACTIVE_DIRECTORY_PASSWORD\":\"xxxxxxxx\"}" ^
--kms-key-id <RDSCustomKMSKey> ^
--tags Key="AWSRDSCustom",Value="custom-sqlserver"
```

Étape 5 : créer ou modifier une instance de base de données personnalisée RDS pour SQL Server

Créez ou modifiez une instance de base de données RDS Custom pour SQL Server à utiliser avec votre annuaire. Vous pouvez utiliser la console, CLI ou l'API RDS pour associer une instance de base de données à un annuaire. Vous pouvez effectuer cette opération de différentes manières :

- Créez une nouvelle instance de base de données SQL Server à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

  Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données SQL Server existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstanceModify RDS API.

  Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Restaurez une instance de base de données SQL Server à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération DBInstance Restore DBSnapshot From RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.

- Restaurez une instance de base de données SQL Server à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération Restore DBInstance ToPointInTime RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

> **Note**
>
> Si votre instance RDS Custom pour SQL Server est déjà jointe manuellement à un AD, vérifiez les paramètres de Règles relatives aux ports de configuration réseauValidation du réseau, et effectuez les étapes 1 à 4. Mettez à jour le `--domain-fqdn``--domain-ou`, et `--domain-auth-secret-arn` sur votre AD, afin que les informations d'identification et les configurations de jonction de domaine soient enregistrées auprès de RDS Custom pour surveiller, enregistrer le CNAME et prendre des mesures de restauration.

Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de base de données puisse utiliser le répertoire que vous avez créé :

- Pour le `--domain-fqdn` paramètre, utilisez le nom de domaine complet de votre AD autogéré.
- Pour le paramètre `--domain-ou`, utilisez l'unité d'organisation que vous avez créée dans votre annuaire AD autogéré.
- Pour le `--domain-auth-secret-arn` paramètre, utilisez la valeur de l'ARN secret que vous avez créé.

> **Important**
>
> Si vous modifiez une instance de base de données pour rejoindre ou supprimer un domaine AD autogéré AWS Managed Microsoft AD, ou si un redémarrage de l'instance de base de données est nécessaire pour que la modification prenne effet. Vous pouvez choisir d'appliquer les modifications immédiatement ou d'attendre la prochaine fenêtre de maintenance. Le choix de l'option Appliquer immédiatement entraîne une interruption de service pour une instance de base de données mono-AZ. Un cluster de base de données multi-AZ effectue un basculement avant de terminer le redémarrage. Pour de plus amples

informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

La commande CLI suivante crée une nouvelle instance de base de données personnalisée RDS pour SQL Server et la joint à une instance autogérée ou AWS Managed Microsoft AD à un domaine.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance  \
--engine custom-sqlserver-se \
--engine-version 15.00.4312.2.v1 \
--db-instance-identifier my-custom-instance \
--db-instance-class db.m5.large \
--allocated-storage 100 --storage-type io1 --iops 1000 \
--master-username my-master-username \
--master-user-password my-master-password \
--kms-key-id  my-RDSCustom-key-id \
--custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance  \
--domain-fqdn "corp.example.com" \
--domain-ou "OU=RDSCustomOU,DC=corp,DC=example,DC=com" \
--domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:do-not-
delete-rds-custom-my-AD-test-secret-123456" \
--db-subnet-group-name my-DB-subnet-grp \
--vpc-security-group-ids  my-securitygroup-id \
--no-publicly-accessible \
--backup-retention-period 3 \
--port 8200 \
--region us-west-2 \
--no-multi-az
```

Dans Windows :

```
aws rds create-db-instance  ^
--engine custom-sqlserver-se ^
--engine-version 15.00.4312.2.v1 ^
--db-instance-identifier my-custom-instance ^
--db-instance-class db.m5.large ^
--allocated-storage 100 --storage-type io1 --iops 1000 ^
--master-usernamemy-master-username ^
--master-user-password my-master-password ^
```

```
--kms-key-id  my-RDSCustom-key-id ^
--custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance  ^
--domain-fqdn "corp.example.com" ^
--domain-ou "OU=RDSCustomOU,DC=corp,DC=example,DC=com" ^
--domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:do-not-delete-rds-custom-my-AD-test-secret-123456" ^
--db-subnet-group-name my-DB-subnet-grp ^
--vpc-security-group-ids  my-securitygroup-id ^
--no-publicly-accessible ^
--backup-retention-period 3 ^
--port 8200 ^
--region us-west-2 ^
--no-multi-az
```

Important

Si votre NetBIOS pour AWS Managed Microsoft AD est corpexample, il apparaît lui-même sous la forme d'une unité d'organisation. Toute nouvelle unité d'organisation créée précédemment apparaîtra sous la forme d'une unité d'organisation imbriquée. Pour AWS Managed Microsoft AD, réglez `--domain-ou` sur`"OU=RDSCustomOU,OU=corpexample,DC=corp,DC=example,DC=com"`.

La commande suivante modifie une instance de base de données RDS Custom for SQL Server existante afin d'utiliser un domaine Active Directory.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --domain-fqdn "corp.example.com" \
    --domain-ou "OU=RDSCustomOU,DC=corp,DC=example,DC=com" \
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:do-not-delete-rds-custom-my-AD-test-secret-123456" \
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --domain-fqdn "corp.example.com" ^
    --domain-ou "OU=RDSCustomOU,DC=corp,DC=example,DC=com" ^
```

```
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:do-not-delete-rds-custom-my-AD-test-secret-123456" ^
```

La commande CLI suivante supprime une instance de base de données RDS Custom pour SQL Server d'un domaine Active Directory.

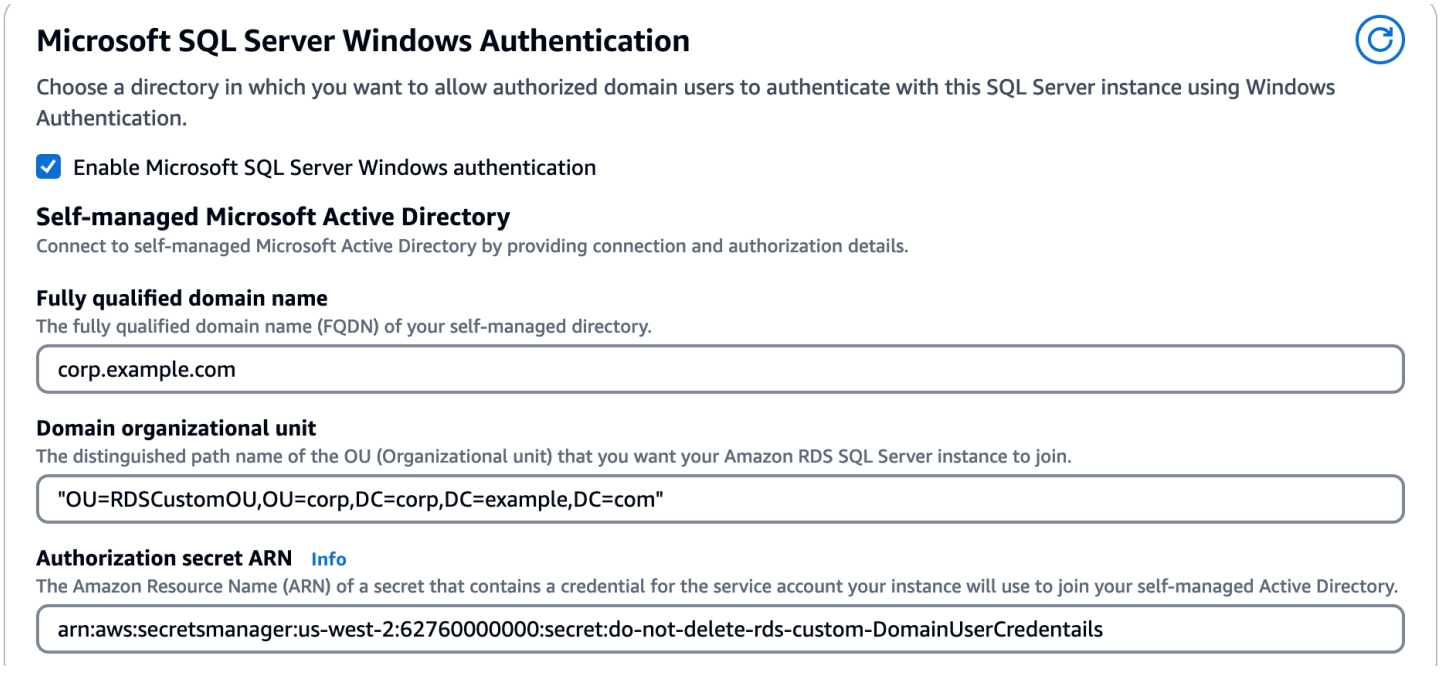
Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my-custom-instance \
    --disable-domain
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my-custom-instance ^
    --disable-domain
```

Lorsque vous utilisez la console pour créer ou modifier votre instance, cliquez sur Activer l'authentification Microsoft SQL Server Windows pour voir les options suivantes.



Il est de votre responsabilité de vous assurer que le nom de domaine complet de votre domaine correspond aux adresses IP des contrôleurs de domaine. Si le contrôleur de domaine IPs ne parvient pas à résoudre le problème, les opérations de jointure de domaine échouent mais la création de

l'instance RDS Custom for SQL Server réussit. Pour plus d'informations sur le dépannage, consultez Dépannage d'Active Directory.

Étape 6 : Création d'une connexion à SQL Server avec authentification Windows

Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données SQL Server de la même manière qu'à n'importe quelle instance de base de données. Comme l'instance de base de données est jointe au domaine AD, vous pouvez configurer des connexions et des utilisateurs SQL Server. Pour ce faire, utilisez l'utilitaire Utilisateurs et groupes AD de votre domaine AD. Les autorisations pour la base de données sont gérées via des autorisations SQL Server standard accordées et révoquées en fonction des connexions Windows.

Pour qu'un utilisateur AD puisse s'authentifier auprès de SQL Server, un identifiant Windows SQL Server doit exister pour l'utilisateur AD ou pour un groupe Active Directory dont il est membre. Un contrôle précis des accès est géré par l'attribution ou la révocation d'autorisations pour ces connexions SQL Server. Un utilisateur AD qui n'a pas de connexion SQL Server ou qui n'appartient pas à un groupe AD avec un tel identifiant ne peut pas accéder à l'instance de base de données SQL Server.

L'`ALTER ANY LOGIN`autorisation est requise pour créer une connexion au serveur AD SQL. Si vous n'avez créé aucune connexion avec cette autorisation, connectez-vous en tant qu'utilisateur principal de l'instance de base de données à l'aide de l'authentification SQL Server et créez vos connexions AD SQL Server dans le contexte de l'utilisateur principal.

Vous pouvez exécuter une commande DDL (Data Definition Language) telle que la suivante pour créer un identifiant SQL Server pour un utilisateur ou un groupe AD.

```
USE [master]
GO
CREATE LOGIN [mydomain\myuser] FROM WINDOWS WITH DEFAULT_DATABASE = [master],
 DEFAULT_LANGUAGE = [us_english];
GO
```

Les utilisateurs (humains et applications) de votre domaine peuvent désormais se connecter à l'instance RDS Custom for SQL Server à partir d'un ordinateur client joint au domaine à l'aide de l'authentification Windows.

Étape 7 : Utilisation de l'authentification Kerberos ou NTLM

Authentification NTLM à l'aide du point de terminaison RDS

Chaque instance de base de données Amazon RDS possède un point de terminaison et chaque point de terminaison possède un nom DNS et un numéro de port pour l'instance de base de données. Pour connecter votre instance de base de données à l'aide d'une application cliente SQL, vous avez besoin du nom DNS et du numéro de port de votre instance de base de données. Pour vous authentifier à l'aide de l'authentification NTLM, vous devez vous connecter au point de terminaison RDS.

Lors de la maintenance planifiée de la base de données ou d'une interruption de service imprévue, Amazon RDS bascule automatiquement vers la base de données up-to-date secondaire afin que les opérations puissent reprendre rapidement sans intervention manuelle. Les instances principale et secondaire utilisent le même point de terminaison, dont l'adresse réseau physique passe au secondaire dans le cadre du processus de basculement. Vous n'avez pas à reconfigurer votre application lorsqu'un basculement se produit.

Authentification Kerberos

L'authentification basée sur Kerberos pour RDS Custom for SQL Server nécessite l'établissement de connexions à un nom principal de service (SPN) spécifique. Cependant, après un événement de basculement, il est possible que l'application ne connaisse pas le nouveau SPN. Pour résoudre ce problème, RDS Custom for SQL Server propose un point de terminaison basé sur Kerberos.

Le point de terminaison basé sur Kerberos suit un format spécifique. Si votre point de terminaison RDS l'est*rds-instance-name*.*account-region-hash*.*aws-region*.rds.amazonaws.com, le point de terminaison basé sur Kerberos correspondant serait. *rds-instance-name*.*account-region-hash*.*aws-region*.awsrds.*fully qualified domain name (FQDN)*

Par exemple, si le point de terminaison RDS est `ad-test.cocv6zwtircu.us-east-1.rds.amazonaws.com` et que le nom de domaine est`corp-ad.company.com`, le point de terminaison basé sur Kerberos serait. `ad-test.cocv6zwtircu.us-east-1.awsrds.corp-ad.company.com`

Ce point de terminaison basé sur Kerberos peut être utilisé pour s'authentifier auprès de l'instance SQL Server à l'aide de Kerberos, même après un événement de basculement, car le point de terminaison est automatiquement mis à jour pour pointer vers le nouveau SPN de l'instance SQL Server principale.

Trouver votre CNAME

Pour trouver votre CNAME, connectez-vous à votre contrôleur de domaine et ouvrez le gestionnaire DNS. Accédez à Forward Lookup Zones et à votre FQDN.

Parcourez les récompenses, l'aws-region et le hachage spécifique au compte et à la région.

Si vous connectez l' EC2 instance personnalisée RDS et que vous essayez de vous connecter à la base de données localement à l'aide de CNAME, votre connexion utilisera l'authentification NTLM au lieu de Kerberos.

Si, après avoir connecté le CNAME à partir d'un client distant, une connexion NTLM est renvoyée, vérifiez si les ports requis sont autorisés.

Pour vérifier si votre connexion utilise Kerberos, exécutez la requête suivante :

```
SELECT net_transport, auth_scheme
    FROM sys.dm_exec_connections
    WHERE session_id = @@SSPID;
```

Gestion d'une instance de base de données dans un domaine

Vous pouvez utiliser la console ou Amazon RDS API pour gérer votre instance de base de données et sa relation avec votre domaine. AWS CLI Par exemple, vous pouvez déplacer l'instance de base de données dans, hors ou entre des domaines.

Par exemple, en utilisant Amazon RDSAPI, vous pouvez effectuer les opérations suivantes :

- Pour réessayer de joindre un domaine en raison d'un échec d'adhésion, utilisez l'odifyDBInstanceAPIopération M et spécifiez l'ID de répertoire de l'adhésion actuelle.
- Pour mettre à jour le nom du IAM rôle pour l'adhésion, utilisez l'`ModifyDBInstance`APIopération et spécifiez l'ID de répertoire du membre actuel et le nouveau IAM rôle.
- Pour supprimer une instance de base de données d'un domaine, utilisez l'`ModifyDBInstance`APIopération et none spécifiez-la comme paramètre de domaine.
- Pour déplacer une instance de base de données d'un domaine à un autre, utilisez l'`ModifyDBInstance`APIopération et spécifiez l'identifiant de domaine du nouveau domaine comme paramètre de domaine.
- Pour répertorier les membres de chaque instance de base de données, utilisez l'escribeDBInstancesAPIopération D.

Restaurer une instance de base de données RDS personnalisée pour le SQL serveur et l'ajouter à un domaine Active Directory

Vous pouvez restaurer un instantané de base de données ou effectuer une point-in-time restauration (PITR) pour une instance de base de données de SQL serveur, puis l'ajouter à un domaine Active Directory. Une fois l'instance de base de données restaurée, modifiez-la à l'aide du processus expliqué dans Étape 5 : créer ou modifier une instance de base de données personnalisée RDS pour SQL Server pour ajouter l'instance de base de données à un domaine AD.

Présentation de l'appartenance au domaine

Après la création ou la modification de votre instance de base de données, l'instance devient un membre du domaine. La AWS console indique le statut de l'appartenance au domaine pour l'instance de base de données. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- joined–L'instance est membre du domaine.
- joining–L'instance est en train de devenir membre du domaine.
- pending-join – L'appartenance de l'instance est en attente.
- pending-maintenance-join— AWS tentera de faire de l'instance un membre du domaine lors de la prochaine fenêtre de maintenance planifiée.
- pending-removal–La suppression de l'instance du domaine est en attente.
- pending-maintenance-removal— AWS tentera de supprimer l'instance du domaine lors de la prochaine fenêtre de maintenance planifiée.
- failed–Un problème de configuration a empêché l'instance d'effectuer la jonction du domaine. Vérifiez et corrigez votre configuration avant d'émettre à nouveau la commande de modification de l'instance.
- removing–La suppression de l'instance du domaine est en cours.

Une demande pour devenir membre d'un domaine peut échouer en raison d'un problème de connectivité réseau ou d'un IAM rôle incorrect. Par exemple, vous pouvez créer une instance de base de données ou modifier une instance existante et faire échouer la tentative pour que l'instance de base de données devienne membre d'un domaine. Dans ce cas, émettez à nouveau la commande pour créer ou modifier l'instance de base de données, ou modifiez l'instance nouvellement créée pour rejoindre le domaine.

Dépannage d’Active Directory

Vous pouvez rencontrer les problèmes suivants lors de la configuration ou de la modification d'un AD.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 2 / 0x2	Le fichier spécifié est introuvable.	Le format ou l'emplacement de l'unité d'organisation (OU) spécifiée avec le paramètre `–domain-ou` est non valide. Le compte de service de domaine spécifié via AWS Secrets Manager ne dispose pas des autorisations requises pour rejoindre l'unité d'organisation.	Passez en revue le paramètre `–domain-ou` . Assurez-vous que le compte de service de domaine dispose des autorisations appropriées pour accéder à l'unité d'organisation.
Erreur 5 / 0x5	Accès refusé.	Autorisations mal configurées pour le compte de service de domaine, ou le compte d'ordinateur existe déjà dans le domaine.	Vérifiez les autorisations du compte de service de domaine dans le domaine et vérifiez que le compte d'RDSordinateur n'est pas dupliqué dans le domaine. Vous pouvez vérifier le nom du compte d'RDSordinateur en l'exécutant `SELECT @@SERVERNAME` sur votre instance de base de données RDS Custom for SQL Server. Si vous utilisez le mode Multi-AZ, essayez de redémarrer avec basculement, puis

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
			vérifiez à nouveau que le compte de l'RDSordinateur est bien enregistré. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.
Erreur 87 / 0x57	Le paramètre est incorrect.	Le compte de service de domaine spécifié via AWS Secrets Manager ne dispose pas des autorisations appropriées. Le profil utilisateur est peut-être également endommagé.	Passez en revue les exigences relatives au compte de service de domaine.
Erreur 234 / 0xEA	L'unité d'organisation (OU) spécifiée n'existe pas.	L'unité d'organisation spécifiée avec le `–domain-ou` paramètre n'existe pas dans votre AD.	Vérifiez le `–domain-ou` paramètre et assurez-vous que l'unité d'organisation spécifiée existe dans votre AD.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 1326 / 0x52E	Le nom d'utilisateur ou le mot de passe est incorrect.	Les informations d'identification du compte de service de domaine fournies dans AWS Secrets Manager contiennent un nom d'utilisateur inconnu ou un mot de passe incorrect. Le compte de domaine peut également être désactivé dans votre annonce.	Assurez-vous que les informations d'identification fournies dans AWS Secrets Manager sont correctes et que le compte de domaine est activé dans votre Active Directory.
Erreur 1355 / 0x54B	Le domaine spécifié n'existe pas ou n'a pas pu être contacté.	Le domaine est en panne, l'ensemble spécifié DNS IPs est inaccessible ou l'ensemble spécifié FQDN est inaccessible.	Vérifiez les paramètres `–domain-dns-ips` et `–domain-fqdn` pour vous assurer qu'ils sont corrects. Vérifiez la configuration réseau de votre instance de base de données RDS personnalisée pour SQL serveur et assurez-vous que votre AD est accessible.
Erreur 1722/0x6BA	Le RPC serveur n'est pas disponible.	Un problème est survenu lors de l'accès au RPC service de votre domaine AD. Il peut s'agir d'un problème de service ou de réseau.	Vérifiez que le RPC service s'exécute sur vos contrôleurs de domaine et que les TCP ports `135` `49152-65535` sont accessibles sur votre domaine depuis votre instance de base de données RDS Custom for SQL Server.

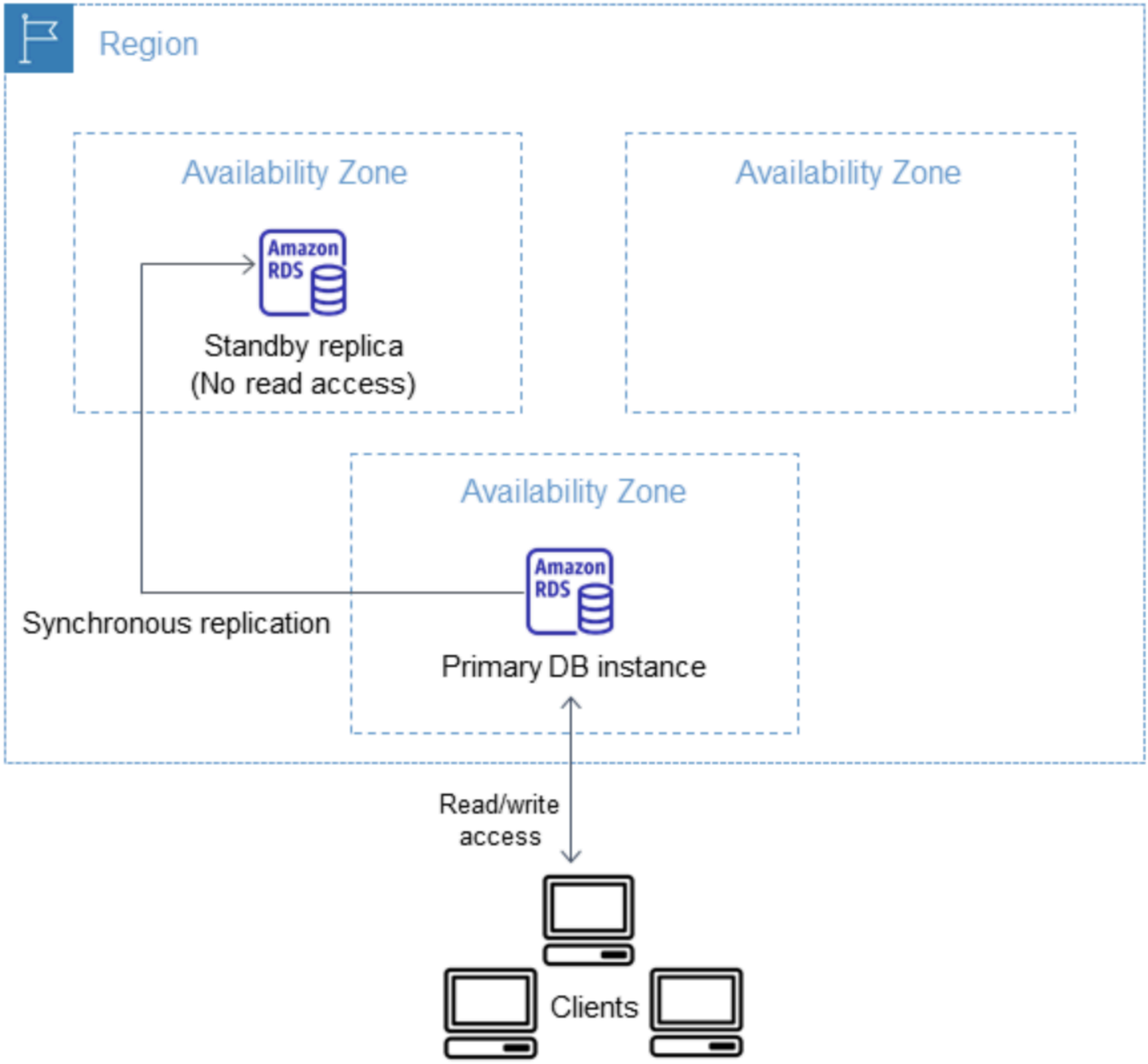
Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 2224 / 0x8B0	Le compte d'utilisateur existe déjà.	Le compte d'ordinateur qui tente d'être ajouté à votre AD existe déjà.	Identifiez le compte d'ordinateur en l'exécutant `SELECT @@SERVERNAME` sur votre instance de base de données RDS Custom for SQL Server, puis supprimez-le soigneusement de votre AD.
Erreur 2242 / 0x8c2	Le mot de passe de cet utilisateur a expiré.	Le mot de passe du compte de service de domaine spécifié via AWS Secrets Manager a expiré.	Mettez à jour le mot de passe du compte de service de domaine utilisé pour joindre votre instance de base de données RDS Custom for SQL Server à votre AD.

Gestion d'un déploiement multi-AZ pour RDS Custom for Server SQL

Dans le cadre d'un déploiement d'instance de base de données multi-AZ pour RDS Custom for SQL Server, Amazon provisionne et gère RDS automatiquement une réplique de secours synchrone dans une autre zone de disponibilité (AZ). L'instance de base de données primaire est répliquée de manière synchrone dans les zones de disponibilité sur un réplica de secours afin d'assurer une redondance des données.

> **Important**
>
> Un déploiement multi-AZ pour RDS Custom for SQL Server est différent d'un déploiement multi-AZ RDS pour SQL Server. Contrairement à Multi-AZ RDS for SQL Server, vous devez configurer les prérequis pour RDS Custom for SQL Server avant de créer votre instance de base de données Multi-AZ, car RDS Custom s'exécute dans votre propre compte, qui nécessite des autorisations.
>
> Si vous ne remplissez pas les conditions préalables, votre instance de base de données multi-AZ risque de ne pas s'exécuter ou de revenir automatiquement à une instance de base de données mono-AZ. Pour plus d'informations sur les conditions préalables, consultez Conditions préalables pour un déploiement multi-AZ avec RDS Custom for Server SQL.

L'exécution d'une instance de base de données en haute disponibilité peut améliorer la disponibilité pendant la maintenance planifiée du système. En cas de maintenance planifiée de la base de données ou d'interruption de service imprévue, Amazon RDS bascule automatiquement vers l'instance de base de données up-to-date secondaire. Cette fonctionnalité permet aux opérations de la base de données de reprendre rapidement sans intervention manuelle. Les instances principales et de secours utilisent le même point de terminaison, l'adresse réseau physique de celui-ci étant transférée vers le réplica secondaire dans le cadre du processus de basculement. Vous n'avez pas à reconfigurer votre application lorsqu'un basculement se produit.



Vous pouvez créer un déploiement multi-AZ RDS personnalisé pour le SQL serveur en spécifiant Multi-AZ lors de la création d'une instance de base de données RDS personnalisée. Vous pouvez utiliser la console pour convertir des instances de base de données RDS personnalisées pour SQL serveur existantes en déploiements multi-AZ en modifiant l'instance de base de données et en spécifiant l'option Multi-AZ. Vous pouvez également spécifier un déploiement d'instance de base de données multi-AZ avec AWS CLI ou Amazon RDSAPI.

La RDS console indique la zone de disponibilité de la réplique de secours (l'AZ secondaire). Vous pouvez également utiliser la `describe-db-instances` CLI commande ou l'`DescribeDBInstances`APIopération pour trouver l'AZ secondaire.

RDSLes instances de base de données personnalisées pour le SQL serveur dotées d'un déploiement multi-AZ peuvent entraîner une latence d'écriture et de validation accrue par rapport à un déploiement mono-AZ. Cette augmentation peut se produire en raison de la réplication de données synchrone entre les instances de base de données. La latence peut changer si votre déploiement bascule vers la réplique de secours, même si elle AWS est conçue avec une connectivité réseau à faible latence entre les zones de disponibilité.

> **Note**
>
> Pour les charges de travail de production, nous vous recommandons d'utiliser une classe d'instance de base de données avec Provisioned IOPS (opérations d'entrée/sortie par seconde) pour des performances rapides et cohérentes. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter Conditions requises et limitations d'Amazon RDS Custom for SQL Server.

Rubriques

- Disponibilité des régions et des versions
- Limites d'un déploiement multi-AZ avec RDS Custom for Server SQL
- Conditions préalables pour un déploiement multi-AZ avec RDS Custom for Server SQL
- Création d'un RDS déploiement multi-AZ personnalisé pour le SQL serveur
- Modification d'un déploiement mono-AZ RDS Custom for SQL Server en un déploiement multi-AZ
- Modification d'un déploiement multi-AZ RDS personnalisé pour SQL serveur vers un déploiement mono-AZ
- Processus de basculement pour un déploiement multi-AZ RDS personnalisé pour SQL serveur

Disponibilité des régions et des versions

Les déploiements multi-AZ pour RDS Custom for SQL Server sont pris en charge pour les éditions de SQL serveur suivantes :

- SQLServer 2022 et 2019 : éditions Enterprise, Standard, Web et Developer

> **Note**
>
> Les déploiements multi-AZ pour RDS Custom for SQL Server ne sont pas pris en charge sur SQL Server 2019 CU8 (15.00.4073.23) ou les versions antérieures.

Les déploiements multi-AZ pour RDS Custom for SQL Server sont disponibles dans toutes les régions où RDS Custom for SQL Server est disponible. Pour plus d'informations sur la disponibilité régionale des déploiements multi-AZ pour RDS Custom for SQL Server, consultez. Régions et moteurs de base de données pris en charge pour RDS Custom pour SQL Server

Limites d'un déploiement multi-AZ avec RDS Custom for Server SQL

Les déploiements multi-AZ avec RDS Custom for SQL Server présentent les limites suivantes :

- Les déploiements multi-AZ entre régions ne sont pas pris en charge.
- Vous ne pouvez pas configurer l'instance de base de données secondaire pour accepter l'activité de lecture de base de données.
- Lorsque vous utilisez une version de moteur personnalisée (CEV) avec un déploiement multi-AZ, votre instance de base de données secondaire l'utilisera également. CEV L'instance de base de données secondaire ne peut pas en utiliser une autreCEV.

Conditions préalables pour un déploiement multi-AZ avec RDS Custom for Server SQL

Si vous disposez déjà d'un déploiement mono-AZ RDS personnalisé pour SQL serveur, les conditions préalables supplémentaires suivantes sont requises avant de le transformer en déploiement multi-AZ. Vous pouvez choisir de remplir les prérequis manuellement ou à l'aide du CloudFormation modèle fourni. Le dernier CloudFormation modèle contient les prérequis pour les déploiements mono-AZ et multi-AZ.

> **Important**
>
> Pour simplifier la configuration, nous vous recommandons d'utiliser le dernier fichier AWS CloudFormation modèle fourni dans les instructions de configuration réseau pour créer les prérequis. Pour de plus amples informations, veuillez consulter Configuration avec AWS CloudFormation.

> **Note**
>
> Lorsque vous modifiez un déploiement mono-AZ RDS personnalisé pour SQL serveur existant en déploiement multi-AZ, vous devez remplir ces conditions préalables. Si vous ne remplissez pas les conditions préalables, la configuration multi-AZ échouera. Pour remplir toutes les conditions préalables, suivez les étapes dans Modification d'un déploiement mono-AZ RDS Custom for SQL Server en un déploiement multi-AZ.

- Mettez à jour les règles entrantes et sortantes du groupe de RDS sécurité pour autoriser le port 1120.
- Ajoutez une règle dans la liste de contrôle d'accès de votre réseau privé (ACL) qui autorise TCP les ports `0-65535` pour l'instance de base de donnéesVPC.
- Créez de nouveaux SQS VPC points de terminaison Amazon qui permettent à l'instance de base de données RDS Custom for SQL Server de communiquer avecSQS.
- Mettez à jour les SQS autorisations dans le rôle de profil d'instance.

Création d'un RDS déploiement multi-AZ personnalisé pour le SQL serveur

Pour créer un déploiement multi-AZ RDS personnalisé pour le SQL serveur, suivez les étapes décrites dansCréation et connexion à une instance de base de données pour Amazon RDS Custom for SQL Server.

> **Important**
>
> Pour simplifier la configuration, nous vous recommandons d'utiliser le dernier fichier AWS CloudFormation modèle fourni dans les instructions de configuration réseau. Pour de plus amples informations, veuillez consulter Configuration avec AWS CloudFormation.

La création d'un déploiement multi-AZ dure quelques minutes.

Modification d'un déploiement mono-AZ RDS Custom for SQL Server en un déploiement multi-AZ

Vous pouvez modifier une instance de base de données RDS Custom for SQL Server existante d'un déploiement mono-AZ en un déploiement multi-AZ. Lorsque vous modifiez l'instance de base de données, Amazon RDS effectue plusieurs actions :

- Prend un instantané de l'instance de base de données primaire.
- Crée de nouveaux volumes pour le réplica en attente à partir de l'instantané. Ces volumes s'initialisent en arrière-plan, et les performances maximales du volume sont atteintes après l'initialisation complète des données.
- Active la réplication synchrone au niveau des blocs entre les instances de base de données primaire et secondaire.

> **Important**
>
> Nous vous recommandons d'éviter de modifier votre instance de base de données RDS Custom for SQL Server pour passer d'un déploiement mono-AZ à un déploiement multi-AZ sur une instance de base de données de production pendant les périodes de pointe.

AWS utilise un instantané pour créer l'instance de secours afin d'éviter les temps d'arrêt lors de la conversion du mode mono-AZ au mode multi-AZ, mais les performances peuvent être affectées pendant et après la conversion au mode multi-AZ. Cet impact peut être significatif pour les charges de travail sensibles à la latence d'écriture. Bien que cette fonctionnalité permette de restaurer rapidement de gros volumes à partir de snapshots, elle peut entraîner une augmentation de la latence des I/O opérations en raison de la réplication synchrone. Cette latence peut avoir un impact sur les performances de votre base de données.

> **Note**
>
> Si vous avez créé votre instance de base de données RDS Custom pour SQL Server avant le 29 août 2024, appliquez un correctif à la dernière version mineure avant de la modifier.
>
> - Pour les instances SQL Server 2019, mettez à niveau la version du moteur de base de données vers `15.00.4410.1.v1` ou une version ultérieure.

- Pour les instances SQL Server 2022, mettez à niveau la version du moteur de base de données vers `16.00.4150.1.v1` ou une version ultérieure.

Rubriques

- Configuration des conditions requises pour modifier un déploiement mono-AZ en déploiement multi-AZ à l'aide de CloudFormation
- Configuration des conditions préalables pour modifier manuellement un déploiement mono-AZ en un déploiement multi-AZ
- Modifiez à l'aide de la console RDS, de la AWS CLI ou de l'API RDS.

Configuration des conditions requises pour modifier un déploiement mono-AZ en déploiement multi-AZ à l'aide de CloudFormation

Pour utiliser un déploiement multi-AZ, vous devez vous assurer que vous avez appliqué le dernier CloudFormation modèle avec les prérequis, ou configurer manuellement les derniers prérequis. Si vous avez déjà appliqué le dernier modèle CloudFormation prérequis, vous pouvez ignorer ces étapes.

Pour configurer les conditions préalables au déploiement multi-AZ de RDS Custom pour SQL Server à l'aide de CloudFormation

1. Ouvrez la CloudFormation console à l'adresse https://console.aws.amazon.com/cloudformation.
2. Pour démarrer l'assistant Créer une pile, sélectionnez la pile existante que vous avez utilisée pour créer un déploiement mono-AZ et choisissez Mettre à jour.

   La page Mettre à jour la pile s'affiche.
3. Pour Prérequis - Préparer le modèle, choisissez Le modèle est prêt.
4. Pour Specify template (Spécifier un modèle), procédez comme suit :

   a. Téléchargez le dernier fichier AWS CloudFormation modèle. Ouvrez le menu contextuel (clic droit) du lien custom-sqlserver-onboard.zip et choisissez Enregistrer le lien sous.

   b. Enregistrez et extrayez le fichier `custom-sqlserver-onboard.json` de votre ordinateur.

   c. Pour Source du modèle, choisissez Charger un fichier de modèle.

   d. Pour Choose file (Choisir un fichier), accédez à `custom-sqlserver-onboard.json` et sélectionnez ce fichier.

5. Choisissez Suivant.

   La page Specify stack details (Spécifier les détails de la pile) s'affiche.
6. Pour conserver les options par défaut, choisissez Next (Suivant).

   La page Options avancées s'affiche.
7. Pour conserver les options par défaut, choisissez Next (Suivant).
8. Pour conserver les options par défaut, choisissez Next (Suivant).
9. Sur la page Vérifier les modifications, procédez comme suit :

   a. Sous Capacités, cochez la case Je sais qu' AWS CloudFormation peut créer des ressources IAM avec des noms personnalisés.

   b. Sélectionnez Envoyer.
10. Vérifiez que la mise à jour est réussie. Le statut d'une opération réussie affiche `UPDATE_COMPLETE`.

Si la mise à jour échoue, toute nouvelle configuration spécifiée dans le processus de mise à jour sera annulée. La ressource existante sera toujours utilisable. Par exemple, si vous ajoutez des règles ACL réseau numérotées 18 et 19, mais que certaines règles existantes portent les mêmes numéros, la mise à jour renverra l'erreur suivante : `Resource handler returned message: "The network acl entry identified by 18 already exists.` (L'entrée ACL réseau numérotée 18 existe déjà). Dans ce scénario, vous pouvez modifier les règles ACL existantes pour utiliser un nombre inférieur à 18, puis réessayez la mise à jour.

Configuration des conditions préalables pour modifier manuellement un déploiement mono-AZ en un déploiement multi-AZ

> **Important**
>
> Pour simplifier la configuration, nous vous recommandons d'utiliser le dernier fichier AWS CloudFormation modèle fourni dans les instructions de configuration réseau. Pour de plus amples informations, veuillez consulter Configuration des conditions requises pour modifier un déploiement mono-AZ en déploiement multi-AZ à l'aide de CloudFormation.

Si vous choisissez de configurer les conditions préalables manuellement, exécutez les tâches suivantes.

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Choisissez Point de terminaison. La page Créer un point de terminaison s'affiche.
3. Pour Catégorie de service, choisissez Services AWS .
4. Dans Services, recherchez *SQS*
5. Dans VPC, choisissez le VPC dans lequel votre instance de base de données RDS Custom for SQL Server est déployée.
6. Dans Sous-réseaux, choisissez les sous-réseaux dans lesquels votre instance de base de données RDS Custom for SQL Server est déployée.
7. Dans Groupes de sécurité, choisissez le *-vpc-endpoint-sg* groupe.
8. Pour Politique, choisissez Personnalisé
9. Dans votre politique personnalisée, remplacez le *AWS partitionRegion*,*accountId*, et *IAM-Instance-role* par vos propres valeurs.

   JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Condition": {
                "StringLike": {
                    "aws:ResourceTag/AWSRDSCustom": "custom-sqlserver"
                }
            },
            "Action": [
                "SQS:SendMessage",
                "SQS:ReceiveMessage",
                "SQS:DeleteMessage",
                "SQS:GetQueueUrl"
            ],
            "Resource": "arn:aws:sqs:us-east-1:111122223333:do-not-delete-
rds-custom-*",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/{IAM-Instance-role}"
            }
        }
    ]
```

```
}
```

10. Mettez à jour le Profil d'instance avec l'autorisation d'accéder à Amazon SQS. Remplacez le *AWS partitionRegion*, et *accountId* par vos propres valeurs.

```
                        {
    "Sid": "SendMessageToSQSQueue",
    "Effect": "Allow",
    "Action": [
      "SQS:SendMessage",
      "SQS:ReceiveMessage",
      "SQS:DeleteMessage",
      "SQS:GetQueueUrl"

    ],
    "Resource": [
      {
        "Fn::Sub": "arn:${AWS::Partition}:sqs:${AWS::Region}:${AWS::AccountId}:do-
not-delete-rds-custom-*"
      }
    ],
    "Condition": {
      "StringLike": {
        "aws:ResourceTag/AWSRDSCustom": "custom-sqlserver"
      }
    }
  }
                        >
```

11. Mettez à jour les règles entrantes et sortantes du groupe de sécurité Amazon RDS pour autoriser le port 1120.

    a. Dans Groupes de sécurité, choisissez le *-rds-custom-instance-sg* groupe.

    b. Pour les règles entrantes, créez une règle TCP personnalisée pour autoriser le port *1120* depuis le groupe source*-rds-custom-instance-sg*.

    c. Pour les règles sortantes, créez une règle TCP personnalisée pour autoriser le port *1120* vers le groupe de destination*-rds-custom-instance-sg*.

12. Ajoutez une règle dans la liste de contrôle d'accès (ACL) de votre réseau privé qui autorise les ports TCP `0-65535` pour le sous-réseau source de l'instance de base de données.

> **Note**
>
> Lorsque vous créez une Règle entrante et une Règle sortante, prenez note du Numéro de règle existant le plus élevé. Les nouvelles règles que vous créez doivent avoir un Numéro de règle inférieur à 100 et ne correspondre à aucun Numéro de règle existant.

a. Dans Réseau ACLs, choisissez le *-private-network-acl* groupe.

b. Pour les règles entrantes, créez une règle entièrement TCP pour autoriser les ports TCP dont la source `0-65535` provient de et. *privatesubnet1* *privatesubnet2*

c. Pour les règles sortantes, créez une règle All TCP pour autoriser les ports TCP `0-65535` à destination et. *privatesubnet1* *privatesubnet2*

Modifiez à l'aide de la console RDS, de la AWS CLI ou de l'API RDS.

Une fois les conditions requises remplies, vous pouvez modifier une instance de base de données personnalisée RDS pour SQL Server d'un déploiement mono-AZ à un déploiement multi-AZ à l'aide de la console RDS, de la AWS CLI ou de l'API RDS.

console

Pour modifier un déploiement mono-AZ RDS Custom for SQL Server existant en un déploiement multi-AZ

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans la console Amazon RDS, choisissez Bases de données.

   Le volet Bases de données s'affiche.
3. Choisissez l'instance de base de données RDS Custom for SQL Server que vous souhaitez modifier.
4. Dans Actions, choisissez Convertir en déploiement multi-AZ.
5. Sur la page Confirmation, choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. Le choix de cette option n'entraîne pas d'interruption de service, mais il existe un impact possible sur les performances. Vous pouvez également choisir d'appliquer la mise

à jour pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

6. Sur la page Confirmation, choisissez Convertir en Multi-AZ.

AWS CLI

Pour passer à un déploiement d'instance de base de données multi-AZ à l'aide de AWS CLI, appelez la modify-db-instancecommande et définissez l'`--multi-az`option. Spécifiez l'identifiant d'instance de base de données et les valeurs des autres options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter Paramètres des instances de base de données.

Example

Le code suivant modifie `mycustomdbinstance` en incluant l'option `--multi-az`. Les modifications sont appliquées dans la prochaine fenêtre de maintenance à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mycustomdbinstance \
    --multi-az \
    --no-apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mycustomdbinstance ^
    --multi-az  \ ^
    --no-apply-immediately
```

API RDS

Pour passer à un déploiement d'instance de base de données multi-AZ avec l'API RDS, appelez l'DBInstanceopération Modify et définissez le `MultiAZ` paramètre sur true.

Modification d'un déploiement multi-AZ RDS personnalisé pour SQL serveur vers un déploiement mono-AZ

Vous pouvez modifier une instance de base de données RDS personnalisée pour SQL serveur existante d'un déploiement multi-AZ à un déploiement mono-AZ.

Console

Pour modifier une instance de base de données RDS personnalisée pour SQL serveur d'un déploiement multi-AZ à un déploiement mono-AZ.

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la RDS console Amazon, sélectionnez Databases.

   Le volet Bases de données s'affiche.
3. Choisissez l'instance de base de données RDS personnalisée pour le SQL serveur que vous souhaitez modifier.
4. Pour Déploiement multi-AZ, choisissez Non.
5. Sur la page Confirmation, choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. Le choix de cette option n'entraîne pas d'interruption de service, mais il existe un impact possible sur les performances. Vous pouvez également choisir d'appliquer la mise à jour pendant le créneau de maintenance suivant. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.
6. Sur la page Confirmation, choisissez Modifier l'instance de base de données.

AWS CLI

Pour modifier un déploiement multi-AZ en un déploiement mono-AZ à l'aide de AWS CLI, appelez la modify-db-instancecommande et incluez l'`--no-multi-az`option. Spécifiez l'identifiant d'instance de base de données et les valeurs des autres options que vous souhaitez modifier. Pour plus d'informations sur chaque option, veuillez consulter Paramètres des instances de base de données.

Example

Le code suivant modifie `mycustomdbinstance` en incluant l'option `--no-multi-az`. Les modifications sont appliquées dans la prochaine fenêtre de maintenance à l'aide de `--no-apply-immediately`. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`.

Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mycustomdbinstance \
    --no-multi-az  \
    --no-apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mycustomdbinstance ^
    --no-multi-az \ ^
    --no-apply-immediately
```

RDS API

Pour modifier un déploiement multi-AZ en un déploiement mono-AZ à l'aide de RDSAPI, appelez l'odifyDBInstanceopération M et définissez le `MultiAZ` paramètre sur. `false`

Processus de basculement pour un déploiement multi-AZ RDS personnalisé pour SQL serveur

Si une panne planifiée ou imprévue de votre instance de base de données résulte d'un défaut d'infrastructure, Amazon passe RDS automatiquement à une réplique de secours dans une autre zone de disponibilité si vous avez activé le mode multi-AZ. La durée du basculement dépend de l'activité de la base de données et d'autres conditions au moment où l'instance de base de données primaire est devenue indisponible. Les temps de basculement oscillent généralement entre 60 et 120 secondes. Cependant, les transactions importantes ou les processus de récupération longs peuvent augmenter le temps de basculement. Lorsque le basculement est terminé, l'affichage de la nouvelle zone de disponibilité par la RDS console peut prendre plus de temps.

> **Note**
>
> Vous pouvez forcer le basculement manuellement lorsque vous redémarrez une instance de base de données avec basculement. Pour plus d'informations sur le redémarrage d'une instance de base de données, consultez Redémarrage d'une instance de base de données.

Amazon RDS gère automatiquement les basculements afin que vous puissiez reprendre les opérations de base de données le plus rapidement possible sans intervention administrative. L'instance de base de données primaire bascule automatiquement vers le réplica de secours si l'une des conditions décrites dans le tableau suivant se produit : Vous pouvez consulter ces raisons de basculement dans le journal des RDS événements.

Raison du basculement	Description
`The operating system for the RDS Custom for SQL Server Multi-AZ DB instance is being patched in an offline operation`	Un basculement a été déclenché pendant la fenêtre de maintenance d'un correctif du système d'exploitation ou d'une mise à jour de sécurité. Pour de plus amples informations, veuillez consulter Entretien d'une instance de base de données.
`The primary host of the RDS Custom for SQL Server Multi-AZ DB instance is unhealthy.`	Le déploiement d'instance de base de données multi-AZ a détecté une instance de base de données primaire déficiente et a opéré un basculement.
`The primary host of the RDS Custom for SQL Server Multi-AZ DB instance is unreachable due to loss of network connectivity.`	RDSla surveillance a détecté une défaillance de l'accessibilité réseau de l'instance de base de données principale et a déclenché un basculement.
`The RDS Custom for SQL Server Multi-AZ DB instance was modified by the customer.`	Une modification d'instance de base de données a déclenché un basculement. Pour de plus amples informations, veuillez consulter Modification d'une instance de base de données RDS personnalisée pour le SQL serveur.

Raison du basculement	Description
`The storage volume of the primary host of the RDS Custom for SQL Server Multi-AZ DB instance experienced a failure.`	Le déploiement d'instance de base de données multi-AZ a détecté un problème de stockage sur l'instance de base de données primaire et a opéré un basculement.
`The user requested a failover of the RDS Custom for SQL Server Multi-AZ DB instance.`	L'instance de base de données Multi-AZ RDS personnalisée pour SQL serveur a été redémarrée avec basculement. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.
`The RDS Custom for SQL Server Multi-AZ primary DB instance is busy or unresponsive.`	L'instance de base de données primaire ne répond pas. Nous vous recommandons d'essayer les étapes suivantes : • Examinez les journaux d'événements et CloudWatch les journaux pour détecter toute utilisation excessive CPU de la mémoire ou de l'espace d'échange. Pour de plus amples informations, veuillez consulter Utilisation des notifications d'RDSévénements Amazon. • Créez une règle qui se déclenche lors d'un RDS événement Amazon. Pour de plus amples informations, veuillez consulter Création d'une règle qui se déclenche sur un événement Amazon RDS. • Évaluez votre charge de travail pour déterminer si vous utilisez la classe d'instance de base de données appropriée. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .

Pour déterminer si votre instance de base de données Multi-AZ a basculé, voici ce que vous pouvez faire :

- Configurez des abonnements aux événements de base de données pour vous informer par e-mail ou SMS qu'un basculement a été initié. Pour plus d'informations sur les événements, consultez Utilisation des notifications d'RDSévénements Amazon.
- Affichez les événements de votre base de données à l'aide de la RDS console ou API des opérations.
- Consultez l'état actuel de votre déploiement d'instance de base de données multi-AZ RDS personnalisée pour SQL serveur à l'aide de la RDS console ou CLI des API opérations.

Paramètres Time to live (TTL) avec des applications utilisant un déploiement multi-AZ RDS personnalisé pour le SQL serveur

Le mécanisme de basculement modifie automatiquement l'enregistrement du système de noms de domaine (DNS) de l'instance de base de données pour qu'il pointe vers l'instance de base de données de secours. Par conséquent, vous devez rétablir toutes les connexions existantes à votre instance de base de données. Assurez-vous que toute valeur de configuration de DNS cache time-to-live (TTL) est faible et vérifiez que votre application ne sera pas mise en cache DNS pendant une période prolongée. Une TTL valeur élevée peut empêcher votre application de se reconnecter rapidement à l'instance de base de données après un basculement.

Sauvegarde et restauration d'une instance de base de données Amazon RDS Custom for SQL Server

Comme AmazonRDS, RDS Custom crée et enregistre des sauvegardes automatisées de votre instance de base de données RDS Custom for SQL Server lorsque la conservation des sauvegardes est activée. Vous pouvez également sauvegarder votre instance de base de données manuellement. Les sauvegardes automatisées comprennent des sauvegardes instantanées et des sauvegardes du journal des transactions. Des sauvegardes instantanées sont effectuées pour l'ensemble du volume de stockage de l'instance de base de données pendant la fenêtre de sauvegarde spécifiée. Des sauvegardes du journal des transactions sont effectuées pour les bases de données PITR éligibles à intervalles réguliers. RDSCustom enregistre les sauvegardes automatisées de votre instance de base de données conformément à la période de conservation des sauvegardes que vous avez spécifiée. Vous pouvez utiliser des sauvegardes automatisées pour restaurer votre instance de base de données à un moment donné pendant la période de conservation des sauvegardes.

Vous pouvez également effectuer des sauvegardes instantanées manuellement. Vous pouvez créer une nouvelle instance de base de données à partir de ces sauvegardes instantanées à tout moment. Pour plus d'informations sur la création manuelle d'un instantané de base de données, consultez Création d'un instantané RDS personnalisé pour SQL le serveur.

Bien que les sauvegardes instantanées soient opérationnelles comme des sauvegardes complètes, vous n'êtes facturée que pour l'utilisation incrémentielle du stockage. Le premier instantané d'une instance de base de données RDS personnalisée contient les données de l'instance de base de données complète. Les instantanés suivants de la même base de données sont incrémentiels, ce qui signifie que seules les données qui ont changé depuis l'instantané le plus récent sont enregistrées.

Rubriques

- Création d'un instantané RDS personnalisé pour SQL le serveur
- Restauration à partir d'un instantané de base de données RDS personnalisé pour le SQL serveur
- Restauration d'une instance RDS Custom for SQL Server à un moment donné
- Suppression d'un instantané RDS personnalisé pour SQL le serveur
- Suppression RDS des sauvegardes automatisées personnalisées pour le SQL serveur

Création d'un instantané RDS personnalisé pour SQL le serveur

RDSCustom for SQL Server crée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de l'instance de base de données et pas uniquement des

bases de données individuelles. Lorsque vous créez un instantané, spécifiez l'instance de base de données RDS personnalisée pour SQL serveur à sauvegarder. Nommez votre instantané afin que vous puissiez restaurer ultérieurement à partir de ce dernier.

Lorsque vous créez un instantané, RDS Custom for SQL Server crée un EBS instantané Amazon pour le volume(`D:`), qui est le volume de base de données attaché à l'instance de base de données. Pour que les instantanés soient faciles à associer à une instance de base de données spécifique, ils sont labelisés `DBSnapshotIdentifier`, `DbiResourceId` et `VolumeType`.

La création d'un instantané de base de données entraîne une brève suspension des I/O. Cette suspension peut durer de quelques secondes à quelques minutes, en fonction de la taille et de la classe de votre instance de base de données. Le temps de création des instantanés varie en fonction du nombre total et de la taille de vos bases de données. Pour en savoir plus sur le nombre de bases de données éligibles à une opération de restauration instantanée (PITR), consultezNombre de bases de données éligibles PITR par type de classe d'instance.

Étant donné que l'instantané inclut l'intégralité du volume de stockage, la taille de fichiers comme les fichiers temporaires a également une incidence sur le temps nécessaire à la création de l'instantané. Pour en savoir plus sur la création des instantanés, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Créez un instantané RDS personnalisé pour le SQL serveur à l'aide de la console ou du AWS CLI.

Console

Pour créer un instantané RDS personnalisé

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Dans la liste des instances de base de données RDS personnalisées, choisissez l'instance pour laquelle vous souhaitez prendre un instantané.
4. Sous Actions, choisissez Take snapshot (Prendre un instantané).

   La fenêtre Capture d'un instantané DB apparaît.
5. Dans Snapshot name (Nom de l'instantané), saisissez le nom de l'instantané.
6. Choisissez Prendre un instantané.

AWS CLI

Vous créez un instantané d'une instance de base de données RDS personnalisée à l'aide de la create-db-snapshot AWS CLI commande.

Spécifiez les options suivantes :

- `--db-instance-identifier`— Identifie l'instance de base de données RDS personnalisée que vous allez sauvegarder
- `--db-snapshot-identifier`— Nomme votre instantané RDS personnalisé afin que vous puissiez le restaurer ultérieurement

Dans cet exemple, vous créez un instantané de base de données appelé *my-custom-snapshot* pour une instance de base de données RDS personnalisée appelée*my-custom-instance*.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-snapshot \
    --db-instance-identifier my-custom-instance \
    --db-snapshot-identifier my-custom-snapshot
```

Dans Windows:

```
aws rds create-db-snapshot ^
    --db-instance-identifier my-custom-instance ^
    --db-snapshot-identifier my-custom-snapshot
```

Restauration à partir d'un instantané de base de données RDS personnalisé pour le SQL serveur

Lorsque vous restaurez une instance de base de données RDS personnalisée pour le SQL serveur, vous fournissez le nom du snapshot de base de données et le nom de la nouvelle instance. Vous ne pouvez pas effectuer de restauration à partir d'un instantané vers une instance de base de données RDS personnalisée existante. Une nouvelle instance de base de données RDS personnalisée pour le SQL serveur est créée lors de la restauration.

La restauration à partir d'un instantané rétablit le volume de stockage au moment où le cliché a été pris. Cela inclura toutes les bases de données et tous les autres fichiers présents sur le `(D:)` volume.

Console

Pour restaurer une instance de base de données RDS personnalisée à partir d'un instantané de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à partir duquel vous voulez restaurer.
4. Pour Actions, choisissez Restaurer l'instantané.
5. Sur la page Restaurer l'instance de base de données, pour l'identifiant de l'instance de base de données, entrez le nom de votre instance de base de données RDS personnalisée restaurée.
6. Choisissez Restore DB Instance (Restaurer une instance de base de données).

AWS CLI

Vous restaurez un instantané de base de données RDS personnalisé à l'aide de la commande restore-db-instance-from AWS CLI -db-snapshot.

Si l'instantané à partir duquel vous restaurez est destiné à une instance de base de données privée, assurez-vous de spécifier le `db-subnet-group-name` correct et `no-publicly-accessible`. Sinon, l'instance de base de données est accessible par défaut au public. Les options suivantes sont requises :

- `db-snapshot-identifier` – Identifie l'instantané à partir duquel restaurer
- `db-instance-identifier`— Spécifie le nom de l'instance de base de données RDS personnalisée à créer à partir du snapshot de base de données
- `custom-iam-instance-profile`— Spécifie le profil d'instance associé à l'EC2instance Amazon sous-jacente d'une instance de base de données RDS personnalisée.

Le code suivant restaure l'instantané nommé `my-custom-snapshot` pour `my-custom-instance`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
  --db-snapshot-identifier my-custom-snapshot \
  --db-instance-identifier my-custom-instance \
  --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance \
  --no-publicly-accessible
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
  --db-snapshot-identifier my-custom-snapshot ^
  --db-instance-identifier my-custom-instance ^
  --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance ^
  --no-publicly-accessible
```

Restauration d'une instance RDS Custom for SQL Server à un moment donné

Vous pouvez restaurer une instance de base de données à un moment précis (PITR), en créant une nouvelle instance de base de données. Pour être prises en chargePITR, la rétention des sauvegardes doit être activée sur vos instances de base de données.

L'heure de restauration la plus récente pour une instance de base de données RDS Custom for SQL Server dépend de plusieurs facteurs, mais elle se situe généralement à moins de 5 minutes de l'heure actuelle. Pour connaître l'heure de restauration la plus récente pour une instance de base de données, utilisez la AWS CLI describe-db-instancescommande et examinez la valeur renvoyée dans le `LatestRestorableTime` champ correspondant à l'instance de base de données. Pour connaître l'heure de restauration la plus récente pour chaque instance de base de données dans la RDS console Amazon, choisissez Sauvegardes automatisées.

Vous pouvez procéder à une restauration à n'importe quel moment dans le passé au cours de la période de rétention des sauvegardes. Pour connaître l'heure de restauration la plus proche pour chaque instance de base de données, choisissez Sauvegardes automatisées dans la RDS console Amazon.

Pour obtenir des informations générales sur PITR, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Rubriques

- PITRconsidérations relatives à RDS Custom for SQL Server
- Nombre de bases de données éligibles PITR par type de classe d'instance
- Rendre les bases de données inéligibles à PITR
- Journaux de transactions dans Amazon S3
- PITRRestaurez à l'aide du AWS Management ConsoleAWS CLI, du ou du RDSAPI.

PITRconsidérations relatives à RDS Custom for SQL Server

Dans RDS Custom for SQL Server, PITR les différences importantes suivantes sont les suivantes par rapport PITR à Amazon RDS :

- PITRrestaure uniquement les bases de données dans l'instance de base de données. Il ne restaure pas le système d'exploitation ou les fichiers du lecteur C:.
- Pour une instance de base de données RDS personnalisée pour SQL serveur, une base de données est sauvegardée automatiquement et PITR n'est éligible que dans les conditions suivantes :
  - La base de données est en ligne.
  - Son modèle de récupération est défini sur `FULL`.
  - Elle est inscriptible.
  - Ses fichiers physiques sont sur le lecteur D:.
  - Elle n'est pas répertoriée dans le tableau `rds_pitr_blocked_databases`. Pour de plus amples informations, veuillez consulter Rendre les bases de données inéligibles à PITR.
- Les bases de données éligibles PITR sont déterminées par l'ordre de leur identifiant de base de données. RDSCustom for SQL Server autorise jusqu'à 5 000 bases de données par instance de base de données. Toutefois, le nombre maximum de bases de données restaurées par une PITR opération pour une instance de base de données RDS Custom for SQL Server dépend du type de classe d'instance. Pour de plus amples informations, veuillez consulter Nombre de bases de données éligibles PITR par type de classe d'instance.

  Les autres bases de données qui n'en font pas partie PITR peuvent être restaurées à partir de snapshots de base de données, y compris les sauvegardes automatiques de snapshots utilisées pourPITR.
- L'ajout d'une nouvelle base de données, le changement de nom d'une base de données ou la restauration d'une base de données éligible à PITR initie un instantané de l'instance de base de données.

- Le nombre maximum de bases de données susceptibles d'être PITR modifiées lorsque l'instance de base de données est soumise à une opération de calcul à l'échelle, en fonction du type de classe d'instance cible. Si l'instance est agrandie, ce qui permet à un plus grand nombre de bases de données de l'instance d'être éligiblesPITR, un nouvel instantané est pris.
- Les bases de données restaurées portent le même nom que dans l'instance de base de données source. Vous ne pouvez pas spécifier un autre nom.
- `AWSRDSCustomSQLServerIamRolePolicy`nécessite l'accès à d'autres AWS services. Pour de plus amples informations, veuillez consulter Ajoutez une politique d'accès à AWSRDSCustom SQLServer InstanceRole.
- Les modifications de fuseau horaire ne sont pas prises en charge pour RDS Custom for SQL Server. Si vous modifiez le système d'exploitation ou le fuseau horaire de l'instance de base de données PITR (et toute autre automatisation), cela ne fonctionne pas.

Nombre de bases de données éligibles PITR par type de classe d'instance

Le tableau suivant indique le nombre maximum de bases de données éligibles en PITR fonction du type de classe d'instance.

Type de classe d'instance	Nombre maximum de bases de données PITR éligibles
db.*.large	100
db.*.xlarge vers db.*.2xlarge	150
db.*.4xlarge vers db.*.8xlarge	300
db.*.12xlarge vers db.*.16xlarge	600

Type de classe d'instance	Nombre maximum de bases de données PITR éligibles				
db.*.24xlarge, db.*32xlarge	1 000				

*Représente différents types de classes d'instance.

Le nombre maximum de bases de données éligibles PITR sur une instance de base de données dépend du type de classe d'instance. Ce nombre est compris entre 100 pour les plus petits et 1 000 pour les plus grands types de classes d'instance pris en charge par RDS Custom for SQL Server. SQLles bases de données (`master, model, msdb, tempdb`) du système serveur ne sont pas incluses dans cette limite. Lorsqu'une instance de base de données est redimensionnée à la hausse ou à la baisse, selon le type de classe d'instance cible, RDS Custom met automatiquement à jour le nombre de bases de données éligiblesPITR. RDSCustom for SQL Server sera envoyé `RDS-EVENT-0352` lorsque le nombre maximum de bases de données éligibles à des PITR modifications sur une instance de base de données est modifié. Pour de plus amples informations, veuillez consulter Événements de version du moteur personnalisés.

> **Note**
>
> PITRla prise en charge de plus de 100 bases de données n'est disponible que sur les instances de base de données créées après le 26 août 2023. Pour les instances créées avant le 26 août 2023, le nombre maximum de bases de données éligibles PITR est de 100, quelle que soit la classe d'instance. Pour activer la PITR prise en charge de plus de 100 bases de données sur des instances de base de données créées avant le 26 août 2023, vous pouvez effectuer l'action suivante :
>
> - Mettez à niveau la version du moteur de base de données vers la version 15.00.4322.2.v1 ou supérieure

Au cours d'une PITR opération, RDS Custom restaure toutes les bases de données qui faisaient partie de l'instance de PITR base de données source au moment de la restauration. Une fois que

l'instance de base de données cible a terminé les opérations de restauration, si la conservation des sauvegardes est activée, l'instance de base de données commence à sauvegarder en fonction du nombre maximum de bases de données éligibles PITR sur l'instance de base de données cible.

Par exemple, si votre instance de base de données s'exécute sur une instance `db.*.xlarge` contenant 200 bases de données :

1. RDSCustom for SQL Server choisira les 150 premières bases de données, classées par leur ID de base de données, à PITR sauvegarder.
2. Vous modifiez l'instance pour la mettre à l'échelle jusqu'à db.*.4xlarge.
3. Une fois l'opération de calcul de l'échelle terminée, RDS Custom for SQL Server choisira les 300 premières bases de données, classées par leur ID de base de données, à PITR sauvegarder. Chacune des 200 bases de données répondant aux conditions PITR requises sera désormais éligiblePITR.
4. Vous modifiez maintenant l'instance pour la réduire à db.*.xlarge.
5. Une fois l'opération de calcul de l'échelle terminée, RDS Custom for SQL Server sélectionnera à nouveau les 150 premières bases de données, classées par leur ID de base de données, à PITR sauvegarder.

Rendre les bases de données inéligibles à PITR

Vous pouvez choisir d'exclure des bases de données individuelles dePITR. Pour ce faire, mettez leurs valeurs `database_id` dans un tableau `rds_pitr_blocked_databases`. Utilisez le SQL script suivant pour créer la table.

Pour créer la table rds_pitr_blocked_databases

- Exécutez le SQL script suivant.

```
create table msdb..rds_pitr_blocked_databases
(
database_id INT NOT NULL,
database_name SYSNAME NOT NULL,
db_entry_updated_date datetime NOT NULL DEFAULT GETDATE(),
db_entry_updated_by SYSNAME NOT NULL DEFAULT CURRENT_USER,
PRIMARY KEY (database_id)
);
```

Pour obtenir la liste des bases de données éligibles et non éligibles, consultez le fichier `RI.End`dans le répertoire `RDSCustomForSQLServer/Instances/`*`DB_instance_resource_ID`*`/TransactionLogMetadata` du compartiment Amazon S3 `do-not-delete-rds-custom-`*`$ACCOUNT_ID`*`-`*`$REGION`*`-`*`unique_identifier`*. Pour plus d'informations sur le fichier `RI.End`, consultez Journaux de transactions dans Amazon S3.

Vous pouvez également déterminer la liste des bases de données éligibles à PITR l'utilisation du SQL script suivant. Définissez la `@limit` variable sur le nombre maximum de bases de données éligibles PITR pour la classe d'instance. Pour de plus amples informations, veuillez consulter Nombre de bases de données éligibles PITR par type de classe d'instance.

Pour déterminer la liste des bases de données éligibles pour une classe PITR d'instance de base de données

- Exécutez le SQL script suivant.

```
DECLARE @Limit INT;
SET @Limit = (insert-database-instance-limit-here);

USE msdb;
IF (EXISTS (SELECT * FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_SCHEMA = 'dbo' AND
 TABLE_NAME = 'rds_pitr_blocked_databases'))
    WITH TABLE0 AS (
        SELECT hdrs.database_id as DatabaseId, sdb.name as DatabaseName,
 'ALWAYS_ON_NOT_WRITABLE_REPLICA' as Reason, NULL as DatabaseNameOnPitrTable
        FROM sys.dm_hadr_database_replica_states hdrs
        INNER JOIN sys.databases sdb ON sdb.database_id = hdrs.database_id
        WHERE (hdrs.is_local = 1 AND hdrs.is_primary_replica = 0)
        OR (sys.fn_hadr_is_primary_replica (sdb.name) = 1 AND DATABASEPROPERTYEX
 (sdb.name, 'Updateability') = 'READ_ONLY')
    ),
    TABLE1 as (
            SELECT dbs.database_id as DatabaseId, sysdbs.name as DatabaseName,
 'OPTOUT' as Reason,
            CASE WHEN dbs.database_name = sysdbs.name THEN NULL ELSE
 dbs.database_name END AS DatabaseNameOnPitrTable
            FROM msdb.dbo.rds_pitr_blocked_databases dbs
            INNER JOIN sys.databases sysdbs ON dbs.database_id = sysdbs.database_id
            WHERE sysdbs.database_id > 4
            ),
    TABLE2 as (
            SELECT
```

```
            db.name AS DatabaseName,
            db.create_date AS CreateDate,
            db.state_desc AS DatabaseState,
            db.database_id AS DatabaseId,
            rs.database_guid AS DatabaseGuid,
            rs.last_log_backup_lsn AS LastLogBackupLSN,
            rs.recovery_fork_guid RecoveryForkGuid,
            rs.first_recovery_fork_guid AS FirstRecoveryForkGuid,
            db.recovery_model_desc AS RecoveryModel,
            db.is_auto_close_on AS IsAutoClose,
            db.is_read_only as IsReadOnly,
            NEWID() as FileName,
            CASE WHEN(db.state_desc = 'ONLINE'
                    AND db.recovery_model_desc != 'SIMPLE'
                    AND((db.is_auto_close_on = 0 and db.collation_name IS NOT NULL)
 OR db.is_auto_close_on = 1))
                    AND db.is_read_only != 1
                    AND db.user_access = 0
                    AND db.source_database_id IS NULL
                    AND db.is_in_standby != 1
                    THEN 1 ELSE 0 END AS IsPartOfSnapshot,
            CASE WHEN db.source_database_id IS NULL THEN 0 ELSE 1 END AS
 IsDatabaseSnapshot
            FROM sys.databases db
            INNER JOIN sys.database_recovery_status rs
            ON db.database_id = rs.database_id
            WHERE DB_NAME(db.database_id) NOT IN('tempdb') AND
            db.database_id NOT IN (SELECT DISTINCT DatabaseId FROM TABLE1) AND
            db.database_id NOT IN (SELECT DISTINCT DatabaseId FROM TABLE0)
        ),
        TABLE3 as(
            Select @Limit+count(DatabaseName) as TotalNumberOfDatabases from TABLE2
 where TABLE2.IsPartOfSnapshot=1 and DatabaseName in ('master','model','msdb')
        )
        SELECT TOP(SELECT TotalNumberOfDatabases from TABLE3)
 DatabaseName,CreateDate,DatabaseState,DatabaseId from TABLE2 where
 TABLE2.IsPartOfSnapshot=1
        ORDER BY TABLE2.DatabaseID ASC
ELSE
    WITH TABLE0 AS (
        SELECT hdrs.database_id as DatabaseId, sdb.name as DatabaseName,
 'ALWAYS_ON_NOT_WRITABLE_REPLICA' as Reason, NULL as DatabaseNameOnPitrTable
        FROM sys.dm_hadr_database_replica_states hdrs
        INNER JOIN sys.databases sdb ON sdb.database_id = hdrs.database_id
```

```
       WHERE (hdrs.is_local = 1 AND hdrs.is_primary_replica = 0)
       OR (sys.fn_hadr_is_primary_replica (sdb.name) = 1 AND DATABASEPROPERTYEX
(sdb.name, 'Updateability') = 'READ_ONLY')
   ),
   TABLE1 as (
           SELECT
           db.name AS DatabaseName,
           db.create_date AS CreateDate,
           db.state_desc AS DatabaseState,
           db.database_id AS DatabaseId,
           rs.database_guid AS DatabaseGuid,
           rs.last_log_backup_lsn AS LastLogBackupLSN,
           rs.recovery_fork_guid RecoveryForkGuid,
           rs.first_recovery_fork_guid AS FirstRecoveryForkGuid,
           db.recovery_model_desc AS RecoveryModel,
           db.is_auto_close_on AS IsAutoClose,
           db.is_read_only as IsReadOnly,
           NEWID() as FileName,
           CASE WHEN(db.state_desc = 'ONLINE'
                   AND db.recovery_model_desc != 'SIMPLE'
                   AND((db.is_auto_close_on = 0 and db.collation_name IS NOT NULL)
OR db.is_auto_close_on = 1))
                   AND db.is_read_only != 1
                   AND db.user_access = 0
                   AND db.source_database_id IS NULL
                   AND db.is_in_standby != 1
                   THEN 1 ELSE 0 END AS IsPartOfSnapshot,
           CASE WHEN db.source_database_id IS NULL THEN 0 ELSE 1 END AS
IsDatabaseSnapshot
           FROM sys.databases db
           INNER JOIN sys.database_recovery_status rs
           ON db.database_id = rs.database_id
           WHERE DB_NAME(db.database_id) NOT IN('tempdb') AND
           db.database_id NOT IN (SELECT DISTINCT DatabaseId FROM TABLE0)
       ),
       TABLE2 as(
           SELECT @Limit+count(DatabaseName) as TotalNumberOfDatabases from TABLE1
where TABLE1.IsPartOfSnapshot=1 and DatabaseName in ('master','model','msdb')
       )
       select top(select TotalNumberOfDatabases from TABLE2)
DatabaseName,CreateDate,DatabaseState,DatabaseId from TABLE1 where
TABLE1.IsPartOfSnapshot=1
       ORDER BY TABLE1.DatabaseID ASC
```

> **Note**
>
> Les bases de données qui ne sont que des liens symboliques sont également exclues des bases de données éligibles aux PITR opérations. La requête ci-dessus ne filtre pas en fonction de ces critères.

Journaux de transactions dans Amazon S3

La période de conservation des sauvegardes détermine si les journaux de transactions des instances de base de données RDS Custom for SQL Server sont automatiquement extraits et chargés sur Amazon S3. Une valeur différente de zéro signifie que des sauvegardes automatiques sont créées et que l'agent RDS personnalisé télécharge les journaux de transactions sur S3 toutes les 5 minutes.

Les journaux des transactions sur S3 sont chiffrés au repos à l'aide de la AWS KMS key que vous avez fournie lors de la création de votre instance de base de données. Pour plus d'informations, consultez la section Protection des données à l'aide du chiffrement côté serveur dans le Guide de l'utilisateur Amazon Simple Storage Service.

Les journaux de transactions de chaque base de données sont chargés dans un compartiment S3 nommé `do-not-delete-rds-custom-`*`$ACCOUNT_ID`*`-`*`$REGION`*`-`*`unique_identifier`*. Le répertoire `RDSCustomForSQLServer/Instances/`*`DB_instance_resource_ID`* dans le compartiment S3 contient deux sous-répertoires :

- `TransactionLogs` – Contient les journaux de transactions de chaque base de données et leurs métadonnées respectives.

  Le nom du fichier journal des transactions suit le modèle *`yyyyMMddHHmm`*`.`*`database_id`*`.`*`timestamp`*, par exemple :

  ```
  202110202230.11.1634769287
  ```

  Le même nom de fichier avec le suffixe `_metadata` contient des informations sur le journal des transactions telles que les numéros de séquence de journaux, le nom de la base de données et `RdsChunkCount`. `RdsChunkCount` détermine le nombre de fichiers physiques qui représentent un fichier journal de transactions unique. Vous pouvez voir des fichiers contenant des suffixes `_0001`,`_0002`, et ainsi de suite, ce qui correspond aux morceaux physiques d'un fichier journal de transactions. Si vous souhaitez utiliser un fichier journal de transactions en morceaux, veillez à fusionner les morceaux après les avoir téléchargés.

Imaginons un scénario dans lequel vous disposez des fichiers suivants :

- `202110202230.11.1634769287`
- `202110202230.11.1634769287_0001`
- `202110202230.11.1634769287_0002`
- `202110202230.11.1634769287_metadata`

Le `RdsChunkCount` est 3. L'ordre de fusion des fichiers est le suivant : `202110202230.11.1634769287`, `202110202230.11.1634769287_0001`, `202110202230.11.1634769287_0002`.

- `TransactionLogMetadata` – Contient des informations de métadonnées sur chaque itération de l'extraction du journal de transactions.

  Le fichier `RI.End` contient des informations sur toutes les bases de données dont les journaux de transactions ont été extraits, et toutes les bases de données existantes mais dont les journaux de transactions n'ont pas été extraits. Le nom de fichier `RI.End` suit le modèle *`yyyyMMddHHmm`*`.RI.End.`*`timestamp`*, par exemple :

  ```
  202110202230.RI.End.1634769281
  ```

PITRRestaurez à l'aide du AWS Management ConsoleAWS CLI, du ou du RDSAPI.

Vous pouvez restaurer une instance de base de données RDS personnalisée pour SQL serveur à un moment donné en utilisant le AWS Management Console AWS CLI, le ou le RDSAPI.

Console

Pour restaurer une instance de base de données RDS personnalisée à une heure spécifiée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Choisissez l'instance de base de données RDS personnalisée que vous souhaitez restaurer.
4. Sous Actions, sélectionnez Restaurer à un moment donné.

   La fenêtre Restaurer à un instant dans le passé s'affiche.

5. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

   Si vous choisissez Custom (Personnalisé), saisissez la date et l'heure auxquelles vous souhaitez restaurer l'instance.

   Les heures sont affichées dans votre fuseau horaire local, qui est indiqué par un décalage par rapport au temps universel coordonné (UTC). Par exemple, UTC -5 correspond à l'heure normale de l'Est/heure avancée du centre.
6. Pour l'identifiant de l'instance de base de données, entrez le nom de l'instance de base de données RDS personnalisée restaurée cible. Le nom doit être unique.
7. Choisissez d'autres options selon vos besoins, comme la classe d'instance de base de données.
8. Choisissez Restaurer à un instant dans le passé.

AWS CLI

Vous restaurez une instance de base de données à une heure spécifiée en utilisant la point-in-time AWS CLI commande restore-db-instance-to- pour créer une nouvelle instance de base de données RDS personnalisée.

Utilisez l'une des options suivantes pour spécifier la sauvegarde à partir de laquelle effectuer la restauration :

- `--source-db-instance-identifier` *`mysourcedbinstance`*
- `--source-dbi-resource-id` *`dbinstanceresourceID`*
- `--source-db-instance-automated-backups-arn` *`backupARN`*

L'option `custom-iam-instance-profile` est obligatoire.

L'exemple suivant restaure `my-custom-db-instance` vers une nouvelle instance de base de données nommée `my-restored-custom-db-instance` au moment spécifié.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-identifier my-custom-db-instance\
    --target-db-instance-identifier my-restored-custom-db-instance \
```

```
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance \
    --restore-time 2022-10-14T23:45:00.000Z
```

Dans Windows:

```
aws rds restore-db-instance-to-point-in-time ^
    --source-db-instance-identifier my-custom-db-instance ^
    --target-db-instance-identifier my-restored-custom-db-instance ^
    --custom-iam-instance-profile AWSRDSCustomInstanceProfileForRdsCustomInstance ^
    --restore-time 2022-10-14T23:45:00.000Z
```

Suppression d'un instantané RDS personnalisé pour SQL le serveur

Vous pouvez supprimer des instantanés de base de données gérés par RDS Custom for SQL Server lorsque vous n'en avez plus besoin. La procédure de suppression est la même pour les instances de base de données Amazon RDS et RDS Custom.

Les EBS instantanés Amazon des volumes binaire et racine restent dans votre compte plus longtemps car ils peuvent être liés à certaines instances exécutées sur votre compte ou à d'autres instantanés RDS personnalisés pour le SQL serveur. Ces EBS instantanés sont automatiquement supprimés une fois qu'ils ne sont plus liés à aucune ressource RDS personnalisée existante pour le SQL serveur (instances de base de données ou sauvegardes).

Console

Pour supprimer un instantané d'une instance de base de données RDS personnalisée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané de base de données à supprimer.
4. Pour Actions, choisissez Delete snapshot (Supprimer la pile).
5. Dans la page de confirmation, sélectionnez Supprimer.

AWS CLI

Pour supprimer un instantané RDS personnalisé, utilisez la AWS CLI commande delete-db-snapshot.

L'option suivante est requise :

- `--db-snapshot-identifier` – L'instantané à supprimer

L'exemple suivant supprime l'instantané de base de données `my-custom-snapshot`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-snapshot \
  --db-snapshot-identifier my-custom-snapshot
```

Dans Windows:

```
aws rds delete-db-snapshot ^
  --db-snapshot-identifier my-custom-snapshot
```

Suppression RDS des sauvegardes automatisées personnalisées pour le SQL serveur

Vous pouvez supprimer les sauvegardes automatiques conservées pour RDS Custom for SQL Server lorsqu'elles ne sont plus nécessaires. La procédure est identique à celle de suppression des RDS sauvegardes Amazon.

Console

Pour supprimer une sauvegarde automatisée conservée

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).
3. Choisissez Retained (Conservées).
4. Choisissez la sauvegarde automatisée conservée que vous souhaitez supprimer.
5. Pour Actions, choisissez Supprimer.
6. Dans la page de confirmation, entrez **`delete me`** et choisissez Delete (Supprimer).

AWS CLI

Vous pouvez supprimer une sauvegarde automatique conservée à l'aide de la AWS CLI commande delete-db-instance-automated-backup.

L'option suivante est utilisée pour supprimer une sauvegarde automatisée conservée.

- `--dbi-resource-id`— L'identifiant de ressource pour l'instance de base de données RDS personnalisée source.

  Vous pouvez trouver l'identifiant de ressource pour l'instance de base de données source d'une sauvegarde automatique conservée à l'aide de la AWS CLI commande describe-db-instance-automated-backups.

L'exemple suivant supprime la sauvegarde automatisée conservée avec l'identifiant de ressource d'instance de base de données `custom-db-123ABCEXAMPLE`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance-automated-backup \
    --dbi-resource-id custom-db-123ABCEXAMPLE
```

Dans Windows:

```
aws rds delete-db-instance-automated-backup ^
    --dbi-resource-id custom-db-123ABCEXAMPLE
```

Copier un instantané de base de données Amazon RDS personnalisé pour SQL Server

Avec RDS Custom pour SQL Server, vous pouvez copier des sauvegardes automatisées et des instantanés de base de données manuels. Après avoir copié un instantané, la copie que vous créez est un instantané manuel. Vous pouvez effectuer plusieurs copies d'une sauvegarde automatique ou d'un instantané manuel, mais chaque copie doit avoir un identifiant unique.

Vous ne pouvez copier un instantané d'un même AWS compte que sur différents sites Régions AWS où RDS Custom for SQL Server est disponible. Les opérations suivantes ne sont actuellement pas prises en charge :

- Copier des instantanés de base de données dans un même fichier Région AWS.
- Copier des instantanés de base de données entre AWS comptes.

RDS Custom pour SQL Server prend en charge la copie incrémentielle de snapshots. Pour de plus amples informations, veuillez consulter Considérations relatives à la copie incrémentielle d'instantanés.

Rubriques

- Limites
- Gestion du chiffrement
- Copie entre régions
- Instantanés d'instances de base de données créées avec des versions de moteur personnalisées (CEV)
- Accorder les autorisations requises à votre principal IAM
- Copie d'un instantané de base de données

Limites

Les limites suivantes s'appliquent à la copie d'un instantané de base de données pour RDS Custom for SQL Server :

- Si vous supprimez un instantané source avant que l'instantané cible ne soit disponible, la copie d'instantané peut échouer. Vérifiez que le cliché cible a un statut égal à `AVAILABLE` avant de supprimer le cliché source.

- Vous ne pouvez pas spécifier un nom de groupe d'options ou copier un groupe d'options dans votre demande de copie instantanée de base de données.
- Si vous supprimez des AWS ressources dépendantes de l'instantané de base de données source avant ou pendant le processus de copie, votre demande de capture instantanée de copie peut échouer de manière asynchrone.
  - Si vous supprimez le fichier de sauvegarde SMK (Service Master Key) de votre instance de base de données source stocké dans le compartiment S3 géré par RDS Custom de votre compte, la copie instantanée de base de données réussit de manière asynchrone. Cependant, les fonctionnalités de SQL Server qui dépendent de SMK, telles que les bases de données compatibles TDE, rencontrent des problèmes. Pour de plus amples informations, veuillez consulter Résolution des problèmes liés à l'état PENDING_RECOVERY pour les bases de données compatibles TDE dans RDS Custom for SQL Server.
- La copie d'instantanés de base de données au sein d'une même base de données n' Région AWS est actuellement pas prise en charge.
- La copie d'instantanés de base de données entre AWS comptes n'est actuellement pas prise en charge.

Les limites liées à la copie d'un instantané de base de données pour Amazon RDS s'appliquent également à RDS Custom for SQL Server. Pour de plus amples informations, veuillez consulter Limites.

Gestion du chiffrement

Toutes les instances de base de données RDS Custom pour SQL Server et les instantanés de base de données sont chiffrés à l'aide de clés KMS. Vous ne pouvez copier un instantané chiffré que sur un instantané chiffré. Vous devez donc spécifier une clé KMS valide dans la destination Région AWS pour votre demande de copie d'instantané de base de données.

L'instantané source reste chiffré pendant tout le processus de copie. Amazon RDS utilise le chiffrement d'enveloppe pour protéger les données lors de l'opération de copie avec la clé Région AWS KMS de destination spécifiée. Pour plus d'informations, consultez Chiffrement d'enveloppe dans le Guide du développeur AWS Key Management Service .

Copie entre régions

Vous pouvez copier les instantanés de base de données entre Régions AWS. Toutefois, il existe certaines contraintes et considérations en ce qui concerne la copie d'instantanés entre régions.

Autoriser RDS à communiquer entre eux pour la copie d' Régions AWS instantanés

Une fois qu'une demande de copie d'instantané de base de données entre régions est traitée avec succès, RDS démarre la copie. Une demande d'autorisation permettant à RDS d'accéder à l'instantané source est créée. Cette demande d'autorisation lie l'instantané de base de données source à l'instantané de base de données cible. Cela permet à RDS de copier uniquement vers le cliché cible spécifié.

RDS vérifie l'autorisation en utilisant l'`rds:CrossRegionCommunication`autorisation dans le rôle IAM lié au service. Si la copie est autorisée, RDS peut communiquer avec la région source et terminer l'opération de copie.

RDS n'a pas accès aux instantanés de base de données qui n'étaient pas autorisés auparavant par une demande de copieDBSnapshot . L'autorisation est révoquée une fois la copie terminée.

RDS utilise le rôle lié au service afin de vérifier l'autorisation dans la région source. La copie échoue si vous supprimez le rôle lié au service pendant le processus de copie.

Pour plus d'informations, consultez la section Utilisation des rôles liés à un service dans le Guide de l'AWS Identity and Access Management utilisateur.

Utilisation des informations d'identification AWS Security Token Service

Les jetons de session provenant du point de terminaison global AWS Security Token Service (AWS STS) ne sont valides Régions AWS que s'ils sont activés par défaut (régions commerciales). Si vous utilisez les informations d'identification issues de l'opération d'`assumeRole`API dans AWS STS, utilisez le point de terminaison régional si la région source est une région optionnelle. Sinon, la demande échoue. Vos informations d'identification doivent être valides dans les deux régions, ce qui n'est vrai pour les régions optionnelles que lorsque vous utilisez le point de AWS STS terminaison régional.

Pour utiliser le point de terminaison global, assurez-vous qu'il est activé dans les opérations pour les deux régions. Définissez le point de terminaison global sur « `Valid` in all » Régions AWS dans les paramètres du AWS STS compte.

Pour plus d'informations, consultez la section Gestion AWS STS dans et Région AWS dans le guide de AWS Identity and Access Management l'utilisateur.

Instantanés d'instances de base de données créées avec des versions de moteur personnalisées (CEV)

Pour un instantané de base de données d'une instance de base de données utilisant une version de moteur personnalisée (CEV), RDS associe le CEV à l'instantané de base de données. Pour copier un instantané de base de données source associé à un CEV Régions AWS, RDS copie le CEV ainsi que le cliché de base de données source dans la région de destination.

Si vous copiez plusieurs instantanés de base de données associés au même CEV vers la même région de destination, la première demande de copie copie le CEV associé. Le processus de copie des demandes suivantes trouve le CEV initialement copié et l'associe aux copies instantanées de base de données suivantes. La copie CEV existante doit être en `AVAILABLE` état pour être associée aux copies instantanées de base de données.

Pour copier un instantané de base de données associé à un CEV, la politique IAM du demandeur doit disposer des autorisations nécessaires pour autoriser à la fois la copie de l'instantané de base de données et la copie CEV associée. Les autorisations suivantes sont nécessaires dans la politique IAM de votre demandeur pour autoriser la copie CEV associée :

- `rds:CopyCustomDBEngineVersion`- Le principal IAM de votre demandeur doit avoir l'autorisation de copier le CEV source dans la région cible avec le snapshot de base de données source. La demande de copie instantanée échoue en raison d'erreurs d'autorisation si le principal IAM de votre demandeur n'est pas autorisé à copier le CEV source.
- `ec2:CreateTags`- L' EC2 AMI sous-jacente du CEV source est copiée dans la région cible dans le cadre de la copie du CEV. RDS Custom tente de baliser l'AMI avec la `AWSRDSCustom` balise avant de copier l'AMI. Assurez-vous que le principal IAM du demandeur est autorisé à créer la balise par rapport à l'AMI sous-jacente au CEV source dans la région source.

Pour plus d'informations sur les autorisations de copie CEV, consultezAccorder les autorisations requises à votre principal IAM.

Accorder les autorisations requises à votre principal IAM

Assurez-vous que vous disposez d'un accès suffisant pour copier un instantané de base de données RDS Custom for SQL Server. Le rôle ou l'utilisateur IAM (appelé principal IAM) chargé de copier un instantané de base de données à l'aide de la console ou de l'interface de ligne de commande doit disposer de l'une des politiques suivantes pour une création d'instance de base de données réussie :

- La `AdministratorAccess` politique ou,
- La politique `AmazonRDSFullAccess` avec les autorisations supplémentaires suivantes :

```
s3:CreateBucket
s3:GetBucketPolicy
s3:PutBucketPolicy
kms:CreateGrant
kms:DescribeKey
ec2:CreateTags
```

RDS Custom utilise ces autorisations lors de la copie d'instantanés. Régions AWS Ces autorisations configurent les ressources de votre compte qui sont requises pour les opérations RDS Custom. Pour plus d'informations sur l'autorisation `kms:CreateGrant`, consultez AWS KMS key gestion.

L'exemple de politique JSON suivant accorde les autorisations requises en plus de la `AmazonRDSFullAccess` politique.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CreateS3BucketAndReadWriteBucketPolicy",
            "Effect": "Allow",
            "Action": [
                "s3:CreateBucket",
                "s3:PutBucketPolicy",
                "s3:GetBucketPolicy"
            ],
            "Resource": "arn:aws:s3:::do-not-delete-rds-custom-*"
        },
        {
            "Sid": "CreateKmsGrant",
            "Effect": "Allow",
            "Action": [
                "kms:CreateGrant",
                "kms:DescribeKey"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "CreateEc2Tags",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": "*"
        }
    ]
}
```

> **Note**
>
> Assurez-vous que les autorisations répertoriées ne sont pas limitées par les politiques de contrôle des services (SCPs), les limites d'autorisation ou les politiques de session associées au principal IAM.

Si vous utilisez des conditions avec des clés contextuelles dans la politique IAM du demandeur, certaines conditions peuvent entraîner l'échec de la demande. Pour plus d'informations sur les pièges courants liés aux conditions de la politique IAM, consultez. Demander une copie d'instantané de bases de données entre régions

Copie d'un instantané de base de données

Utilisez les procédures suivantes pour copier un instantané de base de données. Pour chaque AWS compte, vous pouvez copier jusqu'à 20 instantanés de base de données à la fois de l'un Région AWS à l'autre. Si vous copiez un instantané de base de données vers un autre Région AWS, vous créez un instantané de base de données manuel qui y est conservé Région AWS. La copie d'un instantané de base de données depuis la source Région AWS entraîne des frais de transfert de données Amazon RDS. Pour de plus amples informations sur la tarification du transfert des données, veuillez consulter la Tarification Amazon RDS.

Une fois que la copie instantanée de base de données a été créée dans le nouveau Région AWS, elle se comporte de la même manière que tous les autres instantanés de base de données qu'elle contient. Région AWS

Vous pouvez copier un instantané de base de données à l'aide de AWS Management Console AWS CLI, ou de l'API Amazon RDS.

Console

La procédure suivante copie un instantané de base de données RDS Custom pour SQL Server à l'aide du AWS Management Console.

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots (Instantanés).
3. Sélectionnez le snapshot de base de données RDS Custom pour SQL Server que vous souhaitez copier.
4. Dans le menu déroulant Actions, choisissez Copier un instantané.

RDS > Snapshots > Copy snapshot

Copy snapshot

Settings

Source DB snapshot
DB snapshot identifier for the snapshot being copied.
db1-snapshot

New DB snapshot identifier
DB snapshot identifier for the new snapshot.
copied-db1-snapshot
Must start with a letter and only contain letters, digits, or hyphens.

Destination Region Info
Choose a Region

Resource tags Info
Copy tags from the source DB snapshot

Depending on the amount of data to be copied and the Region you choose, this operation can take several hours to complete. The display on the progress bar might be delayed until setup is complete.

Encryption

Encryption Info
Enable Encryption
Choose to encrypt the copy of the source DB snapshot. Master key IDs and aliases appear in the list after they have been created using KMS. You cannot remove encryption from an encrypted DB snapshot.

AWS KMS key Info
Enter a key ARN

Amazon Resource Name (ARN)
Example: arn:aws:kms:<region>:<accountID>:key/<key-id>

Account

KMS key ID

5. Pour copier le snapshot de base de données vers un autre Région AWS, définissez Destination Region sur la valeur requise.

   > **Note**
   >
   > La destination Région AWS doit disposer de la même version de moteur de base de données disponible que la source Région AWS.

6. Pour Nouvel identifiant de capture de base de données, entrez un nom unique pour l'instantané de base de données. Vous pouvez effectuer plusieurs copies d'une sauvegarde automatique ou d'un instantané manuel, mais chaque copie doit avoir un identifiant unique.
7. (Facultatif) Pour copier les balises et les valeurs de l'instantané vers la copie de cet instantané, choisissez Copier les balises.
8. Pour le chiffrement, spécifiez l'identifiant de clé KMS à utiliser pour chiffrer la copie instantanée de la base de données.

   > **Note**
   >
   > RDS Custom for SQL Server chiffre tous les instantanés de base de données. Vous ne pouvez pas créer un instantané de base de données non chiffré.

9. Choisissez Copy snapshot (Copier un instantané).

RDS Custom for SQL Server crée une copie instantanée de base de données de votre instance de base de données dans celle Région AWS de votre sélection.

AWS CLI

Vous pouvez copier un instantané de base de données RDS Custom pour SQL Server à l'aide de la AWS CLI commande copy-db-snapshot. Si vous copiez le cliché dans un nouveau Région AWS, exécutez la commande dans le nouveau Région AWS. Les options suivantes sont utilisées pour copier un instantané de base de données. Toutes les options ne sont pas requises pour tous les scénarios.

- `--source-db-snapshot-identifier`- Identifiant de l'instantané de base de données source.
  - Si l'instantané source se trouve dans un emplacement différent Région AWS de celui de la copie, spécifiez un ARN d'instantané de base de données valide. Par

exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:instance1-snapshot-12345678`.

- `--target-db-snapshot-identifier`- Identifiant de la nouvelle copie de l'instantané de base de données.
- `--kms-key-id`-Identifiant de clé KMS pour un instantané de base de données chiffré. L'identificateur de clé KMS est l'ARN (Amazon Resource Name), l'identificateur de clé ou l'alias de clé pour la clé KMS.
  - Si vous copiez un instantané chiffré vers une autre Région AWS, vous devez spécifier une clé KMS pour la Région AWS de destination. Les clés KMS sont spécifiques à l'endroit dans Région AWS lequel elles ont été créées et vous ne pouvez pas utiliser les clés de chiffrement les unes Région AWS des autres, Région AWS sauf si une clé multirégionale est utilisée. Pour plus d'informations sur les clés KMS multi-Régions, veuillez consulter la section Utilisation de clés multi-Régions dans AWS KMS.
- `--copy-tags`- Incluez les balises et les valeurs de l'instantané source dans la copie de l'instantané.

Les options suivantes ne sont pas prises en charge pour copier un instantané de base de données RDS Custom pour SQL Server :

- `--copy-option-group`
- `--option-group-name`
- `--pre-signed-url`
- `--target-custom-availability-zone`

L'exemple de code suivant copie un instantané de base de données chiffré de la région USA Ouest (Oregon) vers la région USA Est (Virginie du Nord). Exécutez la commande dans la région de destination (us-east-1).

Pour Linux, macOS ou Unix :

```
aws rds copy-db-snapshot \
     --region us-east-1 \
    --source-db-snapshot-identifier arn:aws:rds:us-
west-2:123456789012:snapshot:instance1-snapshot-12345678 \
    --target-db-snapshot-identifier mydbsnapshotcopy \
    --kms-key-id a1b2c3d4-1234-5678-wxyz-a1b2c3d4d5e6
```

Pour Windows :

```
aws rds copy-db-snapshot ^
     --region us-east-1 ^
    --source-db-snapshot-identifier arn:aws:rds:us-
west-2:123456789012:snapshot:instance1-snapshot-12345678 ^
    --target-db-snapshot-identifier mydbsnapshotcopy ^
    --kms-key-id a1b2c3d4-1234-5678-wxyz-a1b2c3d4d5e6
```

RDS API

Vous pouvez copier un instantané de base de données personnalisé RDS pour SQL Server à l'aide de l'opération Copy de l'API Amazon RDS. DBSnapshot Si vous copiez l'instantané dans une nouvelle Région AWS, exécutez l'action dans cette nouvelle Région AWS. Les paramètres suivants sont utilisés pour copier un instantané de base de données. Tous les paramètres ne sont pas obligatoires :

- `SourceDBSnapshotIdentifier`- Identifiant de l'instantané de base de données source.
  - Si l'instantané source se trouve dans un emplacement différent Région AWS de celui de la copie, spécifiez un ARN d'instantané de base de données valide. Par exemple, `arn:aws:rds:us-west-2:123456789012:snapshot:instance1-snapshot-12345678`.
- `TargetDBSnapshotIdentifier`- Identifiant de la nouvelle copie de l'instantané de base de données.
- `KmsKeyId`- Identifiant de clé KMS pour un instantané de base de données chiffré. L'identificateur de clé KMS est l'ARN (Amazon Resource Name), l'identificateur de clé ou l'alias de clé pour la clé KMS.
  - Si vous copiez un instantané chiffré vers une autre Région AWS, vous devez spécifier une clé KMS pour la Région AWS de destination. Les clés KMS sont spécifiques à l'endroit dans Région AWS lequel elles ont été créées et vous ne pouvez pas utiliser les clés de chiffrement les unes Région AWS des autres, Région AWS sauf si une clé multirégionale est utilisée. Pour plus d'informations sur les clés KMS multi-Régions, veuillez consulter la section Utilisation de clés multi-Régions dans AWS KMS.
- `CopyTags`- Définissez ce paramètre sur `true` pour copier les balises et les valeurs de l'instantané source vers la copie de l'instantané. L'argument par défaut est `false`.

Les options suivantes ne sont pas prises en charge pour copier un instantané de base de données RDS Custom pour SQL Server :

- `CopyOptionGroup`
- `OptionGroupName`
- `PreSignedUrl`
- `TargetCustomAvailabilityZone`

Le code suivant crée une copie d'instantané, avec le nouveau nom `mydbsnapshotcopy`, dans la région US East (N. Virginia).

```
https://rds.us-east-1.amazonaws.com/
    ?Action=CopyDBSnapshot
    &KmsKeyId=a1b2c3d4-1234-5678-wxyz-a1b2c3d4d5e6
    &SourceDBSnapshotIdentifier=arn%3Aaws%3Ards%3Aus-
west-2%3A123456789012%3Asnapshot%3Ainstance1-snapshot-12345678
    &TargetDBSnapshotIdentifier=mydbsnapshotcopy
    &Version=2014-10-31
    &X-Amz-Algorithm=AWS4-HMAC-SHA256
    &X-Amz-Credential=AKIADQKE4SARGYLE/20161117/us-east-1/rds/aws4_request
    &X-Amz-Date=20161117T221704Z
    &X-Amz-SignedHeaders=content-type;host;user-agent;x-amz-content-sha256;x-amz-
date
    &X-Amz-
Signature=da4f2da66739d2e722c85fcfd225dc27bba7e2b8dbea8d8612434378e52adccf
```

Migration d'une base de données sur site vers Amazon RDS Custom for SQL Server

Vous pouvez utiliser le processus suivant pour migrer une base de données Microsoft SQL Server sur site vers Amazon RDS Custom for SQL Server à l'aide de la sauvegarde et de la restauration natives :

1. Effectuez une sauvegarde complète de la base de données sur l'instance de base de données sur site.
2. Chargez le fichier de sauvegarde sur Amazon S3.
3. Téléchargez le fichier de sauvegarde depuis S3 dans votre instance de base de données RDS Custom for SQL Server.
4. Restaurez une base de données à l'aide du fichier de sauvegarde téléchargé sur l'instance de base de données RDS Custom for SQL Server.

Ce processus explique la migration d'une base de données sur site vers RDS Custom for SQL Server en utilisant la sauvegarde et la restauration complètes natives. Pour réduire le temps de basculement pendant le processus de migration, vous pouvez également envisager d'utiliser des sauvegardes différentielles ou de journaux.

Pour obtenir des informations générales sur la sauvegarde et la restauration natives pour RDS for SQL Server, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Rubriques

- Prérequis
- Sauvegarde en cours de la base de données sur site
- Chargement du fichier de sauvegarde sur Amazon S3.
- Téléchargement du fichier de sauvegarde à partir d'Amazon S3
- Restauration du fichier de sauvegarde sur l'instance de base de données RDS Custom for SQL Server

Prérequis

Avant de migrer la base de données, effectuez les tâches suivantes :

1. Configurez la connexion Bureau à distance (RDP) pour votre instance de base de données RDS Custom for SQL Server. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données RDS personnalisée à l'aide de RDP.
2. Configurez l'accès à Amazon S3 afin de pouvoir charger et télécharger le fichier de sauvegarde de la base de données.

Sauvegarde en cours de la base de données sur site

Vous utilisez la sauvegarde native de SQL Server pour effectuer une sauvegarde complète de la base de données sur l'instance de base de données sur site.

L'exemple suivant montre une sauvegarde d'une base de données appelée `mydatabase`, avec l'option `COMPRESSION` spécifiée pour réduire la taille du fichier de sauvegarde.

Pour sauvegarder la base de données sur site

1. À l'aide de SQL Server Management Studio (SSMS), connectez-vous à l'instance SQL Server sur site.
2. Exécutez la commande T-SQL suivante :

```
backup database mydatabase to
disk ='C:\Program Files\Microsoft SQL Server\MSSQL13.MSSQLSERVER\MSSQL\Backup\mydb-full-compressed.bak'
with compression;
```

Chargement du fichier de sauvegarde sur Amazon S3.

Vous utilisez le AWS Management Console pour télécharger le fichier `mydb-full-compressed.bak` de sauvegarde sur Amazon S3.

Chargez le fichier de sauvegarde sur S3

1. Connectez-vous à la console Amazon S3 AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/s3/.
2. Sous Compartiments, sélectionnez le nom du compartiment sur lequel vous souhaitez charger le fichier.
3. Sélectionnez Charger.

4. Dans la fenêtre Charger, procédez de l'une des manières suivantes :

   - Faites glisser `mydb-full-compressed.bak` dans la fenêtre Charger.
   - Sélectionnez Ajouter un fichier, `mydb-full-compressed.bak`, puis Ouvrir.

   Amazon S3 charge votre fichier de sauvegarde en tant qu'objet S3. Lorsque le chargement est terminé, un message de succès s'affiche sur la page Load: status (Charger : statut).

Téléchargement du fichier de sauvegarde à partir d'Amazon S3

Vous utilisez la console pour télécharger le fichier de sauvegarde depuis S3 dans votre instance de base de données RDS Custom for SQL Server.

Pour télécharger le fichier de sauvegarde à partir de S3

1. Connectez-vous à votre instance de base de données RDS Custom for SQL Server à l'aide de RDP.
2. Connectez-vous à la console Amazon S3 AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/s3/.
3. Dans la liste Compartiments, choisissez le nom du compartiment qui contient votre fichier de sauvegarde.
4. Sélectionnez le fichier de sauvegarde `mydb-full-compressed.bak`.
5. Dans Actions, sélectionnez Télécharger en tant que.
6. Ouvrez le menu contextuel (clic droit) pour le lien fourni, puis choisissez Enregistrer sous.
7. Enregistrez `mydb-full-compressed.bak` dans le répertoire `D:\rdsdbdata\BACKUP`.

Restauration du fichier de sauvegarde sur l'instance de base de données RDS Custom for SQL Server

Vous utilisez la restauration native SQL Server pour restaurer le fichier de sauvegarde sur votre instance de base de données RDS Custom for SQL Server.

Dans cet exemple, l'option `MOVE` est spécifiée, car les répertoires de données et de fichiers journaux sont différents de l'instance de base de données sur site.

Pour restaurer le fichier de sauvegarde

1. Connectez-vous à votre instance de base de données RDS Custom for SQL Server à l'aide de SSMS.
2. Exécutez la commande T-SQL suivante :

```
restore database mydatabase from disk='D:\rdsdbdata\BACKUP\mydb-full-
compressed.bak'
with move 'mydatabase' to 'D:\rdsdbdata\DATA\mydatabase.mdf',
move 'mydatabase_log' to 'D:\rdsdbdata\DATA\mydatabase_log.ldf';
```

Mise à niveau d'une instance de base de données Amazon RDS Custom pour SQL Server

Vous pouvez mettre à niveau une instance de base de données Amazon RDS Custom for SQL Server en la modifiant pour utiliser une nouvelle version du moteur de base de données. Pour obtenir des informations générales sur la mise à niveau des instances de base de données, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Rubriques

- Vue d'ensemble des mises à niveau dans RDS Custom pour SQL Server
- Mise à niveau des versions majeures et mineures du moteur
- Niveau de compatibilité de base de données

Vue d'ensemble des mises à niveau dans RDS Custom pour SQL Server

Amazon RDS Custom pour SQL Server prend en charge les mises à niveau des versions majeures et mineures. Les mises à niveau des versions mineures peuvent inclure des correctifs de sécurité, des corrections de bogues et des améliorations du moteur. Microsoft publie ces mises à jour sous forme de mises à jour cumulatives (CUs). Les mises à niveau de versions majeures introduisent de nouvelles fonctionnalités et des modifications du moteur entre les versions, comme la mise à niveau de SQL Server 2019 vers 2022. Vous pouvez appliquer les deux mises à niveau immédiatement ou pendant les fenêtres de maintenance planifiées. Pour éviter d'éventuels problèmes de rétrocompatibilité, nous vous recommandons de tester vos applications dans un environnement hors production avant de procéder à la mise à niveau.

RDS Custom pour SQL Server vous permet de mettre à niveau une version de moteur fournie par RDS (RPEV) ou une version de moteur personnalisée (CEV).

- Les versions du moteur fournies par RDS (RPEV) contiennent des correctifs du système up-to-date d'exploitation (OS) et des mises à jour cumulatives (CU) de SQL Server.
- Pour une version de moteur personnalisée (CEV), vous devez suivre un processus en deux étapes. Créez d'abord un nouveau CEV avec votre version cible de SQL Server, voirPréparation à la création d'une version CEV pour RDS Custom pour SQL Server. Cette version cible doit être égale ou plus récente que votre version actuelle. Une fois le nouveau CEV créé, modifiez votre instance de base de données pour utiliser cette nouvelle version. Pour plus d'informations, consultez Exécution d'une mise à niveau de version mineure pour Amazon RDS Custom pour SQL Server CEV avec Multi-AZ.

N'appliquez pas de mises à jour cumulatives de SQL Server sur place à votre instance personnalisée RDS en cours d'exécution. Une fois que vous avez créé un CEV avec une version spécifique de SQL Server (par exemple, SQL Server 2022 CU16), l'application d'une nouvelle mise à jour cumulative directement à l'instance la fait sortir du périmètre de support et signale une erreur`SP-S3006`. Pour appliquer un correctif à une instance SQL Server existante à l'aide d'un CEV, créez un nouveau CEV incluant la mise à jour cumulative souhaitée, puis modifiez votre instance existante pour passer au nouveau CEV.

Si vous mettez à niveau une instance de base de données RDS Custom pour SQL Server dans le cadre d'un déploiement multi-AZ, RDS Custom for SQL Server effectue des mises à niveau continues pour votre instance. Cette approche minimise les temps d'arrêt en mettant à niveau une instance à la fois. RDS exécute les actions suivantes pour effectuer des mises à niveau progressives :

1. Mettez à niveau l'instance de base de données de secours.
2. Basculez vers l'instance de base de données de secours mise à niveau, ce qui en fait la nouvelle instance de base de données principale.
3. Mettez à niveau la nouvelle instance de base de données de secours.

Le temps d'indisponibilité de l'instance de base de données pour les déploiements multi-AZ correspond au temps nécessaire au basculement.

Les limitations suivantes s'appliquent lors de la mise à niveau d'une instance de base de données RDS Custom pour SQL Server :

- Les options de base de données personnalisées et les groupes de paramètres ne sont pas pris en charge.
- Les volumes de stockage supplémentaires que vous attachez à votre instance de base de données RDS Custom for SQL Server ne sont pas attachés après la mise à niveau.
- En effet CEVs, l'application sur place des mises à jour cumulatives de SQL Server n'est pas prise en charge et entraîne le retrait de l'instance du périmètre de support.

Mise à niveau des versions majeures et mineures du moteur

Les mises à niveau majeures et mineures du moteur sont irréversibles et doivent toujours être effectuées vers une version plus récente. Pour identifier les versions cibles disponibles, utilisez

AWS Management Console et choisissez parmi les versions disponibles lors de la modification de votre instance de base de données. Vous pouvez également utiliser la commande `describe-db-engine-versions`CLI ou la commande API Describe DBEngine Versions RDS.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-engine-versions \
    --engine custom-sqlserver-se \
    --engine-version 15.00.4322.2.v1 \
    --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" \
    --output table
```

Dans Windows :

```
aws rds describe-db-engine-versions ^
    --engine custom-sqlserver-se ^
    --engine-version 15.00.4322.2.v1 ^
    --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" ^
    --output table
```

Le résultat indique les versions du moteur cible disponibles :

```
--------------------------
|DescribeDBEngineVersions|
+------------------------+
|      EngineVersion     |
+------------------------+
|  15.00.4410.1.v1       |
|  15.00.4415.2.v1       |
|  15.00.4430.1.v1       |
|  16.00.4165.4.v1       |
|  16.00.4175.1.v1       |
|  16.00.4185.3.v1       |
+------------------------+
```

Après avoir identifié votre version cible, utilisez le AWS Management Console et suivez les instructions figurant dansModification d'une instance de base de données RDS personnalisée pour le SQL serveur. Vous pouvez également utiliser la commande `modify-db-instance`CLI ou la commande DBInstanceModify RDS API.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier DB_INSTANCE_IDENTIFIER \
    --engine-version ENGINE_VERSION \
    --allow-major-version-upgrade \
    --region Region \
    --no-apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier DB_INSTANCE_IDENTIFIER ^
    --engine-version ENGINE_VERSION ^
    --allow-major-version-upgrade ^
    --region Region ^
    --no-apply-immediately
```

> **Note**
>
> Vous devez inclure le `--allow-major-version-upgrade` paramètre pour effectuer des mises à niveau de version majeures.

Niveau de compatibilité de base de données

Vous pouvez utiliser les niveaux de compatibilité de base de données Microsoft SQL Server afin de régler certains comportements de base de données pour imiter les versions précédentes de SQL Server. Pour de plus amples informations, veuillez consulter Niveau de compatibilité dans la documentation de Microsoft.

Lorsque vous mettez à niveau votre instance de base de données, toutes les bases de données existantes restent à leur niveau de compatibilité initial. Par exemple, si vous effectuez une mise à niveau de SQL Server 2019 vers SQL Server 2022, le niveau de compatibilité de toutes les bases de données existantes est de 150. Toute nouvelle base de données créée après la mise à niveau possède le niveau de compatibilité 160.

Vous pouvez modifier le niveau de compatibilité d'une base de données en utilisant la commande ALTER DATABASE. Par exemple, pour modifier une base de données nommée `customeracct` afin qu'elle soit compatible avec SQL Server 2022, exécutez la commande suivante :

```
ALTER DATABASE customeracct SET COMPATIBILITY_LEVEL = 160
```

Résolution des problèmes de base de données pour Amazon RDS Custom for SQL Server

Le modèle de responsabilité partagée de RDS Custom fournit un accès au niveau du shell du système d'exploitation et un accès administrateur de base de données. RDS Custom exécute les ressources de votre compte, contrairement à Amazon RDS qui exécute les ressources d'un compte système. Un meilleur accès s'accompagne de responsabilités plus importantes. Dans les sections suivantes, vous apprendrez à résoudre les problèmes liés aux instances de base de données Amazon RDS Custom for SQL Server.

> **Note**
>
> Cette section explique comment résoudre les problèmes de RDS Custom for SQL Server. Pour la résolution des problèmes liés à RDS Custom for Oracle, consultez Résolution des problèmes de base de données pour Amazon RDS Custom for Oracle.

Rubriques

- Affichage des événements RDS Custom
- Abonnement aux événements personnalisés RDS
- Résolution des erreurs de version CEV pour RDS Custom for SQL Server
- Correction des configurations non prises en charge dans RDS Custom for SQL Server
- Résolution des problèmes Storage-Full dans RDS Custom pour SQL Server
- Résolution des problèmes liés à l'état PENDING_RECOVERY pour les bases de données compatibles TDE dans RDS Custom for SQL Server

Affichage des événements RDS Custom

La procédure d'affichage est la même pour les instances de base de données RDS Custom et Amazon RDS. Pour de plus amples informations, veuillez consulter Consulter les RDS événements Amazon.

Pour afficher la notification d'événement personnalisée RDS à l'aide de AWS CLI, utilisez la `describe-events` commande. RDS Custom s'accompagne de plusieurs nouveaux événements. Les catégories d'événements sont les mêmes que pour Amazon RDS. Pour obtenir la liste des événements, consultez Catégories d'événements Amazon RDS et messages d'événements .

L'exemple suivant récupère les détails des événements qui se sont produits pour l'instance de base de données RDS Custom spécifiée.

```
aws rds describe-events \
    --source-identifier my-custom-instance \
    --source-type db-instance
```

Abonnement aux événements personnalisés RDS

La procédure d'abonnement à des événements est la même pour les instances de base de données RDS Custom et Amazon RDS. Pour de plus amples informations, veuillez consulter S'abonner aux notifications d'RDSévénements Amazon.

Pour vous abonner à la notification d'événements RDS Custom à l'aide de l'interface de ligne de commande, utilisez la commande `create-event-subscription`. Incluez les paramètres requis suivants :

- `--subscription-name`
- `--sns-topic-arn`

L'exemple suivant montre comment créer un abonnement pour les événements de sauvegarde et de restauration d'une instance de base de données RDS Custom dans le compte AWS actuel. Les notifications sont envoyées à une rubrique Amazon Simple Notification Service (Amazon SNS) spécifiée par `--sns-topic-arn`.

```
aws rds create-event-subscription \
    --subscription-name my-instance-events \
    --source-type db-instance \
    --event-categories '["backup","recovery"]' \
    --sns-topic-arn arn:aws:sns:us-east-1:123456789012:interesting-events
```

Résolution des erreurs de version CEV pour RDS Custom for SQL Server

Lorsque vous essayez de créer une version CEV, il se peut que cela échoue. Dans ce cas, RDS Custom émet le message d'événement `RDS-EVENT-0198`. Pour plus d'informations sur l'affichage des événements RDS, consultez Catégories d'événements Amazon RDS et messages d'événements .

Utilisez les informations suivantes pour vous aider à traiter les causes possibles.

Message	Suggestions de dépannage
`Custom Engine Version creation expected a Sysprep'd AMI. Retry creation using a Sysprep'd AMI.`	Exécutez Sysprep sur l' EC2 instance que vous avez créée à partir de l'AMI. Pour plus d'informations sur la préparation d'une AMI à l'aide de Sysprep, consultez Créer une Amazon Machine Image (AMI) standardisée à l'aide de Sysprep.
`EC2 Image permissions for image (AMI_ID) weren't found for customer (Customer_ID). Verify customer (Customer_ID) has valid permissions on the EC2 Image.`	Vérifiez que le compte et le profil utilisés pour la création disposent des autorisations requises sur `create EC2 Instance` et `Describe Images` pour l'AMI sélectionnée.
`Failed to rebuild databases with server collation (collation name) due to missing setup.exe file for SQL Server.`	Vérifiez que le fichier `setup` est disponible à l'emplacement `C:\Program Files\Microsoft SQL Server\nnn\Setup Bootstrap\SQLnnnn\setup.exe` .
`Image (AMI_ID) doesn't exist in your account (ACCOUNT_ID). Verify (ACCOUNT_ID) is the owner of the EC2 image.`	Assurez-vous que l'AMI existe dans le même compte client.
`Image id (AMI_ID) isn't valid. Specify a valid image id, and try again.`	Le nom de l'AMI est incorrect. Assurez-vous que l'ID d'AMI correct est fourni.
`Image (AMI_ID) operating system platform isn't`	Choisissez une AMI prise en charge dotée de Windows Server avec SQL Server édition Enterprise,

Message	Suggestions de dépannage		
`supported. Specify a valid image, and try again.`	Standard ou Web. Choisissez une AMI avec l'un des codes d'opérati on d'utilisation suivants EC2 sur le Marketplace : • RunInstances:0102 - Windows avec SQL Server Enterprise • RunInstances:0006 - Windows avec SQL Server Standard • RunInstances:0202 - Windows avec SQL Server Web		
`SQL Server Web Edition isn't supported for creating a Custom Engine Version using Bring Your Own Media. Specify a valid image, and try again.`	Utilisez une image AMI qui contient une édition prise en charge de SQL Server. Pour de plus amples informations, veuillez consulter Support de version pour RDS Custom pour SQL Server CEVs.		
`The custom engine version can't be the same as the OEV engine version. Specify a valid CEV, and try again.`	Les versions classiques du moteur RDS Custom for SQL Server ne sont pas prises en charge. Par exemple, la version 15.00.407 3.23.v1. Utilisez un numéro de version pris en charge.		
`The custom engine version isn't in an active state. Specify a valid CEV, and try again.`	La version CEV doit être dans un état `AVAILABLE` pour terminer l'opération. Modifiez la version CEV de `INACTIVE` à `AVAILABLE` .		

Message	Suggestions de dépannage
`The custom engine version isn't valid for an upgrade. Specify a valid CEV with an engine version greater or equal to (X), and try again.`	La version CEV cible n'est pas valide. Vérifiez les exigences relatives à un chemin de mise à niveau valide.
`The custom engine version isn't valid. Names can include only lowercase letters (a-z), dashes (-), underscores (_), and periods (.). Specify a valid CEV, and try again.`	Respectez la convention de dénomination de version CEV requise. Pour de plus amples informations, veuillez consulter Exigences relatives à RDS Custom pour SQL Server CEVs.
`The custom engine version isn't valid. Specify valid database engine version, and try again. Example: 15.00.4073.23-cev123.`	Une version de moteur de base de données non prise en charge a été fournie. Utilisez une version de moteur de base de données prise en charge.
`The expected architecture is (X) for image (AMI_ID), but architecture (Y) was found.`	Utilisez une AMI basée sur l'architecture x86_64.
`The expected owner of image (AMI_ID) is customer account ID (ACCOUNT_ID), but owner (ACCOUNT_ID) was found.`	Créez l' EC2 instance à partir de l'AMI pour laquelle vous êtes autorisé. Exécutez Sysprep sur l' EC2instance pour créer et enregistrer une image de base.
`The expected platform is (X) for image (AMI_ID), but platform (Y) was found.`	Utilisez une AMI créée avec la plateforme Windows.

Message	Suggestions de dépannage
`The expected root device type is (X) for image %s, but root device type (Y) was found.`	Créez l'AMI avec le type de périphérique EBS.
`The expected SQL Server edition is (X), but (Y) was found.`	Choisissez une AMI prise en charge dotée de Windows Server avec SQL Server édition Enterprise, Standard ou Web. Choisissez une AMI avec l'un des codes d'opérati on d'utilisation suivants EC2 sur le Marketplace : • RunInstances:0102 - Windows avec SQL Server Enterprise • RunInstances:0006 - Windows avec SQL Server Standard • RunInstances:0202 - Windows avec SQL Server Web
`The expected state is (X) for image (AMI_ID), but the following state was found: (Y).`	Assurez-vous que l'AMI est dans l'état `AVAILABLE` .
`The provided Windows OS name (X) isn’t valid. Make sure the OS is one of the following: (Y).`	Utilisez un système d'exploitation Windows pris en charge.

Message	Suggestions de dépannage		
`Unable to find bootstrap log file in path.`	Vérifiez que le fichier journal est disponible à l'emplacement `C:\Program Files\Microsoft SQL Server\nnn\Setup Bootstrap\Log\Summary.txt` .		
`RDS expected a Windows build version greater than or equal to (X), but found version (Y)..`	Utilisez une AMI avec une version de build de système d'exploitation minimale de 14393.		
`RDS expected a Windows major version greater than or equal to (X), but found version (Y)..`	Utilisez une AMI avec une version majeure de système d'exploitation minimale de 10.0 ou supérieure.		

Correction des configurations non prises en charge dans RDS Custom for SQL Server

En raison du modèle de responsabilité partagée, il vous incombe de corriger les problèmes de configuration qui redonnent à votre instance de base de données RDS Custom for SQL Server le statut `unsupported-configuration`. Si le problème est lié à l' AWS infrastructure, vous pouvez utiliser la console ou le AWS CLI pour le résoudre. Si le problème concerne le système d'exploitation ou la configuration de la base de données, vous pouvez vous connecter à l'hôte pour le résoudre.

> **Note**
>
> Cette section explique comment corriger les configurations non prises en charge dans RDS Custom for SQL Server. Pour obtenir des informations sur RDS Custom for Oracle, consultez Correction des configurations non prises en charge dans RDS Custom for Oracle.

Dans les tableaux suivants, vous trouverez des descriptions des notifications et des événements envoyés par le périmètre de support et de la manière de les corriger. Ces notifications et le périmètre de prise en charge sont susceptibles d'être modifiés. Pour en savoir plus sur le périmètre de prise en

charge, consultez Périmètre de prise en charge RDS Custom. Pour les descriptions des événements, consultez Catégories d'événements Amazon RDS et messages d'événements .

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S0000	Configuration manuelle non prise en charge	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes :. X	Pour résoudre ce problème, créez un dossier d'assistance.

AWS ressource (infrastructure)

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S1001	EC2 État de l'instance	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : L' EC2 instance sous-jacente %s a été arrêtée sans arrêter l'instance RDS. Vous pouvez résoudre ce problème en démarrant l' EC2 instance sous-	Pour vérifier l'état d'une instance de base de données, utilisez la console ou exécutez la AWS CLI commande suivante :
			`aws rds describe-db-instances \`
			`--db-instance-identifier db-instance-name |grep DBInstanceStatus`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
		jacente et en vous assurant que les volumes binaires et de données sont attachés. Si vous avez l'intention d'arrêter l'instance RDS, assurez-vous d'abord que l' EC2 instance sous-jacente est dans l'état DISPONIBLE, puis utilisez la console RDS ou la CLI pour arrêter l'instance RDS.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S1002	EC2 État de l'instance	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : le statut de l'instance de base de données RDS est défini sur STOPPED mais l' EC2 instance sous-jacente %s a été démarrée. Vous pouvez résoudre ce problème en arrêtant l' EC2instance sous-jacente. Si vous avez l'intention de démarrer l'instance RDS, utilisez la console ou la CLI.	Utilisez la AWS CLI commande suivante pour vérifier l'état d'une instance de base de données :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep DBInstanceStatus`
			Vous pouvez également vérifier l'état de l' EC2 instance à l'aide de la EC2 console.
			Pour démarrer une instance de base de données, utilisez la console ou exécutez la AWS CLI commande suivante :
			`aws rds start-db-instance \ --db-instance-identifier db-instance-name`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1003	EC2 Classe d'instance	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : il existe une incompatibilité entre la classe d'instance de base de données attendue et la classe d'instance de base de données configurée de l'hôte. EC2 Vous pouvez résoudre ce problème en rétablissant le type de classe d'origine de la classe d'instance de base de données.	Utilisez la commande CLI suivante pour vérifier la classe d'instance de base de données attendue : `aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep DBInstanceClass`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1004	Volume de stockage EBS non accessible	L'état de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : le volume de stockage EBS %s d'origine associé à l' EC2 instance n'est actuellement pas accessible.	
SP-S1005	Volume de stockage EBS détaché	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : le volume de stockage EBS « volume-id » d'origine n'est pas attaché. Vous pouvez résoudre ce problème en attachant le volume EBS associé à l' EC2 instance.	Après avoir reconnecté le volume EBS, utilisez les commandes CLI suivantes pour vérifier si le volume EBS « volume-id » est correctement attaché à l'instance RDS : `aws ec2 describe-volumes \ --volume-ids volume-id | grep InstanceId`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S1006	Taille du volume de stockage EBS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : il existe une incompatibilité entre les paramètres attendus et configurés du volume de stockage EBS « volume-id ». La taille du volume a été modifiée manuellement par rapport à EC2 sa ou ses valeurs d'origine de [%s]. Pour résoudre ce problème, créez un dossier d'assistance.	Utilisez la commande CLI suivante pour comparer la taille du volume EBS « volume-id » avec les détails de l'instance RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep AllocatedStorage`
			Utilisez la commande CLI suivante pour afficher la taille réelle du volume alloué :
			`aws ec2 describe-volumes \ --volume-ids |grep Size`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S1007	Configuration du volume de stockage EBS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : il existe une incompatibilité entre les paramètres attendus et configurés du volume de stockage EBS « volume-id ». Vous pouvez résoudre ce problème en modifiant la configuration du volume de stockage EBS [IOPS, débit, type de volume] à sa ou ses valeurs d'origine de [IOPS : %s, débit : %s, type de volume : %s] au niveau. EC2 Pour les futures modifications du stockage, utilisez la console RDS ou la CLI. La taille du volume a également été	Utilisez la commande CLI suivante pour comparer le type de volume des détails « volume-id » du volume EBS et ceux de l'instance RDS. Assurez-vous que les valeurs au niveau EBS correspondent aux valeurs au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep StorageType`
			Pour obtenir la valeur attendue pour le débit de stockage au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep StorageThroughput`
			Pour obtenir la valeur attendue pour le volume IOPS au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep Iops`
			Pour obtenir le type de stockage actuel au EC2 niveau :

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

		modifiée manuellement par rapport à EC2 sa ou ses valeurs d'origine de [%s]. Pour résoudre ce problème, créez un dossier d'assistance.	`aws ec2 describe-volumes \ --volume-ids |grep VolumeType`
			
			Pour obtenir la valeur actuelle du débit de stockage au EC2 niveau :
			
			`aws ec2 describe-volumes \ --volume-ids |grep Throughput`
			
			Pour obtenir la valeur actuelle du volume IOPS au EC2 niveau :
			
			`aws ec2 describe-volumes \ --volume-ids |grep Iops`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S1008	Taille et configuration du volume de stockage EBS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : il existe une incompatibilité entre les paramètres attendus et configurés du volume de stockage EBS « volume-id ». Vous pouvez résoudre ce problème en modifiant la configuration du volume de stockage EBS [IOPS, débit, type de volume] à sa ou ses valeurs d'origine de [IOPS : %s, débit : %s, type de volume : %s] au niveau. EC2 Pour les futures modifications du stockage, utilisez la console RDS ou la CLI. La taille du volume a également été	Utilisez la commande CLI suivante pour comparer le type de volume des détails « volume-id » du volume EBS et ceux de l'instance RDS. Assurez-vous que les valeurs au niveau EBS correspondent aux valeurs au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep StorageType`
			Pour obtenir la valeur attendue pour le débit de stockage au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep StorageThroughput`
			Pour obtenir la valeur attendue pour le volume IOPS au niveau RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep Iops`
			Pour obtenir le type de stockage actuel au EC2 niveau :

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

		modifiée manuellement par rapport à EC2 sa ou ses valeurs d'origine de [%s]. Pour résoudre ce problème, créez un dossier d'assistance.	`aws ec2 describe-volumes \ --volume-ids |grep VolumeType`
			
			Pour obtenir la valeur actuelle du débit de stockage au EC2 niveau :
			
			`aws ec2 describe-volumes \ --volume-ids |grep Throughput`
			
			Pour obtenir la valeur actuelle du volume IOPS au EC2 niveau :
			
			`aws ec2 describe-volumes \ --volume-ids |grep Iops`
			
			Pour obtenir la taille de volume allouée attendue :
			
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name | grep AllocatedStorage`
			
			Pour obtenir la taille réelle du volume alloué :
			
			`aws ec2 describe-volumes \ --volume-ids |grep Size`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1009	Autorisations SQS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : les autorisations Amazon Simple Queue Service (SQS) sont manquantes pour le profil d'instance IAM. Vous pouvez résoudre ce problème en vous assurant que le profil IAM associé à l'hôte dispose des autorisations suivantes : ["SQS : «, "SQS : SendMessage «, "SQS : ReceiveMessage «, "SQS : DeleteMessage «]. GetQueueUrl	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1010	Point de terminaison VPC SQS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : une politique de point de terminaison VPC bloque les opérations Amazon Simple Queue Service (SQS). Vous pouvez résoudre ce problème en modifiant la politique de point de terminaison de votre VPC afin d'autoriser les actions SQS requises.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1011	Politique relative aux bus pour les événements	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : La politique basée sur les ressources pour votre bus d'événements arn:aws:events:region- 1:123456789012:event-bus/default refuse Amazon events : actions. CloudWatch PutEvents Résolvez ce problème en modifiant votre politique basée sur les ressources afin d'autoriser les événements : PutEvents actions pour EventBus %s.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1012	CloudWatch Autorisations VPC	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : une politique de point de terminaison VPC ne permet pas d'accéder aux événements Amazon. CloudWatch Résolvez ce problème en modifiant la politique de point de terminaison de votre VPC pour autoriser les événements : PutEvents on EventBus arn:aws:events:region- 1:123456789012:event-bus/default.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1013	Politique de contrôle des services	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : une politique de contrôle des services de votre partenaire AWS Organizations permet pas d'accéder aux événements Amazon CloudWatch . Résolvez ce problème en modifiant votre politique de contrôle des services pour autoriser les événements : PutEvents on EventBus arn:aws:events:region- 1:123456789012:event-bus/default.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1014	Profil d'instance IAM	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : les autorisations %s de votre profil d'instance IAM refusent les événements Amazon. CloudWatch Résolvez ce problème en réglant ["events : PutEvents «] sur « Autoriser » et en autorisant les événements : PutEvents on EventBus arn:aws:events:region- 1:123456789012:event-bus/default dans votre profil IAM associé à l'instance.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1015	Profil d'instance IAM	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : votre profil d'instance IAM %s ne dispose pas des autorisations d'événement Amazon CloudWatch . Résolvez ce problème en incluant les autorisations ["events : PutEvents «] et en autorisant les événements : PutEvents on EventBus arn:aws:events:region- 1:123456789012:event-bus/default dans votre profil IAM associé à l'instance.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S1016	Limite des autorisations IAM	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : votre profil d'instance IAM %s possède une limite d'autorisations qui refuse les événements Amazon. CloudWatch Résolvez ce problème en réglant ["events : PutEvents «] sur « Autoriser » pour EventBus arn:aws:events:region- 1:123456789012:event-bus/default dans la limite des autorisations de votre profil d'instance IAM.	

Système d’exploitation

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S2001	État du service SQL	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : Le service SQL Server n'est pas démarré. Vous pouvez résoudre ce problème en redémarrant le service SQL Server sur l'hôte. Si cette instance de base de données est une instance de base de données multi-AZ et que le redémarrage échoue, arrêtez et redémarrez l'hôte pour lancer un basculement.	
SP-S2002	État de l'agent personnalisé RDS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons	Connectez-vous à l'hôte et assurez-vous que l'agent RDS Custom est en cours d'exécution. Vous pouvez utiliser les commandes suivantes pour consulter le statut de l'agent.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

		suivantes : le service RDS Custom Agent n'est pas installé ou n'a pas pu être démarré. Vous pouvez résoudre ce problème en consultant le journal des événements Windows pour déterminer pourquoi le service ne démarre pas et en prenant les mesures appropriées pour résoudre le problème. Pour obtenir une assistance supplémentaire, créez un dossier d'assistance.	`$name = "RDSCustomAgent"`
			`$service = Get-Service $name`
			`Write-Host $service.Status`
			
			Si le statut n'est pas `Running`, vous pouvez démarrer le service avec la commande suivante :
			
			`Start-Service $name`
			
			Si l'agent ne peut pas démarrer, consultez les événements Windows pour savoir pourquoi il ne peut pas démarrer. L'agent a besoin d'un utilisateur Windows pour démarrer le service. Assurez-vous qu'un utilisateur Windows existe et dispose des privilèges nécessaires pour exécuter le service.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
`SP-S2003`	État de l'agent SSM	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : Le service Amazon SSM Agent est inaccessible. Vous pouvez résoudre ce problème en vérifiant l'état du service à l'aide de la `Get-Service AmazonSSMAgent` PowerShell commande ou en démarrant le service avec`Start-Service AmazonSSMAgent` . Assurez-vous que le trafic sortant HTTPS (port 443) vers les points de terminaison régionaux ssm, ssmmessages et ec2messages est autorisé.	Pour de plus amples informations, consultez la section Résolution des problèmes de SSM Agent. Pour résoudre les problèmes liés aux points de terminaison SSM, voir Impossible de se connecter aux points de terminaison SSM et Utiliser ssm-cli pour résoudre les problèmes de disponibilité des nœuds gérés.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S2004	Connexion à l'agent personnalisé RDS	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : Un problème inattendu s'est produit lors de la connexion SQL. "$HOSTNAME/RDSAgent" Pour résoudre ce problème, créez un dossier d'assistance.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S2005	Fuseau horaire	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : le fuseau horaire de l' EC2 instance Amazon [%s] a été modifié. Vous pouvez résoudre ce problème en modifiant le fuseau horaire pour revenir au paramètre spécifié lors de la création de l'instance. Si vous souhaitez créer une instance avec un fuseau horaire spécifique, consultez la documentation personnalisée RDS.	Exécutez la `Get-Timezone` PowerShell commande pour confirmer le fuseau horaire. Pour de plus amples informations, veuillez consulter Fuseau horaire local pour les instances de base de données RDS Custom for SQL Server.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S2006	Version de la solution logicielle à haute disponibilité	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : La solution logicielle de haute disponibilité de l'instance actuelle est différente de la version attendue. Pour résoudre ce problème, créez un dossier d'assistance.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S2007	Configuration de la solution logicielle à haute disponibilité	L'état de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : les paramètres de configuration de la solution logicielle de haute disponibilité ont été modifiés à des valeurs inattendues sur l'instance %s. Pour résoudre ce problème, redémarrez l'EC2instance. Lorsque vous redémarrez l'EC2 instance, elle met automatiquement à jour les paramètres selon la configuration requise pour la solution logicielle de haute disponibilité.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S2008	Service SQL Server	L'instance de base de données personnalisée RDS est définie sur [Configuration non prise en charge] : SQLServer (MSSQLServer) le service n'existe pas sur l'hôte. Pour résoudre ce problème, créez un dossier de support.	Vous pouvez utiliser les commandes suivantes pour consulter le statut de l'agent.
			`$name = "MSSQLServer"`
			`$service = Get-Service $name`
			`Write-Host $service.Status`

SP-2009	Certificat SSL	L'instance de base de données personnalisée RDS est définie sur [Configuration non prise en charge] pour les raisons suivantes : des certificats SSL non auto-signés perturbent le RDS. Pour résoudre ce problème, supprimez le ou les certificats non autosignés du magasin de certificats racine approuvé.	Exécutez la PowerShell commande suivante pour vérifier les certificats non auto-signés. `Get-ChildItem cert:\LocalMachine\root -Recurse | Where-Object {$_.Issuer -ne $_.Subject -and $_.Issuer -notlike "*RDSCustomAgentCA*"}` Pour plus d'informations, consultez l'erreur HTTP 403.16 lorsque vous essayez d'accéder à un site Web hébergé sur IIS.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-2010	État du stockage du volume racine	L'instance de base de données personnalisée RDS est définie sur [Configuration non prise en charge] pour les raisons suivantes : le stockage du volume racine est plein. Pour résoudre ce problème, libérez au moins 500 Mo d'espace de stockage dans le volume EBS racine « volume-id » ou augmentez la taille du volume et redimensionnez le lecteur C sur l'« instance-id » de l'instance. EC2 Les modifications de taille du volume racine ne sont pas conservées lorsque vous remplacez l'EC2 instance.	Utilisez la commande suivante pour afficher le stockage disponible sur le volume racine (C :). `(Get-PSDrive -Name C).Free / 1MB` Pour plus d'informations sur la modification du volume racine EBS, consultez Comment

Database (Base de données)

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S3001	Protocole de mémoire partagée SQL Server	L'état de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : Le protocole de mémoire partagée SQL Server est désactivé. Vous pouvez résoudre ce problème en activant le protocole de mémoire partagée dans le Gestionnaire de configuration de SQL Server.	Vous pouvez le valider en vérifiant : Gestionnaire de configuration SQL Server > Configuration réseau SQL Server > Protocoles pour MSSQLSERVER> Mémoire partagée activée. Après avoir activé le protocole, redémarrez le processus SQL Server.
SP-S3002	Clé principale du service	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : RDS Automation n'est pas en mesure de sauvegarder la clé principale de service (SMK) dans le	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
		cadre de la nouvelle génération de SMK. Pour résoudre ce problème, créez un dossier d'assistance.	
SP-S3003	Clé principale du service	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : les métadonnées relatives à la clé principale du service (SMK) sont manquantes ou incomplètes. Pour résoudre ce problème, créez un dossier d'assistance.	

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S3004	Version et édition du moteur de base de données	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : il existe une incompatibilité entre la version et l'édition de SQL Server attendues et installées : la modification de l'édition SQL Server n'est pas prise en charge sur RDS Custom for SQL Server. En outre, la modification manuelle de la version de SQL Server sur l' EC2 instance personnalisée RDS n'est pas prise en charge. Pour résoudre ce problème, créez un dossier d'assistance.	Exécutez la requête suivante pour obtenir la version SQL :
			`select @@version`
			Exécutez la AWS CLI commande suivante pour obtenir la version et l'édition du moteur SQL RDS :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep EngineVersion`
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep Engine`
			Pour plus d'informations, consultez Modification d'une instance de base de données RDS personnalisée pour le SQL serveur et Mise à niveau de la version du moteur d'une instance de base de données.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S3005	Édition DB Engine	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : L'édition actuelle de SQL Server ne correspond pas à l'édition prévue de SQL Server [%s] : la modification de l'édition de SQL Server n'est pas prise en charge sur RDS Custom for SQL Server. Pour résoudre ce problème, créez un dossier d'assistance.	Exécutez la requête suivante pour obtenir l'édition SQL :
			Example
			`select @@version`
			Exécutez la AWS CLI commande suivante pour obtenir l'édition du moteur RDS SQL :
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep Engine`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S3006	DB Engine Version	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : La version actuelle de SQL Server ne correspond pas à la version attendue de SQL Server [%s] : vous ne pouvez pas modifier manuellement la version de SQL Server sur l'instance personnalisée RDS. EC2 Pour résoudre ce problème, créez un dossier d'assistance. Pour toute modification future de la version de SQL Server, vous pouvez modifier l'instance depuis la console AWS RDS ou via la commande modify-db-instance CLI.	Exécutez la requête suivante pour obtenir la version SQL :
			
			Example
			
			`select @@version`
			
			Exécutez la AWS CLI commande suivante pour obtenir la version du moteur SQL RDS :
			
			`aws rds describe-db-instances \ --db-instance-identifier db-instance-name |grep EngineVersion`
			
			Pour plus d'informations, consultez Modification d'une instance de base de données RDS personnalisée pour le SQL serveur et Mise à niveau de la version du moteur d'une instance de base de données.

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation

SP-S3007	Emplacement du fichier de base de données	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : Les fichiers de base de données sont configurés en dehors du lecteur D : \. Vous pouvez résoudre ce problème en vous assurant que tous les fichiers de base de données, y compris ROW, LOG, FILESTREAM, etc... sont stockés sur le lecteur D : \.	Exécutez la requête suivante pour répertorier l'emplacement des fichiers de base de données qui ne figurent pas dans le chemin par défaut :
			`USE master;`
			`SELECT physical_name as files_not_in_default_path`
			`FROM sys.master_files`
			`WHERE SUBSTRING(physical_name,1,3)!='D:\';`

Code de l'événement	Zone de configuration	Message d'événement RDS	Processus de validation
SP-S3008	Nombre limite de base de données dépassé	Le statut de l'instance de base de données personnalisée RDS est défini sur [Configuration non prise en charge] pour les raisons suivantes : le nombre total de bases de données sur l'instance de base de données dépasse la limite maximale de 5 000. Pour résoudre ce problème, réduisez le nombre de bases de données en dessous de la limite maximale prise en charge.	Utilisez la commande suivante pour afficher le nombre total de bases de données : `SELECT COUNT(name) as databaseCount FROM sys.databases WHERE name not in ('tempdb','master','model','msdb','DWDiagnostics','DWConfiguration','DWQueue');`

Résolution des problèmes **`Storage-Full`** dans RDS Custom pour SQL Server

RDS Custom surveille également le volume racine (C :). L'instance de base de données RDS Custom for SQL Server passe à l'`unsupported-configuration`état où le volume racine dispose de moins de 500 Mo d'espace disque disponible. Consultez `Event SP-S2010` dans Correction des configurations non prises en charge dans RDS Custom for SQL Server.

Résolution des problèmes liés à l'état PENDING_RECOVERY pour les bases de données compatibles TDE dans RDS Custom for SQL Server

Les bases de données SQL Server sur lesquelles le chiffrement transparent des données (TDE) est activé peuvent rester en `PENDING_RECOVERY` état si le déchiffrement automatique rencontre des

problèmes. Cela se produit généralement après la restauration d'une instance de base de données si le fichier de sauvegarde SMK (Service Master Key) de l'instance de base de données source stocké dans le compartiment S3 géré par RDS Custom de votre compte a été supprimé avant la fin de la restauration.

Pour activer le déchiffrement automatique et mettre en ligne les bases de données compatibles TDE, vous devez ouvrir la clé principale de base de données (DMK) avec son mot de passe et crypter la DMK à l'aide du SMK.

Utilisez les commandes SQL Server suivantes à titre de référence :

```
-- Identify PENDING_RECOVERY TDE databases
USE MASTER;
GO
SELECT name, is_encrypted, state_desc FROM sys.databases;
GO

-- Open DMK using password
OPEN MASTER KEY DECRYPTION BY PASSWORD = '<password>';
GO

-- Encrypt DMK using SMK
ALTER MASTER KEY ADD ENCRYPTION BY SERVICE MASTER KEY;
GO

-- Close SMK
CLOSE MASTER KEY;
GO

-- Bring the TDE databases online
ALTER DATABASE <database_name> SET ONLINE;
GO

-- Verify TDE databases are now in ONLINE state
SELECT name, is_encrypted, state_desc FROM sys.databases;
GO
```

Amazon RDS sur AWS Outposts

Amazon RDS on AWS Outposts étend les bases de données RDS pour SQL Server, RDS pour MySQL et RDS pour PostgreSQL aux environnements Rack Gen1 et Gen2. AWS Outposts AWS Outposts utilise le même matériel que dans le secteur public Régions AWS pour apporter les AWS services, l'infrastructure et les modèles d'exploitation sur site. Avec RDS sur outposts, vous pouvez allouer des instances de base de données gérées à proximité des applications métier qui doivent être exécutées sur site. Pour plus d'informations AWS Outposts, consultez la AWS Outposts documentation et la page AWS Outposts du produit.

Vous utilisez la même AWS Management Console API RDS pour provisionner et gérer des instances de base de données RDS sur site sur Outposts comme vous le faites pour les instances de base de données RDS exécutées dans le. AWS CLI AWS Cloud RDS sur Outposts automatise des tâches telles que l'approvisionnement de base de données, la mise à jour corrective de système d'exploitation et de base de données, la sauvegarde et l'archivage à long terme dans Amazon S3.

RDS sur outposts prend en charge les sauvegardes automatiques des instances de bases de données. La connectivité réseau entre votre Outpost et votre Région AWS est nécessaire pour sauvegarder et restaurer les instances de base de données. Tous les instantanés de base de données et les journaux de transactions d'un Outpost sont stockés dans votre. Région AWSÀ partir de votre région AWS , vous pouvez restaurer une instance de base de données à partir d'un instantané de bases de données sur un autre Outpost. Pour plus d'informations, consultez Présentation des sauvegardes.

RDS sur outposts prend en charge la maintenance et les mises à niveau automatiques des instances de bases de données. Pour de plus amples informations, veuillez consulter Entretien d'une instance de base de données.

RDS sur Outposts utilise le chiffrement au repos pour les instances et les instantanés de base de données à l'aide de votre AWS KMS key. Pour plus d'informations sur le chiffrement au repos, veuillez consulter Chiffrement des ressources Amazon RDS.

Par défaut, EC2 les instances des sous-réseaux Outposts peuvent utiliser le service DNS Amazon Route 53 pour convertir les noms de domaine en adresses IP. Il se peut que vous rencontriez des temps de résolution DNS plus longs avec Route 53, en fonction de la latence du chemin entre votre Outpost et la Région AWS. Dans ce cas, vous pouvez utiliser les serveurs DNS installés localement dans votre environnement sur site. Pour plus d'informations, consultez DNS dans le AWS Outposts Guide de l'utilisateur d'.

Lorsque la connectivité réseau Région AWS n'est pas disponible, votre instance de base de données continue de s'exécuter localement. Vous pouvez continuer à accéder aux instances de base de données à l'aide de la résolution de noms DNS en configurant un serveur DNS local en tant que serveur secondaire. Cependant, vous ne pouvez pas créer de nouvelles instances de base de données ni modifier des instances de base de données existantes. Les sauvegardes automatiques n'ont pas lieu s'il n'y a pas de connectivité. En cas de défaillance d'une instance de base de données, celle-ci n'est pas automatiquement remplacée tant que la connectivité n'est pas restaurée. Nous vous recommandons de restaurer la connectivité réseau dès que possible.

Rubriques

- Conditions requises pour Amazon RDS sur AWS Outposts
- Amazon RDS parle du AWS Outposts support pour les RDS fonctionnalités Amazon
- Classes d'instances de base de données prises en charge pour Amazon RDS sur AWS Outposts
- Adresses IP appartenant au client pour Amazon RDS on AWS Outposts
- Utilisation des déploiements multi-AZ pour Amazon RDS on AWS Outposts
- Création d'instances de base de données pour Amazon RDS sur AWS Outposts
- Création de répliques de lecture pour Amazon RDS sur AWS Outposts
- Considérations pour la restauration d'instances de base de données sur Amazon RDS on AWS Outposts

Conditions requises pour Amazon RDS sur AWS Outposts

Les prérequis pour utiliser Amazon RDS sur AWS Outposts sont les suivants :

- Installez-le AWS Outposts dans votre centre de données sur site. Pour plus d'informations, consultez la section Installation d'un AWS Outposts serveur dans le guide d'installation AWS Outposts du serveur.
- Assurez-vous de disposer d'au moins un sous-réseau pour RDS sur outposts. Vous pouvez utiliser le même sous-réseau pour d'autres charges de travail.
- Assurez-vous de disposer d'une connexion réseau fiable entre votre Outpost et une région AWS .

Amazon RDS parle du AWS Outposts support pour les RDS fonctionnalités Amazon

Le tableau suivant décrit les RDS fonctionnalités Amazon prises en charge par Amazon RDS sur AWS Outposts.

Fonctionnalité	Pris en charge	Remarques	En savoir plus

Allocation d'instances de base de données	Oui	Vous ne pouvez créer des instances de base de données que RDS pour les moteurs de SQL base de données RDS for SQL ServerSQL, My et Postgre. RDS Les versions suivantes sont prises en charge :
		• Microsoft SQL Server :
		  • 15.00.4043.16.v1 et versions 2019 ultérieures
		  • 14.00.3294.2.v1 et versions 2017 ultérieures
		  • 13.00.5820.21.v1 et versions 2016 ultérieures
		• Mes SQL versions 8.0 et 8.4
		• Toutes les versions de Postgre SQL 16 et 15, 14 et 13, et les versions de Postgre SQL 12.5
			Création d'instances de base de données pour Amazon RDS sur AWS Outposts

Fonctionnalité	Pris en charge	Remarques	En savoir plus
		et supérieures de SQL Postgre 12	
Connectez-vous à une instance de base de données Microsoft SQL Server avec Microsoft SQL Server Management Studio	Oui	Certaines TLS versions et certains chiffrements peuvent ne pas être sécurisés. Pour les désactiver, suivez les instructions de la rubrique Configuration des protocoles de sécurité et des chiffrements du SQL serveur.	Connexion à votre instance de base de données Microsoft SQL Server
Modification du mot de passe utilisateur principal	Oui	Aucun	Modification d'une RDS instance de base de données Amazon
Affectation d'un nouveau nom à une instance DB	Oui	Aucun	Modification d'une RDS instance de base de données Amazon
Redémarrage d'une instance DB	Oui	Aucun	Redémarrage d'une instance de base de données
Arrêt d'une instance de base de données	Oui	Aucun	Arrêt temporaire d'une instance de bases de données Amazon RDS
Démarrage d'une instance de base de données	Oui	Aucun	Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Déploiements multi-AZ	Oui	Les déploiements multi-AZ sont pris en charge sur les instances de base de données My SQL et PostgreSQL. Les déploiements multi-AZ ne prennent pas en charge le VPC routage direct ()DVR.	Création d'instances de base de données pour Amazon RDS sur AWS Outposts Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS
Groupes de paramètres DB	Oui	Aucun	Groupes de paramètres pour (Amazon RDS)
Réplicas en lecture	Oui	Les répliques de lecture sont prises en charge pour les instances de base de données My SQL et PostgreSQL. Les répliques en lecture ne prennent pas en charge le VPC routage direct (DVR).	Création de répliques de lecture pour Amazon RDS sur AWS Outposts
Chiffrement au repos	Oui	RDSon Outposts ne prend pas en charge les instances de base de données non chiffrées.	Chiffrement des ressources Amazon RDS
AWS Identity and Access Management (IAM) authentification de base de données	Non	Aucun	Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Associer un IAM rôle à une instance de base de données	Non	Aucun	add-role-to-dbcommande - instance AWS CLI AddRoleToDBInstanceRDSAPIopération
Authentification Kerberos	Non	Aucun	Authentification Kerberos
Marquer les ressources Amazon RDS	Oui	Aucun	Marquage d' Amazon RDS
Groupes d'options	Oui	Aucun	Utilisation de groupes d'options
Modification de la fenêtre de maintenance	Oui	Aucun	Entretien d'une instance de base de données
Mise à niveau automatique de version mineure	Oui	Aucun	Mise à niveau automatique de la version mineure du moteur
Modification de la fenêtre de sauvegarde	Oui	Aucun	Présentation des sauvegardes Modification d'une RDS instance de base de données Amazon
Modification de la classe d'instance de base de données	Oui	Aucun	Modification d'une RDS instance de base de données Amazon

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Modification du stockage alloué	Oui	Aucun	Modification d'une RDS instance de base de données Amazon
Dimension nement automatique du stockage	Oui	Aucun	Gestion automatique de la capacité avec le dimension nement automatique du stockage Amazon RDS
Instantanés d'instance de base de données manuels et automatiques	Oui	Vous pouvez stocker des sauvegardes automatisées et des instantanés manuels dans votre Région AWS. Vous pouvez également les stocker en local sur votre Outpost. Les sauvegardes locales sont prises en charge sur les SQL instances de base de données My SQL et Postgre. Pour stocker des sauvegardes sur votre Outpost, assurez-vous que Amazon S3 on Outposts est configuré. Les sauvegardes locales ne sont pas prises en charge pour les déploieme nts d'instance multi-AZ.	Création d'instances de base de données pour Amazon RDS sur AWS Outposts Amazon S3 on Outposts Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Restauration à partir d'un instantané de base de données	Oui	Vous pouvez stocker des sauvegardes automatisées et des instantanés manuels pour l'instance de base de données restaurée dans la Région AWS parente ou localement sur votre Outpost.	Considérations pour la restauration d'instances de base de données sur Amazon RDS on AWS Outposts Restauration vers une instance de base de données
Restauration d'une instance de base de données à partir d'Amazon S3	Non	Aucun	Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL
Exportation de données d'instantané vers Amazon S3	Non	Aucun	Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS
Point-in-time rétablissement	Oui	Vous pouvez stocker des sauvegardes automatisées et des instantanés manuels pour l'instance de base de données restaurée dans la Région AWS parente ou localement sur votre Outpost, à une exception.	Considérations pour la restauration d'instances de base de données sur Amazon RDS on AWS Outposts Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Surveillance améliorée	Non	Aucun	Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée
CloudWatch Surveillance d'Amazon	Oui	Vous pouvez afficher le même ensemble de métriques que celles disponibles pour vos bases de données dans la Région AWS.	Surveillance des métriques Amazon RDS () avec Amazon CloudWatch
Publication des journaux du moteur de base de données dans CloudWatch Logs	Oui	Aucun	Publication des journaux de base de données sur Amazon CloudWatch Logs
Notification d'événement	Oui	Aucun	Utilisation des notifications d'RDSévénements Amazon
Amazon RDS Performance Insights	Non	Aucun	Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Affichage ou téléchargement des journaux de base de données	Non	RDSon Outposts ne permet pas d'afficher les journaux de base de données à l'aide de la console ou de décrire les journaux de base de données à l'aide du AWS CLI ou. RDS API RDSsur Outposts ne prend pas en charge le télécharg ement des journaux de base de données à l'aide de la console ou le téléchargement des journaux de base de données à l'aide du AWS CLI ou. RDS API	Surveillance des fichiers journaux RDSAmazon
RDSProxy Amazon	Non	Aucun	Proxy Amazon RDS
Procédures enregistrées pour Amazon RDS for My SQL	Oui	Aucun	RDSpour Ma référence de procédure SQL stockée
Réplication avec des bases de données externes RDS pour for My SQL	Non	Aucun	Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Fonctionnalité	Pris en charge	Remarques	En savoir plus
Sauvegarde et restauration natives pour Amazon RDS pour Microsoft SQL Server	Oui	Aucun	Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives

Classes d'instances de base de données prises en charge pour Amazon RDS sur AWS Outposts

Amazon RDS on AWS Outposts prend en charge les classes d'instances de base de données suivantes :

- Classes d'instances de bases de données à usage général
  - db.m7i.48 x large
  - db.m7i 24 x large
  - db.m7i.16 x large
  - db.m7i 12 x large
  - db.m7i.8xlarge
  - db.m7i.4xlarge
  - db.m7i.2xlarge
  - db.m7i.xlarge
  - db.m7i.large
  - db.m5.24xlarge
  - db.m5.16xlarge
  - db.m5.12xlarge
  - db.m5.8xlarge
  - db.m5.4xlarge
  - db.m5.2xlarge
  - db.m5.xlarge

  - db.m5.large
- Classes d'instances de bases de données à mémoire optimisée
  - db.r7i.48 x large
  - db.r7i 24 x large
  - db.r7i.16 x large
  - db.r7i 12 x large
  - db.r7i.8xlarge
  - db.r7i.4xlarge
  - db.r7i.2xlarge
  - db.r7i.xlarge
  - db.r7i.large
  - db.r5.24xlarge
  - db.r5.16xlarge
  - db.r5.12xlarge
  - db.r5.8xlarge
  - db.r5.4xlarge
  - db.r5.2xlarge
  - db.r5.xlarge
  - db.r5.large

Selon la façon dont vous avez configuré votre Outpost, il se peut que vous n'ayez pas toutes ces classes disponibles. Par exemple, si vous n'avez pas acheté les classes db.r5 pour votre Outpost, vous ne pouvez pas les utiliser avec RDS on Outposts.

Seul le stockage SSD à usage général est pris en charge pour les instances de bases de données RDS sur Outposts. Pour plus d'informations sur les classes d'instance DB, veuillez consulter Classes d'instances de base de données .

Amazon RDS gère la maintenance et la récupération de vos instances de base de données et requiert une capacité active sur l'Outpost pour le faire. Nous vous recommandons de configurer N +1 EC2 instances pour chaque classe d'instance de base de données dans vos environnements de production. RDS on Outposts peut utiliser la capacité supplémentaire de EC2 ces instances pour les opérations de maintenance et de réparation. Par exemple, si vos environnements de production

comportent 3 classes d'instance de base de données db.m5.large et 5 classes d'instances de base de données db.r5.xlarge, nous recommandons qu'ils disposent d'au moins 4 instances m5.large et 6 instances r5.xlarge. EC2 EC2 Pour plus d'informations, reportez-vous à la section Résilience dans AWS Outposts, dans le Guide de l'utilisateur d'AWS Outposts .

Adresses IP appartenant au client pour Amazon RDS on AWS Outposts

Amazon RDS sur AWS Outposts utilise les informations que vous fournissez sur votre réseau sur site pour créer un groupe d'adresses. Ce groupe est connu sous le nom groupe d'adresses IP clients (groupe CoIP). Les adresses IP clients fournissent une connectivité locale ou externe aux ressources de vos sous-réseaux Outpost via votre réseau local. Pour plus d'informations sur les adresses IP clients, consultez Adresses IP clients dans le Guide de l'utilisateur d'AWS Outposts.

Chaque instance de base de données RDS sur outposts dispose d'une adresse IP privée pour le trafic à l'intérieur de son cloud privé virtuel (VPC). Cette adresse IP privée n'est pas accessible au public. Vous pouvez utiliser l'option Public (Publique) pour spécifier si l'instance de base de données possède également une adresse IP publique en plus de l'adresse IP privée. L'utilisation de l'adresse IP publique pour les connexions achemine celles-ci via Internet et peut entraîner de fortes latences dans certains cas.

Au lieu d'utiliser ces adresses IP privées et publiques, RDS sur Outposts prend en charge l'utilisation des CoIP pour les instances de base de données via leurs sous-réseaux. Lorsque vous utilisez une adresse IP client pour une instance de base de données RDS sur Outposts, vous vous connectez à celle-ci avec son point de terminaison. RDS sur Outposts utilise ensuite automatiquement l'adresse IP client pour toutes les connexions tant à l'intérieur qu'à l'extérieur du cloud privé virtuel (VPC).

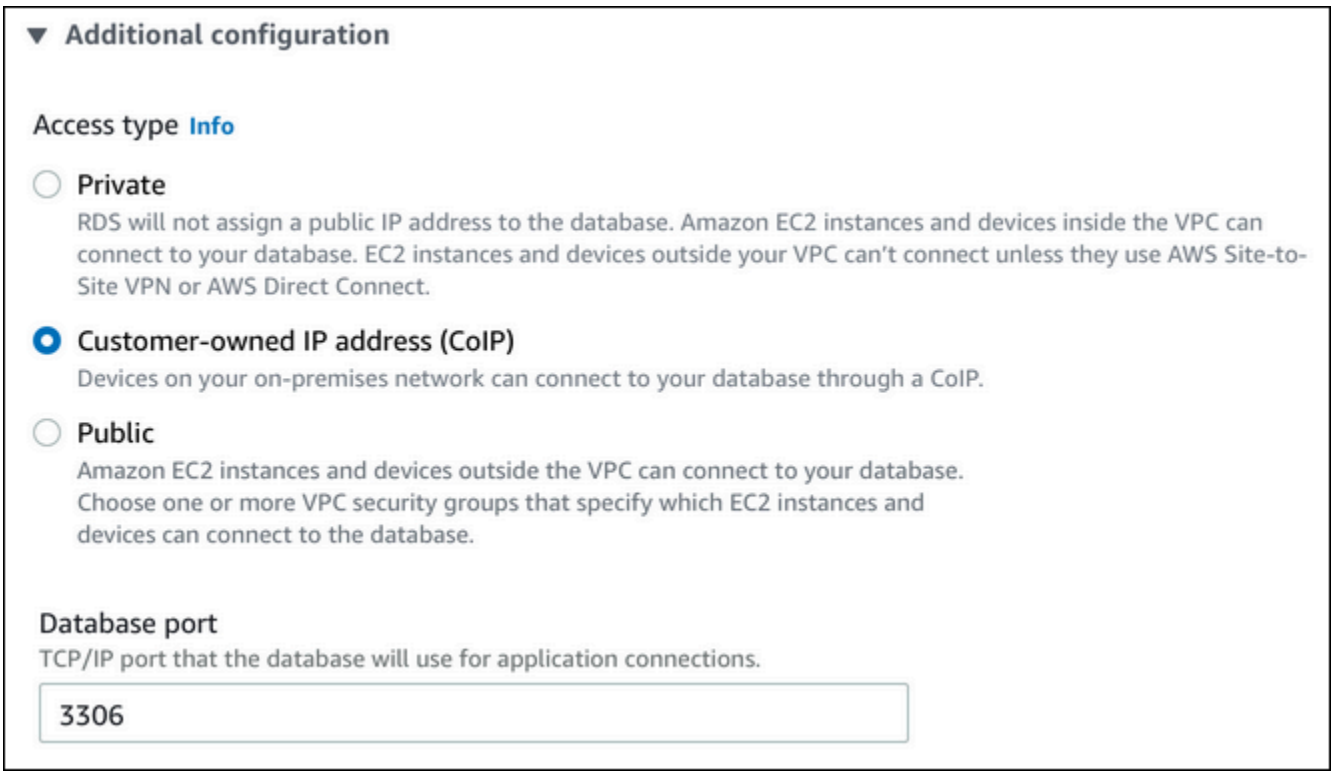
Les adresses IP clients peuvent offrir les avantages suivants aux instances de base de données RDS sur outposts :

- Latence de connexion inférieure
- Sécurité renforcée

Utilisation des CoIP

Vous pouvez activer ou désactiver les CoIP pour une instance de base de données RDS on Outposts à l'aide de la AWS Management Console, de AWS CLI ou de l'API RDS :

- Avec la AWS Management Console, sélectionnez le paramètre Customer-owned IP address (CoIP) (Adresse IP client (CoIP)) dans le champ Access type (Type d'accès) pour utiliser une adresse IP client. Sélectionnez l'un des autres paramètres pour les désactiver.



- Avec l'AWS CLI, utilisez l'option `--enable-customer-owned-ip | --no-enable-customer-owned-ip`.
- Avec l'API RDS, utilisez le paramètre `EnableCustomerOwnedIp`.

Vous pouvez activer ou désactiver les CoIP lorsque vous effectuez l'une des actions suivante :

- Créez une instance de base de données.

  Pour plus d'informations, consultez Création d'instances de base de données pour Amazon RDS sur AWS Outposts.
- Modification d'une instance de base de données

  Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Création d'un réplica en lecture

  Pour de plus amples informations, veuillez consulter Création de répliques de lecture pour Amazon RDS sur AWS Outposts.
- Restauration d'une instance de base de données à partir d'un instantané

Pour plus d'informations, consultez Restauration vers une instance de base de données.

- Restauration d'une instance de base de données à une heure spécifiée

  Pour de plus amples informations, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

> **Note**
>
> Dans certains cas, vous pouvez activer les adresses IP clients pour une instance de base de données, mais Amazon RDS n'est pas en mesure d'allouer une adresse IP client pour cette instance de base de données. Dans ce cas, l'état de l'instance de base de données passe à réseau incompatible. Pour plus d'informations sur l'état de l'instance de base de données, consultez Affichage de l'état de l'instance de base de données dans un cluster Aurora.

Limites

Les limitations suivantes s'appliquent à la prise en charge des adresses IP clients pour les instances de base de données RDS sur outposts :

- Lorsque vous utilisez un CoIP pour une instance de base de données, assurez-vous que l'accessibilité publique est désactivée pour cette instance de base de données.
- Assurez-vous que les règles entrantes de vos groupes de sécurité VPC incluent la plage d'adresses CoIP (bloc d'adresse CIDR). Pour de plus amples informations sur la configuration des groupes de sécurité, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.
- Vous ne pouvez pas attribuer une adresse IP client issue d'un pool d'adresses IP clients à une instance de base de données. Lorsque vous utilisez une adresse IP client pour une instance de base de données, Amazon RDS lui attribue automatiquement une adresse IP client issue d'un groupe de CoIP.
- Vous devez utiliser le Compte AWS qui détient les ressources Outpost (propriétaire) ou partager les ressources suivantes avec d'autres Comptes AWS (consommateurs) de la même organisation :
  - Outpost
  - Table de routage de la passerelle locale (LGW) pour le VPC de l'instance de base de données
  - Pool ou pools d'adresses IP clients pour la table de routage LGW

Pour plus d'informations, consultez la section Utilisation de ressources AWS Outposts partagées du Guide de l'utilisateur d'AWS Outposts.

Utilisation des déploiements multi-AZ pour Amazon RDS on AWS Outposts

Pour les déploiements multi-AZ, Amazon RDS crée une instance de base de données principale sur un AWS Outpost. RDS réplique de manière synchrone les données vers une instance de base de données en veille sur un Outpost différent.

Les déploiements Multi-AZ sur AWS Outposts fonctionnent comme les déploiements Multi-AZ dans Régions AWS, mais avec les différences suivantes :

- Ils nécessitent une connexion locale entre deux Outposts ou plus.
- Ils nécessitent des groupes d'IP appartenant aux clients (CoIP). Pour de plus amples informations, veuillez consulter Adresses IP appartenant au client pour Amazon RDS on AWS Outposts.
- La réplication fonctionne sur votre réseau local.

Les déploiements Multi-AZ sur AWS Outposts sont disponibles pour toutes les versions prises en charge de MySQL et de PostgreSQL sur RDS on Outpost. Les sauvegardes locales ne sont pas prises en charge pour les déploiements Multi-AZ. Pour de plus amples informations, veuillez consulter Création d'instances de base de données pour Amazon RDS sur AWS Outposts.

Utiliser le modèle de responsabilité partagée

Bien que AWS fasse des efforts commercialement raisonnables pour fournir des instances de base de données configurées pour une haute disponibilité, la disponibilité utilise un modèle de responsabilité partagée. La capacité de RDS on Outposts à basculer et à réparer les instances de base de données nécessite que chacun de vos Outposts soit connecté à sa propre Région AWS.

RDS on Outposts nécessite également une connectivité entre l'Outpost qui héberge l'instance de base de données principale et l'Outpost qui héberge l'instance de base de données en veille pour la réplication synchrone. Tout impact sur cette connexion peut empêcher RDS on Outposts d'effectuer un basculement.

Vous pouvez constater des latences élevées pour un déploiement d'instance de base de données standard en raison de la réplication synchrone des données. La bande passante et la latence de la connexion entre l'Outpost hébergeant l'instance de base de données principale et l'Outpost hébergeant l'instance de base de données de secours affectent directement les latences. Pour de plus amples informations, veuillez consulter Prérequis.

Amélioration de la disponibilité

Nous recommandons les actions suivantes pour améliorer la disponibilité :

- Allouez une capacité supplémentaire suffisante à vos applications stratégiques pour permettre la reprise et le basculement en cas de problème d'hôte sous-jacent. Ceci s'applique à tous les Outposts qui contiennent des sous-réseaux dans votre groupe de sous-réseaux de base de données. Pour plus d'informations, consultez Résilience dans AWS Outposts.
- Fournissez une connectivité réseau redondante pour vos Outposts.
- Utilisez plus de deux Outposts. Avoir plus de deux Outposts permet à Amazon RDS de récupérer une instance de base de données. RDS effectue cette restauration en déplaçant l'instance de base de données vers un Outpost si l'Outpost actuel subit une panne.
- Fournissez deux sources d'alimentation et une connectivité réseau redondante pour votre Outpost.

Nous recommandons les éléments suivants pour vos réseaux locaux :

- La latence de durée du cycle (RTT) entre l'Outpost hébergeant votre instance de base de données principale et l'Outpost hébergeant votre instance de base de données de secours affecte directement la latence d'écriture. Maintenez la latence RTT entre les Outposts AWS dans une plage de valeurs en millisecondes la plus basse possible (un chiffre). Nous recommandons de ne pas dépasser cinq millisecondes, mais vos besoins peuvent varier.

  Vous pouvez trouver l'impact net sur la latence du réseau dans les métriques Amazon CloudWatch pour `WriteLatency`. Pour de plus amples informations, veuillez consulter CloudWatch Métriques Amazon pour Amazon RDS.
- La disponibilité de la connexion entre les Outposts affecte la disponibilité globale de vos instances de base de données. Disposez d'une connectivité réseau redondante entre les Outposts.

Prérequis

Les déploiements Multi-AZ sur RDS on Outpost présentent les prérequis suivants :

- Disposez d'au moins deux Outposts, reliés par des connexions locales et attachés à des zones de disponibilité différentes dans une Région AWS
- Assurez-vous que vos groupes de sous-réseau de base de données contiennent les éléments suivants :

  - Au moins deux sous-réseaux dans au moins deux zones de disponibilité au sein d'une Région AWS donnée.
  - Sous-réseaux uniquement dans les Outposts.
  - Au moins deux sous-réseaux dans au moins deux Outposts au sein du même cloud privé virtuel (VPC).
- Associez le VPC de votre instance de base de données à toutes vos tables de routage de passerelles locales. Cette association est nécessaire car la réplication s'effectue sur votre réseau local en utilisant les passerelles locales de vos Outposts.

  Par exemple, supposons que votre VPC contienne le sous-réseau-A dans l'Outpost-A et le sous-réseau-B dans l'Outpost-B. L'Outpost-A utilise LocalGateway-A (LGW-A) et l'Outpost-B utilise LocalGateway-B (LGW-B). LGW-A possède RouteTable-A, et LGW-B possède RouteTable-B. Vous souhaitez utiliser à la fois RouteTable-A et RouteTable-B pour le trafic de réplication. Pour ce faire, associez votre VPC à la fois à RouteTable-A et RouteTable-B.

  Pour plus d'informations sur la création d'une association, consultez la commande AWS CLI Amazon EC2 create-local-gateway-route-table-vpc-association.
- Assurez-vous que vos Outposts utilisent des routages IP appartenant au client (CoIP). Chaque table de routage doit également avoir au moins un groupe d'adresses. Amazon RDS alloue une adresse IP supplémentaire à chacune des instances de base de données principale et de veille pour la synchronisation des données.
- Assurez-vous qu'Compte AWS à qui appartiennent les instances de base de données RDS possède les tables de routage de la passerelle locale et les groupes d'adresses IP clients. Ou assurez-vous qu'il fait partie d'un partage Resource Access Manager ayant accès aux tables de routage de la passerelle locale et aux groupes d'adresses IP clients.
- Assurez-vous que les adresses IP de vos groupes CoIP peuvent être routées d'une passerelle locale d'un Outpost vers les autres.
- Assurez-vous que les blocs d'adresse CIDR du VPC (par exemple, 10.0.0.0/4) et les blocs d'adresse CIDR de votre groupe de CoIP ne contiennent pas d'adresses IP de la classe E (240.0.0.0/4). RDS utilise ces adresses IP en interne.
- Assurez-vous que vous avez correctement configuré le trafic sortant et le trafic entrant correspondant.

  RDS on Outposts établit une connexion de réseau VPN entre les instances de base de données principale et de veille. Pour que cela fonctionne correctement, votre réseau local doit autoriser le trafic sortant et le trafic entrant correspondant pour le protocole ISAKMP (Internet Security

Association and Key Management Protocol). Il utilise pour cela le port 500 du protocole UDP (User Datagram Protocol) et le port 4500 du protocole NAT-T (Network Address Translation Traversal) de la sécurité IP (IPsec).

Pour plus d'informations sur les CoIP, consultez Adresses IP appartenant au client pour Amazon RDS on AWS Outposts dans ce guide et la section Customer-owned IP addresses (Adresses IP appartenant au client) dans le Guide de l'utilisateur AWS Outposts.

Utilisation des opérations API pour les autorisations Amazon EC2

Que vous utilisiez ou non des CoIP pour votre instance de base de données on AWS Outposts, RDS a besoin d'accéder aux ressources de votre groupe de CoIP. RDS peut appeler les opérations suivantes de l'API d'autorisations EC2 pour les CoIP en votre nom pour les déploiements Multi-AZ :

- `CreateCoipPoolPermission` : lorsque vous créez une instance de base de données Multi-AZ sur RDS on Outposts
- `DeleteCoipPoolPermission` : lorsque vous supprimez une instance de base de données Multi-AZ sur RDS on Outposts

Ces opérations API accordent ou retirent aux comptes internes RDS l'autorisation d'allouer des adresses IP élastiques à partir du groupe de CoIP spécifié par l'autorisation. Vous pouvez visualiser ces adresses IP en utilisant l'opération API `DescribeCoipPoolUsage`. Pour plus d'informations sur les CoIP, consultez Adresses IP appartenant au client pour Amazon RDS on AWS Outposts et Customer-owned IP addresses (Adresses IP appartenant au client) dans le Guide de l'utilisateur AWS Outposts.

RDS peut également appeler les opérations suivantes de l'API d'autorisations EC2 pour les tables de routage des passerelles locales en votre nom pour les déploiements Multi-AZ :

- `CreateLocalGatewayRouteTablePermission` : lorsque vous créez une instance de base de données Multi-AZ sur RDS on Outposts
- `DeleteLocalGatewayRouteTablePermission` — lorsque vous supprimez une instance de base de données Multi-AZ sur RDS on Outposts

Ces opérations API accordent ou retirent aux comptes RDS internes l'autorisation d'associer les VPC RDS internes aux tables de routage de votre passerelle locale. Vous

pouvez visualiser ces associations table de routage-VPC à l'aide des opérations API `DescribeLocalGatewayRouteTableVpcAssociations`.

Création d'instances de base de données pour Amazon RDS sur AWS Outposts

La création d'une instance de base de données Amazon RDS sur AWS Outposts ressemble à la création d'une instance de base de données Amazon RDS dans le cloud AWS. Toutefois, veillez à indiquer un groupe de sous-réseaux de base de données associé à votre Outpost.

Un cloud privé virtuel (VPC) basé sur le service Amazon VPC peut couvrir toutes les zones de disponibilité d'une Région AWS. Vous pouvez étendre n'importe quel VPC de la Région AWS à votre Outpost en ajoutant un sous-réseau Outpost. Pour ajouter un sous-réseau Outpost à un VPC, spécifiez l'ARN (Amazon Resource Name) de l'Outpost lorsque vous créez le sous-réseau.

Avant de créer une instance de base de données RDS sur outposts, vous pouvez créer un groupe de sous-réseaux de base de données incluant un sous-réseau associé à votre Outpost. Lorsque vous créez une instance de base de données RDS sur outposts, indiquez ce groupe de sous-réseaux de base de données. Vous pouvez également choisir de créer un nouveau groupe de sous-réseaux de base de données lorsque vous créez votre instance de base de données.

Pour plus d'informations sur la configuration d'AWS Outposts, veuillez consulter AWS Outposts User Guide.

Console

Création d'un groupe de sous-réseaux de base de données

Créez un groupe de sous-réseaux de base de données avec un sous-réseau associé à votre Outpost.

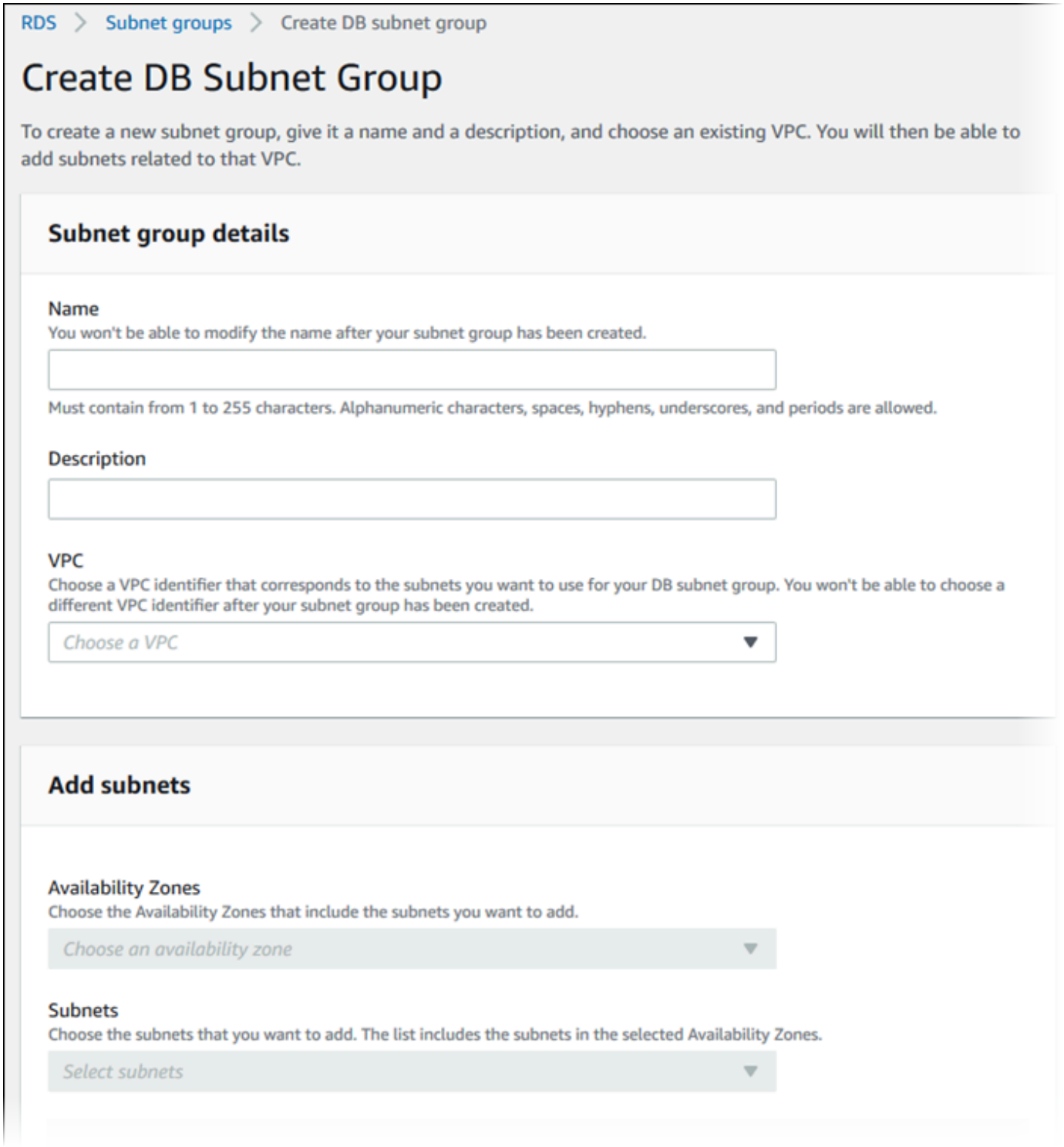
Vous pouvez également créer un nouveau groupe de sous-réseaux de base de données lorsque vous créez votre instance de base de données. Si vous souhaitez le faire, ignorez cette procédure.

> **Note**
>
> Pour créer un groupe de sous-réseaux de base de données pour le AWS Cloud, vous devez indiquer au moins deux sous-réseaux.

Pour créer un groupe de sous-réseaux de base de données pour votre Outpost

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS dans laquelle vous voulez créer le groupe de sous-réseaux de base de données.
3. Sélectionnez Groupes de sous-réseaux, puis Créer un groupe de sous-réseaux DB.

   La page Créer un groupe de sous-réseaux DB s'affiche.



4. Pour Name (Nom), indiquez le nom du groupe de sous-réseaux de base de données.
5. Pour Description, saisissez une description pour le groupe de sous-réseaux de base de données.

6. Pour VPC, choisissez le VPC pour lequel vous créez le groupe de sous-réseaux de base de données.
7. Pour Zones de disponibilité, choisissez la zone de disponibilité pour votre Outpost.
8. Pour Sous-réseaux, choisissez le sous-réseau pour une utilisation par RDS sur outposts.
9. Sélectionnez Créer pour créer le groupe de sous-réseaux de base de données.

Création de l'instance de base de données RDS sur Outposts

Créez l'instance de base de données et choisissez l'Outpost pour votre instance de base de données.

Pour créer une instance de base de données RDS sur outposts à l'aide de la console

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS où l'Outpost sur lequel vous souhaitez créer l'instance de base de données est attaché.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).

   AWS Management Console détecte les Outposts disponibles que vous avez configurés et présente l'option Sur site dans la section Emplacement de la base de données.

   > **Note**
   >
   > Si vous n'avez pas configuré d'Outpost, la section Emplacement de la base de données n'apparaît pas ou l'option RDS sur outposts n'est pas disponible dans la section Choisir une méthode de création sur site.

5. Pour Database location (Emplacement de la base de données), choisissez On-premises (Sur site).
6. Pour On-premises creation method (Méthode de création sur site, choisissez RDS sur Outposts.
7. Spécifiez vos paramètres pour Outposts Connectivity (Connectivité des Outposts). Ces paramètres concernent l'Outpost utilisant le VPC contenant le groupe de sous-réseaux de base de données pour votre instance de base de données. Votre VPC doit être basé sur le service Amazon VPC.

a. Pour Virtual Private Cloud (VPC), choisissez le VPC contenant le groupe de sous-réseaux de base de données pour votre instance de base de données.

b. Pour VPC security group (Groupe de sécurité VPC), choisissez le groupe de sécurité Amazon VPC pour votre instance de base de données.

c. Pour DB Subnet group (Groupe de sous-réseaux de base de données), choisissez le groupe de sous-réseaux de base de données pour votre instance de base de données.

   Vous pouvez sélectionner un groupe de sous-réseaux de base de données existant associé à l'Outpost, par exemple si vous avez effectué la procédure dans Création d'un groupe de sous-réseaux de base de données.

   Vous pouvez également créer un nouveau groupe de sous-réseaux de base de données pour l'Outpost.

8. Pour Multi-AZ deployment (Déploiement Multi-AZ), choisissez Create a standby instance (recommended for production usage) (Créer une instance de secours (recommandé pour une utilisation en production)) pour créer une instance de base de données de secours dans un autre Outpost.

> **Note**
>
> Cette option n'est pas disponible pour Microsoft SQL Server.
> Si vous choisissez de créer un déploiement Multi-AZ, vous ne pouvez pas stocker les sauvegardes sur votre Outpost.

9. Sous Backup (Sauvegarde), procédez comme suit :

   a. Pour Backup target (Cible de sauvegarde), choisissez l'une des options suivantes :

      - AWS Cloud pour stocker les sauvegardes automatiques et les instantanés manuels dans la Région AWS parente.
      - Outposts (on-premises) (Outposts (sur site)) pour créer des sauvegardes locales.

> **Note**
>
> Pour stocker les sauvegardes sur votre Outpost, il doit disposer de la fonctionnalité Amazon S3. Pour de plus amples informations, consultez Amazon S3 on Outposts.

Les sauvegardes locales ne sont pas prises en charge pour les déploiements multi-AZ ou les réplicas en lecture.

b. Choisissez Activer les sauvegardes automatisées pour créer des point-in-time instantanés de votre instance de base de données.

   Si vous activez les sauvegardes automatisées, vous pouvez choisir des valeurs pour Backup retention period (Période de rétention des sauvegardes) et Backup window (Fenêtre de sauvegarde), ou conserver les valeurs par défaut.

10. Spécifiez d'autres paramètres d'instance de base de données si nécessaire.

    Pour plus d'informations sur chaque paramètre lors de la création d'une instance de base de données, veuillez consulter Paramètres des instances de base de données.

11. Choisissez Create database (Créer une base de données).

    La page Databases (Bases de données) s'affiche. Une bannière vous indique que votre instance de base de données est en cours de création et affiche le bouton View credential details (Afficher les détails des informations d'identification).

Affichage des détails de l'instance de base de données

Lorsque vous avez créé votre instance de base de données, vous pouvez en afficher les informations d'identification et d'autres détails.

Pour afficher les informations de l'instance de base de données :

1. Pour afficher le nom d'utilisateur principal et le mot de passe de l'instance de base de données, choisissez View credential details (Afficher les informations d'identification) sur la page Databases (Bases de données).

   Vous pouvez vous connecter à l'instance de base de données en tant qu'utilisateur principal à l'aide de ces informations d'identification.

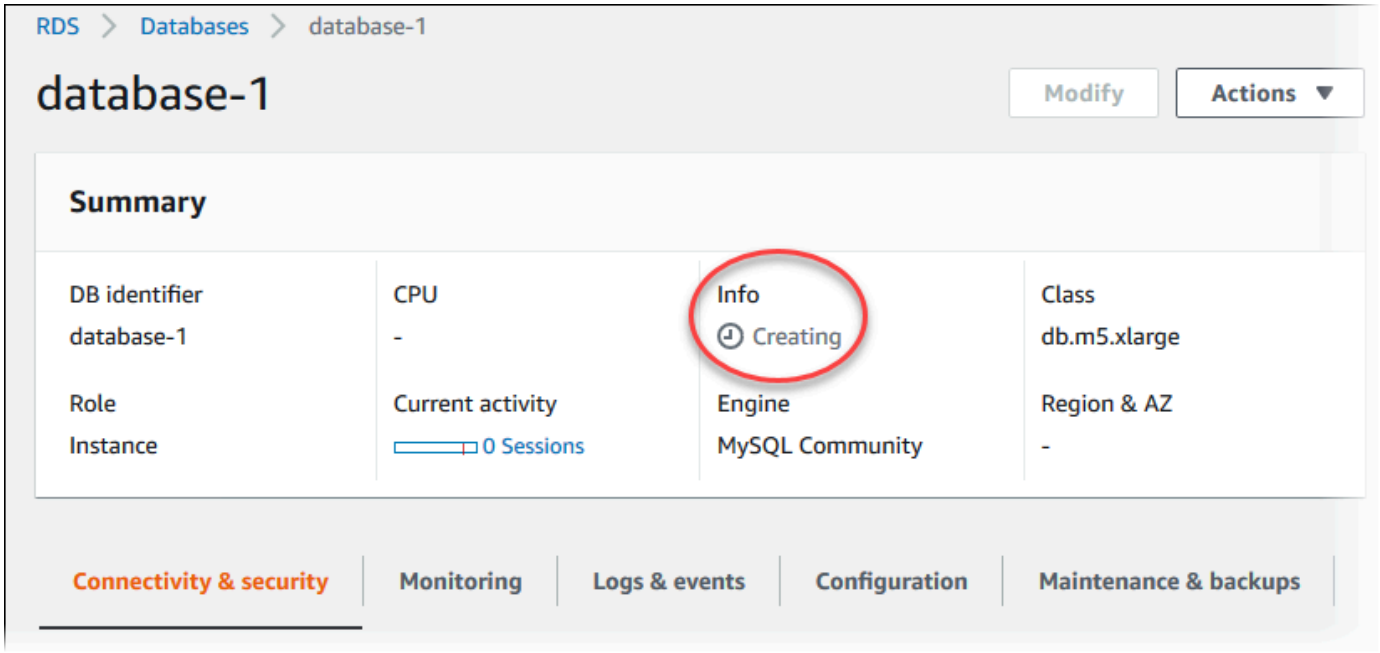
   Important

   Vous ne pourrez pas afficher le mot de passe de l'utilisateur principal de nouveau. Si vous ne l'enregistrez pas, il sera peut-être nécessaire de le modifier. Pour changer le mot de passe de l'utilisateur principal une fois l'instance de base de données disponible, modifiez l'instance de base de données. Pour plus d'informations sur la modification

d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

2. Sur la page Databases (Bases de données), choisissez le nom de la nouvelle instance de base de données.

   Sur la console RDS, les détails de la nouvelle instance de base de données s'affichent. L'instance de base de données a le statut Creating (Création en cours) jusqu'à ce qu'elle soit créée et prête à l'emploi. Lorsque le statut devient Available (Disponible), vous pouvez vous connecter à l'instance de base de données. En fonction du stockage et de la classe d'instance de base de données alloués, la mise à disposition de la nouvelle instance de base de données peut nécessiter plusieurs minutes.



   Une fois l'instance de base de données disponible, vous pouvez la gérer comme vous gérez les instances de bases de données RDS dans le AWS Cloud.

AWS CLI

Avant de créer une instance de base de données dans un Outpost avec la AWS CLI, créez d'abord un groupe de sous-réseaux de base de données à utiliser par RDS on Outposts.

Pour créer un groupe de sous-réseaux de base de données pour votre Outpost

- Utilisez la commande create-db-subnet-group. Pour `--subnet-ids`, spécifiez le groupe de sous-réseaux dans l'Outpost pour une utilisation par RDS sur outposts.

  Pour LinuxmacOS, ou Unix :

  ```
  aws rds create-db-subnet-group \
      --db-subnet-group-name myoutpostdbsubnetgr \
      --db-subnet-group-description "DB subnet group for RDS on Outposts" \
      --subnet-ids subnet-abc123
  ```

  Dans Windows :

  ```
  aws rds create-db-subnet-group ^
      --db-subnet-group-name myoutpostdbsubnetgr ^
      --db-subnet-group-description "DB subnet group for RDS on Outposts" ^
      --subnet-ids subnet-abc123
  ```

Pour créer une instance de base de données RDS sur Outposts à l'aide de la AWS CLI

- Utilisez la commande create-db-instance. Spécifiez une zone de disponibilité pour l'Outpost, un groupe de sécurité Amazon VPC associé à l'Outpost et le groupe de sous-réseaux de base de données que vous avez créé pour l'Outpost. Vous pouvez inclure les options suivantes :
  - `--db-instance-identifier`
  - `--db-instance-class`
  - `--engine` – Moteur de base de données. Utilisez l'une des valeurs suivantes :
    - MySQL – Spécifiez `mysql`.
    - PostgreSQL – Spécifiez `postgres`.
    - Microsoft SQL Server : spécifiez `sqlserver-ee`, `sqlserver-se`, ou `sqlserver-web`.
  - `--availability-zone`
  - `--vpc-security-group-ids`
  - `--db-subnet-group-name`
  - `--allocated-storage`
  - `--max-allocated-storage`

- `--master-username`
- `--master-user-password`
- `--multi-az | --no-multi-az` : (facultatif) indiquez si vous souhaitez créer une instance de base de données de secours dans une zone de disponibilité différente. L'argument par défaut est `--no-multi-az`.

  L'option `--multi-az` n'est pas disponible pour SQL Server.
- `--backup-retention-period`
- `--backup-target` – (Facultatif) Où stocker les sauvegardes automatisées et les instantanés manuels. Utilisez l'une des valeurs suivantes :
  - `outposts` – Stockez-les localement sur votre Outpost.
  - `region` : stockez-les dans la Région AWS parente. C'est la valeur par défaut.

  Si vous utilisez l'option `--multi-az`, vous ne pouvez pas utiliser `outposts` pour `--backup-target`. De plus, l'instance de base de données ne peut pas avoir de réplicas en lecture si vous utilisez `outposts` pour `--backup-target`.
- `--storage-encrypted`
- `--kms-key-id`

Example

L'exemple suivant crée une instance de base de données MySQL nommée `myoutpostdbinstance` avec des sauvegardes stockées sur votre Outpost.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier myoutpostdbinstance \
    --engine-version 8.0.17 \
    --db-instance-class db.m5.large \
    --engine mysql \
    --availability-zone us-east-1d \
    --vpc-security-group-ids outpost-sg \
    --db-subnet-group-name myoutpostdbsubnetgr \
    --allocated-storage 100 \
    --max-allocated-storage 1000 \
    --master-username masterawsuser \
    --manage-master-user-password \
```

```
    --backup-retention-period 3 \
    --backup-target outposts \
    --storage-encrypted \
    --kms-key-id mykey
```

Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier myoutpostdbinstance ^
    --engine-version 8.0.17 ^
    --db-instance-class db.m5.large ^
    --engine mysql ^
    --availability-zone us-east-1d ^
    --vpc-security-group-ids outpost-sg ^
    --db-subnet-group-name myoutpostdbsubnetgr ^
    --allocated-storage 100 ^
    --max-allocated-storage 1000 ^
    --master-username masterawsuser ^
    --manage-master-user-password ^
    --backup-retention-period 3 ^
    --backup-target outposts ^
    --storage-encrypted ^
    --kms-key-id mykey
```

Pour plus d'informations sur chaque paramètre lors de la création d'une instance de base de données, veuillez consulter Paramètres des instances de base de données.

API RDS

Pour créer une nouvelle instance de base de données dans un Outpost avec l'API RDS, créez d'abord un groupe de sous-réseaux de base de données à utiliser par RDS sur Outposts en appelant l'opération CreateDB. SubnetGroup Pour `SubnetIds`, spécifiez le groupe de sous-réseaux dans l'Outpost pour une utilisation par RDS sur outposts.

Ensuite, appelez l'opération CreateDBInstance avec les paramètres suivants. Spécifiez une zone de disponibilité pour l'Outpost, un groupe de sécurité Amazon VPC associé à l'Outpost et le groupe de sous-réseaux de base de données que vous avez créé pour l'Outpost.

- `AllocatedStorage`
- `AvailabilityZone`
- `BackupRetentionPeriod`

- `BackupTarget`

  Si vous créez un déploiement d'instance de base de données multi-AZ, vous ne pouvez pas utiliser `outposts` pour `BackupTarget`. De plus, l'instance de base de données ne peut pas avoir de réplicas en lecture si vous utilisez `outposts` pour `BackupTarget`.
- `DBInstanceClass`
- `DBInstanceIdentifier`
- `VpcSecurityGroupIds`
- `DBSubnetGroupName`
- `Engine`
- `EngineVersion`
- `MasterUsername`
- `MasterUserPassword`
- `MaxAllocatedStorage` (facultatif)
- `MultiAZ` (facultatif)
- `StorageEncrypted`
- `KmsKeyID`

Pour plus d'informations sur chaque paramètre lors de la création d'une instance de base de données, veuillez consulter Paramètres des instances de base de données.

Création de répliques de lecture pour Amazon RDS sur AWS Outposts

Amazon RDS on AWS Outposts utilise la fonctionnalité de réplication intégrée des moteurs de base de données MySQL et PostgreSQL pour créer une réplique en lecture à partir d'une instance de base de données source. L'instance de base de données source devient l'instance de base de données principale. Les mises à jour apportées à l'instance de base de données principale sont copiées de façon asynchrone sur le réplica en lecture. Vous pouvez réduire la charge sur votre instance de base de données principale en acheminant les requêtes en lecture depuis vos applications vers le réplica en lecture. Les réplicas en lecture permettent une montée en puissance basée sur Elastic au-delà des contraintes de capacité d'une seule instance de base de données dans le cas de charges de travail de base de données à lecture intensive.

Lorsque vous créez un réplica en lecture à partir d'une instance de base de données RDS sur Outposts, le réplica en lecture utilise une adresse IP appartenant au client (CoIP). Pour plus d’informations, consultez Adresses IP appartenant au client pour Amazon RDS on AWS Outposts.

Les réplicas en lecture sur RDS sur Outposts présentent les limites suivantes :

- Vous ne pouvez pas créer de réplicas en lecture pour RDS for SQL Server sur les instances de base de données RDS sur Outposts.
- Les réplicas en lecture entre régions ne sont pas pris en charge sur RDS sur Outposts.
- Les réplicas en lecture en cascade ne sont pas pris en charge sur RDS sur Outposts.
- L'instance de base de données RDS sur Outposts source ne peut pas avoir de sauvegardes locales. La cible de sauvegarde pour l'instance de base de données source doit être votre Région AWS.
- Les réplicas en lecture nécessitent des groupes d'IP appartenant aux clients (CoIP). Pour plus d’informations, consultez Adresses IP appartenant au client pour Amazon RDS on AWS Outposts.
- Les répliques de lecture sur RDS on Outposts ne peuvent être créées que dans le même cloud privé virtuel (VPC) que l'instance de base de données source.
- Les répliques de lecture sur RDS on Outposts peuvent être situées sur le même avant-poste ou sur un autre avant-poste du même VPC que l'instance de base de données source.
- Vous ne pouvez pas créer de répliques de lecture pour les instances de base de données chiffrées avec AWS KMS External Key Store (XKS).

Vous pouvez créer une réplique de lecture à partir d'une instance de base de données RDS on Outposts à l'aide de l' AWS Management Console API AWS CLI, ou RDS. Pour plus d'informations sur les réplicas en lecture, consultez Utilisation des réplicas en lecture d'instance de base de données.

Console

Pour créer un réplica en lecture à partir d'une instance de base de données source

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez l'instance de base de données que vous voulez utiliser comme source pour votre réplica en lecture.
4. Sous Actions, choisissez Créer des réplicas en lecture.
5. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture.
6. Spécifiez vos paramètres pour Outposts Connectivity (Connectivité des Outposts). Ces paramètres concernent l'Outpost utilisant le cloud privé virtuel (VPC) contenant le groupe de sous-réseaux de base de données pour votre instance de base de données. Votre VPC doit être basé sur le service Amazon VPC.
7. Choisissez votre classe d'instances de base de données. Nous vous recommandons d'utiliser un type de stockage et une classe d'instances de base de données identiques ou supérieurs à l'instance de base de données source pour le réplica en lecture.
8. Pour Multi-AZ deployment (Déploiement multi-AZ), choisissez Create a standby instance (recommended for production usage) (Créer une instance de secours (recommandé pour une utilisation en production)) afin de créer une instance de base de données de secours dans une autre zone de disponibilité.

   La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.
9. (Facultatif) Sous Connectivity (Connectivité), définissez des valeurs pour Subnet Group (Groupe de sous-réseaux) et Availability Zone (Zone de disponibilité).

   Si vous spécifiez des valeurs à la fois pour Subnet Group (Groupe de sous-réseaux) et Availability Zone (Zone de disponibilité), le réplica en lecture est créé sur un Outpost associé à la zone de disponibilité dans le groupe de sous-réseaux de base de données.

Si vous spécifiez une valeur pour Subnet Group (Groupe de sous-réseaux) et No preference (Aucune préférence) pour Availability Zone (Zone de disponibilité), le réplica en lecture est créé sur un Outpost aléatoire dans le groupe de sous-réseaux de base de données.

10. Pour AWS KMS key, choisissez l' AWS KMS key identifiant de la clé KMS.

    Le réplica en lecture doit être chiffré.
11. Choisissez d'autres options si nécessaire.
12. Choisissez Créer un réplica en lecture.

Une fois le réplica en lecture créé, vous pouvez le voir sur la page Bases de données de la console RDS. Il affiche le réplica dans la colonne Rôle .

AWS CLI

Pour créer une réplique en lecture à partir d'une instance de base de données MySQL ou PostgreSQL source, utilisez AWS CLI la commande -replica. create-db-instance-read

Vous pouvez contrôler l'emplacement de création du réplica en lecture en spécifiant les options `--db-subnet-group-name` et `--availability-zone` :

- Si vous spécifiez les deux options `--db-subnet-group-name` et `--availability-zone`, le réplica en lecture est créé sur un Outpost associé à la zone de disponibilité dans le groupe de sous-réseaux de base de données.
- Si vous spécifiez l'option `--db-subnet-group-name` sans spécifier l'option `--availability-zone`, le réplica en lecture est créé sur un Outpost aléatoire dans le groupe de sous-réseaux de base de données.
- Si vous ne spécifiez aucune option, le réplica en lecture est créé sur le même Outpost que l'instance de base de données RDS sur Outposts source.

L'exemple suivant crée un réplica et spécifie l'emplacement du réplica en lecture en incluant les options `--db-subnet-group-name` et `--availability-zone`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
```

```
    --db-instance-identifier myreadreplica \
    --source-db-instance-identifier mydbinstance \
    --availability-zone us-west-2a
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier myreadreplica ^
    --source-db-instance-identifier mydbinstance ^
    --availability-zone us-west-2a
```

API RDS

Pour créer une réplique en lecture à partir d'une instance de base de données MySQL ou PostgreSQL source, appelez l'opération InstanceReadReplica CreateDB de l'API Amazon RDS avec les paramètres requis suivants :

- `DBInstanceIdentifier`
- `SourceDBInstanceIdentifier`

Vous pouvez contrôler l'emplacement où le réplica en lecture est créé en spécifiant les paramètres `DBSubnetGroupName` et `AvailabilityZone` :

- Si vous spécifiez les deux paramètres `DBSubnetGroupName` et `AvailabilityZone`, le réplica en lecture est créé sur un Outpost associé à la zone de disponibilité dans le groupe de sous-réseaux de base de données.
- Si vous spécifiez le paramètre `DBSubnetGroupName` sans spécifier le paramètre `AvailabilityZone`, le réplica en lecture est créé sur un Outpost aléatoire dans le groupe de sous-réseaux de base de données.
- Si vous ne spécifiez aucun paramètre, le réplica en lecture est créé sur le même Outpost que l'instance de base de données RDS sur Outposts source.

Considérations pour la restauration d'instances de base de données sur Amazon RDS on AWS Outposts

Lorsque vous restaurez une instance de base de données dans Amazon RDS sur AWS Outposts, vous pouvez généralement choisir l'emplacement de stockage des sauvegardes automatisées et des instantanés manuels de l'instance de base de données restaurée.

- Lors de la restauration à partir d'un instantané de base de données manuel, vous pouvez stocker des sauvegardes dans la Région AWS parente ou localement sur votre Outpost.
- En effectuant une restauration à partir d'une sauvegarde automatisée (restauration à un moment donné), vous avez moins de choix :
  - En cas de restauration à partir de la Région AWS parente, vous pouvez stocker des sauvegardes dans la Région AWS ou sur votre Outpost.
  - Si vous effectuez une restauration depuis votre Outpost, vous ne pouvez stocker des sauvegardes que sur votre Outpost.

Proxy Amazon RDS

Amazon RDS Proxy vous permet d'autoriser vos applications à grouper et à partager des connexions de bases de données pour améliorer leur capacité de mise à l'échelle. RDS Proxy rend les applications plus résistantes aux échecs de base de données en les connectant automatiquement à une instance de base de données de secours tout en préservant les connexions des applications. En utilisant le proxy RDS, vous pouvez également appliquer l'authentification AWS Identity and Access Management (IAM) aux bases de données et y stocker les informations d'identification en toute sécurité. AWS Secrets Manager

Avec RDS Proxy, vous pouvez gérer des pics imprévisibles de trafic des bases de données. Dans le cas contraire, ces surtensions peuvent entraîner des problèmes en raison d'un surabonnement ou de la création rapide de nouvelles connexions. RDS Proxy établit un groupe de connexions de bases de données et réutilise les connexions de ce groupe. Cette approche évite la surcharge de mémoire et d'UC liée à l'ouverture d'une nouvelle connexion de base de données à chaque fois. Pour protéger une base de données contre le surabonnement, vous pouvez contrôler le nombre de connexions à la base de données créées.

Le proxy RDS met en file d'attente ou limite les connexions aux applications qui ne peuvent pas être traitées immédiatement à partir du pool de connexions. Bien que les latences puissent augmenter, votre application peut continuer à évoluer sans échouer brusquement ou surcharger la base de données. Si les demandes de connexion dépassent les limites que vous définissez, RDS Proxy rejette les connexions des applications (en d'autres termes, il déleste la charge). Dans le même temps, il maintient des performances prévisibles pour la charge que le RDS peut desservir avec la capacité disponible.

Vous pouvez réduire la surcharge pour traiter des informations d'identification et établir une connexion sécurisée pour chaque nouvelle connexion. RDS Proxy peut gérer une partie de ce travail pour le compte de la base de données.

RDS Proxy est entièrement compatible avec les versions de moteur qu'il prend en charge. Vous pouvez activer RDS Proxy pour la plupart des applications sans modifier le code.

Rubriques

- Disponibilité des régions et des versions
- Quotas et limites pour RDS Proxy
- Planification Où utiliser RDS Proxy

- RDSConcepts et terminologie relatifs aux proxys
- Commencer à utiliser RDS Proxy
- Gestion d'un RDS proxy
- Utilisation des points de terminaison du proxy Amazon RDS
- Surveillance des métriques du proxy RDS avec Amazon CloudWatch
- Utilisation des événements RDS proxy
- Résolution des problèmes liés au RDS Proxy
- Utilisation de RDS Proxy avec AWS CloudFormation

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions d'Amazon RDS avec RDS Proxy, consultez Régions et moteurs de base de données pris en charge pour Amazon RDS Proxy.

Quotas et limites pour RDS Proxy

Voici les quotas et les limites qui s'appliquent à RDS Proxy :

- Chaque Compte AWS identifiant est limité à 20 proxies. Si votre application nécessite davantage de proxys, demandez une augmentation via la page Service Quotas du AWS Management Console. Sur la page Quotas de service, sélectionnez Amazon Relational Database Service (Amazon RDS) et localisez les proxys pour demander une augmentation de quota. AWS peut augmenter automatiquement votre quota ou en attente de l'examen de votre demande par Support.
- Chaque proxy peut avoir jusqu'à 200 secrets Secrets Manager associés. Ainsi, chaque proxy peut se connecter avec 200 comptes d'utilisateur différents maximum à tout moment.
- Chaque proxy possède un point de terminaison par défaut. Vous pouvez également ajouter jusqu'à 20 points de terminaison de proxy pour chaque proxy. Vous pouvez créer, afficher, modifier et supprimer ces points de terminaison.
- Pour les instances de base de données RDS dans les configurations de réplication, vous pouvez associer un proxy uniquement à l'instance de base de données du rédacteur, mais pas à un réplica en lecture.

- Votre RDS Proxy doit se trouver dans le même cloud privé virtuel (VPC) que la base de données. Le proxy peut ne pas être accessible au public, contrairement à la base de données. Par exemple, si vous prototypez votre base de données sur un hôte local, vous ne pouvez pas vous connecter à votre proxy si vous n'avez pas défini la configuration réseau requise pour autoriser la connexion au proxy. Cela est dû au fait que votre hôte local se trouve en dehors du VPC du proxy.
- Vous ne pouvez pas utiliser RDS Proxy avec un VPC dont la location est définie sur `dedicated`.
- Si vous utilisez le proxy RDS avec un de base de données RDS sur lequel l'authentification IAM est activée, vérifiez l'authentification de l'utilisateur. Les utilisateurs qui se connectent via un proxy doivent s'authentifier à l'aide d'informations d'identification. Pour plus d'informations sur la prise en charge de Secrets Manager et d'IAM dans RDS Proxy, consultez Configuration des informations d'identification de base de données AWS Secrets Manager pour le proxy RDS et Configuration de l'authentification IAM pour le proxy RDS.
- Vous ne pouvez pas utiliser RDS Proxy avec DNS personnalisé lorsque vous utilisez la validation du nom d'hôte SSL.
- Chaque proxy peut être associé à une instance de base de données unique. Toutefois, vous pouvez associer plusieurs proxies à la même instance de base de données.
- Le proxy épingle la session à la connexion en cours si la taille de texte de l'instruction est supérieure à 16 ko.
- Certaines régions ont des restrictions de zone de disponibilité (AZ) à prendre en compte lors de la création de votre proxy. La région USA Est (Virginie du Nord) ne prend pas en charge RDS Proxy dans la zone de disponibilité `use1-az3`. La région USA Ouest (Californie du Nord) ne prend pas en charge RDS Proxy dans la zone de disponibilité `usw1-az2`. Lorsque vous sélectionnez des sous-réseaux lors de la création de votre proxy, assurez-vous de ne pas sélectionner de sous-réseaux dans les zones de disponibilité mentionnées ci-dessus.
- Actuellement, RDS Proxy ne prend en charge aucune clé de contexte de condition globale.

  Pour plus d'informations sur les clés de contexte de condition globale, veuillez consulter Clés de contexte de condition globales AWS dans le Guide de l'utilisateur IAM.
- Vous ne pouvez pas utiliser le proxy RDS avec RDS Custom pour SQL Server.
- Pour refléter toute modification du groupe de paramètres de base de données sur votre proxy, un redémarrage de l'instance est nécessaire même si vous avez choisi d'appliquer vos modifications immédiatement. Pour les paramètres au niveau du cluster, un redémarrage à l'échelle du cluster est requis.
- Votre proxy crée automatiquement l'utilisateur `rdsproxyadmin` de base de données lorsque vous enregistrez une cible proxy. Il s'agit d'un utilisateur protégé essentiel à la fonctionnalité du

proxy. Vous devez éviter d'altérer l'`rdsproxyadmin`utilisateur de quelque manière que ce soit. La suppression ou la modification de `rdsproxyadmin` l'utilisateur ou de ses autorisations peut entraîner l'indisponibilité totale du proxy pour votre application.

Pour connaître les limites supplémentaires pour chaque moteur de base de données, consultez les sections suivantes :

- Limites supplémentaires pour RDS for MariaDB
- Limites supplémentaires pour RDS for Microsoft SQL Server
- Limites supplémentaires pour RDS for MySQL
- Limites supplémentaires pour RDS for PostgreSQL

Limites supplémentaires pour RDS for MariaDB

Les limites supplémentaires suivantes s'appliquent à RDS Proxy avec RDS for MariaDB :

- Actuellement, tous les proxies écoutent sur le port 3306 pour MariaDB. Les proxys se connectent toujours à votre base de données à l'aide du port spécifié dans les paramètres de la base de données.
- Vous ne pouvez pas utiliser le proxy RDS avec des bases de données MariaDB autogérées dans des instances Amazon. EC2
- Vous ne pouvez pas utiliser RDS Proxy avec une instance de base de données MariaDB dont le paramètre `read_only` dans son groupe de paramètres de base de données est défini sur `1`.
- Le proxy RDS ne prend pas en charge le mode compressé MariaDB. Par exemple, il ne prend pas en charge la compression utilisée par les options `--compress` ou `-C` de la commande `mysql`.
- Certaines instructions et fonctions SQL peuvent modifier l'état de connexion sans provoquer d'épinglage. Pour connaître le comportement d'épinglage le plus récent, consultez la section Éviter d'épingler un proxy RDS.
- RDS Proxy ne prend pas en charge le plugin `auth_ed25519` MariaDB.
- RDS Proxy ne prend pas en charge le protocole TLS (Transport Layer Security) version 1.3 pour les bases de données MariaDB.
- Les connexions à la base de données qui traitent une commande `GET DIAGNOSTIC` peuvent renvoyer des informations inexactes lorsque le proxy RDS réutilise la même connexion à la base de données pour exécuter une autre requête. Cela peut se produire quand le proxy RDS multiplexe

les connexions à la base de données. Pour de plus amples informations, veuillez consulter Vue d'ensemble des concepts de RDS proxy.

- Le proxy RDS ne prend actuellement pas en charge l'`caching_sha2_password`option `ClientPasswordAuthType` pour MariaDB.

> **Important**
>
> Pour les proxies associés aux bases de données MariaDB, ne définissez pas le paramètre de configuration `sql_auto_is_null` sur `true` ou sur une valeur différente de zéro dans la requête d'initialisation. Cela pourrait entraîner un comportement incorrect de l'application.

Limites supplémentaires pour RDS for Microsoft SQL Server

Les limitations supplémentaires suivantes s'appliquent à RDS Proxy avec les bases de données RDS for Microsoft SQL Server :

- Le nombre de secrets de Secrets Manager que vous devez créer pour un proxy dépend du classement utilisé par votre instance de base de données. Par exemple, supposons que votre instance de base de données utilise un classement sensible à la casse. Si votre application accepte à la fois « Admin » et « admin », votre proxy a besoin de deux secrets distincts. Pour plus d'informations sur le classement SQL Server, consultez la documentation Microsoft SQL Server.
- RDS Proxy ne prend pas en charge les connexions qui utilisent Active Directory.
- Vous ne pouvez pas utiliser l'authentification IAM avec des clients qui ne prennent pas en charge les propriétés des jetons. Pour de plus amples informations, veuillez consulter Considérations relatives à la connexion à un proxy avec Microsoft SQL Server.
- Les résultats `@@IDENTITY`, `@@ROWCOUNT` et `SCOPE_IDENTITY` ne sont pas toujours précis. Pour contourner ce problème, récupérez leurs valeurs dans la même instruction de session pour vous assurer qu'elles renvoient les informations correctes.
- Si la connexion utilise plusieurs ensembles de résultats actifs (MARS), RDS Proxy n'exécute pas les requêtes d'initialisation. Pour plus d'informations, consultez la documentation Microsoft SQL Server.
- Actuellement, le proxy RDS ne prend pas en charge les instances de base de données RDS pour SQL Server qui s'exécutent sur la version majeure de SQL Server 2022.

- Le proxy RDS ne prend pas en charge les instances de base de données RDS pour SQL Server qui s'exécutent sur la version majeure de SQL Server 2014.
- Le proxy RDS ne prend pas en charge les applications clientes qui ne peuvent pas gérer plusieurs messages de réponse dans un seul enregistrement TLS.

Limites supplémentaires pour RDS for MySQL

Les limitations supplémentaires suivantes s'appliquent à RDS Proxy avec RDS for MySQL :

- La prise en charge du proxy RDS pour `caching_sha2_password` l'authentification nécessite une connexion sécurisée (TLS).
- La prise en charge du proxy RDS pour `caching_sha2_password` est connue pour présenter des problèmes de compatibilité avec certaines versions du pilote go-sql.
- Lorsque vous utilisez le pilote MySQL 8.4 C, l'`mysql_stmt_bind_named_param`API peut former des paquets mal formés si le nombre de paramètres dépasse le nombre d'espaces réservés dans une instruction préparée. Cela entraîne des réponses incorrectes. Pour plus d'informations, consultez le rapport de bogue MySQL.
- Actuellement, tous les proxies écoutent sur le port 3306 pour MySQL. Les proxys se connectent toujours à votre base de données à l'aide du port spécifié dans les paramètres de la base de données.
- Vous ne pouvez pas utiliser le proxy RDS avec des bases de données MySQL autogérées dans EC2 des instances.
- Vous ne pouvez pas utiliser RDS Proxy avec une instance de base de données RDS for MySQL dont le paramètre `read_only` dans son groupe de paramètres de base de données est défini sur `1`.
- RDS Proxy ne prend pas en charge le mode compressé de MySQL. Par exemple, il ne prend pas en charge la compression utilisée par les options `--compress` ou `-C` de la commande `mysql`.
- Les connexions à la base de données qui traitent une commande `GET DIAGNOSTIC` peuvent renvoyer des informations inexactes lorsque le proxy RDS réutilise la même connexion à la base de données pour exécuter une autre requête. Cela peut se produire quand le proxy RDS multiplexe les connexions à la base de données.
- Certaines instructions et fonctions SQL `SET LOCAL` peuvent par exemple modifier l'état de la connexion sans provoquer d'épinglage. Pour connaître le comportement d'épinglage le plus récent, consultez la section Éviter d'épingler un proxy RDS.

- L'utilisation de la `ROW_COUNT()` fonction dans une requête à instructions multiples n'est pas prise en charge.
- Le proxy RDS ne prend pas en charge les applications clientes qui ne peuvent pas gérer plusieurs messages de réponse dans un seul enregistrement TLS.
- Le proxy RDS ne prend pas en charge les mots de passe doubles MySQL.
- Le proxy RDS peut ne pas fonctionner comme prévu lorsque vous configurez le `init_connect` paramètre dans votre groupe de paramètres de base de données RDS pour définir les variables d'état de session. Définissez plutôt la requête d'initialisation pour que votre proxy exécute les instructions d'initialisation de session lorsque vous utilisez le proxy pour vous connecter à votre base de données.

> **Important**
>
> Pour les proxies associés aux bases de données MySQL, ne définissez pas le paramètre de configuration `sql_auto_is_null` sur `true` ou sur une valeur différente de zéro dans la requête d'initialisation. Cela pourrait entraîner un comportement incorrect de l'application.

Limites supplémentaires pour RDS for PostgreSQL

Les limitations supplémentaires suivantes s'appliquent à RDS Proxy avec les bases de données RDS for PostgreSQL :

- RDS Proxy ne prend pas en charge les filtres d'épinglage de session pour PostgreSQL.
- Actuellement, tous des proxies écoutent sur le port 5432 pour PostgreSQL.
- Pour PostgreSQL, RDS Proxy ne prend actuellement pas en charge l'annulation d'une requête d'un client en émettant un `CancelRequest`. C'est le cas par exemple lorsque vous annulez une requête longue dans une session psql interactive à l'aide de Ctrl+C.
- Les résultats de la fonction PostgreSQL lastval ne sont pas toujours précis. Pour contourner ce problème, utilisez l'instruction INSERT avec la clause `RETURNING`.
- Le proxy RDS ne prend actuellement pas en charge le mode de réplication de streaming.
- Avec RDS pour PostgreSQL 16, les modifications apportées à `scram_iterations` la valeur ont un impact exclusif sur le processus d'authentification entre le proxy et la base de données. Plus précisément, si vous le configurez`scram-sha-256`, `ClientPasswordAuthType` les

personnalisations apportées à la `scram_iterations` valeur n'ont aucune influence sur l'authentification par client-to-proxy mot de passe. Au lieu de cela, la valeur d'itération pour l'authentification par client-to-proxy mot de passe est fixée à 4096.

- La `default` base de données doit exister.
- Si vous avez l'`ALTER ROLE`habitude de modifier le rôle de l'utilisateur avec`SET ROLE`, les connexions suivantes entre cet utilisateur et le proxy risquent de ne pas utiliser ce paramètre de rôle, si ces connexions sont épinglées. Pour éviter cela, lorsque vous utilisez un proxy, `SET ROLE` utilisez-le dans la requête d'initialisation du proxy. Pour plus d'informations, consultez la section Requête d'initialisation dansCréation d'un proxy pour Amazon RDS (Aurora).

> **Important**
>
> Pour des proxies existants avec des bases de données PostgreSQL, si vous modifiez l'authentification de la base de données pour utiliser uniquement `SCRAM`, le proxy devient indisponible pendant 60 secondes maximum. Pour éviter ce problème, effectuez l'une des actions suivantes :
>
> - Veillez à ce que la base de données permette à la fois l'authentification `SCRAM` et `MD5`.
> - Pour utiliser uniquement l'authentification `SCRAM`, créez un nouveau proxy, migrez le trafic de votre application vers ce nouveau proxy, puis supprimez le proxy précédemment associé à la base de données.

Planification Où utiliser RDS Proxy

Vous pouvez déterminer les instances de base de données, clusters et applications qui pourraient bénéficier le plus de l'utilisation de RDS Proxy. Pour ce faire, tenez compte des facteurs suivants :

- Il est judicieux d'associer à un proxy toute instance de base de données qui rencontre des erreurs liées à un « nombre de connexions trop élevé ». Cela se caractérise souvent par une valeur élevée de la `ConnectionAttempts` CloudWatch métrique. Le proxy permet aux applications d'ouvrir de nombreuses connexions client, tandis que le proxy gère un plus petit nombre de connexions à long terme à l'instance de base de données.
- Pour les d'instances de base de données qui utilisent des classes d'AWSinstance plus petites, telles que T2 ou T3, l'utilisation d'un proxy peut permettre d'éviter certaines out-of-memory conditions. Il peut également contribuer à réduire la surcharge de l'UC lors de l'établissement des

connexions. Ces conditions peuvent se produire lorsque vous faites face à un grand nombre de connexions.

- Vous pouvez surveiller certaines CloudWatch métriques Amazon pour déterminer si un d'instances de base de données approche certains types de limites. Ces limites concernent le nombre de connexions et la mémoire associées à la gestion des connexions. Vous pouvez également surveiller certaines CloudWatch mesures pour déterminer si un d'instances de base de données gère de nombreuses connexions de courte durée. L'ouverture et la fermeture de telles connexions peuvent entraîner une surcharge de performances sur votre base de données. Pour en savoir plus sur les métriques à surveiller, consultez Surveillance des métriques du proxy RDS avec Amazon CloudWatch.
- AWS LambdaIl peut également être judicieux d'utiliser les fonctions avec un proxy. Ces fonctions réalisent fréquemment des connexions de base de données courtes qui bénéficient du regroupement de connexions offert par RDS Proxy. Vous pouvez profiter de toute authentification IAM dont vous disposez déjà pour les fonctions Lambda, plutôt que de gérer les informations d'identification de base de données dans votre code d'application Lambda.
- Les applications qui ouvrent et ferment généralement un grand nombre de connexions à des bases de données et qui ne disposent pas de mécanismes intégrés de regroupement des connexions sont de bons candidats pour l'utilisation d'un proxy.
- Il est souvent judicieux d'utiliser les applications qui maintiennent un grand nombre de connexions ouvertes pendant de longues périodes avec un proxy. Les applications dans des secteurs tels que le logiciel en tant que service (SaaS) ou le e-commerce réduisent souvent la latence pour les demandes de base de données en laissant les connexions ouvertes. Avec RDS Proxy, une application peut maintenir plus de connexions ouvertes qu'elle ne le peut lorsqu'elle se connecte directement au d'instances de base de données.
- Vous n'avez peut-être pas adopté l'authentification IAM et Secrets Manager en raison de la complexité de la configuration d'une telle authentification pour toutes les instances de base de données. Le cas échéant, vous pouvez garder les méthodes d'authentification existantes et déléguer l'authentification à un proxy. Le proxy peut appliquer les politiques d'authentification relatives aux connexions client pour des applications spécifiques. Vous pouvez profiter de toute authentification IAM dont vous disposez déjà pour les fonctions Lambda, plutôt que de gérer les informations d'identification de base de données dans votre code d'application Lambda.
- RDS Proxy peut aider à rendre les applications plus résilientes et plus transparentes face aux pannes de bases de données. RDS Proxy contourne les caches du système de nom de domaine (DNS) afin de réduire les temps de basculement jusqu'à 66 % pour les instances de base de données Amazon RDS Multi-AZ. RDS Proxy achemine automatiquement le trafic vers une nouvelle

instance de base de données tout en préservant les connexions aux applications. Cela rend les basculements plus transparents pour les applications.

RDSConcepts et terminologie relatifs aux proxys

Vous pouvez simplifier la gestion des connexions pour vos instances de base de RDS données Amazon () à l'aide d'RDSun proxy.

RDSLe proxy gère le trafic réseau entre l'application cliente et la base de données. Il le fait d'abord de manière active en comprenant le protocole de la base de données. Il ajuste ensuite son comportement en fonction SQL des opérations de votre application et des jeux de résultats de la base de données.

RDSLe proxy réduit la mémoire et les CPU frais liés à la gestion des connexions sur votre base de données. La base de données a besoin de moins de mémoire et de CPU ressources lorsque les applications ouvrent de nombreuses connexions simultanées. La logique n'est pas non plus nécessaire dans vos applications pour fermer et rouvrir les connexions qui restent inactives pendant longtemps. De même, il faut logique d'application moindre pour rétablir les connexions en cas de problème de base de données.

L'infrastructure du RDS proxy est hautement disponible et déployée sur plusieurs zones de disponibilité (AZs). Le calcul, la mémoire et le stockage du RDS proxy sont indépendants de votre instance de RDS base de données . Cette séparation permet de réduire la surcharge sur vos serveurs de base de données, afin qu'ils puissent dédier leurs ressources à la gestion des charges de travail de base de données. Les ressources de calcul du RDS proxy sont sans serveur et s'adaptent automatiquement en fonction de la charge de travail de votre base de données.

Rubriques

- Vue d'ensemble des concepts de RDS proxy
- Regroupement de connexions
- RDSSécurité du proxy
- Basculement
- Transactions

Vue d'ensemble des concepts de RDS proxy

RDSLe proxy gère l'infrastructure pour effectuer le regroupement des connexions et les autres fonctionnalités décrites dans les sections qui suivent. Les proxys sont représentés dans la RDS console sur la page Proxies.

Chaque proxy gère les connexions à une instance de RDS base de données () unique. Le proxy détermine automatiquement l'instance d'écriture actuelle pour les instances de base de données RDS multi-AZ ou les clusters provisionnés par .

Les connexions qu'un proxy maintient ouvertes et disponibles pour que vos applications de base de données puissent les utiliser constituent le pool de connexions.

Par défaut, RDS Proxy peut réutiliser une connexion après chaque transaction de votre session. « multiplexage » est le terme utilisé pour cette réutilisation au niveau de la transaction. Lorsque le RDS proxy supprime temporairement une connexion du pool de connexions pour la réutiliser, cette opération est appelée emprunt de la connexion. Lorsque cela est possible en toute sécurité, le RDS proxy renvoie cette connexion au pool de connexions.

Dans certains cas, le RDS proxy ne peut pas être sûr de pouvoir réutiliser une connexion à une base de données en dehors de la session en cours. Dans ce cas, il maintient la session sur la même connexion jusqu'à la fin. Ce comportement de secours est appelé épinglage.

Un proxy a un point de terminaison par défaut. Vous vous connectez à ce point de terminaison lorsque vous travaillez avec une instance de base de RDS données Amazon, un . Vous utilisez cette opération plutôt que de vous connecter au point de terminaison en lecture-écriture qui se connecte directement à l'instance . Pour les , vous pouvez également créer des points de terminaison supplémentaires en lecture/écriture et en lecture seule. Pour de plus amples informations, veuillez consulter Présentation des points de terminaison proxy.

Par exemple, vous pouvez toujours vous connecter au point de terminaison du cluster pour les connexions en lecture-écriture sans regroupement de connexions. Vous pouvez toujours vous connecter au point de terminaison du lecteur pour des connexions en lecture seule à charge équilibrée. Vous pouvez toujours vous connecter aux points de terminaison de l'instance pour le diagnostic et le dépannage d'instances de base de données spécifiques d'un cluster. Si vous utilisez d'autres AWS services, par exemple pour vous connecter AWS Lambda à des RDS bases de données, modifiez leurs paramètres de connexion pour utiliser le point de terminaison du proxy. Par exemple, vous spécifiez le point de terminaison du proxy pour autoriser les fonctions Lambda à accéder à votre base de données tout en tirant parti des fonctionnalités du RDS proxy.

Chaque proxy contient un groupe cible. Ce groupe cible incarne le RDScluster de base auquel le proxy peut se connecter. Les RDSinstances de base de associées à un proxy sont appelées cibles de ce proxy. Pour plus de commodité, lorsque vous créez un proxy via la console, le RDS proxy crée également le groupe cible correspondant et enregistre automatiquement les cibles associées.

Une famille de moteurs est un ensemble associé de moteurs de base de données qui utilisent le même protocole de base de données. Vous choisissez la famille de moteurs pour chaque proxy que vous créez.

Regroupement de connexions

Chaque proxy effectue un regroupement de connexions séparément pour l'instance d'écriture et de lecture de sa RDSbase de données associée. Le regroupement de connexions est une optimisation qui réduit la surcharge associée à l'ouverture et à la fermeture des connexions, tout en maintenant plusieurs connexions ouvertes simultanément. Cette surcharge inclut la mémoire nécessaire pour gérer chaque nouvelle connexion. Cela implique également des CPU frais supplémentaires pour fermer chaque connexion et en ouvrir une nouvelle. Les exemples incluent la prise de mains Security/Secure Sockets Layer (TLS/SSL (couche de transport), l'authentification, les capacités de négociation, etc. Le regroupement de connexions simplifie la logique de votre application. Vous n'avez pas besoin d'écrire de code d'application pour minimiser le nombre de connexions ouvertes simultanées.

Chaque proxy effectue également le multiplexage de connexion, également connu sous le nom de réutilisation de connexion. Avec le multiplexage, RDS Proxy exécute toutes les opérations d'une transaction en utilisant une connexion de base de données sous-jacente. RDSvous pouvez ensuite utiliser une autre connexion pour la transaction suivante. Si vous ouvrez de nombreuses connexions simultanées au proxy, celui-ci conserve un plus petit nombre de connexions ouvertes à l'instance ou au cluster de base de données. Cela permet de réduire davantage la surcharge de mémoire pour les connexions sur le serveur de base de données. Cette technique réduit également le risque que des erreurs liées au « nombre de connexions trop élevé » se produisent.

RDSSécurité du proxy

RDSLe proxy utilise les mécanismes RDS de sécurité existants tels queTLS/SSLet AWS Identity and Access Management (IAM). Pour obtenir des informations générales sur ces fonctionnalités de sécurité, reportez-vous à la section Sécurité dans Amazon RDS. Assurez-vous également de vous familiariser avec le fonctionnement d'RDS en matière d'authentification, d'autorisation et d'autres domaines de sécurité.

RDSLe proxy peut servir de couche de sécurité supplémentaire entre les applications clientes et la base de données sous-jacente. Par exemple, vous pouvez vous connecter au proxy à l'aide de la version TLS 1.3, même si l'instance de base de données sous-jacente prend en charge une ancienne version deTLS. Vous pouvez vous connecter au proxy à l'aide d'un IAM rôle. Il en est ainsi même si le proxy se connecte à la base de données à l'aide de la méthode native d'authentification par nom d'utilisateur et mot de passe. Grâce à cette technique, vous pouvez appliquer de fortes exigences d'authentification pour les applications de base de données sans avoir à fournir un effort de migration coûteux pour les instances de base de données elles-mêmes.

Vous stockez les informations d'identification de base de données utilisées par le RDS proxy dans AWS Secrets Manager. Chaque utilisateur de base de données du RDScluster de base de données d'instance auquel un proxy accède doit disposer d'un secret correspondant dans Secrets Manager. Vous pouvez également configurer IAM l'authentification pour les utilisateurs du RDS proxy. Ce faisant, vous pouvez appliquer l'IAMauthentification pour l'accès aux bases de données, même si les bases de données utilisent l'authentification par mot de passe native. Nous vous recommandons d'utiliser ces fonctions de sécurité au lieu d'intégrer les informations d'identification de base de données dans votre code d'application.

UtiliserTLS/SSLavec un RDS proxy

Vous pouvez vous connecter au RDS proxy à l'aide du SSL protocoleTLS/.

> **Note**
>
> RDSLe proxy utilise les certificats du AWS Certificate Manager (ACM). Si vous utilisez un RDS proxy, vous n'avez pas besoin de télécharger les RDS certificats Amazon ni de mettre à jour les applications qui utilisent des connexions RDS proxy.

TLSPour appliquer toutes les connexions entre le proxy et votre base de données, vous pouvez spécifier un paramètre Require Transport Layer Security lorsque vous créez ou modifiez un proxy dans le AWS Management Console.

RDSLe proxy peut également garantir que votre session utiliseTLS/SSLentre votre client et le point de terminaison du RDS proxy. Pour que RDS Proxy le fasse, spécifiez les exigences côté client. SSLles variables de session ne sont pas définies pour les SSL connexions à une base de données à l'aide d'un RDS proxy.

- RDSPour My SQL , spécifiez les exigences côté client avec le `--ssl-mode` paramètre lorsque vous exécutez la `mysql` commande.
- Pour Amazon RDS Postgre SQL , spécifiez-le dans le `sslmode=require` cadre de la `conninfo` chaîne lorsque vous exécutez la `psql` commande.

RDSLe proxy prend en charge les versions de TLS protocole 1.0, 1.1, 1.2 et 1.3. Vous pouvez vous connecter au proxy à l'aide d'une version supérieure à TLS celle que vous utilisez dans la base de données sous-jacente.

Par défaut, les programmes clients établissent une connexion cryptée avec le RDS proxy, avec un contrôle supplémentaire disponible via `--ssl-mode` cette option. Du côté client, RDS Proxy prend en charge tous les SSL modes.

Pour le client, les SSL modes sont les suivants :

PREFERRED

SSLest le premier choix, mais ce n'est pas obligatoire.

DISABLED

Non SSL est autorisé.

REQUIRED

Faire appliquerSSL.

VERIFY_CA

Appliquez SSL et vérifiez l'autorité de certification (CA).

VERIFY_IDENTITY

Appliquez SSL et vérifiez l'autorité de certification et le nom d'hôte de l'autorité de certification.

Lorsque vous utilisez un client avec `--ssl-mode` `VERIFY_CA` ou `VERIFY_IDENTITY`, spécifiez l'option `--ssl-ca` pointant vers une autorité de certification au format `.pem`. Pour utiliser le `.pem` fichier, téléchargez toutes les autorités PEMs de certification racine depuis Amazon Trust Services et placez-les dans un seul `.pem` fichier.

RDSLe proxy utilise des certificats génériques, qui s'appliquent à la fois à un domaine et à ses sous-domaines. Si vous utilisez le `mysql` client pour vous connecter en SSL mode`VERIFY_IDENTITY`, vous devez actuellement utiliser la commande SQL compatible My 8.0`mysql`.

Basculement

Le basculement est une fonction de haute disponibilité qui remplace une instance de base de données par une autre lorsque l'instance d'origine est indisponible. Un problème lié à une instance de base de données peut entraîner un basculement. Celui-ci peut également faire partie de procédures de maintenance normales, lors d'une mise à niveau de la base de données par exemple. Le basculement s'applique aux RDS instances de base de données dans une configuration multi-AZ.

La connexion via un proxy permet à vos applications de mieux résister aux basculements de bases de données. Lorsque l'instance de base de données d'origine devient indisponible, le RDS proxy se connecte à la base de données de secours sans interrompre les connexions aux applications inactives. Cela permet d'accélérer et de simplifier le processus de basculement. Cela perturbe moins votre application qu'un redémarrage classique ou un problème de base de données.

Sans RDS proxy, un basculement implique une brève interruption de service. Pendant la panne, vous ne pouvez pas effectuer d'opérations d'écriture sur la base de données en cas de basculement. Toutes les connexions de base de données existantes sont interrompues et votre application doit les rouvrir. La base de données est ouverte à de nouvelles connexions et opérations d'écriture lorsqu'une instance de base de données en lecture seule est promue pour remplacer celle qui n'est pas disponible.

Pendant les basculements de base de données, le RDS proxy continue d'accepter les connexions à la même adresse IP et dirige automatiquement les connexions vers la nouvelle instance de base de données principale. Les clients qui se connectent via un RDS proxy ne sont pas sensibles aux risques suivants :

- Retards de propagation du système de noms de domaine (DNS) en cas de basculement.
- DNSMise en cache locale.
- Délai d'expiration de connexion.
- Incertitude concernant l'instance de base de données qui est le rédacteur en cours.
- Attente d'une réponse à la requête d'un ancien rédacteur devenu indisponible sans fermer les connexions.

Pour les applications qui gèrent leur propre pool de connexions, le fait de passer par un RDS proxy signifie que la plupart des connexions restent actives en cas de basculement ou d'autres perturbations. Seules les connexions en cours de transaction ou de SQL relevé sont annulées.

RDS Le proxy accepte immédiatement les nouvelles connexions. Lorsque le rédacteur de base de données n'est pas disponible, le RDS proxy place les demandes entrantes dans la file d'attente.

Pour les applications qui ne gèrent pas leurs propres pools de connexions, RDS Proxy offre des débits de connexion plus rapides et des connexions plus ouvertes. Il permet de réduire la surcharge coûteuse des reconnexions fréquentes à la base de données. Pour ce faire, il réutilise les connexions de base de données conservées dans le pool de connexions RDS proxy. Cette approche est particulièrement importante pour les TLS connexions, où les coûts d'installation sont importants.

Transactions

Toutes les instructions d'une seule transaction utilisent toujours la même connexion à la base de données sous-jacente. La connexion devient disponible pour une session différente lorsque la transaction se termine. Voici les conséquences de l'utilisation de la transaction en tant qu'unité de granularité :

- La réutilisation des connexions peut avoir lieu après chaque instruction individuelle lorsque le SQL `autocommit` paramètre RDSfor My SQL est activé.
- Inversement, lorsque le paramètre `autocommit` est désactivé, la première instruction que vous émettez dans une session lance une nouvelle transaction. Supposons, par exemple, que vous saisissiez une séquence d'instructions `SELECT`,`INSERT`,`UPDATE`, et d'autres instructions du langage de manipulation des données (DML). Dans ce cas, la réutilisation de la connexion ne se produit que lorsque vous émettez `COMMIT`, `ROLLBACK` ou que vous mettez fin à la transaction.
- La saisie d'une instruction Data Definition Language (DDL) entraîne la fin de la transaction une fois cette instruction terminée.

RDSLe proxy détecte la fin d'une transaction via le protocole réseau utilisé par l'application cliente de base de données. La détection des transactions ne repose pas sur des mots clés tels que `COMMIT` ou `ROLLBACK` apparaissant dans le texte de la SQL déclaration.

Dans certains cas, le RDS proxy peut détecter une demande de base de données qui rend impossible le déplacement de votre session vers une autre connexion. Dans ce cas, il désactive le multiplexage pour cette connexion pendant le reste de votre session. La même règle s'applique si le RDS proxy ne peut pas être certain que le multiplexage est pratique pour la session. Cette opération est appelée épinglage. Pour savoir comment détecter et réduire l'épinglage, consultez Éviter d'épingler un proxy RDS.

Commencer à utiliser RDS Proxy

Utilisez les informations des pages suivantes pour configurer, gérer Proxy Amazon RDS et définir les options de sécurité associées. Les options de sécurité contrôlent qui peut accéder à chaque proxy et comment chaque proxy se connecte aux instances de base de données.

Si vous utilisez RDS Proxy pour la première fois, nous vous recommandons de suivre les pages dans l'ordre dans lequel nous les présentons.

Rubriques

- Configuration des prérequis réseau pour le proxy RDS
- Configuration des informations d'identification de base de données AWS Secrets Manager pour le proxy RDS
- Configuration de l'authentification IAM pour le proxy RDS
- Création d'un proxy pour Amazon RDS (Aurora)
- Afficher un proxy
- Connexion à une base de données via RDS Proxy

Configuration des prérequis réseau pour le proxy RDS

L'utilisation du proxy RDS nécessite que vous disposiez d'un cloud privé virtuel (VPC) commun entre votre instance de base de données RDS, le cluster de base de données et le proxy RDS. Ce VPC doit avoir au moins deux sous-réseaux situés dans des zones de disponibilité différentes. Votre compte peut posséder ces sous-réseaux ou les partager avec d'autres comptes. Pour plus d'informations sur le partage VPC, consultez la section Travailler avec le partage. VPCs

Les ressources de votre application cliente telles qu'Amazon EC2, Lambda ou Amazon ECS peuvent se trouver dans le même VPC que le proxy. Elles peuvent également se trouver dans un VPC distinct du proxy. Si vous êtes bien connecté à des instances de base de données RDS , vous disposez déjà des ressources réseau requises.

Rubriques

- Obtention d'informations sur vos sous-réseaux
- Planification de la capacité des adresses IP

Obtention d'informations sur vos sous-réseaux

Pour créer un proxy, vous devez fournir les sous-réseaux et le VPC dans lesquels le proxy fonctionne. L'exemple Linux suivant montre des AWS CLI commandes qui examinent les sous-réseaux VPCs et appartenant à votre Compte AWS. En particulier, vous transmettez le sous-réseau IDs en tant que paramètres lorsque vous créez un proxy à l'aide de la CLI.

```
aws ec2 describe-vpcs
aws ec2 describe-internet-gateways
aws ec2 describe-subnets --query '*[].[VpcId,SubnetId]' --output text | sort
```

L'exemple Linux suivant montre des AWS CLI commandes permettant de déterminer le sous-réseau IDs correspondant à un cluster de base de données de base de données RDS spécifique. Trouvez l'ID VPC de l'instance de base de données. Examinez le VPC pour trouver ses sous-réseaux. L'exemple Linux suivant montre comment procéder.

```
$ #From the DB instance, trace through the DBSubnetGroup and Subnets to find the subnet
 IDs.
$ aws rds describe-db-instances --db-instance-identifier my_instance_id --query '*[].
[DBSubnetGroup]||[0]||[0]||[Subnets]||[0]||[*].SubnetIdentifier' --output text
```

```
subnet_id_1
subnet_id_2
subnet_id_3
...
```

```
$ #From the DB instance, find the VPC.
$ aws rds describe-db-instances --db-instance-identifier my_instance_id --query '*[].
[DBSubnetGroup]||[0]||[0].VpcId' --output text
```

```
my_vpc_id
```

```
$ aws ec2 describe-subnets --filters Name=vpc-id,Values=my_vpc_id --query '*[].
[SubnetId]' --output text
```

```
subnet_id_1
subnet_id_2
subnet_id_3
subnet_id_4
```

```
subnet_id_5
subnet_id_6
```

Planification de la capacité des adresses IP

Un RDS Proxy ajuste automatiquement sa capacité selon les besoins en fonction de la taille et du nombre d'instances de base de données enregistrées auprès de lui. Certaines opérations peuvent également nécessiter une capacité de proxy supplémentaire, telle que l'augmentation de la taille d'une base de données enregistrée ou des opérations de maintenance internes du proxy RDS. Au cours de ces opérations, votre proxy peut avoir besoin de plus d'adresses IP pour fournir de la capacité supplémentaire. Ces adresses supplémentaires permettent à votre proxy d'évoluer sans affecter votre charge de travail. L'absence d'adresses IP libres dans vos sous-réseaux empêche d'augmenter un proxy. Cela peut entraîner des latences de requêtes plus élevées ou des échecs de connexion client. RDS vous avertit par le biais de l'événement `RDS-EVENT-0243` lorsqu'il n'y a pas suffisamment d'adresses IP libres dans vos sous-réseaux. Pour plus d'informations sur cet événement, consultez Utilisation des événements RDS proxy.

Réservez le nombre minimum d'adresses IP libres suivant dans vos sous-réseaux pour votre proxy, en fonction de la taille des classes d'instances de base de données.

Classe d'instances de base de données	Nombre minimal d'adresses IP libres
db.*.xlarge ou moins	10
db.*.24xlarge	15
db.*.24xlarge	25
db.*.24xlarge	45
db.*.24xlarge	60
db.*.24xlarge	75
db.*.24xlarge	110

Ces nombres d'adresses IP recommandés sont des estimations pour un proxy avec uniquement le point de terminaison par défaut. Un proxy avec des points de terminaison supplémentaires ou des réplicas en lecture peut avoir besoin d'un plus grand nombre d'adresses IP libres. Pour

chaque point de terminaison supplémentaire, nous vous recommandons de réserver trois adresses IP supplémentaires. Pour chaque réplica en lecture, nous vous recommandons de réserver des adresses IP supplémentaires, comme indiqué dans le tableau, en fonction de la taille de ce réplica en lecture.

> **Note**
>
> Le proxy RDS ne consomme pas plus de 215 adresses IP dans un VPC.

Configuration des informations d'identification de base de données AWS Secrets Manager pour le proxy RDS

Le proxy RDS dans Amazon RDS est utilisé AWS Secrets Manager pour stocker et gérer les informations d'identification de base de données en toute sécurité. Au lieu d'intégrer des informations d'identification dans votre application, vous associez un proxy à un secret Secrets Manager contenant les informations d'authentification nécessaires. Vous créez un secret Secrets Manager distinct pour chaque compte utilisateur de base de données auquel le proxy se connecte sur le RDS.

Rubriques

- Création de secrets à utiliser avec RDS Proxy

Création de secrets à utiliser avec RDS Proxy

Avant de créer un proxy, vous devez d'abord créer au moins un secret qui stocke les informations d'identification de votre base de données.

console

Pour créer un secret

1. Ouvrez la console Secrets Manager à l'adresse https://console.aws.amazon.com/secretsmanager/.
2. Choisissez Store a new secret (Stocker un nouveau secret).
3. Choisissez Credentials for Amazon RDS database.
4. Entrez un nom d'utilisateur et un mot de passe. Les informations d'identification que vous entrez doivent correspondre aux informations d'identification d'un utilisateur de base de données existant dans la base de données RDS associée. Le proxy RDS utilise ces informations

d'identification pour authentifier et établir des connexions à la base de données pour le compte des applications.

En cas de non-concordance, vous pouvez mettre à jour le secret pour qu'il corresponde au mot de passe de la base de données. Tant que vous ne mettez pas à jour le secret, les tentatives de connexion via le proxy utilisant ce secret échouent, mais les connexions utilisant d'autres secrets valides fonctionnent toujours.

> **Note**
>
> Pour RDS for SQL Server, RDS Proxy nécessite un secret distinguant majuscules et minuscules dans Secrets Manager, quels que soient les paramètres de classement des instances de base de données. Si votre application autorise des noms d'utilisateur avec des majuscules différentes, tels que « Admin » et « admin », vous devez créer des secrets distincts pour chacun d'eux. RDS Proxy ne prend pas en charge l'authentification du nom d'utilisateur sans distinction majuscules/majuscules entre le client et le proxy. Pour plus d'informations sur le classement SQL Server, consultez la documentation Microsoft SQL Server.

5. Pour Base de données, sélectionnez la base de données Amazon RDS à laquelle le secret doit accéder.
6. Renseignez les autres paramètres pour le secret, puis choisissez Store. Pour obtenir des instructions complètes, reportez-vous à la section Création d'un AWS Secrets Manager secret dans le guide de AWS Secrets Manager l'utilisateur.

AWS CLI

Lorsque vous créez un proxy via le AWS CLI, vous spécifiez les Amazon Resource Names (ARNs) des secrets correspondants. Vous le faites pour tous les comptes utilisateur de base de données auxquels le proxy peut accéder. Dans le AWS Management Console, vous choisissez les secrets par leurs noms descriptifs.

- Pour créer un secret Secrets Manager à utiliser avec le proxy RDS, utilisez la commande create-secret :

```
aws secretsmanager create-secret \
  --name "secret_name" \
  --description "secret_description" \
```

```
  --region region_name \
  --secret-string '{"username":"db_user","password":"db_user_password"}'
```

- Vous pouvez également créer une clé personnalisée pour chiffrer le secret de votre Secrets Manager. La commande suivante crée un exemple de clé.

```
aws kms create-key --description "test-key" --policy '{
  "Id":"kms-policy",
  "Version":"2012-10-17",
  "Statement":
    [
      {
        "Sid":"Enable IAM User Permissions",
        "Effect":"Allow",
        "Principal":{"AWS":"arn:aws:iam::account_id:root"},
        "Action":"kms:*","Resource":"*"
      },
      {
        "Sid":"Allow access for Key Administrators",
        "Effect":"Allow",
        "Principal":
          {
            "AWS":
              ["$USER_ARN","arn:aws:iam:account_id::role/Admin"]
          },
        "Action":
          [
            "kms:Create*",
            "kms:Describe*",
            "kms:Enable*",
            "kms:List*",
            "kms:Put*",
            "kms:Update*",
            "kms:Revoke*",
            "kms:Disable*",
            "kms:Get*",
            "kms:Delete*",
            "kms:TagResource",
            "kms:UntagResource",
            "kms:ScheduleKeyDeletion",
            "kms:CancelKeyDeletion"
          ],
        "Resource":"*"
```

```
      },
      {
        "Sid":"Allow use of the key",
        "Effect":"Allow",
        "Principal":{"AWS":"$ROLE_ARN"},
        "Action":["kms:Decrypt","kms:DescribeKey"],
        "Resource":"*"
      }
    ]
}'
```

Par exemple, les commandes suivantes créent des secrets Secrets Manager pour deux utilisateurs de base de données :

```
aws secretsmanager create-secret \
  --name secret_name_1 --description "db admin user" \
  --secret-string '{"username":"admin","password":"choose_your_own_password"}'

aws secretsmanager create-secret \
  --name secret_name_2 --description "application user" \
  --secret-string '{"username":"app-user","password":"choose_your_own_password"}'
```

Pour créer ces secrets chiffrés avec votre AWS KMS clé personnalisée, utilisez les commandes suivantes :

```
aws secretsmanager create-secret \
  --name secret_name_1 --description "db admin user" \
  --secret-string '{"username":"admin","password":"choose_your_own_password"}' \
  --kms-key-id arn:aws:kms:us-east-2:account_id:key/key_id

aws secretsmanager create-secret \
  --name secret_name_2 --description "application user" \
  --secret-string '{"username":"app-user","password":"choose_your_own_password"}' \
  --kms-key-id arn:aws:kms:us-east-2:account_id:key/key_id
```

Pour voir les secrets détenus par votre AWS compte, utilisez la commande list-secrets :

```
aws secretsmanager list-secrets
```

Lorsque vous créez un proxy à l'aide de la CLI, vous transmettez les Amazon Resource Names (ARNs) d'un ou de plusieurs secrets au `--auth` paramètre. L'exemple suivant montre comment préparer un rapport avec uniquement le nom et l'ARN de chaque secret détenu par votre AWS compte. Cet exemple utilise le `--output table` paramètre disponible dans la AWS CLI version 2. Si vous utilisez AWS CLI la version 1, utilisez `--output text` plutôt.

```
aws secretsmanager list-secrets --query '*[].[Name,ARN]' --output table
```

Pour vérifier que le secret contient les informations d'identification correctes au format approprié, utilisez la get-secret-valuecommande. Remplacez *your_secret_name* par le nom abrégé ou l'ARN du secret.

```
aws secretsmanager get-secret-value --secret-id your_secret_name
```

La sortie contient une ligne avec une valeur codée en JSON similaire à la suivante :

```
...
"SecretString": "{\"username\":\"your_username\",\"password\":\"your_password\"}",
...
```

Configuration de l'authentification IAM pour le proxy RDS

Pour configurer l'authentification AWS Identity and Access Management (IAM) pour le proxy RDS dans Amazon RDS, créez et configurez une politique IAM qui accorde les autorisations nécessaires. Le proxy RDS permet de gérer les informations d'identification de la base de données de manière sécurisée, ce qui permet aux applications de s'authentifier via le proxy sans gérer directement les informations d'identification. AWS Secrets Manager

Cette rubrique décrit les étapes de configuration de l'authentification IAM pour le proxy RDS, notamment la création de la politique IAM requise et son association à un rôle IAM.

> **Tip**
>
> Cette procédure n'est nécessaire que si vous souhaitez créer votre propre rôle IAM. Sinon, RDS peut créer automatiquement le rôle requis lorsque vous configurez le proxy. Vous pouvez donc ignorer ces étapes.

Prérequis

Avant de configurer l'authentification IAM pour le proxy RDS, assurez-vous que vous disposez des éléments suivants :

- AWS Secrets Manager— Au moins un secret stocké contenant les informations d'identification de la base de données. Pour obtenir des instructions sur la création de secrets, reportez-vous àthe section called “Configuration des informations d'identification de base de données”.
- Autorisations IAM : rôle ou utilisateur IAM autorisé à créer et à gérer des politiques, des rôles et des secrets IAM dans. AWS Secrets Manager

Création d'une politique IAM pour l'accès à Secrets Manager

Pour permettre au proxy RDS de récupérer les informations d'identification de la base de données auprès de Secrets Manager, créez un rôle IAM avec une politique qui accorde les autorisations nécessaires.

console

Pour créer un rôle permettant d'accéder à vos secrets à utiliser avec votre proxy

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Créez une politique d'autorisation pour le rôle. Pour les étapes générales, voir Créer des politiques IAM (console).

   Collez cette politique dans l'éditeur JSON et apportez les modifications suivantes :

   - Indiquez votre propre ID de compte.
   - Remplacez `us-east-2` par la région où résidera le mandataire.
   - Remplacez les noms secrets par ceux que vous avez créés. Pour plus d'informations, consultez la section Spécification des clés KMS dans les déclarations de politique IAM.
   - Remplacez l'ID de clé KMS par celui que vous avez utilisé pour chiffrer les secrets de Secrets Manager, qu'il s'agisse de la clé par défaut ou de votre propre clé.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": [
                "arn:aws:secretsmanager:us-
east-2:111122223333:secret:secret_name_1",
                "arn:aws:secretsmanager:us-
east-2:111122223333:secret:secret_name_2"
            ]
        },
        {
            "Effect": "Allow",
            "Action": "kms:Decrypt",
            "Resource": "arn:aws:kms:us-east-2:111122223333:key/key_id",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-
east-2.amazonaws.com"
                }
            }
        }
    ]
}
```

3. Créez le rôle et associez-y la politique d'autorisations. Pour les étapes générales, voir Créer un rôle pour déléguer des autorisations à un AWS service.

   Pour le type d'entité de confiance, choisissez AWS service. Sous Cas d'utilisation, sélectionnez RDS et choisissez RDS - Ajouter un rôle à la base de données pour le cas d'utilisation.

4. Pour les politiques d'autorisations, choisissez la politique que vous avez créée.

5. Pour Sélectionner les entités de confiance, entrez la politique de confiance suivante pour le rôle :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "rds.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

AWS CLI

Pour créer le rôle à l'aide du AWS CLI, envoyez la demande suivante :

```
aws iam create-role \
  --role-name my_role_name \
  --assume-role-policy-document '{"Version":"2012-10-17","Statement":
[{"Effect":"Allow","Principal":{"Service":
["rds.amazonaws.com"]},"Action":"sts:AssumeRole"}]}'
```

Attachez ensuite la politique au rôle :

```
aws iam put-role-policy \
  --role-name my_role_name \
  --policy-name secret_reader_policy \
  --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "secretsmanager:GetSecretValue",
            "Resource": [
                "arn:aws:secretsmanager:us-east-2:account_id:secret:secret_name_1",
```

```
                "arn:aws:secretsmanager:us-east-2:account_id:secret:secret_name_2"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "kms:Decrypt",
            "Resource": "arn:aws:kms:us-east-2:account_id:key/key_id",
            "Condition": {
                "StringEquals": {
                    "kms:ViaService": "secretsmanager.us-east-2.amazonaws.com"
                }
            }
        }
    ]
}'
```

Une fois le rôle et les autorisations IAM configurés, vous pouvez désormais créer un proxy et l'associer à ce rôle. Cela permet au proxy de récupérer les informations d'identification de la base de données en toute sécurité AWS Secrets Manager et d'activer l'authentification IAM pour vos applications. Pour obtenir des instructions, veuillez consulter the section called “Création d'un proxy”.

Création d'un proxy pour Amazon RDS (Aurora)

Vous pouvez employer un proxy avec une instance de base de données RDS for MariaDB, RDS for Microsoft SQL Server, RDS for MySQL ou RDS for PostgreSQL.

console

Pour créer un proxy

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Choisissez Création d'un proxy.
4. Configurez les paramètres suivants pour votre proxy.

Paramètre	Description
Famille de moteurs	Protocole réseau de base de données reconnu par le proxy lorsqu'il interprète le trafic réseau à destination et en provenance de la base de données. **Note** Pour utiliser RDS pour PostgreSQL PostgreSQL, assurez-vous de conserver la base de données dans votre instance. `postgres` Consultez Résolution des problèmes liés à la `postgres` suppression.
Identifiant du proxy	Un nom unique au sein de votre identifiant de AWS compte et de votre AWS région actuelle.
Expiration de la connexion client inactive	Le proxy ferme une connexion client si celle-ci reste inactive pendant une période définie. Par défaut, il s'agit de 1 800 secondes (30 minutes). Une connexion est inactive lorsque l'application ne soumet pas de nouvelle demande dans le délai spécifié après avoir traité la demande précédente. Le proxy maintient la connexion à la base de données sous-jacente ouverte et la renvoie au pool de connexions, la rendant ainsi disponible pour les nouvelles connexions client. Pour supprimer les connexions périmées de manière proactive , réduisez le délai d'inactivité des connexions client. Pour minimiser les coûts de connexion en cas de pics de charge de travail, augmentez le délai d'expiration.

Paramètre	Description
Database (Base de données)	Le de données RDS auquel accéder via ce proxy. La liste inclut uniquement les instances et les clusters de base de données dotés de moteurs de base de données, de versions de moteur et d'autres paramètres compatibles. Si la liste est vide, créez une instance ou un cluster de base de données compatible avec RDS Proxy. Pour ce faire, suivez la procédure décrite dans Création d'une instance de base de données Amazon RDS. Réessayez ensuite de créer le proxy.
Nombre maximal de connexions du pool de connexions	Une valeur comprise entre 1 et 100 pour définir le pourcentage de `max_connections` limite que le proxy RDS peut utiliser. Si vous avez l'intention d'utiliser un seul proxy avec cette instance ou ce cluster de base de données, définissez cette valeur sur 100. Pour plus d'informations sur la façon dont le proxy RDS utilise ce paramètre, consultezthe section called “MaxConnectionsPercent”.

Paramètre	Description
Filtres d'épinglage de session	Empêche le proxy RDS d'épingler certains états de session détectés, ce qui contourne les mesures de sécurité par défaut pour le multiplexage des connexions. Actuellement, PostgreSQL ne prend pas en charge ce paramètre, et la seule option disponible est. `EXCLUDE_VARIABLE_SETS` Son activation peut affecter les variables de session d'une connexion sur les autres, ce qui peut entraîner des erreurs ou des problèmes d'exactitude si les requêtes reposent sur des variables de session définies en dehors de la transaction en cours. N'utilisez cette option qu'après avoir confirmé que vos applications peuvent partager des connexions de base de données en toute sécurité. Les modèles suivants sont considérés comme sûrs : • `SET`instructions dans lesquelles aucune modification n'est apportée à la valeur effective de la variable de session. En d'autres termes, aucune modification n'est apportée à la variable de session. • Vous modifiez la valeur de la variable de session et exécutez une instruction dans la même transaction. Pour de plus amples informations, veuillez consulter Éviter d'épingler un proxy RDS.
Expiration du délai d'emprunt de connexion	Si vous pensez que le proxy utilisera toutes les connexions de base de données disponibles, définissez le temps d'attente avant qu'il ne renvoie une erreur de temporisation. Vous pouvez spécifier jusqu'à cinq minutes. Ce paramètre s'applique uniquement lorsque le proxy a atteint le nombre maximum de connexions et que toutes sont en cours d'utilisation.

Paramètre	Description
Requête d'initialisation	(Facultatif) Ajoutez une requête d'initialisation ou modifiez la requête en cours. Vous pouvez spécifier une ou plusieurs instructions SQL que le proxy doit exécuter lors de l'ouverture de chaque nouvelle connexion à la base de données. Le paramètre est généralement utilisé avec SET des instructions pour s'assurer que chaque connexion possède des paramètres identiques. Assurez-vous que la requête que vous ajoutez est valide. Pour inclure plusieurs variables dans une seule SET instruction, utilisez des séparateurs de virgules. Par exemple : `SET variable1 =value1, variable2 =value2` Pour plusieurs instructions, utilisez des points-virgules comme séparateur. **Important** Comme vous pouvez accéder à la requête d'initialisation dans le cadre de la configuration du groupe cible, elle n'est pas protégée par des méthodes d'authentification ou cryptographiques. Toute personne ayant accès à l'affichage ou à la gestion de la configuration de votre groupe cible proxy peut consulter la requête d'initialisation. Vous ne devez pas ajouter de données sensibles, telles que des mots de passe ou des clés de chiffrement à longue durée de vie, à cette option.
AWS Identity and Access Management Rôle (IAM)	Rôle IAM autorisé à accéder aux secrets de Secrets Manager, qui représentent les informations d'identification des comptes utilisateur de base de données que le proxy peut utiliser. Vous pouvez également créer un nouveau rôle IAM à partir du AWS Management Console.

Paramètre	Description
Secrets Manager	Choisissez au moins un secret Secrets Manager contenant les informations d'identification de l'utilisateur de base de données permettant au proxy d'accéder au RDS.
Type d'authentification du client	Type d'authentification utilisé par le proxy pour les connexions des clients. Votre choix s'applique à tous les secrets de Secrets Manager que vous associez à ce proxy. Si vous devez spécifier un type d'authentification client différent pour chaque secret, créez votre proxy en utilisant plutôt l'API AWS CLI ou l'API.
Authentification IAM	S'il faut exiger, autoriser ou interdire l'authentification IAM pour les connexions à votre proxy. L'option Autoriser n'est valide que pour les proxies pour RDS for SQL Server. Votre choix s'appliqu e à tous les secrets de Secrets Manager que vous associez à ce proxy. Si vous devez spécifier une authentification IAM différent e pour chaque secret, créez votre proxy en utilisant plutôt l'API AWS CLI ou l'API.
Exigez la sécurité de la couche de transport	Applique le protocole TLS/SSL pour toutes les connexions client. Le proxy utilise le même paramètre de chiffrement pour sa connexion à la base de données sous-jacente, que la connexion du client soit chiffrée ou non chiffrée.
Sous-réseaux	Ce champ est prérempli avec tous les sous-réseaux associés à votre VPC. Vous pouvez supprimer tous les sous-réseaux dont le proxy n'a pas besoin, mais vous devez en conserver au moins deux.

Paramètre	Description
Groupe de sécurité VPC	Choisissez un groupe de sécurité VPC existant ou créez-en un nouveau à partir du. AWS Management Console Configurez les règles entrantes pour permettre à vos applications d'accéder au proxy et les règles sortantes pour autoriser le trafic provenant de vos bases de données cibles. **Note** Le groupe de sécurité doit autoriser les connexions entre le proxy et la base de données. Il sert à la fois à l'entrée de vos applications vers le proxy et à la sortie du proxy vers la base de données. Par exemple, si vous utilisez le même groupe de sécurité pour la base de données et le proxy, assurez-vous que les ressources de ce groupe de sécurité peuvent communiquer entre elles. Lorsque vous utilisez un VPC partagé, évitez d'utiliser le groupe de sécurité par défaut du VPC ou un groupe associé à un autre compte. Sélectionnez plutôt un groupe de sécurité qui appartient à votre compte. S'il n'en existe aucun, créez-en un. Pour plus d'informations, consultez la section Travailler avec le partage VPCs. RDS déploie un proxy dans plusieurs zones de disponibilité pour garantir une haute disponibilité. Pour activer la communica tion inter-AZ, la liste de contrôle d'accès réseau (ACL) de votre sous-réseau proxy doit autoriser la sortie sur le port du moteur et l'entrée sur tous les ports. Pour plus d'informations sur le réseau ACLs, voir Contrôler le trafic vers les sous-réseaux à l'aide du réseau ACLs. Si votre proxy et votre cible ont la même liste ACL réseau, vous devez ajouter une règle de trafic entrant de protocole TCP dans laquelle la source est définie sur le CIDR du VPC. Vous devez également ajouter une règle de sortie du

Paramètre	Description
	protocole TCP spécifique au port du moteur dans laquelle la destination est définie sur le CIDR VPC.
Activer la journalisation améliorée	Activez ce paramètre pour résoudre les problèmes de compatibilité ou de performance du proxy. Lorsqu'il est activé, le proxy RDS enregistre des informations de performance détaillées pour vous aider à déboguer le comportement SQL ou les performances et l'évolutivité des connexions proxy. Activez ce paramètre uniquement pour le débogage et assurez-vous que les mesures de sécurité appropriées sont en place pour protéger les informations sensibles contenues dans les journaux. Pour minimiser les frais, RDS Proxy désactive automatiquement ce paramètre 24 heures après son activation. Utilisez-le temporairement pour résoudre des problèmes spécifiques.

5. Choisissez Création d'un proxy.

AWS CLI

Pour créer un proxy à l'aide de AWS CLI, appelez la create-db-proxycommande avec les paramètres obligatoires suivants :

- `--db-proxy-name`
- `--engine-family`
- `--role-arn`
- `--auth`
- `--vpc-subnet-ids`

La valeur `--engine-family` est sensible à la casse.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-proxy \
    --db-proxy-name proxy_name \
    --engine-family { MYSQL | POSTGRESQL | SQLSERVER } \
    --auth ProxyAuthenticationConfig_JSON_string \
    --role-arn iam_role \
    --vpc-subnet-ids space_separated_list \
    [--vpc-security-group-ids space_separated_list] \
    [--require-tls | --no-require-tls] \
    [--idle-client-timeout value] \
    [--debug-logging | --no-debug-logging] \
    [--tags comma_separated_list]
```

Dans Windows :

```
aws rds create-db-proxy ^
    --db-proxy-name proxy_name ^
    --engine-family { MYSQL | POSTGRESQL | SQLSERVER } ^
    --auth ProxyAuthenticationConfig_JSON_string ^
    --role-arn iam_role ^
    --vpc-subnet-ids space_separated_list ^
    [--vpc-security-group-ids space_separated_list] ^
    [--require-tls | --no-require-tls] ^
    [--idle-client-timeout value] ^
    [--debug-logging | --no-debug-logging] ^
    [--tags comma_separated_list]
```

Voici un exemple de valeur JSON pour l'option `--auth`. Cet exemple applique un type d'authentification client différent à chaque secret.

```
[
  {
    "Description": "proxy description 1",
    "AuthScheme": "SECRETS",
    "SecretArn": "arn:aws:secretsmanager:us-
west-2:123456789123:secret/1234abcd-12ab-34cd-56ef-1234567890ab",
    "IAMAuth": "DISABLED",
    "ClientPasswordAuthType": "POSTGRES_SCRAM_SHA_256"
  },

  {
    "Description": "proxy description 2",
    "AuthScheme": "SECRETS",
```

```
    "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122223333:secret/1234abcd-12ab-34cd-56ef-1234567890cd",
    "IAMAuth": "DISABLED",
    "ClientPasswordAuthType": "POSTGRES_MD5"

  },

  {
    "Description": "proxy description 3",
    "AuthScheme": "SECRETS",
    "SecretArn": "arn:aws:secretsmanager:us-
west-2:111122221111:secret/1234abcd-12ab-34cd-56ef-1234567890ef",
    "IAMAuth": "REQUIRED"
  }

]
```

Tip

Si vous ne connaissez pas encore le sous-réseau IDs à utiliser pour le `--vpc-subnet-ids` paramètre, consultez des exemples Configuration des prérequis réseau pour le proxy RDS pour les trouver.

Note

Le groupe de sécurité doit autoriser l'accès à la base de données à laquelle le proxy se connecte. Le même groupe de sécurité est utilisé pour l'entrée de vos applications vers le proxy, et pour la sortie du proxy vers la base de données. Par exemple, supposons que vous utilisiez le même groupe de sécurité pour votre base de données et votre proxy. Dans ce cas, assurez-vous de spécifier que les ressources de ce groupe de sécurité peuvent communiquer avec d'autres ressources du même groupe de sécurité.
Lorsque vous utilisez un VPC partagé, vous ne pouvez pas utiliser le groupe de sécurité par défaut pour le VPC ni un groupe appartenant à un autre compte. Choisissez un groupe de sécurité qui appartient à votre compte. S'il n'en existe aucun, créez-en un. Pour plus d'informations sur cette limitation, voir Travailler avec le partage VPCs.

Pour créer les associations appropriées pour le proxy, vous devez également utiliser la register-db-proxy-targetscommande. Spécifiez le nom du groupe cible `default`. RDS Proxy crée automatiquement un groupe cible portant ce nom au moment de la création de chaque proxy.

```
aws rds register-db-proxy-targets
    --db-proxy-name value
    [--target-group-name target_group_name]
    [--db-instance-identifiers space_separated_list]  # rds db instances, or
    [--db-cluster-identifiers cluster_id]        # rds db cluster (all instances)
```

API RDS

Pour créer un proxy RDS, appelez l'opération Create de l'API Amazon RDS. DBProxy Vous transmettez un paramètre avec la structure AuthConfigde données.

RDS Proxy crée automatiquement un groupe cible nommé `default` au moment de la création de chaque proxy. Vous associez un de données RDS au groupe cible en appelant la fonction Register DBProxy Targets.

Afficher un proxy

Après avoir créé un ou plusieurs proxys RDS, vous pouvez les afficher et les gérer dans l' AWS Management Console API RDS ou dans l' AWS CLI API RDS. Vous pouvez consulter les détails de leur configuration, surveiller les performances et déterminer les proxys à modifier ou à supprimer selon les besoins.

Pour permettre aux applications de base de données d'acheminer le trafic via un proxy, vous devez spécifier le point de terminaison du proxy dans la chaîne de connexion.

console

Pour afficher un proxy dans la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Sélectionnez le nom du proxy pour en afficher les détails.
4. Sur la page de détails, la section Groupes cibles montre comment le proxy est lié à un d'une instance de base de données RDS spécifique. Vous pouvez accéder à la page du groupe

cible par défaut pour obtenir une vue plus approfondie de cette association, notamment des paramètres de configuration définis lors de la création du proxy. Ces paramètres incluent le pourcentage de connexion maximal, le délai d'emprunt de connexion, la famille de moteurs et les filtres d'épinglage de session.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour afficher votre proxy à l'aide de la CLI, utilisez la describe-db-proxiescommande. Par défaut, la demande renvoie tous les proxys appartenant à votre AWS compte. Pour afficher les détails d'un proxy, spécifiez son nom avec le paramètre `--db-proxy-name`.

```
aws rds describe-db-proxies [--db-proxy-name proxy_name]
```

Pour afficher d'autres informations associées au proxy, utilisez les commandes suivantes.

```
aws rds describe-db-proxy-target-groups  --db-proxy-name proxy_name

aws rds describe-db-proxy-targets --db-proxy-name proxy_name
```

Utilisez la séquence de commandes suivante pour afficher plus de détails sur les éléments associés au proxy :

1. Pour obtenir une liste de proxys, exécutez describe-db-proxies.
2. Pour afficher les paramètres de connexion tels que le pourcentage maximal de connexions que le proxy peut utiliser, exécutez describe-db-proxy-target-groups`--db-proxy-name`. Utilisez le nom du proxy comme valeur de paramètre.
3. Pour voir les détails du de données RDS associé au groupe cible renvoyé, exécutez describe-db-proxy-targets.

API RDS

Pour afficher vos proxys à l'aide de l'API RDS, utilisez l'opération DBProxiesDescribe. Elle renvoie les valeurs du type de DBProxydonnées.

Pour voir les détails des paramètres de connexion du proxy, utilisez les identifiants du proxy issus de cette valeur de retour avec l'DBProxyTargetGroupsopération Describe. Elle renvoie les valeurs du type de DBProxyTargetGroupdonnées.

Pour voir l'instance RDS ou le cluster de base de données Aurora associé au proxy, utilisez l'opération Describe DBProxy Targets. Elle renvoie des valeurs du type de données DBProxyTarget.

Connexion à une base de données via RDS Proxy

La méthode de connexion à une instance de base de données RDS via un proxy ou en se connectant à la base de données est généralement la même. Pour de plus amples informations, veuillez consulter Présentation des points de terminaison proxy.

Rubriques

- Connexion à un proxy à l'aide de l'authentification native
- Connexion à un proxy à l'aide de l'authentification IAM
- Considérations relatives à la connexion à un proxy avec Microsoft SQL Server
- Considérations relatives à la connexion à un proxy avec PostgreSQL

Connexion à un proxy à l'aide de l'authentification native

Procédez comme suit pour vous connecter à un proxy à l'aide de l'authentification native :

1. Recherchez le point de terminaison du proxy. Dans le AWS Management Console, vous pouvez trouver le point de terminaison sur la page de détails du proxy correspondant. Avec le AWS CLI, vous pouvez utiliser la describe-db-proxiescommande. L'exemple suivant montre comment procéder.

```
# Add --output text to get output as a simple tab-separated list.
$ aws rds describe-db-proxies --query '*[*].
{DBProxyName:DBProxyName,Endpoint:Endpoint}'
[
    [
        {
            "Endpoint": "the-proxy.proxy-demo.us-east-1.rds.amazonaws.com",
            "DBProxyName": "the-proxy"
        },
        {
            "Endpoint": "the-proxy-other-secret.proxy-demo.us-
east-1.rds.amazonaws.com",
            "DBProxyName": "the-proxy-other-secret"
        },
        {
```

```
            "Endpoint": "the-proxy-rds-secret.proxy-demo.us-
east-1.rds.amazonaws.com",
            "DBProxyName": "the-proxy-rds-secret"
        },
        {
            "Endpoint": "the-proxy-t3.proxy-demo.us-east-1.rds.amazonaws.com",
            "DBProxyName": "the-proxy-t3"
        }
    ]
]
```

2. Spécifiez le point de terminaison comme paramètre hôte dans la chaîne de connexion de votre application cliente. Par exemple, spécifiez le point de terminaison du proxy comme valeur pour l'option `mysql -h` ou `psql -h`.
3. Fournissez le nom d'utilisateur et le mot de passe de base de données que vous utilisez habituellement.

Connexion à un proxy à l'aide de l'authentification IAM

Lorsque vous utilisez l'authentification IAM avec RDS Proxy, configurez les utilisateurs de votre base de données de sorte qu'ils s'authentifient avec des noms d'utilisateur et des mots de passe normaux. L'authentification IAM s'applique à la récupération RDS Proxy des informations d'identification (nom d'utilisateur et mot de passe) depuis Secrets Manager. La connexion depuis RDS Proxy à la base de données sous-jacente ne passe pas par IAM.

Pour obtenir des informations générales sur l'utilisation d'IAM, consultez Sécurité dans Amazon RDS.

Les principales différences dans l'utilisation d'IAM pour RDS Proxy sont les suivantes :

- Vous ne configurez pas chaque utilisateur de base de données avec un plugin d'autorisation. Les utilisateurs de base de données ont toujours des noms d'utilisateur et des mots de passe réguliers dans la base de données. Vous configurez des secrets Secrets Manager contenant ces noms et mots de passe d'utilisateur, et autorisez RDS Proxy à récupérer les informations d'identification à partir d'Secrets Manager.

  L'authentification IAM s'applique à la connexion entre votre programme client et le proxy. Le proxy s'authentifie ensuite à la base de données à l'aide des informations d'identification (nom d'utilisateur et mot de passe) extraites via Secrets Manager.

- Spécifiez le point de terminaison du proxy plutôt que celui de l'instance, du cluster ou du lecteur. Pour de plus amples informations sur le point de terminaison du proxy, veuillez consulter Connexion à votre instance de base de données à l'aide de l'authentification IAM..
- Dans le cas d'une authentification IAM de base de données directe, vous choisissez de manière sélective les utilisateurs de base de données et les configurez de sorte qu'ils soient identifiés avec un plugin d'authentification spécial. Vous pouvez ensuite vous connecter à ces utilisateurs à l'aide de l'authentification IAM.

  Dans le cas d'utilisation du proxy, vous fournissez au proxy des secrets qui contiennent le nom d'utilisateur et le mot de passe de certains utilisateurs (authentification native). Vous vous connectez ensuite au proxy à l'aide de l'authentification IAM. Pour ce faire, vous générez un jeton d'authentification avec le point de terminaison proxy, et non avec le point de terminaison de base de données. Vous utilisez également un nom d'utilisateur qui correspond à l'un des noms d'utilisateur pour les secrets que vous avez fournis.
- Veillez à utiliser le protocole TLS (Transport Layer Security)/SSL (Secure Sockets Layer) lorsque vous vous connectez à un proxy avec l'authentification IAM.

Vous pouvez accorder l'accès au proxy à un utilisateur spécifique en modifiant la politique IAM. Un exemple suit.

```
"Resource": "arn:aws:rds-db:us-east-2:1234567890:dbuser:prx-ABCDEFGHIJKL01234/db_user"
```

Considérations relatives à la connexion à un proxy avec Microsoft SQL Server

Pour vous connecter à un proxy à l'aide de l'authentification IAM, vous n'utilisez pas le champ du mot de passe. Vous devez plutôt fournir la propriété de jeton appropriée pour chaque type de pilote de base de données dans le champ du jeton. Par exemple, utilisez la propriété `accessToken` pour JDBC ou la propriété `sql_copt_ss_access_token` pour ODBC. Ou utilisez la `AccessToken` propriété du SqlClient pilote .NET. Vous ne pouvez pas utiliser l'authentification IAM avec des clients qui ne prennent pas en charge les propriétés des jetons.

Dans certaines conditions, un proxy ne peut pas partager une connexion à une base de données et épingle la connexion entre votre application cliente et le proxy vers une connexion de base de données dédiée. Pour plus d'informations sur ces conditions, consultez Éviter d'épingler un proxy RDS.

Considérations relatives à la connexion à un proxy avec PostgreSQL

Si vous créez un nouvel utilisateur de base de données PostgreSQL pour vous connecter au proxy RDS, assurez-vous de lui accorder le privilège d'accès à la `CONNECT` base de données. Sans cela, l'utilisateur ne peut pas établir de connexion. Pour de plus amples informations, veuillez consulter the section called “Ajouter un nouvel utilisateur de base de données à une base de données Postgre lors SQL de l'utilisation d'RDSun proxy”.

Lorsqu'un client établit une connexion à une base de données PostgreSQL, il envoie un message de démarrage. Ce message inclut des paires de chaînes de noms de paramètres et de valeurs. Pour plus de détails, veuillez consulter `StartupMessage` dans la section relative aux formats de message PostgreSQL de la documentation PostgreSQL.

Lorsque vous vous connectez via un proxy RDS, le message de démarrage peut inclure les paramètres actuellement reconnus suivants :

- `user`
- `database`

Le message de démarrage peut également inclure les paramètres d'exécution supplémentaires suivants :

- `application_name`
- `client_encoding`
- `DateStyle`
- `TimeZone`
- `extra_float_digits`
- `search_path`

Pour de plus amples informations sur la messagerie PostgreSQL, consultez la section relative au protocole frontend/backend de la documentation PostgreSQL.

Pour PostgreSQL, si vous utilisez JDBC, nous vous recommandons ce qui suit pour éviter le pinning :

- Définissez le paramètre de connexion JDBC `assumeMinServerVersion` sur `9.0` au minimum afin d'éviter l'épinglage. Cela empêche le pilote JDBC d'effectuer un aller-retour supplémentaire lors du démarrage de la connexion lorsqu'il s'exécute. `SET extra_float_digits = 3`

- Définissez le paramètre de connexion JDBC `ApplicationName` sur *`any/your-application-name`* afin d'éviter l'épinglage. Cela empêche le pilote JDBC d'effectuer un aller-retour supplémentaire au démarrage de la connexion lorsqu'il exécute `SET application_name = "PostgreSQL JDBC Driver"`. Notez que le paramètre JDBC est `ApplicationName`, mais que le paramètre PostgreSQL `StartupMessage` est `application_name`.

Pour de plus amples informations, veuillez consulter Éviter d'épingler un proxy RDS. Pour de plus amples informations sur la connexion à l'aide de JDBC, veuillez consulter la section relative à la connexion à la base de données dans la documentation PostgreSQL.

Gestion d'un RDS proxy

Cette section fournit des informations sur la façon de gérer le fonctionnement et la configuration du RDS proxy. Ces procédures aident votre application à utiliser de manière optimale les connexions de base de données et à obtenir une réutilisation maximale des connexions. Plus vous pouvez tirer parti de la réutilisation des connexions, plus CPU vous pouvez économiser de la mémoire. Cela réduit la latence de votre application et permet à la base de données de dédier une plus grande partie de ses ressources au traitement des requêtes d'application.

Rubriques

- Modification d'un RDS Proxy
- Ajout d'un nouvel utilisateur de base de données lors de l'utilisation d'RDSun proxy
- Considérations relatives à la connexion au proxy RDS
- Éviter d'épingler un proxy RDS
- Supprimer un RDS proxy

Modification d'un RDS Proxy

Vous pouvez modifier des paramètres spécifiques associés à un proxy après sa création. Pour ce faire, modifiez le proxy lui-même, son groupe cible associé, ou les deux. Chaque proxy dispose d'un groupe cible associé.

AWS Management Console

> **Important**
>
> Les valeurs des champs Client authentication type (Type d'authentification client) et IAM authentication (Authentification IAM) s'appliquent à tous les secrets de Secrets Manager associés à ce proxy. Pour spécifier des valeurs différentes pour chaque secret, modifiez votre proxy en utilisant plutôt l'API AWS CLI ou l'API.

Modifications des paramètres d'un proxy

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Dans la liste de proxy, choisissez celui dont vous souhaitez modifier les paramètres ou accédez à sa page de détails.
4. Pour Actions, choisissez Modifier.
5. Saisissez ou sélectionnez les propriétés à modifier. Vous pouvez modifier les valeurs suivantes :
   - Identifiant du proxy : renommez le proxy en saisissant un nouvel identifiant.
   - Délai d'inactivité de la connexion client – Saisissez une période pour le délai d'inactivité de la connexion client.
   - Rôle IAM – Modifiez le rôle IAM utilisé pour récupérer les secrets de Secrets Manager.
   - Secrets de Secrets Manager – Ajoutez ou supprimez des secrets Secrets Manager. Ces secrets correspondent aux noms d'utilisateur et mots de passe de la base de données.
   - Client authentication type (Type d'authentification client) – (PostgreSQL uniquement) Modifiez le type d'authentification pour les connexions client au proxy.
   - IAM authentication (Authentification IAM) : exigez ou désactivez l'authentification IAM pour les connexions au proxy.
   - Exiger la Sécurité de la couche transport – Activez ou désactivez l'exigence du protocole TLS (Transport Layer Security).
   - Groupe de sécurité de VPC – Ajoutez ou supprimez des groupes de sécurité de VPC que le proxy doit utiliser.
   - Activation de la journalisation améliorée – Activez ou désactivez la journalisation améliorée.

6. Sélectionnez Modify.

Si vous n'avez pas trouvé les paramètres répertoriés que vous souhaitez modifier, procédez comme suit pour mettre à jour le groupe cible du proxy. Le groupe cible associé à un proxy contrôle les paramètres liés aux connexions à la base de données physique. Chaque proxy dispose d'un groupe cible associé, nommé `default`, qui est créé automatiquement avec le proxy. Vous ne pouvez pas renommer le groupe cible par défaut.

Vous pouvez uniquement modifier le groupe cible à partir de la page de détails du proxy, et non depuis la liste de la page Proxies.

Modification des paramètres d'un groupe cible proxy

1. À partir de la page Proxies, accédez à la page des détails d'un proxy.
2. Pour les Groupes cibles, choisissez le lien `default`. Actuellement, tous les proxy ont un groupe cible unique nommé `default`.
3. Sur la page de détails du groupe cible par défaut, sélectionnez Modifier.
4. Définissez de nouveaux paramètres pour les propriétés que vous pouvez modifier :

   - Base de données : choisissez une autre instance ou un autre cluster de base de données RDS.
   - Nombre maximal de connexions dans le groupe de connexions – Ajustez le pourcentage du nombre de connexions maximum disponibles que le proxy peut utiliser.
   - Filtre d'épinglage de session – (Facultatif) Choisissez un filtre d'épinglage de session. Cela permet de contourner les mesures de sécurité par défaut pour le multiplexage des connexions de base de données entre les connexions client. Actuellement, le paramètre n'est pas pris en charge pour PostgreSQL. Le seul choix est`EXCLUDE_VARIABLE_SETS`.

     L'activation de ce paramètre peut avoir un impact sur les variables de session d'une connexion sur les autres connexions. Cela peut entraîner des erreurs ou des problèmes d'exactitude si vos requêtes dépendent de valeurs de variables de session définies en dehors de la transaction en cours. Vous pouvez utiliser cette option après avoir vérifié que vos applications peuvent partager des connexions de base de données en toute sécurité entre les connexions client.

     Les modèles suivants peuvent être considérés comme sûrs :

- Instructions SET dans lesquelles aucune modification n'est apportée à la valeur effective de la variable de session, c'est-à-dire qu'aucune modification n'est apportée à la variable de session.
- Vous modifiez la valeur de la variable de session et exécutez une instruction dans la même transaction.

Pour de plus amples informations, veuillez consulter Éviter d'épingler un proxy RDS.

- Délai d'expiration d'emprunt de connexion – Ajustez l'intervalle du délai d'attente d'emprunt de connexion. Ce paramètre s'applique lorsque le nombre maximal de connexions est déjà utilisé pour le proxy. Ce paramètre permet de définir combien de temps le proxy doit attendre la disponibilité d'une connexion avant de renvoyer une erreur de dépassement de délai d'attente.
- Requête d'initialisation. (Facultatif) Ajoutez une requête d'initialisation ou modifiez la requête en cours. Vous pouvez spécifier une ou plusieurs instructions SQL que le proxy doit exécuter lors de l'ouverture de chaque nouvelle connexion à la base de données. Le paramètre est généralement utilisé avec SET des instructions pour s'assurer que chaque connexion possède des paramètres identiques. Assurez-vous que la requête que vous ajoutez est valide. Pour inclure plusieurs variables dans une seule SET instruction, utilisez des séparateurs de virgules. Par exemple :

```
SET variable1=value1, variable2=value2
```

Pour plusieurs instructions, utilisez des points-virgules comme séparateur.

Certaines propriétés, telles que l'identifiant du groupe cible et le moteur de base de données, sont corrigées.

5. Sélectionnez Modification du groupe cible.

AWS CLI

Pour modifier un proxy à l'aide de AWS CLI, utilisez les commandes modify-db-proxymodify-db-proxy-target-group deregister-db-proxy-targetset register-db-proxy-targets.

Avec la commande `modify-db-proxy`, vous pouvez modifier des propriétés, par exemple :

- Ensemble des secrets Secrets Manager utilisés par le proxy.
- TLS requis ou non.

- Délai d'expiration de la connexion client inactive.
- Nécessité ou non de consigner des informations supplémentaires des instructions SQL pour le débogage.
- Rôle IAM utilisé pour récupérer les secrets Secrets Manager.
- Groupes de sécurité utilisés par le proxy.

L'exemple suivant montre comment renommer un proxy existant.

```
aws rds modify-db-proxy --db-proxy-name the-proxy --new-db-proxy-name the_new_name
```

Pour modifier les paramètres liés à la connexion ou renommer le groupe cible, utilisez la commande `modify-db-proxy-target-group`. Actuellement, tous les proxy ont un groupe cible unique nommé `default`. Lorsque vous travaillez avec ce groupe cible, vous spécifiez le nom du proxy et `default` le nom du groupe cible. Vous ne pouvez pas renommer le groupe cible par défaut.

L'exemple suivant montre comment vérifier le paramètre `MaxIdleConnectionsPercent` d'un proxy, puis le modifier à l'aide du groupe cible.

```
aws rds describe-db-proxy-target-groups --db-proxy-name the-proxy

{
    "TargetGroups": [
        {
            "Status": "available",
            "UpdatedDate": "2019-11-30T16:49:30.342Z",
            "ConnectionPoolConfig": {
                "MaxIdleConnectionsPercent": 50,
                "ConnectionBorrowTimeout": 120,
                "MaxConnectionsPercent": 100,
                "SessionPinningFilters": []
            },
            "TargetGroupName": "default",
            "CreatedDate": "2019-11-30T16:49:27.940Z",
            "DBProxyName": "the-proxy",
            "IsDefault": true
        }
    ]
}
```

```
aws rds modify-db-proxy-target-group --db-proxy-name the-proxy --target-group-name
 default --connection-pool-config '
{ "MaxIdleConnectionsPercent": 75 }'

{
    "DBProxyTargetGroup": {
        "Status": "available",
        "UpdatedDate": "2019-12-02T04:09:50.420Z",
        "ConnectionPoolConfig": {
            "MaxIdleConnectionsPercent": 75,
            "ConnectionBorrowTimeout": 120,
            "MaxConnectionsPercent": 100,
            "SessionPinningFilters": []
        },
        "TargetGroupName": "default",
        "CreatedDate": "2019-11-30T16:49:27.940Z",
        "DBProxyName": "the-proxy",
        "IsDefault": true
    }
}
```

Grâce aux commandes `deregister-db-proxy-targets` et `register-db-proxy-targets`, vous modifiez les instances de base de données RDS auxquelles le proxy est associé via son groupe cible. Actuellement, chaque proxy peut se connecter à une instance de base de données RDS (). Le groupe cible suit les détails de connexion pour toutes les instances de base de données RDS dans une configuration multi-AZ, .

L'exemple suivant commence par un proxy associé à un cluster Aurora MySQL nommé `cluster-56-2020-02-25-1399`. L'exemple vous explique comment modifier le proxy afin qu'il puisse se connecter à un autre cluster nommé `provisioned-cluster`.

Lorsque vous travaillez avec une instance de base de données RDS, sélectionnez l'option `--db-instance-identifier`.

L'exemple suivant modifie un proxy Aurora MySQL. Un proxy PostgreSQL Aurora dispose du port 5432.

```
aws rds describe-db-proxy-targets --db-proxy-name the-proxy

{
    "Targets": [
        {
```

```
            "Endpoint": "instance-9814.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "instance-9814"
        },
        {
            "Endpoint": "instance-8898.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "instance-8898"
        },
        {
            "Endpoint": "instance-1018.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "instance-1018"
        },
        {
            "Type": "TRACKED_CLUSTER",
            "Port": 0,
            "RdsResourceId": "cluster-56-2020-02-25-1399"
        },
        {
            "Endpoint": "instance-4330.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "instance-4330"
        }
    ]
}

aws rds deregister-db-proxy-targets --db-proxy-name the-proxy --db-cluster-identifier
 cluster-56-2020-02-25-1399

aws rds describe-db-proxy-targets --db-proxy-name the-proxy

{
    "Targets": []
}

aws rds register-db-proxy-targets --db-proxy-name the-proxy --db-cluster-identifier
 provisioned-cluster

{
```

```
    "DBProxyTargets": [
        {
            "Type": "TRACKED_CLUSTER",
            "Port": 0,
            "RdsResourceId": "provisioned-cluster"
        },
        {
            "Endpoint": "gkldje.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "gkldje"
        },
        {
            "Endpoint": "provisioned-1.demo.us-east-1.rds.amazonaws.com",
            "Type": "RDS_INSTANCE",
            "Port": 3306,
            "RdsResourceId": "provisioned-1"
        }
    ]
}
```

API RDS

Pour modifier un proxy à l'aide de l'API RDS, vous devez utiliser les opérations ModifierDBProxy, Modifier DBProxy TargetGroup, Désenregistrer DBProxy les cibles et Enregistrer DBProxy les cibles.

Avec `ModifyDBProxy`, vous pouvez modifier des propriétés, par exemple :

- Ensemble des secrets Secrets Manager utilisés par le proxy.
- TLS requis ou non.
- Délai d'expiration de la connexion client inactive.
- Nécessité ou non de consigner des informations supplémentaires des instructions SQL pour le débogage.
- Rôle IAM utilisé pour récupérer les secrets Secrets Manager.
- Groupes de sécurité utilisés par le proxy.

Avec`ModifyDBProxyTargetGroup`, vous pouvez modifier les paramètres liés à la connexion. Actuellement, tous les proxy ont un groupe cible unique nommé `default`. Lorsque vous travaillez avec ce groupe cible, vous spécifiez le nom du proxy et `default` le nom du groupe cible. Vous ne pouvez pas renommer le groupe cible par défaut.

Avec `DeregisterDBProxyTargets` et`RegisterDBProxyTargets`, vous modifiez l'instance de base de données RDS () à laquelle le proxy est associé via son groupe cible. Actuellement, chaque proxy peut se connecter à une instance de base de données RDS . Le groupe cible suit les détails de connexion des instances de base de données RDS dans une configuration multi-AZ .

Ajout d'un nouvel utilisateur de base de données lors de l'utilisation d'RDSun proxy

Dans certains cas, vous pouvez ajouter un nouvel utilisateur de base de données à une RDSinstance de base de données ou à un cluster associé à un proxy. Le cas échéant, ajoutez ou réaffectez un secret Secrets Manager pour stocker les informations d'identification de cet utilisateur. Pour ce faire, suivez les étapes suivantes :

1. Créez un nouveau secret Secrets Manager en suivant les instructions décrites dans la section Configuration des informations d'identification de base de données AWS Secrets Manager pour le proxy RDS.
2. Mettez à jour le IAM rôle pour autoriser le RDS proxy à accéder au nouveau secret du Gestionnaire de Secrets. Pour ce faire, mettez à jour la section des ressources de la politique de IAM rôle.
3. Modifiez le RDS proxy pour ajouter le nouveau secret de Secrets Manager sous Secrets Manager secrets.
4. Si le nouvel utilisateur remplace un utilisateur existant, mettez à jour les informations d'identification stockées dans le secret Secrets Manager du proxy pour l'utilisateur existant.

Ajouter un nouvel utilisateur de base de données à une base de données Postgre lors SQL de l'utilisation d'RDSun proxy

Lorsque vous ajoutez un nouvel utilisateur à votre SQL base de données Postgre, si vous avez exécuté la commande suivante :

```
REVOKE CONNECT ON DATABASE postgres FROM PUBLIC;
```

Accordez à l'utilisateur `rdsproxyadmin` le privilège `CONNECT` afin qu'il puisse surveiller les connexions sur la base de données cible.

```
GRANT CONNECT ON DATABASE postgres TO rdsproxyadmin;
```

Vous pouvez également autoriser d'autres utilisateurs de la base de données cible à effectuer des surveillances de l'état en modifiant `rdsproxyadmin` pour l'utilisateur de la base de données dans la commande ci-dessus.

Modification du mot de passe d'un utilisateur de base de données lors de l'utilisation d'un RDS proxy

Dans certains cas, vous pouvez modifier le mot de passe d'un utilisateur de base de données dans un RDS d'instance de base de données associé à un proxy. Le cas échéant, mettez à jour le secret Secrets Manager correspondant avec le nouveau mot de passe.

Considérations relatives à la connexion au proxy RDS

Configuration des paramètres de connexion

Pour ajuster le regroupement de connexion RDS Proxy, vous pouvez modifier les paramètres suivants :

- IdleClientTimeout
- MaxConnectionsPercent
- MaxIdleConnectionsPercent
- ConnectionBorrowTimeout

IdleClientTimeout

Vous pouvez spécifier la durée pendant laquelle une connexion client peut être inactive avant que le proxy ne la ferme. La valeur par défaut est de 1 800 secondes (30 minutes).

Une connexion client est considérée comme inactive lorsque l'application ne soumet aucune nouvelle demande dans le délai défini après l'achèvement de la demande précédente. La connexion à la base de données sous-jacente reste ouverte et est renvoyée au regroupement de connexions. Ainsi, elle peut être réutilisée pour de nouvelles connexions client. Si vous souhaitez que le proxy supprime de manière proactive les connexions périmées, réduisez le délai d'expiration des connexions client inactives. Si votre charge de travail établit des connexions fréquentes avec le proxy, augmentez le délai d'inactivité des connexions client afin de réduire les coûts liés à l'établissement des connexions.

Ce paramètre est représenté par le champ Délai d'expiration de la connexion client inactive dans la console RDS et par le `IdleClientTimeout` paramètre dans l'API AWS CLI et. Pour savoir

comment modifier la valeur du champ Idle client connection timeout (Délai d'inactivité de la connexion client) dans la console RDS, veuillez consulter AWS Management Console. Pour savoir comment modifier la valeur du `IdleClientTimeout` paramètre, consultez la commande CLI modify-db-proxyou l'opération Modify de l'APIDBProxy.

MaxConnectionsPercent

Vous pouvez limiter le nombre de connexions qu'un RDS Proxy peut établir avec la base de données cible. Vous indiquez la limite, sous forme de pourcentage, des connexions maximales disponibles pour votre base de données. Ce paramètre est représenté par le champ Nombre maximum de connexions du pool de connexions dans la console RDS et par le `MaxConnectionsPercent` paramètre dans l'API AWS CLI et.

La valeur `MaxConnectionsPercent` est exprimée en pourcentage du paramètre `max_connections` pour l'instance de base de données RDS utilisé par le groupe cible. Le proxy ne crée pas toutes ces connexions à l'avance. Ce paramètre permet au proxy d'établir ces connexions selon les besoins de la charge de travail.

Par exemple, pour une cible de base de données enregistrée avec `max_connections` définies sur 1 000 et `MaxConnectionsPercent` défini sur 95, RDS Proxy définit 950 connexions comme la limite supérieure pour les connexions simultanées à cette cible de base de données.

Le fait que votre charge de travail atteigne le nombre maximum de connexions à la base de données autorisées a souvent pour effet secondaire d'augmenter la latence globale des requêtes, ainsi que d'augmenter la métrique `DatabaseConnectionsBorrowLatency`. Vous pouvez surveiller les connexions à la base de données actuellement utilisées et le nombre total de connexions autorisées en comparant les métriques `DatabaseConnections` et `MaxDatabaseConnectionsAllowed`.

Pour définir ce paramètre, tenez compte des bonnes pratiques suivantes :

- Prévoyez une marge de connexion suffisante pour les modifications du modèle de la charge de travail. Il est recommandé de définir le paramètre afin qu'il soit au moins 30 % supérieur à votre utilisation surveillée maximale récente. Comme RDS Proxy redistribue les quotas de connexion à la base de données entre plusieurs nœuds, les modifications de la capacité interne peuvent nécessiter une marge d'au moins 30 % pour les connexions supplémentaires afin d'éviter des latences d'emprunt plus importantes.
- RDS Proxy réserve un certain nombre de connexions pour une surveillance active afin de permettre un basculement rapide, le routage du trafic et les opérations internes. La métrique `MaxDatabaseConnectionsAllowed` n'inclut pas ces connexions réservées. Elle représente le

nombre de connexions disponibles pour répondre à la charge de travail et peut être inférieure à la valeur dérivée du paramètre `MaxConnectionsPercent`.

Valeurs `MaxConnectionsPercent` minimales recommandées

- db.t3.small : 30
- db.t3.medium ou supérieur : 20

Pour savoir comment modifier la valeur du champ Connection pool maximum connections (Connexions maximales au groupe de connexion) dans la console RDS, veuillez consulter AWS Management Console. Pour savoir comment modifier la valeur du `MaxConnectionsPercent` paramètre, consultez la commande CLI modify-db-proxy-target-group ou l'opération d'API DBProxyTargetGroupModify.

Pour en savoir plus sur les limites de connexion aux bases de données, veuillez consulter Nombre maximal de connexions aux bases de données.

MaxIdleConnectionsPercent

Vous pouvez contrôler le nombre de connexions aux bases de données inactives que RDS Proxy peut conserver dans le groupe de connexion. Par défaut, le proxy RDS considère qu'une connexion à la base de données de son pool est inactive lorsqu'aucune activité n'a été enregistrée pendant cinq minutes.

La `MaxIdleConnectionsPercent` valeur est exprimée en pourcentage du `max_connections` paramètre pour le groupe cible d'instances de base de données RDS. La valeur par défaut est de 50 % de `MaxConnectionsPercent` et la limite supérieure est la valeur de `MaxConnectionsPercent`. Par exemple, si`MaxConnectionsPercent`, est 80, la valeur par défaut de `MaxIdleConnectionsPercent` est 40. Si la valeur de `MaxConnectionsPercent` n'est pas spécifiée, alors pour RDS pour SQL Server, elle `MaxIdleConnectionsPercent` est 5, et pour tous les autres moteurs, la valeur par défaut est 50.

Une valeur élevée permet au proxy de laisser ouvert un pourcentage élevé de connexions inactives à la base de données. Avec une valeur faible, le proxy ferme un pourcentage élevé de connexions de base de données inactives. Si vos charges de travail sont imprévisibles, pensez à définir une valeur élevée pour `MaxIdleConnectionsPercent`. Cela signifie que RDS Proxy peut prendre en charge les vagues d'activité sans ouvrir de nombreuses nouvelles connexions aux bases de données.

Ce paramètre est représenté par le `MaxIdleConnectionsPercent` paramètre de `DBProxyTargetGroup` in the AWS CLI et dans l'API. Pour savoir comment modifier la valeur du

`MaxIdleConnectionsPercent` paramètre, consultez la commande CLI modify-db-proxy-target-group ou l'opération d'API DBProxyTargetGroupModify.

Pour en savoir plus sur les limites de connexion aux bases de données, veuillez consulter Nombre maximal de connexions aux bases de données.

ConnectionBorrowTimeout

Vous pouvez choisir combien de temps le RDS Proxy doit attendre la disponibilité d'utilisation d'une connexion à une base de données dans le groupe de connexion avant de renvoyer une erreur de dépassement de délai d'attente. La durée par défaut est de 120 secondes. Ce paramètre s'applique lorsque le nombre maximal de connexions est atteint et qu'aucune connexion n'est disponible dans le groupe de connexion. Cela s'applique également lorsqu'aucune instance de base de données appropriée n'est disponible pour traiter la demande, par exemple lorsqu'une opération de basculement est en cours. À l'aide de ce paramètre, vous pouvez définir le meilleur délai d'attente pour votre application sans modifier le délai d'attente des requêtes dans le code de votre application.

Ce paramètre est représenté par le champ Connection borrow timeout dans la console RDS ou par le `ConnectionBorrowTimeout` paramètre de `DBProxyTargetGroup` l'API AWS CLI or. Pour savoir comment modifier la valeur du champ Connection borrow timeout (Délai d'expiration d'emprunt de connexion) dans la console RDS, veuillez consulter AWS Management Console. Pour savoir comment modifier la valeur du `ConnectionBorrowTimeout` paramètre, consultez la commande CLI modify-db-proxy-target-group ou l'opération d'API DBProxyTargetGroupModify.

Connexions client et connexions aux bases de données

Les connexions entre votre application et RDS Proxy sont appelées connexions client. Les connexions d'un proxy à la base de données sont appelées connexions à la base de données. Lorsque vous utilisez RDS Proxy, les connexions client s'arrêtent au niveau du proxy tandis que les connexions à la base de données sont gérées au sein de RDS Proxy.

Le regroupement des connexions côté application peut offrir l'avantage de réduire l'établissement de connexions récurrentes entre votre application et le proxy RDS.

Tenez compte des aspects de configuration suivants avant d'implémenter un pool de connexions côté application :

- Durée de vie maximale de la connexion client : le proxy RDS impose une durée de vie maximale de 24 heures aux connexions client. Cette valeur n'est pas configurable. Configurez votre pool

avec une durée de vie maximale de connexion inférieure à 24 heures afin d'éviter les interruptions inattendues de la connexion client.

- Délai d'inactivité de la connexion client : le proxy RDS impose une durée d'inactivité maximale pour les connexions client. Configurez votre regroupement avec un délai d'inactivité inférieur au délai d'inactivité de votre connexion client pour RDS Proxy afin d'éviter les interruptions de connexion inattendues.

Le nombre maximum de connexions client configurées dans votre pool de connexions côté application ne doit pas nécessairement être limité au paramètre max_connections pour le proxy RDS.

Le regroupement des connexions client prolonge la durée de vie des connexions client. Si vos connexions sont épinglées, le regroupement des connexions client peut réduire l'efficacité du multiplexage. Les connexions client bloquées mais inactives dans le pool de connexions côté application continuent de conserver une connexion à la base de données et empêchent la réutilisation de la connexion à la base de données par d'autres connexions client. Consultez les journaux de votre proxy pour vérifier si vos connexions sont épinglées.

> **Note**
>
> RDS Proxy ferme les connexions à la base de données après 24 heures lorsqu'elles ne sont plus utilisées. Le proxy effectue cette action indépendamment de la valeur du paramètre de connexions inactives maximum.

Éviter d'épingler un proxy RDS

Le multiplexage est plus efficace lorsque les demandes de base de données ne dépendent pas des informations d'état issues de demandes précédentes. Dans ce cas, RDS Proxy peut réutiliser une connexion à la fin de chaque transaction. Ces informations d'état incluent la plupart des variables et des paramètres de configuration que vous pouvez modifier à l'aide des instructions `SET` ou `SELECT`. Les transactions SQL sur une connexion client peuvent se multiplexer entre les connexions de base de données sous-jacentes par défaut.

Vos connexions au proxy peuvent entrer dans un état appelé épinglage. Lorsqu'une connexion est épinglée, chaque transaction ultérieure utilise la même connexion de base de données sous-jacente jusqu'à la fin de la session. De même, les autres connexions client ne peuvent pas réutiliser cette connexion à la base de données tant que la session n'est pas terminée. La session se termine lorsque la connexion client est supprimée.

RDS Proxy épingle automatiquement une connexion client à une connexion de base de données spécifique lorsqu'il détecte un changement d'état de session qui n'est pas approprié pour d'autres sessions. L'épinglage réduit l'efficacité de la réutilisation des connexions. Si la totalité, ou presque, de vos connexions font l'objet d'un épinglage, pensez à modifier le code de votre application ou votre charge de travail afin de réduire les conditions à l'origine de l'épinglage.

Par exemple, votre application modifie une variable de session ou un paramètre de configuration. Dans ce cas, les instructions ultérieures peuvent reposer sur la nouvelle variable ou le nouveau paramètre pour entrer en vigueur. Ainsi, lorsque le RDS Proxy traite des demandes de modification des variables ou des paramètres de configuration de session, il épingle cette session à la connexion de base de données. De cette manière, l'état de session reste en vigueur pour toutes les transactions ultérieures de la même session.

Pour les moteurs de bases de données, cette règle ne s'applique pas à tous les paramètres que vous pouvez définir. RDS Proxy suit certaines instructions et variables. Ainsi, le proxy RDS n'épingle pas la session lorsque vous les modifiez. Dans ce cas, RDS Proxy réutilise la connexion uniquement pour les autres sessions dont les valeurs de ces paramètres sont identiques. Pour plus de détails sur ce que RDS Proxy suit pour un moteur de base de données, consultez ce qui suit :

- Ce que RDS Proxy suit pour les bases de données RDS for SQL Server
- Ce que RDS Proxy suit pour les bases de données RDS for MariaDB et RDS for MySQL

Ce que RDS Proxy suit pour les bases de données RDS for SQL Server

Le proxy RDS suit les instructions SQL Server suivantes :

- `USE`
- `SET ANSI_NULLS`
- `SET ANSI_PADDING`
- `SET ANSI_WARNINGS`
- `SET ARITHABORT`
- `SET CONCAT_NULL_YIELDS_NULL`
- `SET CURSOR_CLOSE_ON_COMMIT`
- `SET DATEFIRST`
- `SET DATEFORMAT`
- `SET LANGUAGE`

- SET LOCK_TIMEOUT
- SET NUMERIC_ROUNDABORT
- SET QUOTED_IDENTIFIER
- SET TEXTSIZE
- SET TRANSACTION ISOLATION LEVEL

Ce que RDS Proxy suit pour les bases de données RDS for MariaDB et RDS for MySQL

RDS Proxy suit les instructions MariaDB et MySQL suivantes :

- DROP DATABASE
- DROP SCHEMA
- USE

RDS Proxy suit les variables MySQL et MariaDB suivantes :

- AUTOCOMMIT
- AUTO_INCREMENT_INCREMENT
- CHARACTER SET (or CHAR SET)
- CHARACTER_SET_CLIENT
- CHARACTER_SET_DATABASE
- CHARACTER_SET_FILESYSTEM
- CHARACTER_SET_CONNECTION
- CHARACTER_SET_RESULTS
- CHARACTER_SET_SERVER
- COLLATION_CONNECTION
- COLLATION_DATABASE
- COLLATION_SERVER
- INTERACTIVE_TIMEOUT
- NAMES
- NET_WRITE_TIMEOUT

- QUERY_CACHE_TYPE
- SESSION_TRACK_SCHEMA
- SQL_MODE
- TIME_ZONE
- TRANSACTION_ISOLATION (or TX_ISOLATION)
- TRANSACTION_READ_ONLY (or TX_READ_ONLY)
- WAIT_TIMEOUT

> **Note**
>
> Le proxy RDS suit les modifications apportées aux TRANSACTION_READ_ONLY variables TRANSACTION_ISOLATION et lorsque vous les définissez au cours de la session. Toutefois, si vous les définissez lors de la prochaine transaction, le proxy RDS épingle les connexions. Ce comportement s'applique que vous utilisiez une SET instruction ou une SET TRANSACTION instruction pour configurer ces valeurs.

Minimiser l'épinglage

Le réglage des performances RDS Proxy entraîne une tentative d'optimisation de la réutilisation des connexions au niveau de la transaction (multiplexage) en réduisant l'épinglage.

Vous pouvez minimiser l'épinglage en procédant comme suit :

- Évitez les requêtes de base de données inutiles qui pourraient provoquer l'épinglage.
- Définissez les variables et les paramètres de configuration de manière cohérente sur toutes les connexions. De cette façon, les sessions ultérieures sont plus susceptibles de réutiliser les connexions qui ont ces paramètres particuliers.

  En revanche, pour PostgreSQL, la définition d'une variable entraîne l'épinglage de la session.
- Pour une base de données de la famille de moteur MySQL, appliquez un filtre d'épinglage de session au proxy. Vous pouvez configurer certains types d'opérations pour qu'elles n'épinglent pas la session si vous savez que cela n'affecte pas le bon fonctionnement de votre application.
- Découvrez la fréquence de l'épinglage en surveillant la CloudWatch métrique DatabaseConnectionsCurrentlySessionPinned Amazon. Pour en savoir plus sur ce sujet

et sur d'autres métriques CloudWatch, consultez la section Surveillance des métriques du proxy RDS avec Amazon CloudWatch.

- Si vous utilisez des instructions SET pour exécuter une initialisation identique pour chaque connexion client, vous pouvez conserver le multiplexage au niveau de la transaction. Dans ce cas, vous déplacez les instructions qui définissent l'état initial de la session vers la requête d'initialisation utilisée par un proxy. Cette propriété est une chaîne contenant une ou plusieurs instructions SQL, séparées par des points-virgules.

  Par exemple, vous pouvez définir une requête d'initialisation pour un proxy qui établit certains paramètres de configuration. RDS Proxy applique ensuite ces paramètres dès qu'il configure une nouvelle connexion pour ce proxy. Vous pouvez supprimer les instructions SET correspondantes de votre code d'application, afin qu'elles n'interfèrent pas avec le multiplexage au niveau de la transaction.

  Pour les métriques relatives à la fréquence d'épinglage d'un proxy, veuillez consulter Surveillance des métriques du proxy RDS avec Amazon CloudWatch.

Conditions qui entraînent l'épinglage pour toutes les familles de moteurs

Le proxy épingle la session à la connexion en cours dans les situations suivantes où le multiplexage peut entraîner un comportement inattendu :

- Le proxy épingle la session si la taille de texte de l'instruction est supérieure à 16 Ko.

Conditions qui entraînent l'épinglage pour RDS for Microsoft SQL Server

Pour RDS for SQL Server, les interactions suivantes entraînent également l'épinglage :

- Utilisation de plusieurs ensembles de résultats actifs (MARS). Pour plus d'informations sur MARS, consultez la documentation Microsoft SQL Server.
- Utilisation de la communication DTC (Distributed Transaction Coordinator).
- Création de tables temporaires, de transactions, de curseurs ou d'instructions préparées.
- À l'aide des instructions SET suivantes :
  - `SET ANSI_DEFAULTS`
  - `SET ANSI_NULL_DFLT`
  - `SET ARITHIGNORE`

- `SET DEADLOCK_PRIORITY`
- `SET FIPS_FLAGGER`
- `SET FMTONLY`
- `SET FORCEPLAN`
- `SET IDENTITY_INSERT`
- `SET NOCOUNT`
- `SET NOEXEC`
- `SET OFFSETS`
- `SET PARSEONLY`
- `SET QUERY_GOVERNOR_COST_LIMIT`
- `SET REMOTE_PROC_TRANSACTIONS`
- `SET ROWCOUNT`
- `SET SHOWPLAN_ALL`, `SHOWPLAN_TEXT` et `SHOWPLAN_XML`
- `SET STATISTICS`
- `SET XACT_ABORT`

Conditions qui entraînent l'épinglage pour RDS for MariaDB et RDS for MySQL

Pour MariaDB et MySQL, les interactions suivantes sont également à l'origine du pinning :

- Le proxy épingle la session en cas d'instructions de verrouillage de table `LOCK TABLE`, `LOCK TABLES` ou `FLUSH TABLES WITH READ LOCK` explicites.
- La création de verrous nommés à l'aide de `GET_LOCK` entraîne le proxy à épingler la session.
- La définition d'une variable utilisateur ou système (à quelques exceptions près) associe la session au proxy. Si cela limite considérablement la réutilisation des connexions, vous pouvez configurer les `SET` opérations pour éviter l'épinglage. Pour ce faire, ajustez la propriété des filtres d'épinglage de session. Pour plus d’informations, consultez Création d'un proxy pour Amazon RDS (Aurora) et Modification d'un RDS Proxy.
- Le proxy épingle la session lors de la création d'une table temporaire. De cette façon, le contenu de la table temporaire est conservé tout au long de la session, sans tenir compte des limites de transaction.
- L'appel des `FOUND_ROWS` fonctions `ROW_COUNT` et entraîne parfois un épinglage.

- Le proxy épingle la session en cas d'instructions préparées. Cette règle s'applique si l'instruction préparée utilise du texte SQL ou le protocole binaire.
- RDS Proxy n'épingle pas les connexions lorsque vous utilisez SET LOCAL.
- L'appel de procédures et de fonctions stockées ne provoque pas d'épinglage. RDS Proxy ne détecte aucun changement d'état de session résultant de tels appels. Assurez-vous que votre application ne modifie pas l'état de session dans les routines stockées si vous comptez sur cet état de session pour qu'il persiste dans toutes les transactions. Par exemple, le proxy RDS n'est actuellement pas compatible avec une procédure stockée qui crée une table temporaire qui persiste dans toutes les transactions.

Si vous avez des connaissances avancées sur le comportement de votre application, vous pouvez ignorer le comportement d'épinglage de certaines instructions d'application. Pour ce faire, sélectionnez l'option Filtres d'épinglage de session lors de la création du proxy. Actuellement, vous pouvez désactiver l'épinglage de session pour définir des variables de session et des paramètres de configuration.

Conditions qui entraînent l'épinglage pour RDS for PostgreSQL

Pour PostgreSQL, les interactions suivantes provoquent également l'épinglage :

- À l'aide de `SET` commandes.
- Utilisation `PREPARE` des `EXECUTE` commandes `DISCARDDEALLOCATE`,, ou pour gérer les instructions préparées.
- Création de séquences, de tables ou de vues temporaires.
- Déclarer des curseurs.
- Suppression de l'état de session.
- Écoute sur un canal de notification.
- Chargement d'un module de bibliothèque tel que`auto_explain`.
- Manipulation de séquences à l'aide de fonctions telles que `nextval` et. `setval`
- Interaction avec les verrous à l'aide de fonctions telles que `pg_advisory_lock` et`pg_try_advisory_lock`.

> **Note**
>
> RDS Proxy n'attache pas aux verrous consultatifs au niveau des transactions, en particulier `pg_advisory_xact_lockpg_advisory_xact_lock_shared,pg_try_advisory_xact_lock etpg_try_advisory_xact_lock_shared`.

- Définition d'un paramètre ou réinitialisation d'un paramètre à sa valeur par défaut. Plus précisément, utiliser `SET` des `set_config` commandes et pour attribuer des valeurs par défaut aux variables de session.
- L'appel de procédures et de fonctions stockées ne provoque pas d'épinglage. RDS Proxy ne détecte aucun changement d'état de session résultant de tels appels. Assurez-vous que votre application ne modifie pas l'état de session dans les routines stockées si vous comptez sur cet état de session pour qu'il persiste dans toutes les transactions. Par exemple, le proxy RDS n'est actuellement pas compatible avec une procédure stockée qui crée une table temporaire qui persiste dans toutes les transactions.
- Suppression de l'état de la session. Si vous utilisez des bibliothèques de regroupement de connexions avec une `DISCARD ALL` requête configurée comme une requête de réinitialisation, le proxy RDS épingle votre connexion client lors de la publication. Cela réduit l'efficacité du multiplexage du proxy et peut entraîner des résultats inattendus car la `DISCARD ALL` commande peut interférer avec la gestion de l'état de session.

Supprimer un RDS proxy

Vous pouvez supprimer un proxy lorsque vous n'en avez plus besoin. Vous pouvez également supprimer un proxy si vous mettez hors service l'instance de base de données ou le cluster qui lui est associé.

AWS Management Console

Suppression d'un proxy

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Choisissez le proxy à supprimer de la liste.
4. Sélectionnez Suppression du proxy.

AWS CLI

Pour supprimer un proxy de base de données, utilisez la AWS CLI commande delete-db-proxy. Pour supprimer les associations associées, utilisez également la deregister-db-proxy-targetscommande.

```
aws rds delete-db-proxy --name proxy_name
```

```
aws rds deregister-db-proxy-targets
    --db-proxy-name proxy_name
    [--target-group-name target_group_name]
    [--target-ids comma_separated_list]       # or
    [--db-instance-identifiers instance_id]       # or
    [--db-cluster-identifiers cluster_id]
```

RDS API

Pour supprimer un proxy de base de données, appelez la RDS API fonction Amazon eleteDBProxyD. Pour supprimer des éléments et des associations associés, vous pouvez également appeler les fonctions D eleteDBProxy TargetGroup et D eregisterDBProxy Targets.

Utilisation des points de terminaison du proxy Amazon RDS

Les points de terminaison RDS Proxy offrent des moyens flexibles et efficaces de gérer les connexions aux bases de données, ce qui améliore l'évolutivité, la disponibilité et la sécurité. Avec les points de terminaison proxy, vous pouvez :

- Simplifiez la surveillance et le dépannage : utilisez plusieurs points de terminaison pour suivre et gérer indépendamment les connexions provenant de différentes applications.

Rubriques

- Présentation des points de terminaison proxy
- Limites pour les points de terminaison proxy
- Points de terminaison proxy pour les clusters de bases de données multi-AZ
- Accès aux bases de données RDS via VPCs
- Création d'un point de terminaison proxy
- Affichage des points de terminaison proxy

- Modification d'un point de terminaison proxy
- Suppression d'un point de terminaison proxy

Présentation des points de terminaison proxy

Travailler avec des points de terminaison RDS Proxy implique les mêmes types de procédures qu'avec les points de terminaison d'instances RDS. Pour vous familiariser avec les points de terminaison RDS, consultez les informations dans Connexion à une instance de base de données exécutant le moteur de base de données MySQL et Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL.

Lorsque vous créez un point de terminaison proxy, vous pouvez l'associer à un cloud privé virtuel (VPC) différent du VPC du proxy. Cela vous permet de vous connecter au proxy depuis un autre VPC, tel qu'un VPC utilisé par une autre application au sein de votre organisation.

Pour plus d'informations sur les limites associées aux points de terminaison proxy, consultez Limites pour les points de terminaison proxy.

Les journaux du proxy RDS préfixent chaque entrée avec le nom du point de terminaison du proxy associé. Il peut s'agir du nom que vous avez spécifié pour un point de terminaison défini par l'utilisateur ou du nom spécial du point de terminaison `default` de lecture/écriture par défaut du proxy.

Chaque point de terminaison du proxy possède son propre ensemble de CloudWatch mesures. Surveillez les métriques de tous les points de terminaison du proxy, d'un point de terminaison spécifique ou de tous les points de terminaison en lecture/écriture ou en lecture seule d'un proxy. Pour de plus amples informations, veuillez consulter Surveillance des métriques du proxy RDS avec Amazon CloudWatch.

Un point de terminaison de proxy utilise le même mécanisme d'authentification que le proxy auquel il est associé. RDS Proxy configure automatiquement des permissions et des autorisations pour le point de terminaison défini par l'utilisateur, conformément aux propriétés du proxy associé.

Limites pour les points de terminaison proxy

Les points de terminaison du proxy RDS présentent les limites suivantes :

- Le point de terminaison par défaut du proxy RDS ne peut pas être modifié.

- Le nombre maximal de points de terminaison définis par l'utilisateur pour un proxy est de 20. Un proxy peut ainsi avoir jusqu'à 21 points de terminaison : le point de terminaison par défaut, plus 20 que vous créez.
- Lorsque vous associez des points de terminaison supplémentaires à un proxy, RDS Proxy détermine automatiquement les instances de base de données à utiliser dans votre cluster pour chaque point de terminaison.

Lorsque vous créez un proxy, RDS crée automatiquement un point de terminaison VPC pour une communication sécurisée entre les applications et la base de données. Le point de terminaison VPC est visible et est accessible depuis la console Amazon VPC.

L'ajout d'un nouveau point de terminaison proxy provisionne un point de terminaison d' AWS PrivateLink interface. Si vous ajoutez un ou plusieurs points de terminaison à votre proxy, des frais supplémentaires vous seront facturés. Pour plus d'informations, consultez la section Tarification du proxy RDS.

Points de terminaison proxy pour les clusters de bases de données multi-AZ

Par défaut, le point de terminaison auquel vous vous connectez lorsque vous utilisez RDS Proxy avec un cluster de base de données multi-AZ a une capacité de lecture/écriture. Par conséquent, ce point de terminaison envoie toutes les demandes à l'instance d'écriture du cluster. Toutes ces connexions sont prises en compte dans la valeur `max_connections` de l'instance d'écriture. Si votre proxy est associé à un cluster de base de données multi-AZ, vous pouvez créer des points de terminaison supplémentaires en lecture/écriture ou en lecture seule pour ce proxy.

Vous pouvez utiliser un point de terminaison en lecture seule avec votre proxy pour les requêtes en lecture seule. Vous le faites de la même façon que vous utilisez le point de terminaison de lecteur pour un cluster de base de données multi-AZ. Cela vous permet de tirer avantage de l'évolutivité de lecture d'un cluster de base de données multi-AZ avec une ou plusieurs instances de base de données de lecteur. Vous pouvez exécuter plus de requêtes et créer plus de connexions simultanément à l'aide d'un point de terminaison en lecture seule et en ajoutant d'autres instances de base de données de lecteur à votre cluster de base de données multi-AZ selon vos besoins. Ces points de terminaison du lecteur peuvent améliorer la capacité de mise à l'échelle de lecture de vos applications exigeantes en requêtes. Les points de terminaison du lecteur peuvent également améliorer la disponibilité de vos connexions si une instance de base de données de lecteur dans votre cluster devient indisponible.

Points de terminaison de lecteur pour les clusters de base de données multi-AZ

Avec RDS Proxy, vous pouvez créer et utiliser des points de terminaison de lecteur. Cependant, ces points de terminaison ne fonctionnent que pour des proxies associés à des clusters de base de données multi-AZ. Si vous utilisez l'interface de ligne de commande (CLI) ou l'API RDS, vous pouvez voir l'attribut `TargetRole` avec une valeur de `READ_ONLY`. Vous pouvez tirer parti de ces proxys en modifiant la cible d'un proxy d'une instance de base de données RDS à un cluster de base de données multi-AZ.

Vous pouvez créer des points de terminaison en lecture seule appelés points de terminaison de lecteur et y accéder lorsque vous utilisez RDS Proxy avec des clusters de base de données multi-AZ.

Amélioration de la disponibilité des applications par les points de terminaison du lecteur

Il peut arriver qu'une instance de lecteur de votre cluster devienne indisponible. Dans ce cas, les connexions qui utilisent un point de terminaison de lecteur d'un proxy de base de données peuvent récupérer plus rapidement que celles qui utilisent le point de terminaison de lecteur d'un cluster de base de données multi-AZ. RDS Proxy achemine les connexions uniquement vers l'instance de lecteur disponible du cluster. La mise en cache DNS ne provoque pas de retard lorsqu'une instance devient indisponible.

Si la connexion est multiplexée, RDS Proxy dirige les requêtes suivantes vers une autre instance de lecteur sans aucune interruption de votre application. Si une instance de lecteur est indisponible, toutes les connexions client au point de terminaison de cette instance sont fermées.

Si la connexion est épinglée, la requête suivante sur la connexion retourne une erreur. Toutefois, votre application peut se reconnecter immédiatement au même point de terminaison du proxy. RDS Proxy achemine la connexion vers une autre instance de base de données du lecteur qui se trouve dans l'état `available`. Lorsque vous vous reconnectez manuellement, RDS Proxy ne vérifie pas le décalage de réplication entre l'ancienne instance de lecteur et la nouvelle.

Si votre cluster de base de données multi-AZ n'a pas d'instances de lecteur disponibles, RDS Proxy tente de se connecter à un point de terminaison de lecteur disponible. Si aucune instance de lecteur n'est disponible pendant la période de délai d'expiration de l'emprunt de connexion, la tentative de connexion échoue. Si une instance de lecteur devient disponible, la tentative de connexion aboutit.

Amélioration de la capacité de mise à l'échelle des requêtes par les points de terminaison du lecteur

Les points de terminaison de lecteur d'un proxy peuvent améliorer l'évolutivité des requêtes du cluster de base de données multi-AZ de la manière suivante :

- Dans la mesure du possible, RDS Proxy utilise la même instance de base de données de lecteur pour tous les problèmes de requêtes utilisant une connexion de point de terminaison de lecteur particulière. De cette façon, un ensemble de requêtes associées sur les mêmes tables peut tirer avantage de la mise en cache, de l'optimisation du plan, etc., sur une instance de base de données particulière.
- Si une instance de base de données de lecteur devient indisponible, l'effet sur votre application sera différent selon que la séance est multiplexée ou épinglée. Si la séance est multiplexée, RDS Proxy achemine toutes les requêtes ultérieures vers une autre instance de base de données de lecteur sans que vous ayez à intervenir. Si la séance est épinglée, votre application obtient une erreur et doit se reconnecter. Vous pouvez vous reconnecter au point de terminaison du lecteur immédiatement et RDS Proxy achemine la connexion vers une instance de base de données de lecteur disponible. Pour plus d'informations sur le multiplexage et l'épinglage des séances proxy, consultez Vue d'ensemble des concepts de RDS proxy.

Accès aux bases de données RDS via VPCs

Par défaut, les composants de votre pile technologique RDS sont tous dans le même VPC Amazon. Supposons, par exemple, qu'une application exécutée sur une EC2 instance Amazon se connecte à une instance de base de données Amazon RDS d'un cluster de bases . Dans ce cas, le serveur d'applications et la base de données doivent se trouver tous les deux dans le même VPC.

Avec RDS Proxy, vous pouvez configurer l'accès à une instance de base de données Amazon RDS d'un cluster de bases dans un VPC à partir des ressources d'un autre VPC, telles que des instances. EC2 Par exemple, votre organisation peut avoir plusieurs applications qui accèdent aux mêmes ressources de base de données. Chaque application peut se trouver dans son propre VPC.

Pour activer l'accès entre VPC, vous créez un nouveau point de terminaison pour le proxy. Le proxy lui-même réside dans le même VPC que l'instance de base de données Amazon RDS. Cependant, le point de terminaison inter-VPC réside dans l'autre VPC, avec les autres ressources telles que les instances. EC2 Le point de terminaison inter-VPC est associé à des sous-réseaux et à des groupes de sécurité provenant du même VPC que les autres ressources. EC2 Ces associations vous permettent de vous connecter au point de terminaison à partir des applications qui, autrement, ne peuvent pas accéder à la base de données en raison des restrictions de VPC.

Les étapes suivantes vous expliquent comment créer et accéder à un point de terminaison entre VPC via RDS Proxy :

1. Créez-en deux VPCs ou choisissez-en deux VPCs que vous utilisez déjà pour le travail avec RDS. Chaque VPC doit disposer de ses propres ressources réseau associées, telles qu'une passerelle Internet, des tables de routage, des sous-réseaux et des groupes de sécurité. Si vous n'avez qu'un seul VPC, vous pouvez consulter Mise en route avec Amazon RDS à propos des étapes de configuration d'un autre VPC à suivre pour utiliser RDS avec succès. Vous pouvez également examiner votre VPC existant dans la EC2 console Amazon pour voir les types de ressources à connecter entre elles.
2. Créez un proxy de base de données associé à l'instance de base de données Amazon RDS à laquelle vous voulez vous connecter. Suivez la procédure décrite dans Création d'un proxy pour Amazon RDS (Aurora).
3. Sur la page Details (Détails) de votre proxy dans la console RDS, dans la section Proxy endpoints (Points de terminaison proxy), cliquez sur Create endpoint (Créer un point de terminaison). Suivez la procédure décrite dans Création d'un point de terminaison proxy.
4. Choisissez si le point de terminaison entre VPC doit être en lecture/écriture ou en lecture seule.
5. Au lieu d'accepter la valeur par défaut du même VPC que l'instance de base de données RDS, sélectionnez un autre VPC. Ce VPC doit se trouver dans la même AWS région que le VPC où réside le proxy.
6. Maintenant, au lieu d'accepter les valeurs par défaut pour les sous-réseaux et les groupes de sécurité du même VPC que l'instance de base de données RDS, effectuez de nouvelles sélections. Basez vos sélections sur les sous-réseaux et des groupes de sécurité du VPC que vous avez sélectionné.
7. Vous n'avez pas besoin de modifier les paramètres des secrets Secrets Manager. Les mêmes informations d'identification fonctionnent pour tous les points de terminaison de votre proxy, indépendamment du VPC dans lequel réside chaque point de terminaison.
8. Attendez que le nouveau point de terminaison atteigne l'état Available (Disponible).
9. Notez le nom complet du point de terminaison. Il s'agit de la valeur se terminant par *`Region_name`*`.rds.amazonaws.com` que vous fournissez dans le cadre de la chaîne de connexions pour votre application de base de données.
10. Accédez au nouveau point de terminaison à partir d'une ressource située dans le même VPC que le point de terminaison. Un moyen simple de tester ce processus consiste à créer une nouvelle EC2 instance dans ce VPC. Connectez-vous ensuite à l' EC2 instance et exécutez les `psql` commandes `mysql` or pour vous connecter en utilisant la valeur du point de terminaison dans votre chaîne de connexion.

Création d'un point de terminaison proxy

Pour créer un point de terminaison proxy, suivez ces instructions :

Console

Pour créer un point de terminaison proxy

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Cliquez sur le nom du proxy pour lequel vous voulez créer un nouveau point de terminaison.

   La page de détails concernant ce proxy apparaît.
4. Dans la section Proxy endpoints (Points de terminaison proxy), cliquez sur Create proxy endpoint (Créer un point de terminaison proxy).

   La fenêtre Create proxy endpoint (Créer un point de terminaison proxy) apparaît.
5. Pour Proxy endpoint name (Nom du point de terminaison proxy), saisissez le nom descriptif de votre choix.
6. Pour Target role (Rôle cible), choisissez si le point de terminaison doit être en lecture/écriture ou en lecture seule.

   Les connexions qui utilisent des points de terminaison en lecture/écriture peuvent effectuer toutes sortes d'opérations, telles que des instructions en langage de définition des données (DDL), des instructions en langage de manipulation des données (DML) et des requêtes. Ces points de terminaison se connectent toujours à l'instance principale du cluster de base de données du . Vous pouvez utiliser des points de terminaison de lecteur/écriture pour les opérations générales de base de données lorsque vous utilisez un seul point de terminaison dans votre application. Vous pouvez également utiliser des points de terminaison en lecture/écriture pour les opérations administratives, les applications de traitement des transactions en ligne (OLTP) et les tâches extract-transform-load (ETL).

   Les connexions qui utilisent un point de terminaison en lecture seule ne peuvent effectuer que des requêtes. RDSLe proxy peut utiliser l'une des instances du lecteur pour chaque connexion au point de terminaison. De cette façon, une application exigeante en requêtes peut tirer avantage de la capacité de clustering du cluster de base de données multi-AZ. Ces connexions en lecture seule n'imposent aucune surcharge à l'instance principale du cluster. Ainsi, vos

requêtes de reporting et d'analyse ne ralentissent pas les opérations d'écriture de vos OLTP applications.

7. Pour Virtual Private Cloud (VPC), choisissez la valeur par défaut pour accéder au point de terminaison à partir des mêmes EC2 instances ou d'autres ressources que celles utilisées normalement pour accéder au proxy ou à sa base de données associée. Pour configurer l'VPCaccès croisé pour ce proxy, choisissez-en un VPC autre que celui par défaut. Pour plus d'informations sur l'VPCaccès croisé, consultezAccès aux bases de données RDS via VPCs.
8. Pour les sous-réseaux, le RDS proxy remplit les mêmes sous-réseaux que le proxy associé par défaut. Pour limiter l'accès au point de terminaison à une partie seulement VPC de la plage d'adresses pouvant s'y connecter, supprimez un ou plusieurs sous-réseaux.
9. Pour le groupe VPC de sécurité, vous pouvez choisir un groupe de sécurité existant ou en créer un nouveau. RDSLe proxy remplit le ou les mêmes groupes de sécurité que le proxy associé par défaut. Si les règles d'entrée et de sortie du proxy sont appropriées pour ce point de terminaison, conservez le choix par défaut.

   Si vous choisissez de créer un nouveau groupe de sécurité, donnez-lui un nom sur cette page. Modifiez ensuite les paramètres du groupe de sécurité depuis la EC2 console.
10. Cliquez sur Create proxy endpoint (Créer un point de terminaison proxy).

AWS CLI

Pour créer un point de terminaison proxy, utilisez la AWS CLI create-db-proxy-endpointcommande.

Incluez les paramètres requis suivants :

- `--db-proxy-name` *`value`*
- `--db-proxy-endpoint-name` *`value`*
- `--vpc-subnet-ids` *`list_of_ids`*. Séparez le sous-réseau IDs par des espaces. Vous ne spécifiez pas son identifiant VPC lui-même.

Vous pouvez également ajouter les paramètres facultatifs suivants :

- `--target-role { READ_WRITE | READ_ONLY }`. Ce paramètre a pour valeur par défaut `READ_WRITE`. Lorsque le proxy est associé à un cluster de base de données multi-AZ, un cluster contenant uniquement une instance de base de données d'écriture, vous ne pouvez pas le spécifier`READ_ONLY`. Pour plus d'informations sur l'utilisation prévue des points de terminaison en

lecture seule avec les de bases de données multi-AZ, consultez. Points de terminaison de lecteur pour les clusters de base de données multi-AZ

- `--vpc-security-group-ids` *value*. Séparez le groupe de sécurité IDs par des espaces. Si vous omettez ce paramètre, RDS Proxy utilise le groupe de sécurité par défaut pour. VPC RDSLe proxy le détermine VPC en fonction du sous-réseau IDs que vous spécifiez pour le `--vpc-subnet-ids` paramètre.

Example

L'exemple suivant crée un point de terminaison proxy nommé `my-endpoint`.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-proxy-endpoint \
  --db-proxy-name my-proxy \
  --db-proxy-endpoint-name my-endpoint \
  --vpc-subnet-ids subnet_id subnet_id subnet_id ... \
  --target-role READ_ONLY \
  --vpc-security-group-ids security_group_id ]
```

Dans Windows :

```
aws rds create-db-proxy-endpoint ^
  --db-proxy-name my-proxy ^
  --db-proxy-endpoint-name my-endpoint ^
  --vpc-subnet-ids subnet_id_1 subnet_id_2 subnet_id_3 ... ^
  --target-role READ_ONLY ^
  --vpc-security-group-ids security_group_id
```

RDS API

Pour créer un point de terminaison proxy, utilisez l'action reateDBProxyPoint de terminaison RDS API C.

Affichage des points de terminaison proxy

Pour afficher les points de terminaison du proxy existants, suivez ces instructions :

Console

Pour afficher les détails d'un point de terminaison proxy

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Dans la liste, sélectionnez le proxy dont vous souhaitez afficher le point de terminaison. Cliquez sur le nom du proxy pour afficher la page contenant ses détails.
4. Dans la section Proxy endpoints (Points de terminaison proxy), sélectionnez le point de terminaison que vous voulez afficher. Cliquez sur son nom pour afficher la page contenant ses détails.
5. Examinez les paramètres dont les valeurs vous intéressent. Vous pouvez vérifier les propriétés suivantes :

   - Indique si le point de terminaison est en lecture/écriture ou en lecture seule.
   - Adresse de point de terminaison que vous utilisez dans une chaîne de connexions à la base de données.
   - Les VPC sous-réseaux et les groupes de sécurité associés au point de terminaison.

AWS CLI

Pour afficher un ou plusieurs points de terminaison du proxy, utilisez la AWS CLI describe-db-proxy-endpointscommande.

Vous pouvez ajouter les paramètres facultatifs suivants :

- `--db-proxy-endpoint-name`
- `--db-proxy-name`

L'exemple suivant décrit le point de terminaison proxy `my-endpoint`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-proxy-endpoints \
```

```
  --db-proxy-endpoint-name my-endpoint
```

Dans Windows :

```
aws rds describe-db-proxy-endpoints ^
  --db-proxy-endpoint-name my-endpoint
```

RDS API

Pour décrire un ou plusieurs points de terminaison du proxy, utilisez l'opération RDS API DescribeDBProxy Endpoints.

Modification d'un point de terminaison proxy

Pour modifier les points de terminaison de votre proxy, suivez ces instructions :

Console

Pour modifier un ou plusieurs points de terminaison proxy

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, sélectionnez Proxies.
3. Dans la liste, sélectionnez le proxy dont vous voulez modifier le point de terminaison. Cliquez sur le nom du proxy pour afficher la page contenant ses détails.
4. Dans la section Proxy endpoints (Points de terminaison proxy), sélectionnez le point de terminaison que vous voulez modifier. Vous pouvez le sélectionner dans la liste ou cliquer sur son nom pour afficher la page contenant ses détails.
5. Sur la page de détails du proxy, dans la section Proxy endpoints (Points de terminaison proxy), cliquez sur Edit (Modifier). Ou, sur la page de détails du point de terminaison du proxy, pour Actions, sélectionnez Modifier.
6. Modifiez les valeurs des paramètres que vous souhaitez remplacer.
7. Sélectionnez Enregistrer les modifications.

AWS CLI

Pour modifier un point de terminaison du proxy, utilisez la AWS CLI modify-db-proxy-endpointcommande avec les paramètres obligatoires suivants :

- `--db-proxy-endpoint-name`

Précisez les modifications apportées aux propriétés du point de terminaison à l'aide d'un ou plusieurs des paramètres suivants :

- `--new-db-proxy-endpoint-name`
- `--vpc-security-group-ids`. Séparez le groupe de sécurité IDs par des espaces.

L'exemple suivant renomme le point de terminaison proxy `my-endpoint` à `new-endpoint-name`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-proxy-endpoint \
  --db-proxy-endpoint-name my-endpoint \
  --new-db-proxy-endpoint-name new-endpoint-name
```

Dans Windows:

```
aws rds modify-db-proxy-endpoint ^
  --db-proxy-endpoint-name my-endpoint ^
  --new-db-proxy-endpoint-name new-endpoint-name
```

RDS API

Pour modifier un point de terminaison proxy, utilisez l'opération RDS API M odifyDBProxy Endpoint.

Suppression d'un point de terminaison proxy

Pour supprimer un point de terminaison pour votre proxy, suivez ces instructions :

> **Note**
>
> Vous ne pouvez pas supprimer le point de terminaison du proxy par défaut que RDS Proxy crée automatiquement pour chaque proxy.
> Lorsque vous supprimez un proxy, le RDS proxy supprime automatiquement tous les points de terminaison associés.

Console

Pour supprimer un point de terminaison proxy à l'aide du AWS Management Console

1. Dans le panneau de navigation, sélectionnez Proxies.
2. Dans la liste, sélectionnez le proxy dont vous voulez supprimer un point de terminaison. Cliquez sur le nom du proxy pour afficher la page contenant ses détails.
3. Dans la section Proxy endpoints (Points de terminaison proxy), sélectionnez le point de terminaison que vous voulez supprimer. Vous pouvez sélectionner un ou plusieurs points de terminaison dans la liste ou cliquer sur le nom d'un seul point de terminaison pour afficher sa page de détails.
4. Sur la page de détails du proxy, dans la section Proxy endpoints (Points de terminaison proxy), cliquez sur Delete (Supprimer). Ou, sur la page de détails du point de terminaison du proxy, pour Actions, choisissez Supprimer.

AWS CLI

Pour supprimer un point de terminaison proxy, exécutez la delete-db-proxy-endpointcommande avec les paramètres obligatoires suivants :

- `--db-proxy-endpoint-name`

La commande suivante supprime le point de terminaison proxy nommé `my-endpoint`.

Pour LinuxmacOS, ou Unix :

```
aws rds delete-db-proxy-endpoint \
  --db-proxy-endpoint-name my-endpoint
```

Dans Windows :

```
aws rds delete-db-proxy-endpoint ^
  --db-proxy-endpoint-name my-endpoint
```

RDS API

Pour supprimer un point de terminaison proxy doté du RDSAPI, exécutez l'opération DeleteDBProxy Endpoint. Spécifiez le nom du point de terminaison proxy pour le paramètre `DBProxyEndpointName`.

Surveillance des métriques du proxy RDS avec Amazon CloudWatch

Vous pouvez surveiller le proxy RDS à l'aide d'Amazon CloudWatch. CloudWatch collecte et traite les données brutes des proxys en near-real-time métriques lisibles. Pour trouver ces métriques dans la CloudWatch console, choisissez Metrics, puis RDS, puis Per-Proxy Metrics. Pour plus d'informations, consultez la section Utilisation CloudWatch des métriques Amazon dans le guide de CloudWatch l'utilisateur Amazon.

> **Note**
>
> RDS publie ces métriques pour chaque EC2 instance Amazon sous-jacente associée à un proxy. Un seul proxy peut être desservi par plusieurs EC2 instances. Utilisez CloudWatch les statistiques pour agréger les valeurs d'un proxy sur toutes les instances associées. Certaines de ces métriques peuvent ne pas être visibles avant la première connexion réussie par un proxy.

Dans les journaux RDS Proxy, chaque entrée est préfixée avec le nom du point de terminaison proxy associé. Ce nom peut être le nom que vous avez spécifié pour un point de terminaison défini par l'utilisateur ou le nom spécial `default` pour le point de terminaison par défaut d'un proxy qui exécute des read/write demandes.

Toutes les métriques RDS Proxy sont dans le groupe `proxy`.

Chaque point de terminaison du proxy possède ses propres CloudWatch métriques. Vous pouvez surveiller l'utilisation de chaque point de terminaison proxy individuellement. Pour plus d'informations sur les points de terminaison proxy, veuillez consulter Utilisation des points de terminaison du proxy Amazon RDS.

Vous pouvez agréger les valeurs de chaque métrique à l'aide de l'un des jeux de dimensions suivants. Par exemple, en utilisant le jeu de dimensions `ProxyName`, vous pouvez analyser l'ensemble du trafic d'un proxy particulier. En utilisant les autres jeux de dimensions, vous pouvez

fractionner les métriques de différentes manières. Vous pouvez diviser les métriques en fonction des différents points de terminaison ou bases de données cibles de chaque proxy, ou du read/write trafic en lecture seule vers chaque base de données.

- Ensemble de dimensions 1 : `ProxyName`
- Ensemble de dimensions 2 : `ProxyName`, `EndpointName`
- Ensemble de dimensions 3 : `ProxyName`, `TargetGroup`, `Target`
- Ensemble de dimensions 4 : `ProxyName`, `TargetGroup`, `TargetRole`

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`AvailabilityPercentage`	Pourcentage de temps pendant lequel le groupe cible était disponible dans le rôle indiqué par la dimension. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 4
`ClientConnections`	Le nombre actuel de connexions client. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 2
`ClientConnectionsClosed`	Le nombre de connexions client	1 minute et plus	Dimension set 1, Dimension set 2

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
	fermées. est la statistique la plus utile pour cette métriqu Sum.		
ClientConnectionsInSetup	Le nombre actuel de connexions client ouvertes mais dont l'installation n'est pas terminée. Cette métrique est donnée toutes les minutes. Sum est la statistique la plus utile pour cette métrique.	1 minute	Dimension set 1, Dimension set 2
ClientConnectionsNoTLS	Nombre actuel de connexions client sans TLS (Transport Layer Security). Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique Sum.	1 minute	Dimension set 1, Dimension set 2
ClientConnectionsReceived	Le nombre de demandes de connexion client reçues. est la statistiq ue la plus utile pour cette métriqu Sum.	1 minute et plus	Dimension set 1, Dimension set 2

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`ClientConnectionsSetupFailedAuth`	Nombre de tentatives de connexion client ayant échoué en raison d'une mauvaise configuration de l'authentification ou du protocole TLS. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2
`ClientConnectionsSetupSucceeded`	Le nombre de connexions client établies avec succès via n'importe quel mécanisme d'authentification avec ou sans protocole TLS. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2
`ClientConnectionsTLS`	Nombre actuel de connexions client avec TLS. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 2

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`DatabaseConnectionRequests`	Le nombre de demandes de création d'une connexion à une base de données. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 3, Dimension set 4
`DatabaseConnectionRequestsWithTLS`	Nombre de demandes de création d'une connexion à une base de données avec TLS. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 3, Dimension set 4
`DatabaseConnections`	Le nombre actuel de connexions à une base de données. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`DatabaseConnectionsBorrowLatency`	Temps, en microsecondes, nécessaire au proxy surveillé pour obtenir une connexion à la base de données. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2
`DatabaseConnectionsCurrentlyBorrowed`	Le nombre actuel de connexions à une base de données en état d'emprunt. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4
`DatabaseConnectionsCurrentlyInTransaction`	Nombre actuel de connexions à la base de données dans une transaction. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`DatabaseConnectionsCurrentlySessionPinned`	Nombre de connexions à la base de données actuellement épinglées en raison d'opérations dans les demandes client qui modifient l'état de session. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4
`DatabaseConnectionsSetupFailed`	Le nombre de demandes de connexion à une base de données qui ont échoué. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 3, Dimension set 4
`DatabaseConnectionsSetupSucceeded`	Le nombre de connexions à une base de données correctement établies avec ou sans protocole TLS. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 3, Dimension set 4

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`DatabaseConnectionsWithTLS`	Nombre actuel de connexions à une base de données avec TLS. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4
`MaxDatabaseConnectionsAllowed`	Le nombre maximal de connexions à une base de données autorisées. Cette métrique est donnée toutes les minutes. est la statistique la plus utile pour cette métrique `Sum`.	1 minute	Dimension set 1, Dimension set 3, Dimension set 4
`QueryDatabaseResponseLatency`	Temps, en microsecondes, pris par la base de données pour répondre à la requête. est la statistique la plus utile pour cette métriqu `Average`.	1 minute et plus	Dimension set 1, Dimension set 2, Dimension set 3, Dimension set 4

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`QueryRequests`	Le nombre de requêtes reçues. Une requête comprenant plusieurs instructions est comptée comme étant une seule et même requête. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2
`QueryRequestsNoTLS`	Nombre de requêtes reçues de connexions non TLS. Une requête comprenant plusieurs instructions est comptée comme étant une seule et même requête. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2

Métrique	Description	Période de validité	CloudWatch ensemble de dimensions
`QueryRequestsTLS`	Nombre de requêtes reçues des connexions TLS. Une requête comprenant plusieurs instructions est comptée comme étant une seule et même requête. est la statistique la plus utile pour cette métriqu `Sum`.	1 minute et plus	Dimension set 1, Dimension set 2
`QueryResponseLatency`	Temps, en microsecondes, entre l'obtention d'une demande de requête et le proxy qui y répond. est la statistique la plus utile pour cette métriqu `Average`.	1 minute et plus	Dimension set 1, Dimension set 2

Vous trouverez les journaux de l'activité du proxy RDS CloudWatch dans le AWS Management Console. Chaque proxy dispose d'une entrée dans la page Groupes de journaux.

> **Important**
>
> Ces journaux sont destinés à la consommation humaine à des fins de dépannage et non à un accès par programmation. Le format et le contenu des journaux sont susceptibles d'être modifiés.
>
> En particulier, les journaux plus anciens ne contiennent pas de préfixes indiquant le point de terminaison pour chaque requête. Dans les journaux plus récents, chaque entrée est préfixée avec le nom du point de terminaison proxy associé. Ce nom peut être celui que vous avez

spécifié pour un point de terminaison défini par l'utilisateur, ou le nom spécial `default` pour les demandes utilisant le point de terminaison par défaut d'un proxy.

Utilisation des événements RDS proxy

Un événement indique un changement dans un environnement tel qu'un AWS environnement, un service ou une application d'un partenaire de logiciel en tant que service (SaaS). Il peut également s'agir de l'une de vos propres applications ou services personnalisés. Par exemple, Amazon RDS génère un événement lorsque vous créez ou modifiez un RDS proxy. Amazon RDS diffuse des événements à Amazon EventBridge en temps quasi réel. Vous trouverez ci-dessous une liste des événements RDS Proxy auxquels vous pouvez vous abonner ainsi qu'un exemple d'événement RDS Proxy.

Pour de plus amples informations sur l'utilisation des événements, veuillez consulter les éléments suivants :

- Pour obtenir des instructions sur la façon d'afficher les événements à l'aide du AWS Management Console AWS CLI, ou RDSAPI, voir Consulter les RDS événements Amazon.
- Pour savoir comment configurer RDSAmazon pour envoyer des événements à EventBridge, consultez Création d'une règle qui se déclenche sur un événement Amazon RDS.

RDSÉvénements liés au proxy

Le tableau suivant indique la catégorie d'événements et la liste des événements dont le type source est un RDS proxy.

Catégorie	ID d'événement RDS	Message	Remarques
modification de configuration	RDS- EVENT -0204	RDSproxy de base de données modifié *name*.	Aucun
modification de configuration	RDS- EVENT -0207	RDSa modifié le point final du proxy de base de données *name*.	Aucun

Catégorie	ID d'événement RDS	Message	Remarques
modification de configuration	RDS- EVENT -0213	RDSa détecté l'ajout de l'instance de base de données et l'a automatiquement ajoutée au groupe cible du proxy de base de données *name*.	Aucun
modification de configuration	RDS- EVENT -0214	RDSsuppression détectée d'une instance de base de données *name* et l'a automatiquement retiré du groupe cible *name* du proxy de base de données *name*.	Aucun
modification de configuration	RDS- EVENT -0215	RDSsuppression détectée du cluster de base de données *name* et l'a automatiquement retiré du groupe cible *name* du proxy de base de données *name*.	Aucun
création	RDS- EVENT -0203	RDSproxy de base de données créé *name*.	Aucun
création	RDS- EVENT -0206	RDSpoint de terminaison créé *name* pour le proxy de base de données *name*.	Aucun
suppression	RDS- EVENT -0205	RDSproxy de base de données supprimé *name*.	Aucun
suppression	RDS- EVENT -0208	RDSpoint de terminaison supprimé *name* pour le proxy de base de données *name*.	Aucun

Catégorie	ID d'événement RDS	Message	Remarques
échec	RDS- EVENT -0243	RDSéchec du provision nement de la capacité pour le proxy *name* car il n'y a pas assez d'adresses IP disponibles dans vos sous-réseaux : *name*. Pour résoudre le problème, assurez-vous que vos sous-réseaux comporten t le nombre minimum d'adresses IP inutilisées, comme recommandé dans la documentation du RDS proxy.	Pour déterminer le nombre recommandé pour votre classe d'instances, consultez Planification de la capacité des adresses IP.
échec	RDS- EVENT -0275	RDSlimité certaines connexions au proxy de base de données *name*. Le nombre de demandes de connexion simultanées du client au proxy a dépassé la limite.	Aucun

Voici un exemple d'événement RDS Proxy au JSON format. L'événement indique que le point de terminaison nommé `my-endpoint` du RDS proxy a été RDS modifié`my-rds-proxy`. L'ID de l'événement est RDS - EVENT -0207.

```
{
  "version": "0",
  "id": "68f6e973-1a0c-d37b-f2f2-94a7f62ffd4e",
  "detail-type": "RDS DB Proxy Event",
  "source": "aws.rds",
  "account": "123456789012",
  "time": "2018-09-27T22:36:43Z",
```

```
  "region": "us-east-1",
  "resources": [
     "arn:aws:rds:us-east-1:123456789012:db-proxy:my-rds-proxy"
  ],
  "detail": {
    "EventCategories": [
      "configuration change"
    ],
    "SourceType": "DB_PROXY",
    "SourceArn": "arn:aws:rds:us-east-1:123456789012:db-proxy:my-rds-proxy",
    "Date": "2018-09-27T22:36:43.292Z",
    "Message": "RDS modified endpoint my-endpoint of DB Proxy my-rds-proxy.",
    "SourceIdentifier": "my-endpoint",
    "EventID": "RDS-EVENT-0207"
  }
}
```

Résolution des problèmes liés au RDS Proxy

Vous trouverez ci-dessous des idées pour résoudre certains problèmes courants liés au proxy RDS et des informations sur les CloudWatch journaux du proxy RDS.

Dans les journaux RDS Proxy, chaque entrée est préfixée avec le nom du point de terminaison proxy associé. Ce nom peut être celui que vous avez spécifié pour un point de terminaison défini par l'utilisateur. Il peut également s'agir du nom spécial du point `default` de terminaison par défaut d'un proxy qui exécute des demandes de lecture/écriture. Pour plus d'informations sur les points de terminaison proxy, veuillez consulter Utilisation des points de terminaison du proxy Amazon RDS.

Rubriques

- Vérification de la connectivité pour un proxy
- Problèmes courants et solutions correspondantes
- Résolution des problèmes liés au proxy RDS avec RDS pour MySQL
- Résolution des problèmes liés au proxy RDS avec RDS pour PostgreSQL

Vérification de la connectivité pour un proxy

Vous pouvez utiliser les commandes suivantes pour vérifier que tous les composants tels que le proxy, la base de données et les instances de calcul de la connexion peuvent communiquer entre eux.

Examinez le proxy lui-même à l'aide de la describe-db-proxiescommande. Examinez également le groupe cible associé à l'aide de la commande describe-db-proxy-target-groups. Vérifiez que les détails des cibles correspondent à l'instance de base de données RDS que vous prévoyez d'associer au proxy. Utilisez des commandes telles que les suivantes.

```
aws rds describe-db-proxies --db-proxy-name $DB_PROXY_NAME
aws rds describe-db-proxy-target-groups --db-proxy-name $DB_PROXY_NAME
```

Pour vérifier que le proxy peut se connecter à la base de données sous-jacente, examinez les cibles spécifiées dans les groupes cibles à l'aide de la describe-db-proxy-targetscommande. Utilisez une commande telle que la suivante.

```
aws rds describe-db-proxy-targets --db-proxy-name $DB_PROXY_NAME
```

La sortie de la describe-db-proxy-targetscommande inclut un `TargetHealth` champ. Vous pouvez examiner les champs `State`, `Reason` et `Description` dans `TargetHealth` pour vérifier si le proxy peut communiquer avec l'instance de base de données sous-jacente.

- Si la valeur `State` est `AVAILABLE`, cela indique que le proxy peut se connecter à l'instance de base de données.
- Si la valeur `State` est `UNAVAILABLE`, cela signale un problème de connexion temporaire ou permanent. Dans ce cas, examinez les champs `Reason` et `Description`. Par exemple, si `Reason` a une valeur `PENDING_PROXY_CAPACITY`, essayez de vous connecter à nouveau une fois que le proxy a terminé son opération de mise à l'échelle. Si `Reason` a une valeur `UNREACHABLE`, `CONNECTION_FAILED` ou `AUTH_FAILURE`, utilisez l'explication du champ `Description` pour vous aider à diagnostiquer le problème.
- Le champ `State` peut avoir une valeur `REGISTERING` pendant une courte période avant de passer à `AVAILABLE` ou `UNAVAILABLE`.

Si la commande Netcat (`nc`) suivante aboutit, vous pouvez accéder au point de terminaison du proxy depuis l' EC2 instance ou un autre système sur lequel vous êtes connecté. Cette commande signale un échec si vous n'êtes pas dans le même VPC que le proxy et la base de données associée. Vous pouvez peut-être vous connecter directement à la base de données sans vous trouver dans le même VPC. Cependant, vous ne pouvez pas vous connecter au proxy sauf si vous êtes dans le même VPC.

```
nc -zx MySQL_proxy_endpoint 3306
```

```
nc -zx PostgreSQL_proxy_endpoint 5432
```

Vous pouvez utiliser les commandes suivantes pour vous assurer que votre EC2 instance possède les propriétés requises. En particulier, le VPC de l' EC2 instance doit être le même que le VPC du cluster de l' auquel le proxy se connecte.

```
aws ec2 describe-instances --instance-ids your_ec2_instance_id
```

Examinez les secrets Secrets Manager utilisés pour le proxy.

```
aws secretsmanager list-secrets
aws secretsmanager get-secret-value --secret-id your_secret_id
```

Assurez-vous que le `SecretString` champ affiché par `get-secret-value` est codé sous la forme d'une chaîne JSON incluant les `password` champs `username` et. L'exemple suivant illustre le format du champ `SecretString`.

```
{
  "ARN": "some_arn",
  "Name": "some_name",
  "VersionId": "some_version_id",
  "SecretString": '{"username":"some_username","password":"some_password"}',
  "VersionStages": [ "some_stage" ],
  "CreatedDate": some_timestamp
}
```

Problèmes courants et solutions correspondantes

Cette section décrit certains problèmes courants et les solutions potentielles lors de l'utilisation du proxy RDS.

Après avoir exécuté la commande `aws rds describe-db-proxy-targets` CLI, si la `TargetHealth` description l'indique`Proxy does not have any registered credentials`, vérifiez les points suivants :

- Des informations d'identification sont enregistrées pour permettre à l'utilisateur d'accéder au proxy.
- Le rôle IAM permettant d'accéder au secret du Gestionnaire de Secrets Manager utilisé par le proxy est valide.

Vous pouvez rencontrer les événements RDS suivants lors de la connexion à un proxy de base de données ou de sa création.

Catégorie	ID d'évènement RDS	Description
échec	RDS-EVENT-0243	RDS n'a pas pu allouer la capacité pour le proxy car il n'y a pas suffisamment d'adresses IP disponibles dans vos sous-réseaux. Pour résoudre ce problème, veillez à ce que vos sous-réseaux aient le nombre minimum d'adresses IP inutilisées. Pour déterminer le nombre recommandé pour votre classe d'instances, consultez Planification de la capacité des adresses IP.
échec	RDS-EVENT-0275	RDS a limité certaines connexions au proxy de base de données. *name* Le nombre de demandes de connexion simultanées du client au proxy a dépassé la limite.

Vous pouvez rencontrer les problèmes suivants lors de la création d'un proxy ou de la connexion à un proxy.

Erreur	Causes ou solutions de contournement
`403: The security token included in`	Sélectionnez un rôle IAM existant au lieu d'en créer un nouveau.

Erreur	Causes ou solutions de contournement
`the request is invalid`	

Résolution des problèmes liés au proxy RDS avec RDS pour MySQL

Vous pouvez rencontrer les problèmes suivants lors de la connexion à un proxy MySQL.

Erreur	Causes ou solutions de contournement
`ERROR 1040 (HY000): Connections rate limit exceeded (`*`limit_value`*` )`	Le taux de demandes de connexion du client au proxy a dépassé la limite.
`ERROR 1040 (HY000): IAM authentication rate limit exceeded`	Le nombre de demandes de connexion simultanée avec authentification IAM du client au proxy a dépassé la limite.
`ERROR 1040 (HY000): Number simultaneous connections exceeded (`*`limit_value`*` )`	Le nombre de demandes de connexion simultanée du client au proxy a dépassé la limite.
`ERROR 1045 (28000): Access denied for user '`*`DB_USER`*`'@'%' (using password: YES)`	Le secret Secrets Manager utilisé par le proxy ne correspond pas au nom d'utilisateur et au mot de passe d'un utilisateur de base de données existant. Mettez à jour les informations d'identification dans le secret Secrets Manager ou assurez-vous que l'utilisateur de base de données existe et possède le même mot de passe que celui du secret.

Erreur	Causes ou solutions de contournement
`ERROR 1105 (HY000): Unknown error`	Une erreur inconnue s'est produite.
`ERROR 1231 (42000): Variable ''character_set_client'' can't be set to the value of value`	La valeur définie pour le paramètre `character_set_client` n'est pas valide. Par exemple, la valeur `ucs2` n'est pas valide, car elle peut bloquer le serveur MySQL.
`ERROR 3159 (HY000): This RDS Proxy requires TLS connections.`	Vous avez activé le paramètre Exiger la sécurité de la couche de transport dans le proxy, mais votre connexion a inclus le paramètre `ssl-mode=DISABLED` dans le client MySQL. Effectuez l'une des actions suivantes : • Désactivez le paramètre Exiger la sécurité de la couche de transport pour le proxy. • Connectez-vous à la base de données en utilisant le paramètre minimal de `ssl-mode=REQUIRED` dans le client MySQL.
`ERROR 2026 (HY000): SSL connection error: Internal Server Error`	La négociation TLS avec le proxy a échoué. Les causes possibles sont notamment les suivantes : • SSL est requis, mais le serveur ne le prend pas en charge. • Une erreur interne du serveur s'est produite. • Une mauvaise négociation s'est produite.

Erreur	Causes ou solutions de contournement
`ERROR 9501 (HY000): Timed-out waiting to acquire database connection`	Le proxy a expiré en attendant l'obtention d'une connexion à la base de données. Les causes possibles sont notamment les suivantes : • Le proxy n'est pas en mesure d'établir une connexion à la base de données, car le nombre maximal de connexions a été atteint. • Le proxy n'est pas en mesure d'établir une connexion à la base de données, car la base de données n'est pas disponible.

Résolution des problèmes liés au proxy RDS avec RDS pour PostgreSQL

Vous pouvez rencontrer les problèmes suivants lors de la connexion à un proxy PostgreSQL.

Erreur	Cause	Solution
`ERROR 28000: IAM authentication is allowed only with SSL connections.`	L'utilisateur a essayé de se connecter à la base de données à l'aide de l'authentification IAM en utilisant le paramètre `sslmode=disable` du client PostgreSQL.	L'utilisateur doit se connecter à la base de données en utilisant le paramètre minimal `sslmode=require` du client PostgreSQL. Pour de plus amples informations, veuillez consulter la documentation PostgreSQL SSL Support .
`ERROR 28000: This RDS proxy has no credentials for the role` *`role_name`* `. Check the credentials for this role and try again.`	Il n'y a pas de secret Secrets Manager pour ce rôle.	Ajoutez un secret Secrets Manager pour ce rôle. Pour de plus amples informations, veuillez consulter Configuration de l'authentification IAM pour le proxy RDS.
`ERROR 28000: RDS supports only IAM, MD5, or SCRAM authentication.`	Le client de base de données utilisé pour se connecter au proxy utilise un mécanisme d'authentification qui n'est	Si vous n'utilisez pas l'authentification IAM, utilisez l'authentification par mot de passe MD5 ou SCRAM.

Erreur	Cause	Solution
	actuellement pas pris en charge par le proxy.	
`ERROR 28000: A user name is missing from the connectio n startup packet. Provide a user name for this connection.`	Le client de base de données utilisé pour la connexion au proxy n'envoie pas de nom d'utilisateur lorsqu'il tente d'établir une connexion.	Veillez à définir un nom d'utilisateur lors de la configuration d'une connexion au proxy à l'aide du client PostgreSQL de votre choix.
`ERROR 28000: IAM is allowed only with SSL connections.`	Un client a essayé de se connecter à l'aide de l'authent ification IAM, mais le protocole SSL n'était pas activé.	Activez SSL sur le client PostgreSQL.
`ERROR 28000: This RDS Proxy requires TLS connections.`	L'utilisateur a activé l'option Exiger la sécurité de la couche de transport mais a essayé de se connecter en utilisant `sslmode=disable` dans le client PostgreSQL.	Pour corriger cette erreur, effectuez l'une des opérations suivantes : • Désactivez l'option Exiger la sécurité de la couche de transport du proxy. • Connectez-vous à la base de données en utilisant le paramètre minimal `sslmode=allow` du client PostgreSQL.

Erreur	Cause	Solution
`ERROR 28P01: IAM authentication failed for user` *`user_name`* `. Check the IAM token for this user and try again.`	Cette erreur peut être due aux raisons suivantes : • Le client a indiqué un nom d'utilisateur IAM incorrect. • Le client a fourni un jeton d'autorisation IAM incorrect pour l'utilisateur • Le client utilise une politique IAM qui ne dispose pas des autorisations nécessaires. • Le client a fourni un jeton d'autorisation IAM expiré pour l'utilisateur.	Pour corriger cette erreur, procédez comme suit : 1. Vérifiez que l'utilisateur IAM indiqué existe. 2. Vérifiez que le jeton d'autorisation IAM appartien t à l'utilisateur IAM indiqué. 3. Vérifiez que la politique IAM dispose des autorisations adéquates pour RDS. 4. Vérifiez la validité du jeton d'autorisation IAM utilisé.
`ERROR 28P01: The password that was provided for the role` *`role_name`* `is wrong.`	Le mot de passe de ce rôle ne correspond pas au secret Secrets Manager.	Vérifiez le secret de ce rôle dans Secrets Manager pour voir si le mot de passe est le même que celui utilisé sur votre client PostgreSQL.
`ERROR 28P01: The IAM authentication failed for the role` *`role_name`* `. Check the IAM token for this role and try again.`	Il y a un problème avec le jeton IAM utilisé pour l'authent ification IAM.	Générez un nouveau jeton d'authentification et utilisez-le dans une nouvelle connexion.

Erreur	Cause	Solution
`ERROR 0A000: Feature not supported: RDS Proxy supports only version 3.0 of the PostgreSQL messaging protocol.`	Le client PostgreSQL utilisé pour la connexion au proxy utilise un protocole antérieur à la version 3.0.	Utilisez un client PostgreSQL plus récent qui prend en charge le protocole de messagerie 3.0. Si vous utilisez l'interface de ligne de commande `psql` de PostgreSQL, utilisez une version supérieure ou égale à 7.4.
`ERROR 0A000: Feature not supported: RDS Proxy currently doesn't support streaming replication mode.`	Le client PostgreSQL utilisé pour la connexion au proxy essaie d'utiliser le mode de réplication de streaming, lequel n'est actuellement pas pris en charge par RDS Proxy.	Désactivez le mode de réplication de streaming sur le client PostgreSQL utilisé pour la connexion.
`ERROR 0A000: Feature not supported: RDS Proxy currently doesn't support the option` *`option_name`* `.`	Par le biais du message de démarrage, le client PostgreSQL utilisé pour la connexion au proxy demande une option qui n'est pas actuellement prise en charge par RDS Proxy.	Désactivez l'option indiquée comme non prise en charge dans le message ci-dessus sur le client PostgreSQL utilisé pour la connexion.
`ERROR 53300: The IAM authentication failed because of too many competing requests.`	Le nombre de demandes de connexion simultanée avec authentification IAM du client au proxy a dépassé la limite.	Réduisez le taux d'établissement de connexions à l'aide de l'authentification IAM à partir d'un client PostgreSQL.
`ERROR 53300: The maximum number of client connections to the proxy exceeded` *`number_value`* `.`	Le nombre de demandes de connexion simultanée du client au proxy a dépassé la limite.	Réduisez le nombre de connexions actives des clients PostgreSQL à ce proxy RDS.

Erreur	Cause	Solution
`ERROR 53300: Rate of connection to proxy exceeded` *`number_value`* `.`	Le taux de demandes de connexion du client au proxy a dépassé la limite.	Réduisez le taux d'établissement de connexions à partir d'un client PostgreSQL.
`ERROR XX000: Unknown error.`	Une erreur inconnue s'est produite.	Contactez le AWS Support pour étudier le problème.
`ERROR 08000: Timed-out waiting to acquire database connection.`	Le proxy a expiré en attendant d'établir une connexion à la base de données pendant la durée spécifiée par le `ConnectionBorrowTimeout` paramètre. Les causes possibles sont notamment les suivantes : • Le proxy ne peut pas établir une connexion à la base de données, car le nombre maximal de connexions a été atteint. • Le proxy ne peut pas établir de connexion à la base de données parce que la base de données n'est pas disponible ou si l'établissement de la connexion à la base de données prend plus de temps que la configuration`ConnectionBorrowTimeout` .	Les solutions possibles sont les suivantes : • Évitez d'épingler les connexions proxy. Consultez Éviter d'épingler un proxy RDS. • Révision `ConnectionBorrowTimeout` et `MaxConnectionsPercent` paramètres. Consultez Considérations relatives à la connexion au proxy RDS. • Vérifiez la disponibilité des cibles. Consultez `AvailabilityPercentage` dans Surveillance des métriques du proxy RDS avec Amazon CloudWatch.

Erreur	Cause	Solution
`ERROR XX000: Request returned an error: database_error .`	La connexion à la base de données établie à partir du proxy a renvoyé une erreur.	La solution dépend de l'erreur de base de données spécifique. Par exemple : `Request returned an error: database "your-database-name" does not exist`. Cela signifie que le nom de base de données spécifié n'existe pas sur le serveur de base de données. Ou cela signifie que le nom d'utilisateur utilisé comme nom de base de données (si un nom de base de données n'est pas spécifié) n'existe pas sur le serveur.

Résolution des problèmes liés à la **`postgres`** suppression

Si vous supprimez par erreur la `postgres` base de données de votre instance, vous devez restaurer la base de données pour rétablir la connectivité à votre instance. Exécutez les commandes suivantes dans votre instance de base de données :

```
CREATE DATABASE postgres;
GRANT CONNECT ON DATABASE postgres TO rdsproxyadmin;
```

Utilisation de RDS Proxy avec AWS CloudFormation

Vous pouvez utiliser RDS Proxy avec AWS CloudFormation. Cela vous permet de créer des groupes de ressources connexes. Un tel groupe peut inclure un proxy qui peut se connecter à une instance de base de données Amazon RDS que vous venez de créer. La prise en charge de RDS Proxy dans AWS CloudFormation implique deux nouveaux types de registres : `DBProxy` et `DBProxyTargetGroup`.

La liste suivante présente un exemple de modèle AWS CloudFormation pour RDS Proxy.

```
Resources:
 DBProxy:
   Type: AWS::RDS::DBProxy
   Properties:
     DBProxyName: CanaryProxy
     EngineFamily: MYSQL
     RoleArn:
      Fn::ImportValue: SecretReaderRoleArn
     Auth:
       - {AuthScheme: SECRETS, SecretArn: !ImportValue ProxySecret, IAMAuth: DISABLED}
     VpcSubnetIds:
       Fn::Split: [",", "Fn::ImportValue": SubnetIds]

 ProxyTargetGroup:
   Type: AWS::RDS::DBProxyTargetGroup
   Properties:
     DBProxyName: CanaryProxy
     TargetGroupName: default
     DBInstanceIdentifiers:
       - Fn::ImportValue: DBInstanceName
   DependsOn: DBProxy
```

Pour plus d'informations sur les ressources de cet exemple, consultez DBProxy et DBProxyTargetGroup.

Pour de plus amples informations sur les ressources que vous pouvez créer avec AWS CloudFormation, consultez Référence de type de ressource RDS.

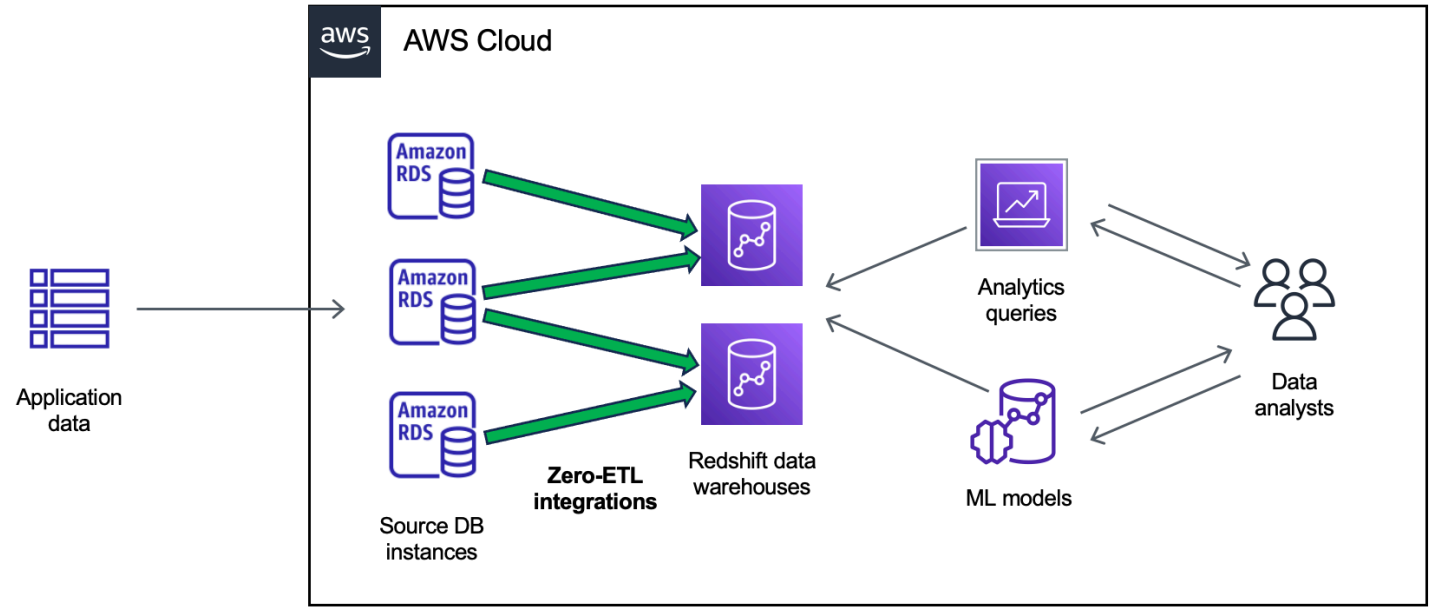
Intégrations Amazon RDS Zero-ETL

Il s'agit d'une solution entièrement gérée permettant de rendre les données transactionnelles disponibles dans votre destination d'analyse après leur écriture dans un cluster de base de données RDS Aurora . L'extraction, la transformation et le chargement (ETL) sont le processus qui consiste à combiner des données provenant de sources multiples dans un vaste entrepôt de données central.

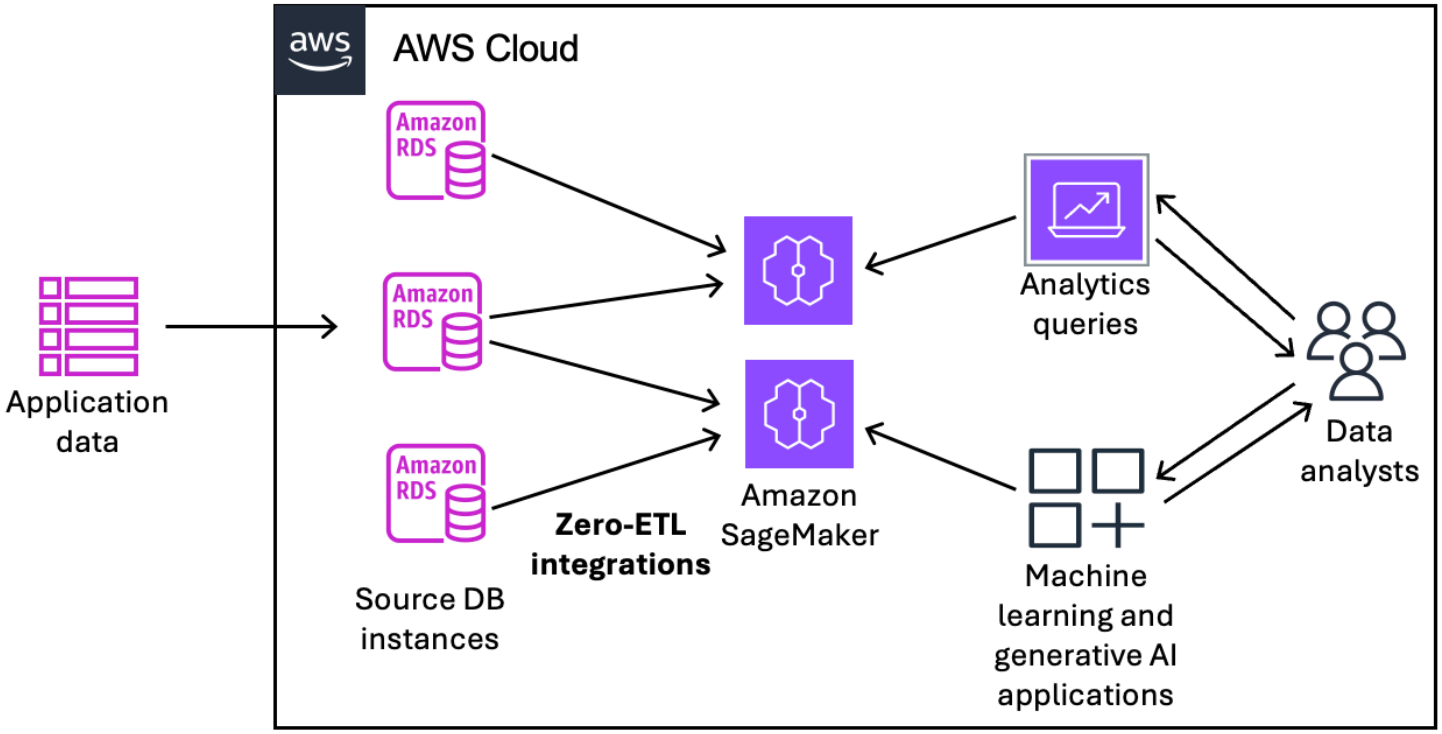
Une intégration zéro ETL rend les données de votre disponibles dans Amazon Redshift ou dans Amazon SageMaker AI un Lakehouse en temps quasi réel. Une fois que ces données se trouvent dans l'entrepôt de données ou le lac de données cible, vous pouvez optimiser vos charges de travail d'analyse, de machine learning et d'IA à l'aide des fonctionnalités intégrées, telles que l'apprentissage automatique, les vues matérialisées, le partage de données, l'accès fédéré à plusieurs magasins de données et lacs de données, ainsi que les intégrations avec SageMaker Amazon AI QuickSight, etc. Services AWS

Pour créer une intégration zéro ETL, vous devez spécifier un comme source et un entrepôt de données ou un lakehouse pris en charge comme cible. L'intégration réplique les données de la base de données source vers l'entrepôt de données ou le lakehouse cible.

Le schéma suivant illustre cette fonctionnalité pour une intégration zéro ETL avec Amazon Redshift :



Le schéma suivant illustre cette fonctionnalité pour une intégration sans ETL dans un Amazon SageMaker AI lakehouse :



L'intégration surveille l'état du pipeline de données et effectue la récupération en cas de problèmes, lorsque cela est possible. Vous pouvez créer des intégrations à partir de plusieurs bases de données RDS () dans un seul entrepôt de données cible ou un lakehouse, ce qui vous permet d'obtenir des informations sur plusieurs applications.

Rubriques

- Avantages
- Concepts clés
- Limites
- Quotas
- Régions prises en charge
- Démarrez avec les intégrations Amazon RDS
- Création d'intégrations zéro ETL d'Amazon RDS à Amazon Redshift
- Création d'intégrations Amazon RDS Zero-ETL avec un lakehouse Amazon SageMaker
- Filtrage des données pour les intégrations Amazon RDS
- Ajouter des données à une base de données RDS source () et les interroger
- Affichage et surveillance des intégrations Amazon RDS
- Modification des intégrations Amazon RDS

- Suppression des intégrations Amazon RDS Zero-ETL
- Résolution des problèmes liés aux intégrations Amazon RDS Zero-ETL

Avantages

Les intégrations RDS Zero-ETL présentent les avantages suivants :

- Elles vous aident à dériver des informations holistiques de plusieurs sources de données.
- Elles éliminent la nécessité de créer et de gérer des pipelines de données complexes qui effectuent des opérations d'extraction, de transformation et de chargement (ETL). Les intégrations zéro ETL suppriment les défis liés à la création et à la gestion de pipelines en les provisionnant et en les gérant pour vous.
- Elles réduisent la charge opérationnelle et les coûts, et vous permettent de vous concentrer sur l'amélioration de vos applications.
- Vous pouvez tirer parti des capacités d'analyse et de machine learning de la destination cible pour obtenir des informations à partir de données transactionnelles et autres, afin de répondre efficacement aux événements critiques et urgents.

Concepts clés

Lorsque vous commencez à utiliser des intégrations zéro ETL, tenez compte des concepts suivants :

Integration

Un pipeline de données entièrement géré qui réplique automatiquement les données transactionnelles et les schémas d'un cluster de base de données RDS vers un entrepôt de données ou un catalogue.

données source

Le duquel les données sont répliquées. Vous pouvez spécifier une instance de base de données mono-AZ ou multi-AZ, ou un cluster de base de données multi-AZ.

Cible

L'entrepôt de données ou le lakehouse dans lequel les données sont répliquées. Il existe deux types d'entrepôts de données : l'entrepôt de données en cluster provisionné et l'entrepôt de données sans serveur. Un entrepôt de données en cluster provisionné est une collection de ressources informatiques appelées nœuds, qui sont organisées en un groupe appelé cluster. Un

entrepôt de données sans serveur est composé d'un groupe de travail qui stocke les ressources de calcul et d'un espace de noms qui héberge les utilisateurs et les objets de base de données. Les deux entrepôts de données exécutent un moteur d'analyse et contiennent une ou plusieurs bases de données.

Un lakehouse cible comprend des catalogues, des bases de données, des tables et des vues. Pour plus d'informations sur l'architecture de Lakehouse, consultez le guide SageMaker Lakehouse componentsde l'Amazon SageMaker AI Unified Studioutilisateur.

Plusieurs bases de données sources de données peuvent écrire sur la même cible.

Pour plus d'informations, consultez Architecture système de l'entrepôt de données dans le Guide du développeur de base de données Amazon Redshift.

Limites

Les limitations suivantes s'appliquent aux intégrations RDS Zero-ETL.

Rubriques

- Limitations générales
- Limitations propres à RDS for MySQL
- Limites de RDS pour PostgreSQL
- Limitations RDS for Oracle
- Limitations propres à Amazon Redshift
- Amazon SageMaker AI limites du lakehouse

Limitations générales

- Le de base de données source doit se trouver dans la même région que la cible.
- Vous ne pouvez pas renommer un s'il possède des intégrations existantes.
- Vous ne pouvez pas créer plusieurs intégrations entre les mêmes bases de données source et cible.
- Vous ne pouvez pas supprimer un de base de données doté d'intégrations existantes. Vous devez d’abord supprimer toutes les intégrations associées.
- Si vous arrêtez le de base de données source, les dernières transactions risquent de ne pas être répliquées vers la cible tant que vous ne reprenez pas le de bases de données.

- Vous ne pouvez pas supprimer une intégration si la base de données source est arrêtée.
- Si votre de base de données est à l'origine d'un déploiement bleu/vert, les environnements bleu et vert ne peuvent pas comporter d'intégrations zéro ETL existantes lors du passage au numérique. Vous devez d'abord supprimer l'intégration et basculer, puis la recréer.
- Vous ne pouvez pas créer d'intégration pour une base de données source dont une autre intégration est activement créée.
- Lors de la création initiale d'une intégration ou lors de la resynchronisation d'une table, l'ensemencement des données de la source vers la cible peut prendre 20 à 25 minutes, voire plus, selon la taille de la base de données source. Ce délai peut entraîner une augmentation du délai de réplication.
- Certains types de données ne sont pas pris en charge. Pour de plus amples informations, veuillez consulter the section called “Différences de type de données”.
- Les tables système, les tables temporaires et les vues ne sont pas répliquées vers les entrepôts cibles.
- L'exécution de commandes DDL (par exemple`ALTER TABLE`) sur une table source peut déclencher une resynchronisation de la table, rendant la table indisponible pour les requêtes pendant la resynchronisation. Pour de plus amples informations, veuillez consulter the section called “Une ou plusieurs de mes tables Amazon Redshift nécessitent une resynchronisation”.

Limitations propres à RDS for MySQL

- Votre base de données source doit exécuter une version prise en charge de RDS pour MySQL. Pour une liste de versions prises en charge, consultez the section called “Intégrations zéro ETL ”.
- Les intégrations Zero-ETL ne sont pas prises en charge à la fois sur l'instance principale et sur une instance de réplication en lecture dans la même AWS région.
- Les intégrations zéro ETL s'appuient sur la journalisation binaire MySQL (binlog) pour capturer les modifications continues des données. N'utilisez pas le filtrage des données basé sur le binlog, car cela peut entraîner des incohérences entre les bases de données source et cible.
- Les intégrations zéro ETL sont prises en charge uniquement pour les bases de données configurées pour utiliser le moteur de stockage InnoDB.
- Les références de clé étrangère avec des mises à jour de table prédéfinies ne sont pas prises en charge. Plus précisément, `ON DELETE` les `ON UPDATE` règles ne sont pas prises en charge par `CASCADESET NULL`, et `SET DEFAULT` les actions. Toute tentative de création ou de mise à jour d'une table contenant de telles références à une autre table entraînera l'échec de la table.

Limites de RDS pour PostgreSQL

- La base de données source doit être une instance RDS pour PostgreSQL exécutant les versions 15.7+, 16.3+ ou 17.1+. Les versions antérieures ne sont pas prises en charge.
- Vous ne pouvez pas créer d'intégration zéro ETL à partir d'une instance de réplique en lecture RDS pour PostgreSQL.
- Les tables non enregistrées et les vues matérialisées de PostgreSQL ne sont pas répliquées sur Amazon Redshift.
- La réplication de certains types de données PostgreSQL, tels que les types de données géométriques et les données supérieures à 64 Ko, n'est pas prise en charge en raison des limites d'Amazon Redshift. Pour plus d'informations sur les différences de type de données entre RDS pour PostgreSQL et Amazon Redshiftthe section called “”, consultez la section Différences entre les types de données.
- Vous ne pouvez pas effectuer de mise à niveau de version majeure sur l'instance source RDS pour PostgreSQL si celle-ci possède une intégration Zero-ETL active. Pour mettre à niveau l'instance source, vous devez d'abord supprimer toutes les intégrations Zero-ETL existantes. Une fois la mise à niveau de la version majeure terminée, vous pouvez recréer les intégrations Zero-ETL.
- Si vous effectuez des transactions de partitionnement déclaratif sur l'instance de base de données source, toutes les tables concernées entrent dans un état d'échec et ne sont plus accessibles.

Limitations RDS for Oracle

- La base de données source doit être une instance RDS pour Oracle exécutant la version 19c Enterprise Edition ou Standard Edition 2, mise à jour de juillet 2019 ou ultérieure. Les versions antérieures ne sont pas prises en charge.
- Vous ne pouvez pas créer d'intégration zéro ETL à partir d'une instance de réplique de lecture RDS pour Oracle.
- Vous ne pouvez pas renommer une base de données mutualisée lorsqu'elle comporte une intégration zéro ETL.
- Une base de données mutualisée ne peut comporter qu'une seule intégration zéro ETL.
- Lorsque vous créez une intégration zéro ETL à partir de RDS pour Oracle, vous ne pouvez pas modifier cette intégration. Pour modifier une intégration, vous devez la supprimer, puis en créer une nouvelle.

- RDS pour Oracle et Amazon Redshift présentent certaines différences de type de données. Pour plus d'informations, consultez the section called “RDS for Oracle” la section Différences entre les types de données.

Limitations propres à Amazon Redshift

Pour obtenir la liste des limitations d'Amazon Redshift liées aux intégrations sans ETL, consultez la section Considérations relatives à l'utilisation d'intégrations sans ETL avec Amazon Redshift dans le guide de gestion Amazon Redshift.

Amazon SageMaker AI limites du lakehouse

Voici une limite pour les intégrations Amazon SageMaker AI Lakehouse Zero-ETL.

- Les noms de catalogue sont limités à 19 caractères.

Quotas

Votre compte possède les quotas suivants liés aux intégrations RDS Zero-ETL. Chaque quota s'applique par région, sauf indication contraire.

Nom	Par défaut	Description
Intégrations	100	Nombre total d'intégrations au sein d'un Compte AWS.
Intégrations par cible	50	Le nombre d'intégrations envoyant des données à un seul entrepôt de données ou à un lakehouse cible.
Intégrations par instance source	5	Nombre d'intégrations envoyant des données à partir d'un de base de données source unique.

En outre, l'entrepôt cible impose certaines limites au nombre de tables autorisées dans chaque instance de base de données ou nœud de cluster. Pour plus d'informations sur les quotas et les limites d'Amazon Redshift, consultez la section Quotas et limites dans Amazon Redshift dans le guide de gestion Amazon Redshift.

Régions prises en charge

Les intégrations RDS Zero-ETL sont disponibles dans un sous-ensemble de. Régions AWS Pour obtenir une liste des régions prises en charge, consultez the section called “Intégrations zéro ETL ”.

Démarrez avec les intégrations Amazon RDS

Avant de créer une intégration zéro ETL, configurez votre et votre entrepôt de données avec les paramètres et autorisations requis. Au cours de la configuration, vous allez suivre les étapes suivantes :

1. Création d'un groupe personnalisé de paramètres de base de données.
2. Créez un de base de données source.
3. Créez un entrepôt de données cible pour Amazon Redshift ou créez un lakehouse Amazon SageMaker AI cible.

Une fois ces tâches terminées, passez à the section called “Création d'intégrations sans ETL avec Amazon Redshift” outhe section called “Création d'intégrations sans ETL avec un lakehouse Amazon SageMaker”.

> **Tip**
>
> Vous pouvez demander à RDS d'effectuer ces étapes de configuration pour vous pendant que vous créez l'intégration, plutôt que de les exécuter manuellement. Pour commencer immédiatement la création d'une intégration, consultez the section called “Création d'intégrations sans ETL avec Amazon Redshift”.

Pour l'étape 3, vous pouvez choisir de créer un entrepôt de données cible (étape 3a) ou un lakehouse cible (étape 3b) en fonction de vos besoins :

- Choisissez un entrepôt de données si vous avez besoin de fonctionnalités d'entreposage de données traditionnelles avec des analyses basées sur SQL.
- Choisissez un Amazon SageMaker AI Lakehouse si vous avez besoin de fonctionnalités d'apprentissage automatique et souhaitez utiliser les fonctionnalités de Lakehouse pour la science des données et les flux de travail de machine learning.

Étape 1 : Créer un groupe de paramètres de base de données personnalisé

Les intégrations Amazon RDS Zero-ETL nécessitent des valeurs spécifiques pour les paramètres de base de données qui contrôlent la réplication des données. Les paramètres spécifiques dépendent de votre moteur de base de données source. Pour configurer ces paramètres, vous devez d'abord créer un groupe de paramètres de base de données personnalisé, puis l'associer à la base de données source. Configurez les valeurs de paramètres suivantes en fonction de votre moteur de base de données source. Pour obtenir des instructions sur la création d'un groupe de paramètres, consultez the section called “Groupes de paramètres DB”. Nous vous recommandons de configurer toutes les valeurs de paramètres dans la même demande afin d'éviter les problèmes de dépendance.

RDS pour MySQL :

- `binlog_format=ROW`
- `binlog_row_image=full`

Assurez-vous également que le paramètre `binlog_row_value_options` n'est pas défini sur `PARTIAL_JSON`. Si la base de données source est un cluster de base de données multi-AZ, assurez-vous que le `binlog_transaction_compression` paramètre n'est pas défini sur. `ON`

Certains de ces paramètres (tels que`binlog_format`) sont dynamiques, ce qui signifie que vous pouvez appliquer des modifications au paramètre sans déclencher de redémarrage. Cela signifie que certaines sessions existantes peuvent continuer à utiliser l'ancienne valeur du paramètre. Pour éviter que cela ne pose de problèmes lors de la création d'une intégration zéro ETL, activez le schéma de performance. Le schéma de performance garantit l'exécution de pré-contrôles zéro ETL, ce qui permet de détecter les paramètres manquants au début du processus.

RDS pour PostgreSQL :

- `rds.logical_replication = 1`
- `rds.replica_identity_full = 1`
- `session_replication_role = origin`
- `wal_sender_timeout ≥ 20000 or = 0`
- `max_wal_senders ≥ 20`
- `max_replication_slots ≥ 20`

Pour plusieurs intégrations PostgreSQL, un emplacement de réplication logique sera utilisé par intégration. Vérifiez les `max_wal_senders` paramètres `max_replication_slots` et en fonction de votre utilisation.

Pour une synchronisation efficace des données dans les intégrations sans ETL, configurez-la `rds.replica_identity_full` dans votre instance de base de données source. Cela indique à la base de données de consigner les données de ligne complètes dans le journal d'écriture anticipée (WAL) pendant `UPDATE` les `DELETE` opérations, plutôt que simplement les informations clés primaires. Zero-ETL nécessite des données de ligne complètes, même lorsque toutes les tables répliquées doivent disposer de clés primaires. Pour déterminer quelles données sont visibles lors des requêtes, Amazon Redshift utilise une stratégie anti-jointure spécialisée pour comparer vos données à celles d'une table de suivi des suppressions interne. L'enregistrement d'images en ligne complète permet à Amazon Redshift de réaliser efficacement ces anti-jointures. Sans données de ligne complètes, Amazon Redshift devrait effectuer des recherches supplémentaires, ce qui pourrait ralentir les performances lors d'opérations à haut débit dans le moteur en colonnes utilisé par Amazon Redshift.

> **Important**
>
> La configuration de l'identité de réplique pour enregistrer les lignes complètes augmente le volume de votre WAL, ce qui peut entraîner une augmentation de l'amplification et de I/O l'utilisation de l'écriture, en particulier pour les tables larges ou les mises à jour fréquentes. Pour vous préparer à ces impacts, planifiez votre capacité de stockage et vos I/O besoins, surveillez la croissance de votre WAL et suivez le retard de réplication dans les charges de travail intensives en écriture.

RDS pour Oracle :

Aucune modification de paramètre n'est requise pour RDS pour Oracle.

Étape 2 : sélectionner ou créer un de base de données source

Après avoir créé un groupe de paramètres de de base de données personnalisé, choisissez ou créez un RDS. Ce de base de données sera la source de réplication des données vers l'entrepôt de données cible. Pour obtenir des instructions sur la création d'un cluster de base de données multi-AZ, consultezthe section called “Création d'un cluster de base de données multi-AZ”.

La base de données doit exécuter une version du moteur de base de données prise en charge. Pour une liste de versions prises en charge, consultez the section called “Intégrations zéro ETL ”.

Lorsque vous créez la base de données, sous Configuration supplémentaire, remplacez le groupe de paramètres du de base de données par défaut par le groupe de paramètres personnalisé que vous avez créé à l'étape précédente.

> **Note**
>
> Pour obtenir des instructions, consultez the section called “Redémarrage d'une instance DB” ou the section called “Redémarrage d'un cluster de base de données multi-AZ”.

Assurez-vous également que les sauvegardes automatisées sont activées sur la base de données. Pour de plus amples informations, veuillez consulter the section called “Activation des sauvegardes automatiques”.

Étape 3a : créer un entrepôt de données cible

Après avoir créé votre de base de données source, vous devez créer et configurer un entrepôt de données cible. L'entrepôt de données doit respecter les exigences suivantes :

- En utilisant un type de RA3 nœud avec au moins deux nœuds, ou Redshift Serverless.
- Chiffré (si vous utilisez un cluster provisionné). Pour plus d’informations, consultez Chiffrement de base de données Amazon Redshift.

Pour obtenir des instructions sur la création d'un entrepôt des données, consultez Création d'un cluster pour les clusters provisionnés ou Création d'un groupe de travail avec un espace de noms pour Redshift sans serveur.

Activer la sensibilité à la casse sur l'entrepôt des données

Pour que l'intégration réussisse, le paramètre de sensibilité à la casse (`enable_case_sensitive_identifier`) doit être activé pour l'entrepôt des données. Par défaut, la sensibilité à la casse est désactivée sur tous les clusters provisionnés et les groupes de travail Redshift sans serveur.

Pour activer la sensibilité à la casse, effectuez les étapes suivantes en fonction du type de votre entrepôt des données :

- Cluster provisionné : pour activer la sensibilité à la casse sur un cluster provisionné, créez un groupe de paramètres personnalisé en activant le paramètre `enable_case_sensitive_identifier`. Associez ensuite le groupe de paramètres au cluster. Pour obtenir des instructions, consultez Gestion des groupes de paramètres à l'aide de la console ou Configuration des valeurs des paramètres à l'aide de l' AWS CLI.

  > **Note**
  >
  > N'oubliez pas de redémarrer le cluster après lui avoir associé le groupe de paramètres personnalisé.

- Groupe de travail sans serveur : pour activer la sensibilité à la casse sur un groupe de travail Redshift sans serveur, vous devez utiliser l' AWS CLI. La console Amazon Redshift ne prend actuellement pas en charge la modification des valeurs des paramètres Redshift sans serveur. Envoyez la demande de mise à jour du groupe de travail suivante :

  ```
  aws redshift-serverless update-workgroup \
    --workgroup-name target-workgroup \
    --config-parameters
   parameterKey=enable_case_sensitive_identifier,parameterValue=true
  ```

  Vous n'avez pas besoin de redémarrer un groupe de travail après avoir modifié ses valeurs de paramètres.

Configuration de l'autorisation pour l'entrepôt des données

Après avoir créé un entrepôt de données, vous devez configurer le de la base de données RDS source en tant que source d'intégration autorisée. Pour obtenir des instructions, consultez Configuration de l'autorisation pour votre entrepôt des données Amazon Redshift.

Configurez une intégration à l'aide du AWS SDKs

Plutôt que de configurer chaque ressource manuellement, vous pouvez exécuter le script Python suivant pour configurer automatiquement les ressources requises pour vous. L'exemple de code utilise le AWS SDK pour Python (Boto3)pour créer une instance de base de données source RDS pour MySQL et un entrepôt de données cible, chacun avec les valeurs de paramètres requises. Il attend ensuite que les bases de données soient disponibles avant de créer une intégration zéro ETL

entre elles. Vous pouvez commenter différentes fonctions en fonction des ressources que vous devez configurer.

Pour installer les dépendances requises, exécutez les commandes suivantes :

```
pip install boto3
pip install time
```

Dans le script, modifiez éventuellement les noms de la source, de la cible et des groupes de paramètres. La fonction finale crée une intégration nommée d'`my-integration`après la configuration des ressources.

Exemple de code Python

```
import boto3
import time

# Build the client using the default credential configuration.
# You can use the CLI and run 'aws configure' to set access key, secret
# key, and default Region.

rds = boto3.client('rds')
redshift = boto3.client('redshift')
sts = boto3.client('sts')

source_db_name = 'my-source-db' # A name for the source database
source_param_group_name = 'my-source-param-group' # A name for the source parameter
 group
target_cluster_name = 'my-target-cluster' # A name for the target cluster
target_param_group_name = 'my-target-param-group' # A name for the target parameter
 group

def create_source_db(*args):
    """Creates a source RDS for MySQL DB instance"""

    response = rds.create_db_parameter_group(
        DBParameterGroupName=source_param_group_name,
        DBParameterGroupFamily='mysql8.0',
        Description='RDS for MySQL zero-ETL integrations'
    )
    print('Created source parameter group: ' + response['DBParameterGroup']
['DBParameterGroupName'])
```

```
    response = rds.modify_db_parameter_group(
        DBParameterGroupName=source_param_group_name,
        Parameters=[
            {
                'ParameterName': 'binlog_format',
                'ParameterValue': 'ROW',
                'ApplyMethod': 'pending-reboot'
            },
            {
                'ParameterName': 'binlog_row_image',
                'ParameterValue': 'full',
                'ApplyMethod': 'pending-reboot'
            }
        ]
    )
    print('Modified source parameter group: ' + response['DBParameterGroupName'])

    response = rds.create_db_instance(
        DBInstanceIdentifier=source_db_name,
        DBParameterGroupName=source_param_group_name,
        Engine='mysql',
        EngineVersion='8.0.32',
        DBName='mydb',
        DBInstanceClass='db.m5.large',
        AllocatedStorage=15,
        MasterUsername='username',
        MasterUserPassword='Password01**'
    )
    print('Creating source database: ' + response['DBInstance']
['DBInstanceIdentifier'])
    source_arn = (response['DBInstance']['DBInstanceArn'])
    create_target_cluster(target_cluster_name, source_arn, target_param_group_name)
    return(response)

def create_target_cluster(target_cluster_name, source_arn, target_param_group_name):
    """Creates a target Redshift cluster"""

    response = redshift.create_cluster_parameter_group(
        ParameterGroupName=target_param_group_name,
        ParameterGroupFamily='redshift-1.0',
        Description='RDS for MySQL zero-ETL integrations'
    )
    print('Created target parameter group: ' + response['ClusterParameterGroup']
['ParameterGroupName'])
```

```
response = redshift.modify_cluster_parameter_group(
    ParameterGroupName=target_param_group_name,
    Parameters=[
        {
            'ParameterName': 'enable_case_sensitive_identifier',
            'ParameterValue': 'true'
        }
    ]
)
print('Modified target parameter group: ' + response['ParameterGroupName'])

response = redshift.create_cluster(
    ClusterIdentifier=target_cluster_name,
    NodeType='ra3.4xlarge',
    NumberOfNodes=2,
    Encrypted=True,
    MasterUsername='username',
    MasterUserPassword='Password01**',
    ClusterParameterGroupName=target_param_group_name
)
print('Creating target cluster: ' + response['Cluster']['ClusterIdentifier'])

# Retrieve the target cluster ARN
response = redshift.describe_clusters(
    ClusterIdentifier=target_cluster_name
)
target_arn = response['Clusters'][0]['ClusterNamespaceArn']

# Retrieve the current user's account ID
response = sts.get_caller_identity()
account_id = response['Account']

# Create a resource policy granting access to source database and account ID
response = redshift.put_resource_policy(
    ResourceArn=target_arn,
    Policy='''
    {
        \"Version\":\"2012-10-17\",
        \"Statement\":[
            {\"Effect\":\"Allow\",
            \"Principal\":{
                \"Service\":\"redshift.amazonaws.com\"
            },
```

```
                \"Action\":[\"redshift:AuthorizeInboundIntegration\"],
                \"Condition\":{
                    \"StringEquals\":{
                        \"aws:SourceArn\":\"%s\"}
                    }
                },
                {\"Effect\":\"Allow\",
                \"Principal\":{
                    \"AWS\":\"arn:aws:iam::%s:root\"},
                \"Action\":\"redshift:CreateInboundIntegration\"}
            ]
        }
        ''' % (source_arn, account_id)
    )
    return(response)

def wait_for_db_availability(*args):
    """Waits for both databases to be available"""

    print('Waiting for source and target to be available...')

    response = rds.describe_db_instances(
        DBInstanceIdentifier=source_db_name
    )
    source_status = response['DBInstances'][0]['DBInstanceStatus']
    source_arn = response['DBInstances'][0]['DBInstanceArn']

    response = redshift.describe_clusters(
        ClusterIdentifier=target_cluster_name
    )
    target_status = response['Clusters'][0]['ClusterStatus']
    target_arn = response['Clusters'][0]['ClusterNamespaceArn']

    # Every 60 seconds, check whether the databases are available
    if source_status != 'available' or target_status != 'available':
        time.sleep(60)
        response = wait_for_db_availability(
            source_db_name, target_cluster_name)
    else:
        print('Databases available. Ready to create zero-ETL integration.')
        create_integration(source_arn, target_arn)
        return

def create_integration(source_arn, target_arn):
```

```
    """Creates a zero-ETL integration using the source and target databases"""

    response = rds.create_integration(
        SourceArn=source_arn,
        TargetArn=target_arn,
        IntegrationName='my-integration'
    )
    print('Creating integration: ' + response['IntegrationName'])

def main():
    """main function"""
    create_source_db(source_db_name, source_param_group_name)
    wait_for_db_availability(source_db_name, target_cluster_name)

if __name__ == "__main__":
    main()
```

Étape 3b : Création d'un AWS Glue catalogue pour l'intégration Amazon SageMaker AI Zero-ETL

Lorsque vous créez une intégration zéro ETL avec un Amazon SageMaker AI Lakehouse, vous devez créer un catalogue AWS Glue géré dans. AWS Lake Formation Le catalogue cible doit être un catalogue géré par Amazon Redshift. Pour créer un catalogue géré Amazon Redshift, créez d'abord le rôle lié à un `AWSServiceRoleForRedshift` service. Dans la console Lake Formation, ajoutez le `AWSServiceRoleForRedshift` en tant qu'administrateur en lecture seule.

Pour plus d'informations sur les tâches précédentes, consultez les rubriques suivantes.

- Pour plus d'informations sur la création d'un catalogue géré Amazon Redshift, consultez la section Création d'un catalogue géré Amazon Redshift dans AWS Glue Data CatalogAWS Lake Formation le manuel du développeur.
- Pour plus d'informations sur le rôle lié à un service pour Amazon Redshift, consultez la section Utilisation des rôles liés à un service pour Amazon Redshift dans le guide de gestion Amazon Redshift.
- Pour plus d'informations sur les autorisations d'administrateur en lecture seule pour Lake Formation, consultez les références relatives aux personas de Lake Formation et aux autorisations IAM dans le Guide du développeur.AWS Lake Formation

Configurer les autorisations pour le AWS Glue catalogue cible

Avant de créer un catalogue cible pour une intégration Zero-ETL, vous devez créer le rôle de création de cible Lake Formation et le rôle de transfert de AWS Glue données. Utilisez le rôle de création de cibles Lake Formation pour créer le catalogue cible. Lors de la création du catalogue cible, entrez le rôle de transfert de données Glue dans le champ Rôle IAM de la section Accès depuis les moteurs.

Rôle de création de la cible de Lake Formation

Le rôle de création de la cible doit être un administrateur de Lake Formation et nécessite les autorisations suivantes.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "lakeformation:RegisterResource",
            "Resource": "*"
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": [
                "s3:PutEncryptionConfiguration",
                "iam:PassRole",
                "glue:CreateCatalog",
                "glue:GetCatalog",
                "s3:PutBucketTagging",
                "s3:PutLifecycleConfiguration",
                "s3:PutBucketPolicy",
                "s3:CreateBucket",
                "redshift-serverless:CreateNamespace",
                "s3:DeleteBucket",
                "s3:PutBucketVersioning",
                "redshift-serverless:CreateWorkgroup"
            ],
            "Resource": [
                "arn:aws:glue:*:111122223333:catalog",
```

```
                "arn:aws:glue:*:111122223333:catalog/*",
                "arn:aws:s3:::*",
                "arn:aws:redshift-serverless:*:111122223333:workgroup/*",
                "arn:aws:redshift-serverless:*:111122223333:namespace/*",
                "arn:aws:iam::111122223333:role/GlueDataCatalogDataTransferRole"
            ]
        }
    ]
}
```

Le rôle de création de la cible doit avoir la relation de confiance suivante.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "glue.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        },
        {
          "Effect": "Allow",
          "Principal": {
            "AWS": "arn:aws:iam::111122223333:user/Username"
          },
          "Action": "sts:AssumeRole"
        }
    ]
}
```

Rôle de transfert de données Glue

Le rôle de transfert de données Glue est requis pour les opérations du catalogue MySQL et doit disposer des autorisations suivantes.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "DataTransferRolePolicy",
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey",
                "kms:Decrypt",
                "glue:GetCatalog",
                "glue:GetDatabase"
            ],
            "Resource": [
                "*"
            ]
        }
    ]
}
```

Le rôle de transfert de données Glue doit avoir la relation de confiance suivante.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "glue.amazonaws.com",
                    "redshift.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Étapes suivantes

Avec un source et un entrepôt de données cible Amazon Redshift ou SageMaker Amazon AI Lakehouse, vous pouvez créer une intégration zéro ETL et répliquer les données. Pour obtenir des instructions, veuillez consulter the section called “Création d'intégrations sans ETL avec Amazon Redshift”.

Création d'intégrations zéro ETL d'Amazon RDS à Amazon Redshift

Lorsque vous créez une intégration Amazon RDS Zero-ETL, vous spécifiez le source et l'entrepôt de données Amazon Redshift cible. Vous pouvez également personnaliser les paramètres de chiffrement et ajouter des balises. Amazon RDS crée une intégration entre le de base de données source et sa cible. Une fois l'intégration active, toutes les données que vous insérez dans le de base de données source seront répliquées dans la cible Amazon Redshift configurée.

Prérequis

Avant de créer une intégration zéro ETL, vous devez créer un de base de données source et un entrepôt de données Amazon Redshift cible. Vous devez également autoriser la réplication dans l'entrepôt de données en ajoutant le de base de données en tant que source d'intégration autorisée.

Pour obtenir des instructions sur la réalisation de chacune de ces étapes, consultez the section called “Bien démarrer avec les intégrations zéro ETL”.

Autorisations requises

Certaines autorisations IAM sont requises pour créer une intégration zéro ETL. Vous avez au moins besoin des autorisations requises pour effectuer les actions suivantes :

- Créez des intégrations zéro ETL pour le cluster DB de la base de données RDS source.
- Afficher et supprimer toutes les intégrations zéro ETL.
- Créer des intégrations entrantes dans l'entrepôt de données cible.

Les exemples de politiques suivants illustrent les autorisations les moins privilégiées requises pour créer et gérer des intégrations. Il se peut que vous n'ayez pas besoin de ces autorisations exactes si votre utilisateur ou votre rôle dispose d'autorisations plus étendues, telles qu'une politique `AdministratorAccess` gérée.

> **Note**
>
> Les noms de ressources Redshift Amazon (ARNs) ont le format suivant. Notez l'utilisation d'une barre oblique (`/`) à la place du caractère deux-points (`:`) avant l'UUID de l'espace de noms sans serveur.
>
> - Cluster provisionné – `arn:aws:redshift:{region}:{account-id}:namespace:namespace-uuid`
> - Sans serveur – `arn:aws:redshift-serverless:{region}:{account-id}:namespace/namespace-uuid`

Exemple de politique pour Redshift Target

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "CreateIntegration",
      "Effect": "Allow",
      "Action": [
        "rds:CreateIntegration"
      ],
      "Resource": [
        "arn:aws:rds:us-east-1:123456789012:db:source-db",
        "arn:aws:rds:us-east-1:123456789012:integration:*"
      ]
    },
    {
      "Sid": "DescribeIntegrationDetails",
      "Effect": "Allow",
      "Action": [
        "rds:DescribeIntegrations"
      ],
      "Resource": [
      "arn:aws:rds:us-east-1:123456789012:integration:*"
  ]
    },
```

```
    {
      "Sid": "ChangeIntegrationDetails",
      "Effect": "Allow",
      "Action": [
        "rds:DeleteIntegration",
        "rds:ModifyIntegration"
      ],
      "Resource": [
        "arn:aws:rds:us-east-1:123456789012:integration:*"
      ]
    },
    {
      "Sid": "AllowRedShiftIntegration",
      "Effect": "Allow",
      "Action": [
        "redshift:CreateInboundIntegration"
      ],
      "Resource": [
        "arn:aws:redshift:us-east-1:123456789012:namespace:namespace-uuid"
      ]
    }
  ]
}
```

Choix d'un entrepôt de données cible dans un autre compte

Si vous prévoyez de spécifier un entrepôt de données Amazon Redshift cible situé dans un autre Compte AWS, vous devez créer un rôle permettant aux utilisateurs du compte courant d'accéder aux ressources du compte cible. Pour plus d'informations, consultez la section Fournir un accès à un utilisateur IAM dans un autre utilisateur Compte AWS dont vous êtes le propriétaire.

Le rôle doit disposer des autorisations suivantes, qui permettent à l'utilisateur de consulter les clusters provisionnés Amazon Redshift et les espaces de noms Redshift sans serveur disponibles dans le compte cible.

Autorisations requises et politique d'approbation

JSON

```
{
```

```
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "redshift:DescribeClusters",
            "redshift-serverless:ListNamespaces"
         ],
         "Resource":[
            "*"
         ]
      }
   ]
}
```

Le rôle doit respecter la politique d'approbation suivante, qui spécifie l'ID du compte cible.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Pour obtenir des instructions quant à la création du rôle, consultez Création d'un rôle à l'aide de politiques d'approbation personnalisées.

Création d'intégrations zéro ETL

Vous pouvez créer une intégration zéro ETL à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

> **Important**
>
> Les intégrations sans ETL ne prennent pas en charge les opérations d'actualisation ou de resynchronisation. Si vous rencontrez des problèmes avec une intégration après sa création, vous devez la supprimer et en créer une nouvelle.

Par défaut, RDS for MySQL purge immédiatement les fichiers journaux binaires. Étant donné que les intégrations sans ETL reposent sur des journaux binaires pour répliquer les données de la source vers la cible, la période de conservation de la base de données source doit être d'au moins une heure. Dès que vous créez une intégration, Amazon RDS vérifie la période de conservation du fichier journal binaire pour la base de données source sélectionnée. Si la valeur actuelle est de 0 heure, Amazon RDS la remplace automatiquement par 1 heure. Sinon, la valeur reste la même.

Console RDS

Pour créer une intégration zéro ETL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation de gauche, choisissez Intégrations Zero-ETL.
3. Choisissez Créer une intégration Zero-ETL.
4. Dans Identifiant d'intégration, saisissez un nom pour l'intégration. Ce nom peut comporter jusqu'à 63 caractères alphanumériques et peut inclure des traits d'union.

   > **Important**
   >
   > Les noms de catalogue sont limités à 19 caractères. Assurez-vous que votre identifiant d'intégration répond à cette exigence s'il doit être utilisé comme nom de catalogue.

5. Choisissez Suivant.
6. Pour Source, sélectionnez le d'où proviendront les données.

   > **Note**
   >
   > RDS vous avertit si les paramètres de base de données ne sont pas configurés correctement. Si vous recevez ce message, vous pouvez soit choisir Fix it for me, soit les configurer manuellement. Pour obtenir des instructions pour les corriger manuellement,

reportez-vous à the section called “Étape 1 : Créer un groupe de paramètres de base de données personnalisé”.
La modification des paramètres de base de données nécessite un redémarrage. Avant de créer l'intégration, le redémarrage doit être terminé et les nouvelles valeurs de paramètres doivent être correctement appliquées au de bases de données.

7. Une fois que le de base de données de votre base de données source est correctement configuré, choisissez Next.
8. Pour Cible, procédez comme suit :
   1. (Facultatif) Pour utiliser un autre Compte AWS compte pour la cible Amazon Redshift, choisissez Spécifier un autre compte. Saisissez ensuite l'ARN d'un rôle IAM doté d'autorisations pour afficher vos entrepôts des données. Pour obtenir des instructions sur la création du rôle IAM, consultez the section called “Choix d'un entrepôt de données cible dans un autre compte”.
   2. Vous pouvez choisir un cluster Amazon Redshift provisionné ou un espace de noms Redshift sans serveur comme cible.

> **Note**
>
> RDS vous avertit si la politique de ressources ou les paramètres de sensibilité à la casse pour l'entrepôt des données spécifié ne sont pas correctement configurés. Si vous recevez ce message, vous pouvez soit choisir Fix it for me, soit les configurer manuellement. Pour obtenir des instructions pour les corriger manuellement, consultez Activation de la sensibilité à la casse pour votre entrepôt des données et Configuration de l'autorisation pour votre entrepôt des données dans le Guide de gestion Amazon Redshift.
> La modification de la sensibilité à la casse pour un cluster Redshift provisionné nécessite un redémarrage. Avant de créer l'intégration, le redémarrage doit être terminé et la nouvelle valeur de paramètre doit être correctement appliquée au cluster.
> Si la source et la cible que vous avez sélectionnées se trouvent dans des Comptes AWS différents, Amazon RDS ne peut pas corriger ces paramètres pour vous. Vous devez accéder à l'autre compte et les corriger manuellement dans Amazon Redshift.

9. Une fois que votre entrepôt des données cible est correctement configuré, choisissez Suivant.

10. (Facultatif) Pour Balises, ajoutez une ou plusieurs balises à l'intégration. Pour de plus amples informations, veuillez consulter the section called “Marquage d' RDS de ses ressources”.
11. Pour Chiffrement, spécifiez la manière dont vous souhaitez que votre intégration soit chiffrée. Par défaut, RDS chiffre toutes les intégrations avec un. Clé détenue par AWS Pour choisir plutôt une clé gérée par le client, activez Personnaliser les paramètres de chiffrement et choisissez une clé KMS à utiliser pour le chiffrement. Pour de plus amples informations, veuillez consulter the section called “Chiffrement des ressources Amazon RDS”.

    Ajoutez éventuellement un contexte de chiffrement. Consultez Contexte de chiffrement dans le AWS Key Management Service guide du développeur pour en savoir plus.

    > **Note**
    >
    > Amazon RDS ajoute les paires de contextes de chiffrement suivantes en plus de celles que vous ajoutez :
    >
    > - `aws:redshift:integration:arn` - `IntegrationArn`
    > - `aws:servicename:id` - `Redshift`
    >
    > Cela réduit le nombre total de paires que vous pouvez ajouter de 8 à 6 et contribue à la limite de caractères globale de la contrainte de subvention. Pour plus d'informations, consultez la section Utilisation des contraintes de subvention dans le Guide du AWS Key Management Service développeur.

12. Choisissez Suivant.
13. Vérifiez vos paramètres d'intégration et choisissez Créer une intégration zéro ETL.

    Si la création échoue, consultez the section called “Je ne parviens pas à créer une intégration zéro ETL” pour obtenir les étapes de résolution des problèmes.

L'intégration a un statut de `Creating` lors de sa création et l'entrepôt de données Amazon Redshift cible a un statut de `Modifying`. Pendant ce temps, vous ne pouvez pas interroger l'entrepôt de données ni y apporter aucune modification de configuration.

Quand l'intégration est créée avec succès, le statut de l'intégration et celui de l'entrepôt de données Amazon Redshift cible passent tous deux à `Active`.

AWS CLI

Pour créer une intégration zéro ETL à l'aide de AWS CLI, utilisez la commande create-integration avec les options suivantes :

> **Note**
>
> N'oubliez pas que les noms de catalogue sont limités à 19 caractères. Choisissez le nom de votre intégration en conséquence s'il doit être utilisé comme nom de catalogue.

- `--integration-name` : spécifiez le nom de l'intégration.
- `--source-arn`— Spécifiez l'ARN du qui sera la source de l'intégration.
- `--target-arn` : spécifiez l'ARN de l'entrepôt des données Amazon Redshift qui sera la cible de l'intégration.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-integration \
    --integration-name my-integration \
    --source-arn arn:aws:rds:{region}:{account-id}:my-db \
    --target-arn arn:aws:redshift:{region}:{account-id}:namespace:namespace-uuid
```

Dans Windows :

```
aws rds create-integration ^
    --integration-name my-integration ^
    --source-arn arn:aws:rds:{region}:{account-id}:my-db ^
    --target-arn arn:aws:redshift:{region}:{account-id}:namespace:namespace-uuid
```

API RDS

Pour créer une intégration zéro ETL à l'aide de l'API Amazon RDS, utilisez l'opération `CreateIntegration` avec les paramètres suivants :

> **Note**
>
> Les noms de catalogue sont limités à 19 caractères. Assurez-vous que votre IntegrationName paramètre répond à cette exigence s'il doit être utilisé comme nom de catalogue.

- `IntegrationName` : spécifiez le nom de l'intégration.
- `SourceArn`— Spécifiez l'ARN du qui sera la source de l'intégration.
- `TargetArn` : spécifiez l'ARN de l'entrepôt des données Amazon Redshift qui sera la cible de l'intégration.

Chiffrer les intégrations à l'aide d'une clé gérée par le client

Si vous spécifiez une clé KMS personnalisée plutôt qu'une clé Clé détenue par AWS lors de la création d'une intégration, la politique de clé doit fournir au service Amazon Redshift l'accès principal à l'`CreateGrant`action. En outre, il doit autoriser l'utilisateur actuel à effectuer les `CreateGrant` actions `DescribeKey` et.

L'exemple de politique suivant montre comment fournir les autorisations requises dans la politique clé. Il inclut des clés contextuelles pour réduire davantage la portée des autorisations.

Exemple de politique clé

JSON

```
{
    "Version": "2012-10-17",
    "Id": "Key policy",
    "Statement": [
        {
            "Sid": "Enables IAM user permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
```

```
            "Sid": "Allows the Redshift service principal to add a grant to a KMS
key",
            "Effect": "Allow",
            "Principal": {
                "Service": "redshift.amazonaws.com"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:{context-key}": "{context-value}"
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "GenerateDataKey",
                        "CreateGrant"
                    ]
                }
            }
        },
        {
            "Sid": "Allows the current user or role to add a grant to a KMS key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/{role-name}"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:{context-key}": "{context-value}",
                    "kms:ViaService": "rds.us-east-1.amazonaws.com"
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "GenerateDataKey",
                        "CreateGrant"
                    ]
                }
            }
        },
        {
```

```
            "Sid": "Allows the current uer or role to retrieve information about
 a KMS key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/{role-name}"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        }
    ]
}
```

Pour plus d'informations, consultez Création d'une stratégie de clé dans le Guide du développeur AWS Key Management Service .

Étapes suivantes

Une fois que vous avez réussi à créer une intégration zéro ETL, vous devez créer une base de données de destination au sein de votre cluster ou groupe de travail Amazon Redshift cible. Vous pouvez ensuite commencer à ajouter des données au de la base de données RDS source et à les interroger dans Amazon Redshift. Pour obtenir des instructions, consultez Création de bases de données de destination dans Amazon Redshift.

Création d'intégrations Amazon RDS Zero-ETL avec un lakehouse Amazon SageMaker

Lorsque vous créez une intégration Amazon RDS Zero-ETL avec un Amazon SageMaker Lakehouse, vous spécifiez le source et le catalogue géré cible. AWS Glue Vous pouvez également personnaliser les paramètres de chiffrement et ajouter des balises. Amazon RDS crée une intégration entre le de base de données source et sa cible. Une fois l'intégration active, toutes les données que vous insérez dans le de base de données source seront répliquées dans la cible configurée.

Prérequis

Avant de créer une intégration zéro ETL avec un Amazon SageMaker Lakehouse, vous devez créer un de base de données source et un catalogue géré cible AWS Glue . Vous devez également autoriser la réplication dans le catalogue en ajoutant le de base de données en tant que source d'intégration autorisée.

Pour obtenir des instructions sur la réalisation de chacune de ces étapes, consultez the section called “Bien démarrer avec les intégrations zéro ETL”.

Autorisations requises

Certaines autorisations IAM sont requises pour créer une intégration zéro ETL avec un Amazon SageMaker lakehouse. Vous avez au moins besoin des autorisations requises pour effectuer les actions suivantes :

- Créez des intégrations zéro ETL pour le cluster DB de la base de données RDS source.
- Afficher et supprimer toutes les intégrations zéro ETL.
- Créez des intégrations entrantes dans le catalogue AWS Glue géré cible.
- Accédez aux compartiments Amazon S3 utilisés par le catalogue AWS Glue géré.
- Utilisez AWS KMS des clés pour le chiffrement si le chiffrement personnalisé est configuré.
- Enregistrez des ressources auprès de Lake Formation.
- Appliquez une politique de ressources au catalogue AWS Glue géré pour autoriser les intégrations entrantes.

L'exemple de politique suivant illustre les autorisations de moindre privilège requises pour créer et gérer des intégrations avec un Amazon SageMaker lakehouse. Il se peut que vous n'ayez pas besoin de ces autorisations exactes si votre utilisateur ou votre rôle dispose d'autorisations plus étendues, telles qu'une politique `AdministratorAccess` gérée.

En outre, vous devez configurer une politique de ressources sur le catalogue AWS Glue géré cible afin d'autoriser les intégrations entrantes. Utilisez la AWS CLI commande suivante pour appliquer la politique de ressources.

Exemple de AWS CLI commande pour autoriser les intégrations entrantes sur le catalogue cible

```
aws glue put-resource-policy \
      --policy-in-json  '{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Principal": {
            "Service": "glue.amazonaws.com"
        },
```

```
        "Action": [
            "glue:AuthorizeInboundIntegration"
        ],
        "Resource": ["arn:aws:glue:region:account_id:catalog/catalog_name"],
        "Condition": {
            "StringEquals": {
                "aws:SourceArn": "arn:aws:rds:region:account_id:db:source_name"
            }
        }
    },
    {
        "Effect": "Allow",
        "Principal": {
            "AWS": "account_id"
        },
        "Action": ["glue:CreateInboundIntegration"],
        "Resource": ["arn:aws:glue:region:account_id:catalog/catalog_name"]
    }
    ]
}' \
      --region region
```

> **Note**
>
> Les noms de ressources Amazon du catalogue Glue (ARNs) ont le format suivant :
>
> - Catalogue Glue — `arn:aws:glue:{region}:{account-id}:catalog/catalog-name`

Choix d'un catalogue AWS Glue géré cible dans un autre compte

Si vous envisagez de spécifier un catalogue AWS Glue géré cible qui se trouve dans un autre Compte AWS, vous devez créer un rôle permettant aux utilisateurs du compte courant d'accéder aux ressources du compte cible. Pour plus d'informations, consultez la section Fournir un accès à un utilisateur IAM dans un autre utilisateur Compte AWS dont vous êtes le propriétaire.

Le rôle doit disposer des autorisations suivantes, qui permettent à l'utilisateur de consulter les AWS Glue catalogues disponibles dans le compte cible.

Autorisations requises et politique d'approbation

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "glue:GetCatalog"
         ],
         "Resource":[
            "*"
         ]
      }
   ]
}
```

Le rôle doit respecter la politique d'approbation suivante, qui spécifie l'ID du compte cible.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Principal":{
            "AWS": "arn:aws:iam::111122223333:root"
         },
         "Action":"sts:AssumeRole"
      }
   ]
}
```

Pour obtenir des instructions quant à la création du rôle, consultez Création d'un rôle à l'aide de politiques d'approbation personnalisées.

Création d'intégrations sans ETL avec un lakehouse Amazon SageMaker

Vous pouvez créer une intégration zéro ETL avec un Amazon SageMaker lakehouse à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

> **Important**
>
> Les intégrations sans ETL avec un Amazon SageMaker Lakehouse ne prennent pas en charge les opérations d'actualisation ou de resynchronisation. Si vous rencontrez des problèmes avec une intégration après sa création, vous devez la supprimer et en créer une nouvelle.

Par défaut, RDS for MySQL purge immédiatement les fichiers journaux binaires. Étant donné que les intégrations sans ETL reposent sur des journaux binaires pour répliquer les données de la source vers la cible, la période de conservation de la base de données source doit être d'au moins une heure. Dès que vous créez une intégration, Amazon RDS vérifie la période de conservation du fichier journal binaire pour la base de données source sélectionnée. Si la valeur actuelle est de 0 heure, Amazon RDS la remplace automatiquement par 1 heure. Sinon, la valeur reste la même.

Console RDS

Pour créer une intégration zéro ETL avec un lakehouse Amazon SageMaker

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation de gauche, choisissez Intégrations Zero-ETL.
3. Choisissez Créer une intégration Zero-ETL.
4. Dans Identifiant d'intégration, saisissez un nom pour l'intégration. Ce nom peut comporter jusqu'à 63 caractères alphanumériques et peut inclure des traits d'union.
5. Choisissez Suivant.
6. Pour Source, sélectionnez le d'où proviendront les données.

> **Note**
>
> RDS vous avertit si les paramètres de base de données ne sont pas configurés correctement. Si vous recevez ce message, vous pouvez soit choisir Fix it for me, soit les configurer manuellement. Pour obtenir des instructions pour les corriger manuellement,

reportez-vous à the section called “Étape 1 : Créer un groupe de paramètres de base de données personnalisé”.
La modification des paramètres de base de données nécessite un redémarrage. Avant de créer l'intégration, le redémarrage doit être terminé et les nouvelles valeurs de paramètres doivent être correctement appliquées au de bases de données.

7. Une fois que le de base de données de votre base de données source est correctement configuré, choisissez Next.
8. Pour Cible, procédez comme suit :

   1. (Facultatif) Pour utiliser un autre Compte AWS compte pour la cible Amazon SageMaker Lakehouse, choisissez Spécifier un autre compte. Entrez ensuite l'ARN d'un rôle IAM autorisé à afficher vos AWS Glue catalogues. Pour obtenir des instructions sur la création du rôle IAM, consultez the section called “Choix d'un catalogue AWS Glue géré cible dans un autre compte”.
   2. Pour le AWS Glue catalogue, sélectionnez la cible pour les données répliquées à partir du de base de données source. Vous pouvez choisir un catalogue AWS Glue géré existant comme cible.
   3. Le rôle IAM cible doit décrire les autorisations sur le catalogue cible et doit disposer des autorisations suivantes :

      JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": "glue:GetCatalog",
            "Resource": [
                "arn:aws:glue:us-east-1:111122223333:catalog/*",
                "arn:aws:glue:us-east-1:111122223333:catalog"
            ]
        }
    ]
}
```

      Le rôle IAM cible doit avoir la relation de confiance suivante :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "glue.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

4. Vous devez accorder au rôle IAM cible des autorisations de description pour le catalogue AWS Glue géré cible avec le rôle d'administrateur de Lake Formation créé dansÉtape 3b : Création d'un AWS Glue catalogue pour l'intégration Amazon SageMaker AI Zero-ETL.

> **Note**
>
> RDS vous avertit si la politique de ressources ou les paramètres de configuration du catalogue AWS Glue géré spécifié ne sont pas correctement configurés. Si vous recevez ce message, vous pouvez soit choisir Fix it for me, soit les configurer manuellement.
> Si la source et la cible que vous avez sélectionnées se trouvent dans des Comptes AWS différents, Amazon RDS ne peut pas corriger ces paramètres pour vous. Vous devez accéder à l'autre compte et les corriger manuellementSageMaker Unified Studio.

9. Une fois que votre catalogue AWS Glue géré cible est correctement configuré, choisissez Next.
10. (Facultatif) Pour Balises, ajoutez une ou plusieurs balises à l'intégration. Pour de plus amples informations, veuillez consulter the section called “Marquage d' RDS de ses ressources”.
11. Pour Chiffrement, spécifiez la manière dont vous souhaitez que votre intégration soit chiffrée. Par défaut, RDS chiffre toutes les intégrations avec un. Clé détenue par AWS Pour choisir plutôt une clé gérée par le client, activez Personnaliser les paramètres de chiffrement et choisissez une clé KMS à utiliser pour le chiffrement. Pour de plus amples informations, veuillez consulter the section called “Chiffrement des ressources Amazon RDS”.

Ajoutez éventuellement un contexte de chiffrement. Consultez Contexte de chiffrement dans le AWS Key Management Service guide du développeur pour en savoir plus.

> **Note**
>
> Amazon RDS ajoute les paires de contextes de chiffrement suivantes en plus de celles que vous ajoutez :
>
> - `aws:glue:integration:arn` - `IntegrationArn`
> - `aws:servicename:id` - `glue`
>
> Cela réduit le nombre total de paires que vous pouvez ajouter de 8 à 6 et contribue à la limite de caractères globale de la contrainte de subvention. Pour plus d'informations, consultez la section Utilisation des contraintes de subvention dans le Guide du AWS Key Management Service développeur.

12. Choisissez Suivant.
13. Vérifiez vos paramètres d'intégration et choisissez Créer une intégration zéro ETL.

    Si la création échoue, consultez the section called “Résolution des problèmes liés aux intégrations zéro ETL” pour obtenir les étapes de résolution des problèmes.

L'intégration a un statut `Creating` pendant sa création, et le Amazon SageMaker lakehouse cible a un statut de. `Modifying` Pendant ce temps, vous ne pouvez pas interroger le catalogue ni y apporter de modifications de configuration.

Lorsque l'intégration est créée avec succès, le statut de l'intégration et celui du Amazon SageMaker lakehouse cible passent tous deux à. `Active`

AWS CLI

Pour préparer un catalogue AWS Glue géré cible pour une intégration zéro ETL à l'aide de AWS CLI, vous devez d'abord utiliser la create-integration-resource-propertycommande avec les options suivantes :

- `--resource-arn`— Spécifiez l'ARN du catalogue AWS Glue géré qui sera la cible de l'intégration.

- `--target-processing-properties`— Spécifiez l'ARN du rôle IAM pour accéder au catalogue AWS Glue géré cible

```
aws glue create-integration-resource-property --region us-east-1
 --resource-arn arn:aws:glue:region:account_id:catalog/catalog_name \
 --target-processing-properties '{"RoleArn" : "arn:aws:iam::account_id:role/
TargetIamRole"}'
```

Pour créer une intégration zéro ETL avec un Amazon SageMaker lakehouse à l'aide de AWS CLI, utilisez la commande create-integration avec les options suivantes :

- `--integration-name` : spécifiez le nom de l'intégration.
- `--source-arn`— Spécifiez l'ARN du qui sera la source de l'intégration.
- `--target-arn`— Spécifiez l'ARN du catalogue AWS Glue géré qui sera la cible de l'intégration.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-integration \
    --integration-name my-sagemaker-integration \
    --source-arn arn:aws:rds:{region}:{account-id}:my-db \
    --target-arn arn:aws:glue:{region}:{account-id}:catalog/catalog-name
```

Dans Windows :

```
aws rds create-integration ^
    --integration-name my-sagemaker-integration ^
    --source-arn arn:aws:rds:{region}:{account-id}:my-db ^
    --target-arn arn:aws:glue:{region}:{account-id}:catalog/catalog-name
```

API RDS

Pour créer une intégration zéro ETL à Amazon SageMaker l'aide de l'API Amazon RDS, utilisez l'`CreateIntegration`opération avec les paramètres suivants :

> **Note**
>
> Les noms de catalogue sont limités à 19 caractères. Assurez-vous que votre IntegrationName paramètre répond à cette exigence s'il doit être utilisé comme nom de catalogue.

- `IntegrationName` : spécifiez le nom de l'intégration.
- `SourceArn`— Spécifiez l'ARN du qui sera la source de l'intégration.
- `TargetArn`— Spécifiez l'ARN du catalogue AWS Glue géré qui sera la cible de l'intégration.

Chiffrer les intégrations à l'aide d'une clé gérée par le client

Si vous spécifiez une clé KMS personnalisée plutôt qu'une clé Clé détenue par AWS lorsque vous créez une intégrationAmazon SageMaker, la politique de clé doit fournir au SageMaker Unified Studio service principal un accès à l'`CreateGrant`action. En outre, il doit autoriser l'utilisateur actuel à effectuer les `CreateGrant` actions `DescribeKey` et.

L'exemple de politique suivant montre comment fournir les autorisations requises dans la politique clé. Il inclut des clés contextuelles pour réduire davantage la portée des autorisations.

Exemple de politique clé

JSON

```
{
    "Version": "2012-10-17",
    "Id": "Key policy",
    "Statement": [
        {
            "Sid": "EnablesIAMUserPermissions",
            "Effect": "Allow",
            "Principal": {
            "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "AllowsGlueServicePrincipalAddGrant AWS KMS Key",
```

```
            "Effect": "Allow",
            "Principal": {
                "Service": "glue.amazonaws.com"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:{context-key}":"{context-value}"
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "GenerateDataKey",
                        "CreateGrant"
                    ]
                }
            }
        },
        {
            "Sid": "AllowsCurrentUserRoleAddGrantKMSKey",
            "Effect": "Allow",
            "Principal": {
            "AWS": "arn:aws:iam::111122223333:role/{role-name}"
            },
            "Action": "kms:CreateGrant",
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:{context-key}":"{context-value}",
                    "kms:ViaService": "rds.us-east-1.amazonaws.com"
                },
                "ForAllValues:StringEquals": {
                    "kms:GrantOperations": [
                        "Decrypt",
                        "GenerateDataKey",
                        "CreateGrant"
                    ]
                }
            }
        },
        {
            "Sid": "AllowsCurrentUserRoleRetrieveKMSKeyInformation",
            "Effect": "Allow",
```

```
            "Principal": {
            "AWS": "arn:aws:iam::111122223333:role/{role-name}"
            },
            "Action": "kms:DescribeKey",
            "Resource": "*"
        }
    ]
}
```

Pour plus d'informations, consultez Création d'une stratégie de clé dans le Guide du développeur AWS Key Management Service .

Étapes suivantes

Une fois que vous avez créé avec succès une intégration Zero-ETL avecAmazon SageMaker, vous pouvez commencer à ajouter des données au de la base de données RDS source et à les interroger dans votre Lakehouse. Amazon SageMaker Les données seront automatiquement répliquées et mises à disposition pour les charges de travail d'analyse et d'apprentissage automatique.

Filtrage des données pour les intégrations Amazon RDS

Les intégrations Amazon RDS Zero-ETL prennent en charge le filtrage des données, ce qui vous permet de contrôler quelles données sont répliquées depuis votre source vers votre entrepôt de données cible. Au lieu de répliquer l'intégralité de la base de données, vous pouvez appliquer un ou plusieurs filtres pour inclure ou exclure des tables spécifiques de manière sélective. Cela vous permet d'optimiser les performances de stockage et de requête en garantissant que seules les données pertinentes sont transférées. Actuellement, le filtrage est limité aux niveaux de la base de données et de la table. Le filtrage au niveau des colonnes et des lignes n'est pas pris en charge.

Le filtrage des données peut être utile lorsque vous souhaitez :

- Joignez certaines tables provenant d'au moins deux de bases de données sources différents, et vous n'avez pas besoin de données complètes provenant de l'un ou l'autre des de bases de données.
- Réduisez les coûts en effectuant des analyses en utilisant uniquement un sous-ensemble de tables plutôt qu'un parc complet de bases de données.
- Filtrez les informations sensibles, telles que les numéros de téléphone, les adresses ou les informations de carte de crédit, de certains tableaux.

Vous pouvez ajouter des filtres de données à une intégration sans ETL à l' AWS Management Console aide de l' AWS Command Line Interface API,AWS CLI the () ou Amazon RDS.

Si l'intégration a un cluster provisionné comme cible, le cluster doit être sur le correctif 180 ou supérieur pour utiliser le filtrage des données.

Rubriques

- Format d'un filtre de données
- Logique de filtrage
- Priorité du filtre
- Exemples d'de RDS pour MySQL
- Exemples d'RDS pour PostgreSQL
- Exemples de RDS pour Oracle
- Ajouter des filtres de données à une intégration
- Supprimer les filtres de données d'une intégration

Format d'un filtre de données

Vous pouvez définir plusieurs filtres pour une seule intégration. Chaque filtre inclut ou exclut les tables de base de données existantes et futures qui correspondent à l'un des modèles de l'expression du filtre. Les intégrations Amazon RDS Zero-ETL utilisent la syntaxe du filtre Maxwell pour le filtrage des données.

Chaque filtre comporte les éléments suivants :

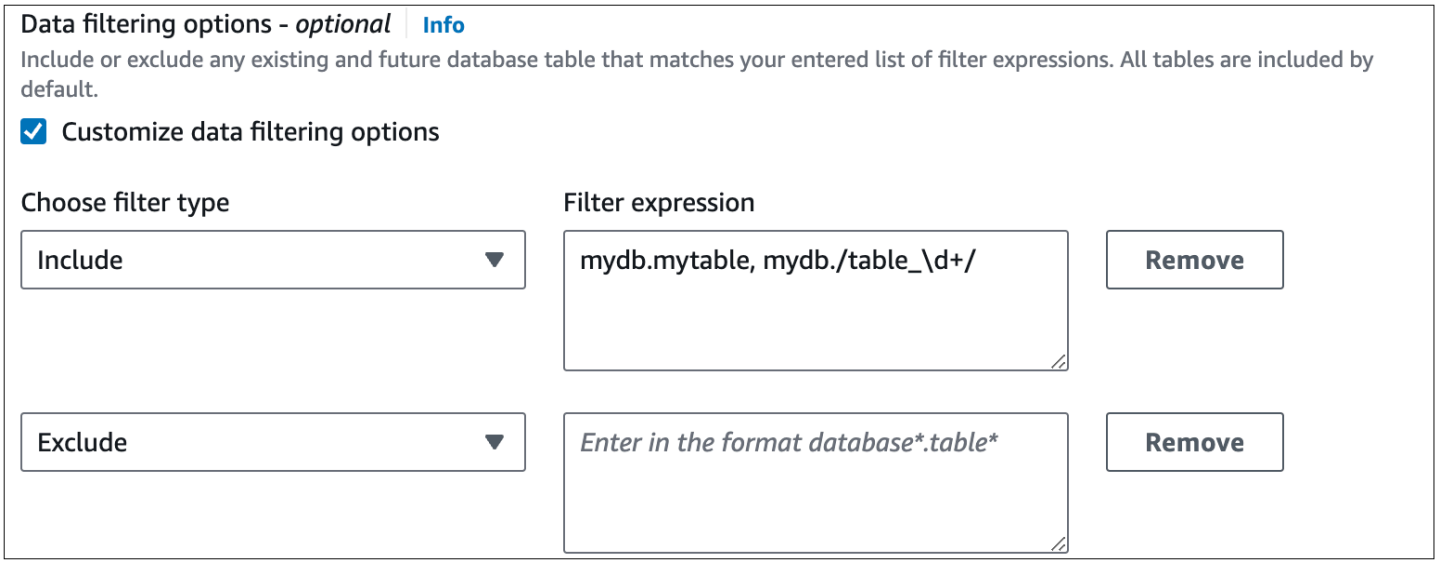
Element	Description
Type de filtre	Un type de `Include` filtre inclut toutes les tables qui correspondent à l'un des modèles de l'expression du filtre. Un type de `Exclude` filtre exclut toutes les tables correspondant à l'un des modèles.
Expression de filtrage	Liste de modèles séparés par des virgules. Les expressions doivent utiliser la syntaxe du filtre Maxwell.

Element	Description
Modèle	Un modèle de filtre au format *`database`*`.`*`table`* pour RDS pour MySQL ou pour . Vous pouvez spécifier des noms littéraux ou définir des expressions régulière s. **Note** Pour RDS for MySQL, les expressio ns régulières sont prises en charge à la fois dans le nom de la base de données et dans le nom de la table. Pour , les expressions régulières ne sont prises en charge que dans le schéma et le nom de la table, et non dans le nom de la base de données. Vous ne pouvez pas inclure de filtres ou de listes de refus au niveau des colonnes. Une seule intégration peut avoir un maximum de 99 modèles au total. Dans la console, vous pouvez saisir des modèles dans une seule expression de filtre ou les répartir entre plusieurs expressions. Un seul modèle ne peut pas dépasser 256 caractères.

Important

Si vous sélectionnez un cluster de bases de données source PostgreSQL, vous devez spécifier au moins un modèle de filtre de données. Le modèle doit au minimum inclure une seule base de données (*`database-name`*`.*.*`) pour la réplication vers l'entrepôt de données cible.

L'image suivante montre la structure des filtres de données RDS pour MySQL dans la console :



> **Important**
>
> N'incluez pas d'informations d'identification personnelle, confidentielles ou sensibles dans vos modèles de filtrage.

Filtres de données dans le AWS CLI

Lorsque vous utilisez le AWS CLI pour ajouter un filtre de données, la syntaxe est légèrement différente de celle de la console. Vous devez attribuer un type de filtre (`Include`ou`Exclude`) à chaque modèle individuellement, afin de ne pas pouvoir regrouper plusieurs modèles sous un même type de filtre.

Par exemple, dans la console, vous pouvez regrouper les modèles suivants, séparés par des virgules, sous une seule `Include` instruction :

RDS pour MySQL

```
mydb.mytable, mydb./table_\d+/
```

PostgreSQL

```
mydb.myschema.mytable, mydb.myschema./table_\d+/
```

Toutefois, lorsque vous utilisez le AWS CLI, le même filtre de données doit être au format suivant :

RDS pour MySQL

```
'include: mydb.mytable, include: mydb./table_\d+/'
```

PostgreSQL

```
'include: mydb.myschema.mytable, include: mydb.myschema./table_\d+/'
```

Logique de filtrage

Si vous ne spécifiez aucun filtre de données dans votre intégration, Amazon RDS suppose un filtre par défaut de`include:*.*`, qui réplique toutes les tables dans l'entrepôt de données cible. Toutefois, si vous ajoutez au moins un filtre, la logique par défaut passe à zéro`exclude:*.*`, ce qui exclut toutes les tables par défaut. Cela vous permet de définir explicitement les bases de données et les tables à inclure dans la réplication.

Par exemple, si vous définissez le filtre suivant :

```
'include: db.table1, include: db.table2'
```

Amazon RDS évalue le filtre comme suit :

```
'exclude:*.*, include: db.table1, include: db.table2'
```

Par conséquent, Amazon RDS réplique uniquement `table1` et `table2` depuis la base de données nommée db vers l'entrepôt de données cible.

Priorité du filtre

Amazon RDS évalue les filtres de données dans l'ordre que vous spécifiez. Dans le AWS Management Console, il traite les expressions de filtre de gauche à droite et de haut en bas. Un deuxième filtre ou un modèle individuel qui suit le premier peut le remplacer.

Par exemple, si le premier filtre l'est`Include books.stephenking`, il inclut uniquement la `stephenking` table de la `books` base de données. Toutefois, si vous ajoutez un deuxième filtre`Exclude books.*`, il remplace le premier filtre. Cela empêche les tables de l'`books`index d'être répliquées vers l'entrepôt de données cible.

Lorsque vous spécifiez au moins un filtre, la logique commence par l'hypothèse `exclude:*.*` par défaut, ce qui exclut automatiquement toutes les tables de la réplication. Il est recommandé de définir

des filtres du plus large au plus spécifique. Commencez par une ou plusieurs `Include` instructions pour spécifier les données à répliquer, puis ajoutez des `Exclude` filtres pour supprimer certaines tables de manière sélective.

Le même principe s'applique aux filtres que vous définissez à l'aide du AWS CLI. Amazon RDS évalue ces modèles de filtre dans l'ordre dans lequel vous les spécifiez, de sorte qu'un modèle peut remplacer celui que vous avez spécifié avant lui.

Exemples d'de RDS pour MySQL

Les exemples suivants montrent comment fonctionne le filtrage des données pour d'intégrations Zero-ETL avec RDS pour MySQL :

- Incluez toutes les bases de données et toutes les tables :

```
'include: *.*'
```

- Incluez toutes les tables de la `books` base de données :

```
'include: books.*'
```

- Excluez toutes les tables nommées `mystery` :

```
'include: *.*, exclude: *.mystery'
```

- Incluez deux tables spécifiques dans la `books` base de données :

```
'include: books.stephen_king, include: books.carolyn_keene'
```

- Incluez toutes les tables de la `books` base de données, à l'exception de celles contenant la sous-chaîne `mystery` :

```
'include: books.*, exclude: books./.*mystery.*/'
```

- Incluez toutes les tables de la `books` base de données, à l'exception de celles commençant par `mystery` :

```
'include: books.*, exclude: books./mystery.*/'
```

- Incluez toutes les tables de la `books` base de données, à l'exception de celles se terminant par `mystery` :

```
'include: books.*, exclude: books./.*mystery/'
```

- Incluez toutes les tables de la `books` base de données qui commencent par`table_`, à l'exception de celle nommée`table_stephen_king`. Par exemple, `table_movies` ou `table_books` serait répliqué, mais non`table_stephen_king`.

```
'include: books./table_.*/, exclude: books.table_stephen_king'
```

Exemples d'RDS pour PostgreSQL

Les exemples suivants montrent comment fonctionne le filtrage des données pour les intégrations RDS pour PostgreSQL Zero-ETL :

- Incluez toutes les tables de la `books` base de données :

```
'include: books.*.*'
```

- Excluez toutes les tables nommées `mystery` dans la `books` base de données :

```
'include: books.*.*, exclude: books.*.mystery'
```

- Incluez une table dans la `books` base de données dans le `mystery` schéma, et une table dans la `employee` base de données dans le `finance` schéma :

```
'include: books.mystery.stephen_king, include: employee.finance.benefits'
```

- Incluez toutes les tables de la `books` base de données et `science_fiction` du schéma, à l'exception de celles contenant la sous-chaîne `king` :

```
'include: books.science_fiction.*, exclude: books.*./.*king.*/
```

- Incluez toutes les tables de la `books` base de données, à l'exception de celles dont le nom de schéma commence par `sci` :

```
'include: books.*.*, exclude: books./sci.*/.*'
```

- Incluez toutes les tables de la `books` base de données, à l'exception de celles du `mystery` schéma se terminant par `king` :

```
'include: books.*.*, exclude: books.mystery./.*king/'
```

- Incluez toutes les tables de la `books` base de données qui commencent par`table_`, à l'exception de celle nomméé`table_stephen_king`. Par exemple, `table_movies` dans le `fiction` schéma et `table_books` dans le `mystery` schéma sont répliqués, mais pas `table_stephen_king` dans l'un ou l'autre des schémas :

```
'include: books.*./table_.*/, exclude: books.*.table_stephen_king'
```

Exemples de RDS pour Oracle

Les exemples suivants montrent comment fonctionne le filtrage des données pour les intégrations RDS pour Oracle Zero-ETL :

- Incluez toutes les tables de la base de données des livres :

```
'include: books.*.*'
```

- Excluez toutes les tables nommées mystère dans la base de données des livres :

```
'include: books.*.*, exclude: books.*.mystery'
```

- Incluez une table dans la base de données des livres dans le schéma mystère, et une table dans la base de données des employés dans le schéma financier :

```
'include: books.mystery.stephen_king, include: employee.finance.benefits'
```

- Incluez toutes les tables du schéma mystère de la base de données des livres :

```
'include: books.mystery.*'
```

> **Note**
>
> L'utilisation d'expressions régulières dans la valeur du filtre pour le nom de base de données, le schéma ou le nom de table n'est pas prise en charge dans les intégrations RDS pour Oracle.

Ajouter des filtres de données à une intégration

Vous pouvez configurer le filtrage des données à l'aide de l' AWS Management Console API AWS CLI, de, ou de l'API Amazon RDS.

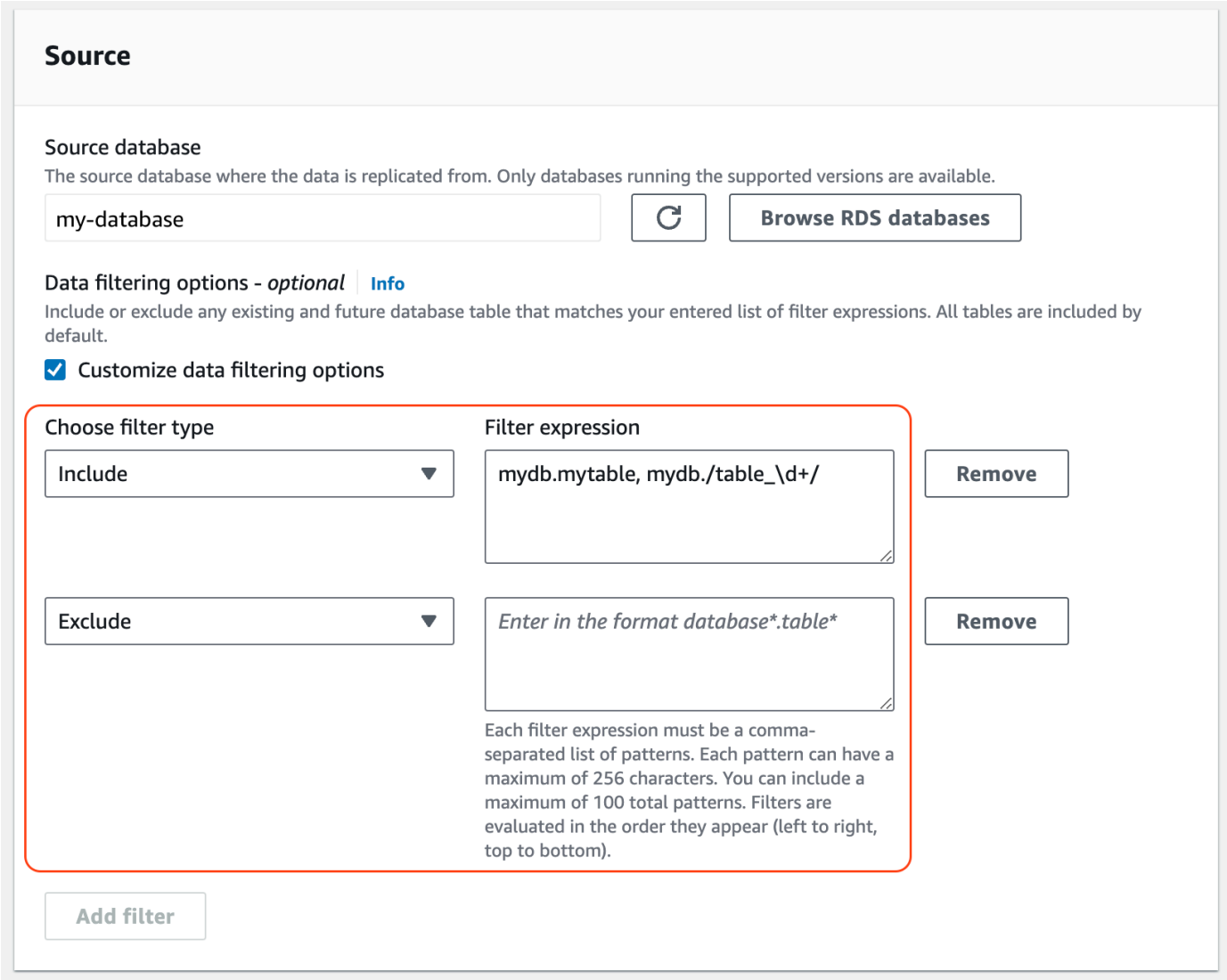
> **Important**
>
> Si vous ajoutez un filtre après avoir créé une intégration, Amazon RDS le traite comme s'il avait toujours existé. Il supprime toutes les données de l'entrepôt de données cible qui ne correspondent pas aux nouveaux critères de filtrage et resynchronise toutes les tables concernées.

Console RDS

Pour ajouter des filtres de données à une intégration sans ETL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Zero-ETL integrations. Sélectionnez l'intégration à laquelle vous souhaitez ajouter des filtres de données, puis choisissez Modifier.
3. Sous Source, ajoutez une ou plusieurs `Exclude` déclarations `Include` et.

   L'image suivante montre un exemple de filtres de données pour une intégration MySQL :



4. Lorsque vous êtes satisfait des modifications, choisissez Continuer et Enregistrer les modifications.

AWS CLI

Pour ajouter des filtres de données à une intégration zéro ETL à l'aide de AWS CLI, appelez la commande modify-integration. Outre l'identifiant d'intégration, spécifiez le `--data-filter` paramètre à l'aide d'une liste séparée par des virgules de filtres `Include` et de `Exclude` Maxwell.

Example

L'exemple suivant ajoute des modèles de filtre à`my-integration`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-integration \
    --integration-identifier my-integration \
    --data-filter 'include: foodb.*, exclude: foodb.tbl, exclude: foodb./table_\d+/'
```

Dans Windows :

```
aws rds modify-integration ^
    --integration-identifier my-integration ^
    --data-filter 'include: foodb.*, exclude: foodb.tbl, exclude: foodb./table_\d+/'
```

API RDS

Pour modifier une intégration zéro ETL à l'aide de l'API RDS, appelez l'ModifyIntegrationopération. Spécifiez l'identifiant d'intégration et fournissez une liste de modèles de filtre séparés par des virgules.

Supprimer les filtres de données d'une intégration

Lorsque vous supprimez un filtre de données d'une intégration, Amazon RDS réévalue les filtres restants comme si le filtre supprimé n'avait jamais existé. Il réplique ensuite toutes les données précédemment exclues qui répondent désormais aux critères dans l'entrepôt de données cible. Cela déclenche une resynchronisation de toutes les tables concernées.

Ajouter des données à une base de données RDS source () et les interroger

Pour terminer la création d'une intégration zéro ETL qui réplique les données d'Amazon RDS Amazon Aurora vers Redshift, vous devez créer une base de données dans la destination cible.

Pour les connexions avec Amazon Redshift, connectez-vous à votre cluster ou groupe de travail Amazon Redshift et créez une base de données avec une référence à votre identifiant d'intégration. Vous pouvez ensuite ajouter des données à votre source et les voir répliquées dans Amazon Redshift ou. Amazon SageMaker

Rubriques

- Création d'une base de données cible
- Ajout de données au de base de données source

- Interrogation de vos données Amazon RDS dans Amazon Redshift
- Différences de type de données entre les bases de données RDS et Amazon Redshift
-

Création d'une base de données cible

Avant de pouvoir commencer à répliquer des données dans Amazon Redshift, après avoir créé une intégration, vous devez créer une base de données dans votre entrepôt de données cible. Cette base de données doit inclure une référence à l'identifiant d'intégration. Vous pouvez utiliser la console Amazon Redshift ou l'éditeur de requête v2 pour créer la base de données.

Pour obtenir des instructions sur la création d'une base de données de destination, consultez Création d'une base de données de destination dans Amazon Redshift.

Ajout de données au de base de données source

Après avoir configuré votre intégration, vous pouvez remplir le source avec les données que vous souhaitez répliquer dans votre entrepôt de données.

> **Note**
>
> Il existe des différences entre les types de données dans Amazon RDS, et dans l'entrepôt d'analyse cible. Pour un tableau des mappages de types de données, consultez the section called “Différences de type de données”.

Connectez-vous d'abord au de base de données source à l'aide du client MySQL de votre choix. Pour obtenir des instructions, veuillez consulter the section called “Connexion à une instance de base de données exécutant MySQL”.

Ensuite, créez une table et insérez une ligne d'exemples de données.

> **Important**
>
> Assurez-vous que la table possède une clé primaire. Sinon, elle ne peut pas être répliquée vers l'entrepôt de données cible.

RDS pour MySQL

L'exemple suivant utilise l'utilitaire MySQL Workbench.

```
CREATE DATABASE my_db;

USE my_db;

CREATE TABLE books_table (ID int NOT NULL, Title VARCHAR(50) NOT NULL, Author
 VARCHAR(50) NOT NULL,
Copyright INT NOT NULL, Genre VARCHAR(50) NOT NULL, PRIMARY KEY (ID));

INSERT INTO books_table VALUES (1, 'The Shining', 'Stephen King', 1977, 'Supernatural
 fiction');
```

RDS pour PostgreSQL

L'exemple suivant utilise le terminal psql interactif PostgreSQL. Lorsque vous vous connectez à la base de données, incluez le nom de la base de données que vous souhaitez répliquer.

```
psql -h mydatabase.123456789012.us-east-2.rds.amazonaws.com -p 5432 -U username -
d named_db;

named_db=> CREATE TABLE books_table (ID int NOT NULL, Title VARCHAR(50) NOT NULL,
 Author VARCHAR(50) NOT NULL,
Copyright INT NOT NULL, Genre VARCHAR(50) NOT NULL, PRIMARY KEY (ID));

named_db=> INSERT INTO books_table VALUES (1, 'The Shining', 'Stephen King', 1977,
 'Supernatural fiction');
```

RDS pour Oracle

L'exemple suivant utilise SQL*Plus pour se connecter à votre base de données RDS pour Oracle.

```
sqlplus 'user_name@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dns_name)(PORT=port))
(CONNECT_DATA=(SID=database_name)))'

SQL> CREATE TABLE books_table (ID int NOT NULL, Title VARCHAR(50) NOT NULL, Author
 VARCHAR(50) NOT NULL,
Copyright INT NOT NULL, Genre VARCHAR(50) NOT NULL, PRIMARY KEY (ID));
```

```
SQL> INSERT INTO books_table VALUES (1, 'The Shining', 'Stephen King', 1977,
 'Supernatural fiction');
```

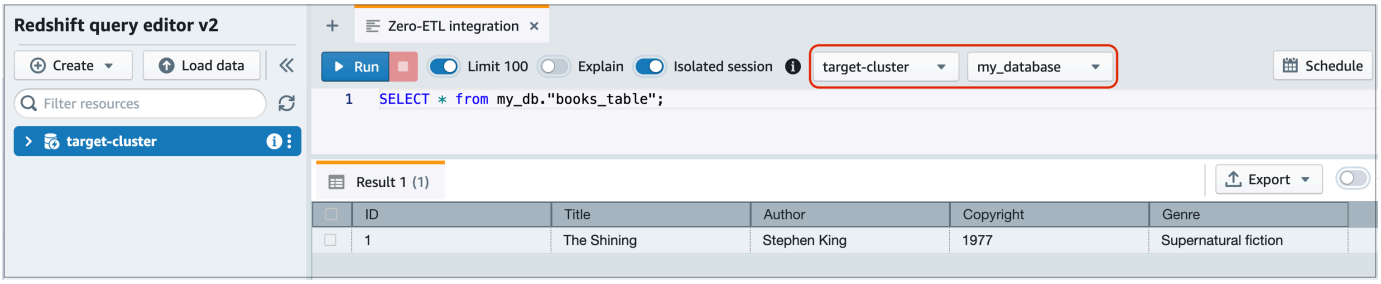
Interrogation de vos données Amazon RDS dans Amazon Redshift

Une fois que vous avez ajouté des données au de la base de données RDS, celles-ci sont répliquées dans la base de données de destination et sont prêtes à être interrogées.

Pour interroger les données répliquées

1. Accédez à la console Amazon Redshift et choisissez Éditeur de requête v2 dans le panneau de navigation de gauche.
2. Connectez-vous à votre cluster ou groupe de travail et choisissez votre base de données de destination (que vous avez créée à partir de l'intégration) dans le menu déroulant (destination_database dans cet exemple). Pour obtenir des instructions sur la création d'une base de données de destination, consultez Création d'une base de données de destination dans Amazon Redshift.
3. Utilisez une instruction SELECT pour interroger vos données. Dans cet exemple, vous pouvez exécuter la commande suivante pour sélectionner toutes les données de la table que vous avez créée dans le de la base de données RDS source :

   ```
   SELECT * from my_db."books_table";
   ```

   - *my_db* est le nom du schéma de base de données RDS.
   - *books_table* est le nom de la table RDS.

Vous pouvez également interroger les données à l'aide d'un client de ligne de commande. Par exemple :

```
destination_database=# select * from my_db."books_table";
```

```
 ID |       Title |        Author |   Copyright |                 Genre |  txn_seq |
 txn_id
----+-------------+---------------+-------------+-----------------------+----------
+--------+
  1 | The Shining |  Stephen King |        1977 |   Supernatural fiction |        2 |
 12192
```

> **Note**
>
> Pour appliquer la sensibilité à la casse, utilisez des guillemets doubles (« ») pour les noms de schéma, de table et de colonne. Pour plus d'informations, consultez enable_case_sensitive_identifier.

Différences de type de données entre les bases de données RDS et Amazon Redshift

Les tableaux suivants présentent les mappages des tables RDS pour MySQL, RDS pour PostgreSQL et RDS pour Oracle. Les tables montrent les mappages des types de données Aurora MySQL et correspondants. Amazon RDS ne prend actuellement en charge que ces types de données pour les intégrations sans ETL.

Si une table du de base de données de votre base de données source inclut un type de données non pris en charge, la table est désynchronisée et n'est pas consommable par la cible de destination. Le streaming de la source vers la cible se poursuit, mais le tableau contenant le type de données non pris en charge n'est pas disponible. Pour corriger le tableau et le rendre disponible dans la destination cible, vous devez annuler manuellement la modification la plus importante, puis actualiser l'intégration en exécutant`ALTER DATABASE...INTEGRATION REFRESH`.

> **Note**
>
> Vous ne pouvez pas actualiser les intégrations Zero-ETL avec un Amazon SageMaker Lakehouse. Supprimez plutôt l'intégration et réessayez de la créer.

Rubriques

- RDS pour MySQL MySQL

-
- RDS for Oracle

RDS pour MySQL MySQL

Type de données RDS for MySQL	Type de données cible	Description	Limites
INT	INTEGER	Entier signé sur quatre octets	Aucun
SMALLINT	SMALLINT	Entier signé sur deux octets	Aucun
TINYINT	SMALLINT	Entier signé sur deux octets	Aucun
MEDIUMINT	INTEGER	Entier signé sur quatre octets	Aucun
BIGINT	BIGINT	Entier signé sur huit octets	Aucun
INT UNSIGNED	BIGINT	Entier signé sur huit octets	Aucun
TINYINT UNSIGNED	SMALLINT	Entier signé sur deux octets	Aucun
MEDIUMINT UNSIGNED	INTEGER	Entier signé sur quatre octets	Aucun
BIGINT UNSIGNED	DECIMAL(20,0)	Valeur numérique exacte avec précision sélectionnable	Aucun

Type de données RDS for MySQL	Type de données cible	Description	Limites
DÉCIMAL (p, s) = NUMÉRIQUE (p, s)	DECIMAL(p,s)	Valeur numérique exacte avec précision sélectionnable	La précision supérieure à 38 et l'échelle supérieure à 37 ne sont pas prises en charge
DÉCIMAL (p, s) NON SIGNÉ = NUMÉRIQUE (p, s) NON SIGNÉ	DECIMAL(p,s)	Valeur numérique exacte avec précision sélectionnable	La précision supérieure à 38 et l'échelle supérieure à 37 ne sont pas prises en charge
FLOAT4/RÉEL	REAL	Nombre à virgule flottante simple précision	Aucun
FLOAT4/REAL NON SIGNÉ	REAL	Nombre à virgule flottante simple précision	Aucun
DOUBLE/REAL/FLOAT8	DOUBLE PRECISION	Nombre à virgule flottante de double précision	Aucun
DOUBLE/REAL/FLOAT8 NON SIGNÉ	DOUBLE PRECISION	Nombre à virgule flottante de double précision	Aucun
BIT (n)	VARBYTE(8)	Valeur binaire de longueur variable	Aucun
BINAIRE (n)	VARBYTE (n)	Valeur binaire de longueur variable	Aucun

Type de données RDS for MySQL	Type de données cible	Description	Limites
VARBINAIRE (n)	VARBYTE (n)	Valeur binaire de longueur variable	Aucun
CHAR(n)	VARCHAR(n)	Valeur de chaîne de longueur variable	Aucun
VARCHAR(n)	VARCHAR(n)	Valeur de chaîne de longueur variable	Aucun
TEXT	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
TINYTEXT	VARCHAR(255)	Valeur de chaîne de longueur variable jusqu'à 255 caractères	Aucun
MEDIUMTEXT	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
LONGTEXT	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun

Type de données RDS for MySQL	Type de données cible	Description	Limites
ENUM	VARCHAR(1020)	Valeur de chaîne de longueur variable jusqu'à 1 020 caractères	Aucun
SET	VARCHAR(1020)	Valeur de chaîne de longueur variable jusqu'à 1 020 caractères	Aucun
DATE	DATE	Date calendaire (année, mois, jour)	Aucun
DATETIME	TIMESTAMP	Date et heure (sans fuseau horaire)	Aucun
HORODATAGE (p)	TIMESTAMP	Date et heure (sans fuseau horaire)	Aucun
TIME	VARCHAR(18)	Valeur de chaîne de longueur variable jusqu'à 18 caractères	Aucun
YEAR	VARCHAR(4)	Valeur de chaîne de longueur variable jusqu'à 4 caractères	Aucun

Type de données RDS for MySQL	Type de données cible	Description	Limites
JSON	SUPER	Données ou documents semi-structurés sous forme de valeurs	Aucun

Intégrations sans ETL pour RDS pour PostgreSQL ne prend pas en charge les types de données personnalisés ni les types de données créés par des extensions.

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
array	SUPER	Données ou documents semi-structurés sous forme de valeurs	Aucun
bigint	BIGINT	Entier signé sur huit octets	Aucun
grande série	BIGINT	Entier signé sur huit octets	Aucun
bit variable (n)	VARBYTE (n)	Valeur binaire de longueur variable jusqu'à 16 777 216 octets	Aucun
bit (n)	VARBYTE (n)	Valeur binaire de longueur variable jusqu'à 16 777 216 octets	Aucun
un peu, un peu variable	VARBYTE (16777216)	Valeur binaire de longueur variable	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
		jusqu'à 16 777 216 octets	
boolean	BOOLEAN	Booléen logique (vrai/faux)	Aucun
par le thé	VARBYTE (16777216)	Valeur binaire de longueur variable jusqu'à 16 777 216 octets	Aucun
chaise (n)	CHAR(n)	Valeur de chaîne de caractères de longueur fixe jusqu'à 65 535 octets	Aucun
caractère variable (n)	VARCHAR(65535)	Valeur de chaîne de caractères de longueur variable jusqu'à 65 535 caractères	Aucun
cid	BIGINT	Entier signé sur huit octets	Aucun
cidr	GUERRE (19)	Valeur de chaîne de longueur variable jusqu'à 19 caractères	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
date	DATE	Date calendaire (année, mois, jour)	Les valeurs supérieures à 294 276 après Jésus-Christ ne sont pas prises en charge
double precision	DOUBLE PRECISION	Chiffres à virgule flottante à double précision	Les valeurs inférieures à la normale ne sont pas entièrement prises en charge
vecteur gts	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
inet	GUERRE (19)	Valeur de chaîne de longueur variable jusqu'à 19 caractères	Aucun
entier	INTEGER	Entier signé sur quatre octets	Aucun
vecteur int2	SUPER	Données ou documents semi-structurés sous forme de valeurs.	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
interval	INTERVAL	Durée du temps	Seuls les types INTERVAL qui spécifient un qualificatif annuel par mois ou un qualifica tif journalier par seconde sont pris en charge.
json	SUPER	Données ou documents semi-structurés sous forme de valeurs	Aucun
jsonb	SUPER	Données ou documents semi-structurés sous forme de valeurs	Aucun
chemin json	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractère s	Aucun
macaddr	GUERRE (17)	Valeur de chaîne de longueur variable jusqu'à 17 caractères	Aucun
macaddr8	GUERRE (23)	Valeur de chaîne de longueur variable jusqu'à 23 caractères	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
money	DÉCIMAL (20,3)	Montant en devise	Aucun
name	GUERRE (64)	Valeur de chaîne de longueur variable jusqu'à 64 caractères	Aucun
numeric(p,s)	DECIMAL(p,s)	Valeur de précision fixe définie par l'utilisateur	• `NaN`valeurs non prises en charge • La précision et l'échelle doivent être définies de manière explicite et ne pas dépasser 38 (précision) et 37 (échelle) • Échelle négative non prise en charge
oid	BIGINT	Entier signé sur huit octets	Aucun
vecteur oid	SUPER	Données ou documents semi-structurés sous forme de valeurs.	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
pg_brin_bloom_summary	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
pg_dependencies	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
pg_lsn	GUERRE (17)	Valeur de chaîne de longueur variable jusqu'à 17 caractères	Aucun
pg_mcv_list	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
pg_ndistinct	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
pg_node_tree	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
pg_snapshot	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
real	REAL	Nombre à virgule flottante simple précision	Les valeurs inférieures à la normale ne sont pas entièrement prises en charge
récurseur	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
smallint	SMALLINT	Entier signé sur deux octets	Aucun
petite série	SMALLINT	Entier signé sur deux octets	Aucun
serial	INTEGER	Entier signé sur quatre octets	Aucun
text	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
tid	GUERRE (23)	Valeur de chaîne de longueur variable jusqu'à 23 caractères	Aucun
heure [(p)] sans fuseau horaire	GUERRE (19)	Valeur de chaîne de longueur variable jusqu'à 19 caractères	`Infinity`et `-Infinity` valeurs non prises en charge
heure [(p)] avec fuseau horaire	GUERRE (22)	Valeur de chaîne de longueur variable jusqu'à 22 caractères	`Infinity`et `-Infinity` valeurs non prises en charge
horodatage [(p)] sans fuseau horaire	TIMESTAMP	Date et heure (sans fuseau horaire)	• `Infinity`et `-Infinity` valeurs non prises en charge • Valeurs supérieures à celles qui `9999-12-31` ne sont pas prises en charge • Les valeurs de la Colombie-Britannique ne sont pas prises en charge

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
horodatage [(p)] avec fuseau horaire	TIMESTAMPTZ	Date et heure (avec fuseau horaire)	• `Infinity`et `-Infinity` valeurs non prises en charge • Valeurs supérieures à celles qui `9999-12-31` ne sont pas prises en charge • Les valeurs de la Colombie-Britannique ne sont pas prises en charge
tsquery	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
vecteur ts	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun

Type de données RDS pour PostgreSQL PostgreSQL	Type de données Amazon Redshift	Description	Limites
txid_instantané	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun
uuid	GUERRE (36)	Chaîne de 36 caractères de longueur variable	Aucun
xid	BIGINT	Entier signé sur huit octets	Aucun
xid8	DÉCIMAL (20, 0)	Décimal à précision fixe	Aucun
xml	VARCHAR(65535)	Valeur de chaîne de longueur variable jusqu'à 65 535 caractères	Aucun

RDS for Oracle

Types de données non pris en charge

Les types de données RDS pour Oracle suivants ne sont pas pris en charge par Amazon Redshift :

- `ANYDATA`
- `BFILE`
- `REF`
- `ROWID`
- `UROWID`
- `VARRAY`

- SDO_GEOMETRY
- Types de données définis par l'utilisateur

Différences entre les types de données

Le tableau suivant montre les différences de type de données qui affectent une intégration zéro ETL lorsque RDS pour Oracle est la source et Amazon Redshift est la cible.

Type de données RDS pour Oracle	Type de données Amazon Redshift
BINARY_FLOAT	FLOAT4
BINARY_DOUBLE	FLOAT8
BINAIRE	VARCHAR (Length)
FLOAT (P)	Si la précision est =< 24, alors FLOAT4. Si la précision est > 24, alors FLOAT8.
NUMBER (P,S)	Si l'échelle est => 0 et =< 37, alors NUMERIC (p, s). Si l'échelle est => 38 et =< 127, alors VARCHAR (Longueur). Si l'échelle est égale à 0 : • Et la précision est donc inférieure ou égale à 2 INT2. • Et la précision est donc supérieure à 2 et inférieure ou égale à 4 INT2. • Et la précision est donc supérieure à 4 et inférieure ou égale à 9 INT4. • Et la précision est supérieure à 9, puis NUMERIC (p, s). • Et la précision est supérieure ou égale à l'échelle, puis NUMERIC (p, s).

Type de données RDS pour Oracle	Type de données Amazon Redshift
	Si l'échelle est inférieure à 0, alors INT8.
DATE	Si l'échelle est => 0 et =< 6, selon le type de colonne cible Redshift, alors l'une des valeurs suivantes : • TIMESTAMP (s) • TIMESTAMPTZ (s) : si l'horodatage source contient un décalage de zone (comme dans SQL Server ou Oracle), il est converti en UTC lors de l'insertion/de la mise à jour. S'il ne contient pas de décalage, l'heure est déjà prise en compte en UTC. Si l'échelle est => 7 et =< 9, alors VARCHAR (37).
INTERVAL_YEAR TO MONTH	Si la longueur est comprise entre 1 et 65 535, alors VARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors VARCHAR (65535).
INTERVAL_DAY TO SECOND	Si la longueur est comprise entre 1 et 65 535, alors VARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors VARCHAR (65535).

Type de données RDS pour Oracle	Type de données Amazon Redshift
TIMESTAMP	Si l'échelle est => 0 et =< 6, selon le type de colonne cible Redshift, alors l'une des valeurs suivantes : • TIMESTAMP (s) • TIMESTAMPTZ (s) : si l’horodatage source contient un décalage de zone (comme dans SQL Server ou Oracle), il est converti en UTC lors de l’insertion/de la mise à jour. S'il ne contient pas de décalage, l'heure est déjà prise en compte en UTC. Si l'échelle est => 7 et =< 9, alors VARCHAR (37).
TIMESTAMP WITH TIME ZONE	Si la longueur est comprise entre 1 et 65 535, alors VARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors VARCHAR (65535).
TIMESTAMP WITH LOCAL TIME ZONE	Si la longueur est comprise entre 1 et 65 535, alors VARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors VARCHAR (65535).
CHAR	Si la longueur est comprise entre 1 et 65 535, alors VARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors VARCHAR (65535).

Type de données RDS pour Oracle	Type de données Amazon Redshift
VARCHAR2	Lorsque la longueur est supérieure à 4 000 octets, VARCHAR (taille LOB maximale). La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieure à 64 Ko. Lorsque la longueur est inférieure ou égale à 4 000 octets, VARCHAR (longueur en octets).
NCHAR	Si la longueur est comprise entre 1 et 65 535, alors NVARCHAR (longueur en octets). Si la longueur est comprise entre 65 536 et 2 147 483 647, alors NVARCHAR (65535).
NVARCHAR2	Lorsque la longueur est supérieure à 4 000 octets, NVARCHAR (taille LOB maximale). La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieure à 64 Ko. Lorsque la longueur est inférieure ou égale à 4 000 octets, NVARCHAR (longueur en octets).
RAW	VARCHAR (Length)
REAL	FLOAT8
BLOB	VARCHAR (taille de LOB maximale *2) La taille de LOB maximale ne peut pas dépasser 31 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.

Type de données RDS pour Oracle	Type de données Amazon Redshift
CLOB	VARCHAR (taille de LOB maximale) La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.
NCLOB	NVARCHAR (taille de LOB maximale) La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.
LONG	VARCHAR (taille de LOB maximale) La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.
LONG RAW	VARCHAR (taille de LOB maximale *2) La taille de LOB maximale ne peut pas dépasser 31 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.
XMLTYPE	VARCHAR (taille de LOB maximale) La taille de LOB maximale ne peut pas dépasser 63 Ko. Amazon Redshift ne prend pas en charge une VARCHARs taille supérieur e à 64 Ko.

Les intégrations Zero-ETL tirent parti de ces similitudes pour rationaliser la réplication des données de RDS pour PostgreSQL Aurora PostgreSQL et en utilisant la base de données, le schéma et la structure de table partagés.

Tenez compte des points suivants lors de la gestion des intégrations RDS pour PostgreSQL :

- L'isolation est gérée au niveau de la base de données.
- La réplication s'effectue au niveau de la base de données.
- RDS pour PostgreSQL Les bases de données PostgreSQL sont mappées aux bases de données Amazon Redshift par leur nom, les données étant transmises à la base de données Redshift renommée correspondante si l'original est renommé.

Les sections suivantes décrivent les réponses du système Amazon Redshift pour les opérations DDL courantes.

Rubriques

- Opérations sur les bases
- Opérations sur le schéma
- Opérations de table

Opérations sur les bases

Le tableau suivant indique les réponses du système pour les opérations DDL de base de données.

Fonctionnement DDL	Réponse du système Redshift
`CREATE DATABASE`	Aucune opération
`DROP DATABASE`	Amazon Redshift supprime toutes les données de la base de données Redshift cible.
`RENAME DATABASE`	Amazon Redshift supprime toutes les données de la base de données cible d'origine et resynchronise les données de la nouvelle base de données cible. Si la nouvelle base de données n'existe pas, vous devez la créer

Fonctionnement DDL	Réponse du système Redshift
	manuellement. Pour obtenir des instructions, consultez Créer une base de données de destination dans Amazon Redshift.

Opérations sur le schéma

Le tableau suivant indique les réponses du système pour les opérations DDL du schéma.

Fonctionnement DDL	Réponse du système Redshift
`CREATE SCHEMA`	Aucune opération
`DROP SCHEMA`	Amazon Redshift supprime le schéma d'origine.
`RENAME SCHEMA`	Amazon Redshift supprime le schéma d'origine puis resynchronise les données dans le nouveau schéma.

Opérations de table

Le tableau suivant indique les réponses du système pour les opérations DDL dans les tables.

Fonctionnement DDL	Réponse du système Redshift
`CREATE TABLE`	Amazon Redshift crée le tableau. Certaines opérations entraînent l'échec de la création de tables, telles que la création d'une table sans clé primaire ou le partitionnement déclaratif. Pour plus d’informations, consultez the section called “Limites” et the section called “Résolution des problèmes liés aux intégrations zéro ETL”.
`DROP TABLE`	Amazon Redshift abandonne la tendance.

Fonctionnement DDL	Réponse du système Redshift
`TRUNCATE TABLE`	Amazon Redshift tronque le tableau.
`ALTER TABLE (RENAME... )`	Amazon Redshift renomme la table ou la colonne.
`ALTER TABLE (SET SCHEMA)`	Amazon Redshift supprime la table dans le schéma d'origine et resynchronise la table dans le nouveau schéma.
`ALTER TABLE (ADD PRIMARY KEY)`	Amazon Redshift ajoute une clé primaire et resynchronise le tableau.
`ALTER TABLE (ADD COLUMN)`	Amazon Redshift ajoute une colonne au tableau.
`ALTER TABLE (DROP COLUMN)`	Amazon Redshift supprime la colonne s'il ne s'agit pas d'une colonne de clé primaire. Dans le cas contraire, il resynchronise le tableau.
`ALTER TABLE (SET LOGGED/UNLOGGED )`	Si vous remplacez la table par log, Amazon Redshift la resynchronise. Si vous définissez le tableau sur Unlogged, Amazon Redshift le supprime.

Affichage et surveillance des intégrations Amazon RDS

Vous pouvez afficher les détails d'une intégration zéro ETL d'Amazon RDS pour voir ses informations de configuration et son statut actuel. Vous pouvez également surveiller le statut de votre intégration en interrogeant des vues système spécifiques dans Amazon Redshift. En outre, Amazon Redshift publie certaines métriques liées à l'intégration sur Amazon CloudWatch, que vous pouvez consulter dans la console Amazon Redshift.

Rubriques

- Affichage des intégrations
- Surveillance des intégrations à l'aide des tables système pour Amazon Redshift

- Surveillance des intégrations avec Amazon EventBridge pour Amazon Redshift

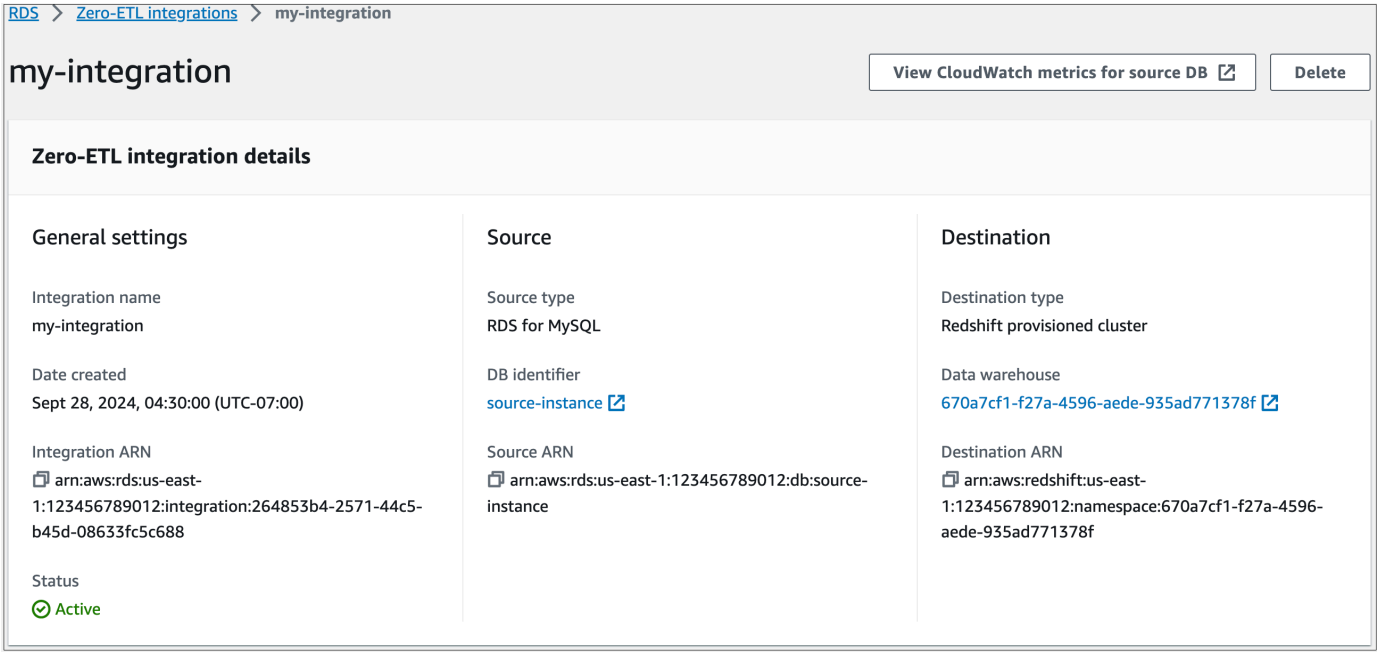
Affichage des intégrations

Vous pouvez consulter les intégrations Amazon RDS Zero-ETL à l'aide de l'API AWS Management Console, de ou de l' AWS CLI API RDS.

console

Pour afficher les détails d’une intégration zéro ETL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation de gauche, choisissez Intégrations Zero-ETL.
3. Sélectionnez une intégration pour afficher plus de détails à son sujet, tels que sa base de données source, son de bases de données et son entrepôt de données cible.



Une intégration peut avoir les statuts suivants :

- `Creating` : l'intégration est en cours de création.
- `Active` : l'intégration envoie des données transactionnelles à l'entrepôt des données cible.

- `Syncing` : l'intégration a rencontré une erreur récupérable et réensemence les données. Les tables concernées ne sont pas disponibles pour les requêtes tant que leur resynchronisation n'est pas terminée.
- `Needs attention` : l'intégration a rencontré un événement ou une erreur nécessitant une intervention manuelle pour être résolu. Pour corriger le problème, suivez les instructions du message d'erreur dans la page des détails relatifs à l'intégration.
- `Failed` : l'intégration a rencontré un événement ou une erreur irrécupérable qui ne peut pas être corrigé. Vous devez supprimer et recréer l'intégration.
- `Deleting` : l'intégration est en cours de suppression.

AWS CLI

Pour afficher toutes les intégrations Zero-ETL du compte courant à l'aide de AWS CLI, utilisez la commande describe-integrations et spécifiez l'option. `--integration-identifier`

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-integrations \
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374
```

Dans Windows :

```
aws rds describe-integrations ^
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374
```

API RDS

Pour afficher une intégration zéro ETL à l'aide de l'API Amazon RDS, utilisez l'opération `DescribeIntegrations` avec le paramètre `IntegrationIdentifier`.

Surveillance des intégrations à l'aide des tables système pour Amazon Redshift

Amazon Redshift comporte des tables et vues système contenant des informations sur le fonctionnement du système. Vous pouvez interroger ces tables et vues système de la même manière

que vous interrogez toute autre table de base de données. Pour plus d'informations sur les tables et les vues système dans Amazon Redshift, consultez la référence relative aux tables et vues système dans le manuel Amazon Redshift Database Developer Guide.

Vous pouvez interroger les vues système et les tables suivantes pour obtenir des informations sur vos intégrations Zero-ETL :

- SVV_INTEGRATION : fournit les détails de configuration de vos intégrations.
- SVV_INTEGRATION_TABLE_STATE : décrit l'état de chaque table au sein d'une intégration.
- SYS_INTEGRATION_TABLE_STATE_CHANGE : affiche les journaux de changement d'état des tables pour une intégration.
- SYS_INTEGRATION_ACTIVITY : fournit des informations sur les cycles d'intégration terminés.

Toutes les CloudWatch métriques Amazon liées à l'intégration proviennent d'Amazon Redshift. Pour plus d'informations, consultez la section Mesures pour les intégrations sans ETL dans le guide de gestion Amazon Redshift. Actuellement, Amazon RDS () ne publie aucune métrique d'intégration sur CloudWatch.

Surveillance des intégrations avec Amazon EventBridge pour Amazon Redshift

Amazon Redshift envoie des événements liés à l'intégration à Amazon. EventBridge Pour obtenir la liste des événements et de l'événement correspondant IDs, consultez la section Notifications d'événements d'intégration Zero-ETL avec Amazon EventBridge dans le guide de gestion Amazon Redshift.

Modification des intégrations Amazon RDS

Vous pouvez uniquement modifier le nom, la description et les options de filtrage des données pour une intégration sans ETL dans un entrepôt de données pris en charge. Vous ne pouvez pas modifier la AWS KMS clé utilisée pour chiffrer l'intégration, ni les bases de données source ou cible.

Si vous ajoutez un filtre de données à une intégration existante, Amazon RDS réévalue le filtre comme s'il avait toujours existé. Il supprime toutes les données qui se trouvent actuellement dans l'entrepôt de données cible et qui ne correspondent pas aux nouveaux critères de filtrage. Si vous supprimez un filtre de données d'une intégration, il réplique toutes les données qui ne répondaient pas auparavant aux critères de filtrage (mais qui le sont désormais) dans l'entrepôt de données cible.

Pour de plus amples informations, veuillez consulter the section called “Filtrage des données pour les intégrations sans ETL”.

Vous pouvez modifier une intégration Zero-ETL à l'aide de l' AWS Management Console API, de AWS CLI, ou de l'API Amazon RDS.

Console RDS

Pour modifier une intégration zéro ETL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Zero-ETL integrations, puis choisissez l'intégration que vous souhaitez modifier.
3. Choisissez Modifier et apportez des modifications aux paramètres disponibles.
4. Lorsque toutes les modifications sont telles que vous le souhaitez, choisissez Modifier.

AWS CLI

Pour modifier une intégration Zero-ETL à l'aide de AWS CLI, appelez la commande modify-integration. En plus de`--integration-identifier`, spécifiez l'une des options suivantes :

- `--integration-name`— Spécifiez un nouveau nom pour l'intégration.
- `--description`— Spécifiez une nouvelle description pour l'intégration.
- `--data-filter`— Spécifiez les options de filtrage des données pour l'intégration. Pour de plus amples informations, veuillez consulter the section called “Filtrage des données pour les intégrations sans ETL”.

Example

La demande suivante modifie une intégration existante.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-integration \
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374 \
    --integration-name my-renamed-integration
```

Dans Windows :

```
aws rds modify-integration ^
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374 ^
    --integration-name my-renamed-integration
```

API RDS

Pour modifier une intégration zéro ETL à l'aide de l'API RDS, appelez l'ModifyIntegrationopération. Spécifiez l'identifiant d'intégration et les paramètres que vous souhaitez modifier.

Suppression des intégrations Amazon RDS Zero-ETL

Lorsque vous supprimez une intégration zéro ETL, Amazon RDS (Amazon Aurora) la supprime du cluster de base de de la base de données source. Vos données transactionnelles ne sont pas supprimées d'Amazon RDS, d' ou de la destination d'analyse, mais Amazon RDS n'envoie pas de nouvelles données à Amazon Redshift ou. Amazon SageMaker

Vous ne pouvez supprimer une intégration que si son statut est `ActiveFailed`,`Syncing`, ou`Needs attention`.

Vous pouvez supprimer les intégrations Zero-ETL à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

console

Pour supprimer une intégration zéro ETL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation de gauche, choisissez Intégrations Zero-ETL.
3. Sélectionnez l'intégration zéro ETL que vous souhaitez supprimer.
4. Choisissez Actions et Supprimer, puis confirmez la suppression.

AWS CLI

Pour supprimer une intégration zéro ETL, utilisez la commande delete-integration et spécifiez l'option `--integration-identifier`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds delete-integration \
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374
```

Dans Windows :

```
aws rds delete-integration ^
    --integration-identifier ee605691-6c47-48e8-8622-83f99b1af374
```

API RDS

Pour supprimer une intégration zéro ETL à l'aide de l'API Amazon RDS, utilisez l'opération `DeleteIntegration` avec le paramètre `IntegrationIdentifier`.

Résolution des problèmes liés aux intégrations Amazon RDS Zero-ETL

Vous pouvez vérifier l'état d'une intégration zéro ETL en interrogeant la table système SVV_INTEGRATION dans la destination d'analyse. Si la valeur de la colonne `state` est `ErrorState`, cela signifie que quelque chose ne va pas. Pour de plus amples informations, veuillez consulter the section called “Surveillance à l'aide des tables système”.

Utilisez les informations suivantes pour résoudre les problèmes courants liés aux intégrations Amazon RDS Aurora .

> **Important**
>
> Les opérations de resynchronisation et d'actualisation ne sont pas disponibles pour les intégrations sans ETL avec un lakehouse. Amazon SageMaker AI En cas de problème avec une intégration, vous devez la supprimer et en créer une nouvelle. Vous ne pouvez pas actualiser ou resynchroniser une intégration existante.

Rubriques

- Je ne parviens pas à créer une intégration zéro ETL

- Mon intégration est bloquée dans un état de Syncing
- Mes tables ne sont pas répliquées sur Amazon Redshift
- Une ou plusieurs de mes tables Amazon Redshift nécessitent une resynchronisation
- Problèmes d'intégration pour les intégrations Amazon SageMaker AI Lakehouse Zero-ETL

Je ne parviens pas à créer une intégration zéro ETL

Si vous ne parvenez pas à créer une intégration zéro ETL, assurez-vous que les informations suivantes sont correctes pour votre base de données source :

- Votre base de données source doit exécuter une version du moteur de base de données prise en charge. Pour une liste de versions prises en charge, consultez the section called “Intégrations zéro ETL ”.
- Vous avez correctement configuré les paramètres de base de données. Si les paramètres requis ne sont pas définis correctement ou ne sont pas associés à la base de données, la création échoue. Consultez the section called “Étape 1 : Créer un groupe de paramètres de base de données personnalisé”.

En outre, assurez-vous que les informations suivantes sont correctes pour votre entrepôt de données cible :

- La sensibilité à la casse est activée. Consultez Activation de la sensibilité à la casse pour votre entrepôt de données.
- Vous avez ajouté le principal autorisé et la source d'intégration appropriés. Consultez Configurer l'autorisation pour votre entrepôt de données Amazon Redshift.
- L'entrepôt de données est chiffré (s'il s'agit d'un cluster provisionné). Consultez la section Chiffrement de base de données Amazon Redshift.

Mon intégration est bloquée dans un état de **`Syncing`**

Il est possible que votre intégration affiche systématiquement le statut `Syncing` si vous modifiez la valeur de l'un des paramètres de base de données requis.

Pour résoudre ce problème, vérifiez les valeurs des paramètres du groupe de paramètres associé au de bases de données de base de données source et assurez-vous qu'elles correspondent aux

valeurs requises. Pour de plus amples informations, veuillez consulter the section called “Étape 1 : Créer un groupe de paramètres de base de données personnalisé”.

Si vous modifiez des paramètres, veillez à redémarrer le de base de données pour appliquer les modifications.

Mes tables ne sont pas répliquées sur Amazon Redshift

Si aucune ou plusieurs tables ne sont reflétées dans Amazon Redshift, vous pouvez exécuter la commande suivante pour les resynchroniser :

```
ALTER DATABASE dbname INTEGRATION REFRESH TABLES table1, table2;
```

Pour plus d'informations, consultez ALTER DATABASE dans le manuel Amazon Redshift SQL Reference.

Vos données ne sont peut-être pas répliquées car une ou plusieurs de vos tables sources ne possèdent pas de clé primaire. Le tableau de bord de surveillance d'Amazon Redshift affiche l'état de ces tables au fur `Failed` et à mesure que l'état de l'intégration Zero-ETL globale passe à. `Needs attention` Pour résoudre ce problème, vous pouvez identifier une clé existante dans votre table qui peut devenir une clé primaire, ou vous pouvez ajouter une clé primaire synthétique. Pour des solutions détaillées, consultez Gérer les tables sans clés primaires lors de la création d'intégrations Amazon Aurora MySQL ou Amazon RDS for MySQL Zero-ETL avec Amazon Redshift.

Une ou plusieurs de mes tables Amazon Redshift nécessitent une resynchronisation

L'exécution de certaines commandes sur votre base de données source peut nécessiter la resynchronisation de vos tables. Dans ce cas, la vue système SVV_INTEGRATION_TABLE_STATE affiche un `table_state` de `ResyncRequired`, ce qui signifie que l'intégration doit complètement recharger les données de cette table spécifique depuis MySQL vers Amazon Redshift.

Lorsque la table commence à se resynchroniser, elle passe à l'état `Syncing`. Aucune action manuelle n'est requise pour resynchroniser une table. Pendant la resynchronisation des données des tables, vous ne pouvez pas y accéder dans Amazon Redshift.

Vous trouverez ci-dessous quelques exemples d'opérations permettant de mettre une table dans un état `ResyncRequired` et les alternatives possibles à envisager.

Opération	exemple	Autrement
Ajout d'une colonne à une position spécifique	`ALTER TABLE table_name ADD COLUMN column_name INTEGER NOT NULL first;`	Amazon Redshift ne prend pas en charge l'ajout de colonnes à des positions spécifiques à l'aide des mots clés `first` et `after`. Si l'ordre des colonnes de la table cible n'est pas critique, ajoutez la colonne à la fin de la table à l'aide d'une commande plus simple : `ALTER TABLE table_name ADD COLUMN column_name column_type ;`
Ajout d'une colonne d'horodatage avec la valeur par défaut	`ALTER TABLE table_name ADD COLUMN column_name TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP;`	La `CURRENT_TIMESTAMP` valeur des lignes

Opération	exemple	Autrement
de `CURRENT_TIMESTAMP`		de table existantes est calculée par RDS pour MySQL et ne peut pas être simulée dans Amazon Redshift sans resynchronisation complète des données de table. Si possible, remplacez la valeur par défaut par une constante littérale comme `2023-01-01 00:00:15` afin d'éviter toute latence dans la disponibilité de la table.

Opération	exemple	Autrement
Réalisation d'opérations sur plusieurs colonnes au sein d'une seule commande	`ALTER TABLE table_name ADD COLUMN column_1, RENAME COLUMN column_2 TO column_3;`	Envisagez de diviser la commande en deux opérations distinctes, ADD et RENAME, qui ne nécessiteront pas de resynchronisation.

Problèmes d'intégration pour les intégrations Amazon SageMaker AI Lakehouse Zero-ETL

Si vous rencontrez des problèmes avec une intégration Zero-ETL existante avec un Amazon SageMaker AI Lakehouse, la seule solution consiste à supprimer l'intégration et à en créer une nouvelle. Contrairement à d'autres AWS services, les intégrations Zero-ETL ne prennent pas en charge les opérations d'actualisation ou de resynchronisation.

Pour résoudre les problèmes d'intégration :

1. Supprimez l'intégration Zero-ETL problématique à l'aide de la console, de la CLI ou de l'API.
2. Vérifiez que les configurations de la base de données source et de l'entrepôt de données cible sont correctes.
3. Créez une nouvelle intégration Zero-ETL avec la même configuration ou une configuration mise à jour.

Ce processus entraînera une réinitialisation complète du pipeline de données, ce qui peut prendre du temps en fonction de la taille de votre base de données source.

Amazon RDS pour DB2

Amazon RDS prend en charge les instances de base de données qui exécutent les éditions suivantes de IBM Db2 :

- Db2 Standard Edition
- Db2 Advanced Edition

Amazon RDS prend en charge les instances de base de données qui exécutent les versions suivantes de Db2 :

- DB2 11,5

Pour plus d'informations sur la prise en charge des versions mineures, consultez Versions de DB2 sur Amazon RDS.

Avant de créer une instance de base de données, suivez les étapes décrites dans la Configuration de votre environnement Amazon RDS section de ce guide de l'utilisateur. Lorsque vous créez une instance de base de données à l'aide de votre utilisateur principal, celui-ci obtient `DBADM` des droits, avec certaines limites. Utilisez cet utilisateur pour des tâches administratives telles que la création de comptes de base de données supplémentaires. Vous ne pouvez pas utiliser l'`SYSADM`autorité `SYSMAINT` au niveau de l'instance ou au niveau de la base de `SECADM` données. `SYSCTRL`

Vous pouvez créer ce qui suit :

- Instances DB
- Instantanés de base de données
- Point-in-time restaure
- Sauvegardes de stockage automatisées
- Sauvegardes de stockage manuelles

Vous pouvez utiliser des instances de base de données exécutant Db2 dans un cloud privé virtuel (VPC). Vous pouvez également ajouter des fonctionnalités à votre instance de base de données Amazon RDS pour DB2 en activant différentes options. Amazon RDS prend en charge les déploiements multi-AZ pour RDS for Db2 en tant que solution de basculement à haute disponibilité.

> **Important**
>
> Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures et tables du système qui nécessitent des privilèges élevés. Vous pouvez accéder à votre base de données à l'aide de clients SQL standard tels queIBM Db2 CLP. Toutefois, vous ne pouvez pas accéder directement à l'hôte en utilisant Telnet ou Secure Shell (SSH).

Rubriques

- Présentation de Db2 sur Amazon RDS
- Conditions préalables à la création d'une instance de base de données Amazon RDS pour DB2
- Plusieurs bases de données sur une instance de base de données Amazon RDS pour DB2
- Connexion à votre instance de base de données DB2
- Sécurisation des connexions aux instances de base de données Amazon RDS pour DB2
- Administration de votre instance de base de données Amazon RDS pour DB2
- Intégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3
- Migration de données vers Amazon RDS pour DB2
- Amazon RDS pour la fédération DB2
- Utilisation de répliques pour Amazon RDS pour DB2
- Options pour Amazon RDS pour les instances de base de données DB2
- Procédures stockées externes pour Amazon RDS pour DB2
- Problèmes connus et limites d'Amazon RDS pour DB2
- Référence de procédure stockée Amazon RDS pour DB2
- Référence RDS des fonctions définies par l'utilisateur Amazon pour Db2
- Résolution des problèmes liés à Amazon RDS pour DB2

Présentation de Db2 sur Amazon RDS

Vous pouvez lire les sections suivantes pour avoir une vue d'ensemble de Db2 sur Amazon RDS.

Rubriques

- Fonctionnalités d'Amazon RDS pour DB2

- Versions de DB2 sur Amazon RDS
- Options de licence Amazon RDS pour DB2
- Amazon RDS pour les classes d'instance DB2
- Rôles par défaut RDS d'Amazon pour DB2
- Paramètres Amazon RDS pour DB2
- EBCDICclassement pour les bases de données DB2 sur Amazon RDS
- Fuseau horaire local pour Amazon RDS pour les instances de base de données DB2

Fonctionnalités d'Amazon RDS pour DB2

Amazon RDS pour Db2 prend en charge la plupart des fonctionnalités et capacités de la base de données. IBM Db2 Certaines fonctions peuvent avoir une prise en charge limitée ou des privilèges restreints. Pour plus d'informations sur les fonctionnalités de base de données DB2 pour des versions spécifiques de DB2, consultez la IBM Db2 documentation.

Vous pouvez filtrer les nouvelles fonctions de Amazon RDS sur la page Nouveautés en matière de base de données. Pour Produits, choisissez Amazon RDS. Ensuite, vous pouvez effectuer une recherche en utilisant des mots clés tels que**`Db2 2023`**.

> **Note**
>
> Les listes suivantes ne sont pas exhaustives.

Rubriques

- Fonctionnalités prises en charge dans RDS pour DB2
- Fonctionnalités non prises en charge dans RDS pour DB2

Fonctionnalités prises en charge dans RDS pour DB2

RDS pour Db2 prend en charge des fonctionnalités qui incluent des fonctionnalités natives IBM Db2 et des fonctionnalités essentielles à Amazon RDS.

Fonctionnalités natives de IBM Db2

RDS pour DB2 prend en charge les fonctionnalités de base de données Db2 suivantes :

- Création d'une base de données standard qui utilise un jeu de codes, un classement, un format de page et un territoire définis par le client. Utilisez la procédure rdsadmin.create_database stockée Amazon RDS.
- Ajout, suppression ou modification d'utilisateurs et de groupes locaux. Utilisez les procédures stockées Amazon RDS pourProcédures stockées pour l'octroi et la révocation de privilèges pour RDS pour Db2.
- Création de rôles avec la procédure rdsadmin.create_role stockée Amazon RDS.
- Support pour les tables standard organisées en rangées.
- Support de la charge de travail analytique pour les tables organisées en colonnes.
- Possibilité de définir des fonctionnalités de compatibilité DB2 telles Oracle que et. MySQL
- Support pour les procédures stockées externes Java basées sur des bases.
- Support pour le chiffrement des données en transit à l'aide du protocole SSL/TLS.
- Surveillance de l'état d'une base de données (`ALIVEDOWNSTORAGE_FULL`,,`UNKNOWN`, et`STANDBY_CONNECTABLE`).
- Restauration d'une base de données en ligne ou hors ligne fournie par le client. Linux (LE) Utilisez les procédures stockées Amazon RDS pourProcédures stockées pour les bases de données pour RDS pour Db2.
- Application de journaux d'archivage DB2 fournis par le client pour maintenir la base de données synchronisée avec les bases de données DB2 autogérées. Utilisez les procédures stockées Amazon RDS pourProcédures stockées pour les bases de données pour RDS pour Db2.
- Support pour l'audit au niveau de l'instance DB2 et au niveau de la base de données.
- Support à une fédération homogène.
- Possibilité de charger une table à partir de fichiers de données dans Amazon Simple Storage Service (Amazon S3).
- Autorisations accordées aux utilisateurs, aux groupes ou aux rôles, tels que `CONNECT`,`SYSMON`,`ACCESSCTRL`,`DATAACCESS`,,`SQLADM`,`WLMADM`, `EXPLAIN`,`LOAD`, ou`IMPLICIT_SCHEMA`.
- Création de plusieurs bases de données.

> **Note**
>
> Une instance de base de données RDS pour DB2 peut contenir jusqu'à 50 bases de données. Pour de plus amples informations, veuillez consulter the section called “Plusieurs bases de données DB2”.

Fonctionnalités essentielles à Amazon RDS

RDS pour DB2 prend en charge les fonctionnalités principales d'Amazon RDS suivantes :

- Groupes de paramètres personnalisés à attribuer aux instances de base de données
- Création, modification et suppression d'instances de base de données
- Restauration d'une sauvegarde de base de données DB2 autogérée hors ligne ou en ligne Linux (LE)

> **Note**
>
> Pour pouvoir restaurer votre sauvegarde, ne donnez pas de nom à votre base de données lorsque vous créez une instance de base de données. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

- Support des types de stockage gp3, io2 et io1
- Utilisation de AWS Managed Microsoft AD pour l'Kerberosauthentification et autorisation de groupe LDAP pour RDS pour DB2
- Modification des groupes de sécurité, des ports, des types d'instances, du stockage, des périodes de conservation des sauvegardes et d'autres paramètres pour les instances Db2 existantes
- Protection contre la suppression pour les instances de base de données
- Restauration entre régions point-in-time (PITR), y compris pour les sauvegardes chiffrées
- Utilisation de AWS Key Management Service (AWS KMS) pour le chiffrement du stockage et le chiffrement au repos
- Instances de base de données multi-AZ avec une seule instance de secours pour une haute disponibilité
- Redémarrages d'instances de base de données
- Mises à jour des mots de passe principaux

- Restauration des instances de base de données à une heure précise
- Backup et restauration d'instances de base de données à l'aide de sauvegardes au niveau du stockage
- Démarrage et arrêt des instances de base de données
- Maintenance des instances de base de données

Fonctionnalités non prises en charge dans RDS pour DB2

RDS pour DB2 ne prend pas en charge les fonctionnalités de base de données Db2 suivantes :

- `SYSADMSECADM`, et `SYSMAINT` accès pour l'utilisateur principal.
- Procédures stockées externes écrites en C, C++ ou Cobol.
- Plusieurs instances de base de données DB2 sur un seul hôte.
- Plug-ins GSS-API externes pour l'authentification.
- Plug-ins tiers externes pour sauvegarder ou restaurer les bases de données DB2.
- Traitement massivement parallèle (MPP) sur plusieurs nœuds, tel que. IBM Db2 Warehouse
- IBM Db2 pureScale.
- Configuration manuelle de High Availability Disaster Recovery (HADR) pour RDS pour DB2.

> **Note**
>
> Amazon RDS prend en charge et gère le HADR pour RDS pour DB2 par le biais de répliques. Pour de plus amples informations, veuillez consulter Utilisation de répliques pour Amazon RDS pour DB2.
> RDS for Db2 prend en charge les déploiements multi-AZ, les sauvegardes automatisées entre régions et la réplication. Pour plus d'informations, consultez Déploiements d'instances de base de données multi-AZ pour Amazon RDS et Réplication des sauvegardes automatisées vers une autre Région AWS.

- Chiffrement de base de données natif.
- Fédération hétérogène avec Informix, Sybase et Teradata. Pour de plus amples informations, veuillez consulter the section called “Fédération”.
- Création de routines non clôturées et migration de routines non clôturées existantes en sauvegardant et en restaurant les données. Pour de plus amples informations, veuillez consulter Routines non clôturées.

- Création de nouveaux tablespaces de stockage non automatiques. Pour de plus amples informations, veuillez consulter Tablespaces de stockage non automatiques pendant la migration.
- Tables extérieures.

Versions de DB2 sur Amazon RDS

Pour Db2, les numéros de version prennent la forme major.minor.build.revision, par exemple 11.5.9.0.sb00000000.r1. Notre implémentation de version correspond à celle de Db2.

majeur

Le numéro de version principal est à la fois le nombre entier et la première partie fractionnaire du numéro de version, par exemple 11,5. Un changement de version est considéré comme majeur si le numéro de version principale change, par exemple si vous passez de la version 11.5 à la version 12.1.

mineur

Le numéro de version secondaire est à la fois la troisième et la quatrième partie du numéro de version, par exemple, 9.0 dans la version 11.5.9.0. La troisième partie indique le modpack Db2, par exemple, 9 dans la version 9.0. La quatrième partie indique le fixpack Db2, par exemple, 0 dans la version 9.0. Un changement de version est considéré comme mineur si le modpack Db2 ou le fixpack Db2 change, par exemple en passant de la version 11.5.9.0 à la version 11.5.9.1, ou de la version 11.5.9.0 à la version 11.5.10.0, avec des exceptions pour fournir des mises à jour des tables de catalogue. (Amazon RDS prend en charge ces exceptions.)

construire

Le numéro de version est la cinquième partie du numéro de version, par exemple, sb00000000 dans la version 11.5.9.0.sb00000000. Un numéro de version dont la partie numérique est entièrement composée de zéros indique une version standard. Un numéro de build dont la partie numérique n'est pas entièrement composée de zéros indique une version spéciale. Un numéro de version change s'il existe un correctif de sécurité ou une version spéciale d'une version DB2 existante. Un changement de numéro de version indique également qu'Amazon RDS a automatiquement appliqué une nouvelle version mineure.

révision

Le numéro de révision est la sixième partie du numéro de version, par exemple, r1 dans 11.5.9.0.sb00000000.r1. Une révision est une révision Amazon RDS d'une version DB2 existante.

Un changement de numéro de révision indique qu'Amazon RDS a automatiquement appliqué une nouvelle version mineure.

Rubriques

- Versions mineures de DB2 prises en charge sur Amazon RDS
- Versions majeures de DB2 prises en charge sur Amazon RDS

Versions mineures de DB2 prises en charge sur Amazon RDS

Le tableau suivant indique les versions mineures de DB2 11.5 actuellement prises en charge par Amazon RDS.

> **Note**
>
> Les dates avec seulement un mois et une année sont approximatives et sont mises à jour avec une date exacte quand elles sont connues.

Version du moteur DB2	IBMdate de sortie	Date de parution de RDS	
11,5,9.0	15 novembre 2023	27 novembre 2023	

Vous pouvez spécifier n'importe quelle version de DB2 actuellement prise en charge lors de la création d'une nouvelle instance de base de données. Vous pouvez spécifier la version principale (telle que DB2 11.5) et toute version mineure prise en charge pour la version principale spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version prise en charge, généralement la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée. Pour voir la liste des versions prises en charge, ainsi que les valeurs par défaut pour les instances de base de données nouvellement créées, utilisez la commande describe-db-engine-versions AWS Command Line Interface (AWS CLI).

Par exemple, pour répertorier les versions de moteur prises en charge pour Amazon RDS pour Db2, exécutez la commande suivante AWS CLI . *region*Remplacez-le par votre Région AWS.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-engine-versions \
    --filters Name=engine,Values=db2-ae,db2-se \
    --query "DBEngineVersions[].{Engine:Engine, EngineVersion:EngineVersion,
 DBParameterGroupFamily:DBParameterGroupFamily}" \
    --region region
```

Dans Windows :

```
aws rds describe-db-engine-versions ^
    --filters Name=engine,Values=db2-ae,db2-se ^
    --query "DBEngineVersions[].{Engine:Engine, EngineVersion:EngineVersion,
 DBParameterGroupFamily:DBParameterGroupFamily}" ^
    --region region
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
[
    {
    "Engine": "db2-ae",
    "EngineVersion": "11.5.9.0.sb00000000.r1",
    "DBParameterGroupFamily": "db2-ae-11.5"
    },
    {
    "Engine": "db2-se",
    "EngineVersion": "11.5.9.0.sb00000000.r1",
    "DBParameterGroupFamily": "db2-se-11.5"
    }
]
```

La version par défaut de DB2 peut varier de Région AWS. Pour créer une instance de base de données avec une version mineure spécifique, spécifiez la version mineure lors de la création de l'instance de base de données. Vous pouvez déterminer la version par défaut d'un Région AWS for `db2-ae` et d'un moteur `db2-se` de base de données en exécutant la `describe-db-engine-versions` commande. L'exemple suivant renvoie la version par défaut pour `db2-ae` dans l'est des États-Unis (Virginie du Nord).

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-engine-versions \
```

```
    --default-only --engine db2-ae \
    --query "DBEngineVersions[].{Engine:Engine, EngineVersion:EngineVersion,
 DBParameterGroupFamily:DBParameterGroupFamily}" \
    --region us-east-1
```

Dans Windows :

```
aws rds describe-db-engine-versions ^
    --default-only --engine db2-ae ^
    --query "DBEngineVersions[].{Engine:Engine, EngineVersion:EngineVersion,
 DBParameterGroupFamily:DBParameterGroupFamily}" ^
    --region us-east-1
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
[
    {
    "Engine": "db2-ae",
    "EngineVersion": "11.5.9.0.sb00000000.r1",
    "DBParameterGroupFamily": "db2-ae-11.5"
    }
]
```

Avec Amazon RDS, vous pouvez contrôler le moment de la mise à niveau de votre instance Db2 vers une nouvelle version majeure prise en charge par Amazon RDS. Vous pouvez maintenir la compatibilité avec des versions spécifiques de DB2, tester de nouvelles versions avec votre application avant de les déployer en production et effectuer des mises à niveau de versions majeures aux moments qui correspondent le mieux à votre calendrier.

Lorsque la mise à niveau automatique des versions mineures est activée, Amazon RDS met automatiquement à niveau vos instances de base de données vers les nouvelles versions mineures de DB2, car elles sont prises en charge par Amazon RDS. Ces correctifs sont appliqués pendant le créneau de maintenance planifié. Vous pouvez modifier une instance de base de données pour activer ou désactiver les mises à niveau automatiques des versions mineures.

À l'exception des versions 11.5.9.1 et 11.5.10.0 de DB2, les mises à niveau automatiques vers la nouvelle version mineure de DB2 incluent les mises à niveau automatiques vers les nouvelles versions et révisions. Pour les versions 11.5.9.1 et 11.5.10.0, mettez à niveau manuellement les versions mineures.

Si vous refusez les mises à niveau automatiques planifiées, vous pouvez procéder manuellement à une mise à niveau vers une version mineure prise en charge en suivant la même procédure que pour une mise à jour de la version majeure. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Versions majeures de DB2 prises en charge sur Amazon RDS

Les versions majeures de RDS pour DB2 sont disponibles dans le cadre du support standard au moins jusqu'à IBM la fin du support (base) pour la version correspondanteIBM. Le tableau suivant indique les dates que vous pouvez utiliser pour planifier vos cycles de test et de mise à niveau. Si Amazon prolonge le support d'une version de RDS pour DB2 pendant une période plus longue que celle initialement indiquée, nous prévoyons de mettre à jour ce tableau pour tenir compte de cette date ultérieure.

Vous pouvez utiliser les dates suivantes pour planifier vos cycles de test et de mise à niveau.

> **Note**
>
> Les dates avec seulement un mois et une année sont approximatives et sont mises à jour avec une date exacte quand elles sont connues.
> Vous pouvez consulter les versions principales de vos bases de données DB2 en exécutant la AWS CLI commande describe-db-major-engine-versions ou en utilisant l'opération d'API Describe DBMajor EngineVersions RDS.

Version majeure de DB2	IBMdate de sortie	Date de parution de RDS	IBMfin du support (éditions Standard et Advanced)	IBMfin du support (prolongé)	
DB2 11,5	27 juin 2019	27 novembre 2023	Pas encore annoncé parIBM. [1]	4 ans après la fin du support	

[1] Pour obtenir les dernières informations, consultez les end-of-supportdates des produits DB2 dans la IBM Db2 documentation.

Options de licence Amazon RDS pour DB2

Amazon RDS pour Db2 propose deux options de licence : bring your own license (BYOL) et Db2 license through. AWS Marketplace

Rubriques

- Apportez votre propre licence (BYOL) pour Db2
- Licence DB2 via AWS Marketplace
- Basculer entre les licences DB2

Apportez votre propre licence (BYOL) pour Db2

Dans le modèle BYOL, vous utilisez vos licences de base de données DB2 existantes pour déployer des bases de données sur Amazon RDS. Vérifiez que vous disposez de la licence de base de données DB2 appropriée pour la classe d'instance de base de données et l'édition de base de données DB2 que vous souhaitez exécuter. Vous devez également suivre les IBM politiques de licence des logiciels IBM de base de données dans l'environnement de cloud computing.

> **Note**
>
> Les instances de base de données multi-AZ sont des instances de secours à froid car la base de données DB2 est installée mais ne fonctionne pas. Les standbys ne sont pas lisibles, ne sont pas en cours d'exécution ou ne répondent pas aux demandes. Pour plus d'informations, consultez les informations IBM Db2 relatives aux licences sur le site Web d'IBM.

Dans ce modèle, vous continuez à utiliser votre compte de IBM support actif et vous contactez IBM directement pour les demandes de service de base de données DB2. Si vous avez un Support compte avec support de dossier, vous pouvez le contacter Support pour tout problème concernant Amazon RDS. Amazon Web Services et IBM disposent d'un processus de support multifournisseur pour les cas nécessitant l'assistance des deux organisations.

Amazon RDS prend en charge le modèle BYOL pour Db2 Standard Edition et. Db2 Advanced Edition

Rubriques

- IBM IDs pour apporter votre propre licence (BYOL) pour Db2
- Ajout IBM IDs à un groupe de paramètres pour les instances de base de données RDS pour DB2

- Intégration avec AWS License Manager

IBM IDs pour apporter votre propre licence (BYOL) pour Db2

Dans le modèle BYOL, vous IBM Customer ID devez créer, modifier ou restaurer RDS pour les instances de base de données DB2. IBM Site ID Vous devez créer un groupe de paramètres personnalisé avec votre IBM Customer ID et votre IBM Site ID avant de créer une instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Ajout IBM IDs à un groupe de paramètres pour les instances de base de données RDS pour DB2. Vous pouvez exécuter plusieurs instances de base de données RDS pour DB2 avec des instances différentes IBM Customer IDs et IBM Site IDs dans le même Compte AWS ou. Région AWS

> **Important**
>
> Si nous ne pouvons pas vérifier votre licence par vous-même IBM Customer ID et par vous-mêmeIBM Site ID, nous pouvons résilier toutes les instances de base de données exécutées avec ces licences non vérifiées.

Si vous êtes un nouveau IBM Db2 client, vous devez d'abord acheter une licence logicielle DB2 auprès de IBM. Après avoir acheté une licence logicielle DB2, vous recevrez une preuve d'admissibilité de la part de IBM laquelle vous IBM Customer ID et votre. IBM Site ID

Si vous êtes déjà IBM Db2 client, vous pouvez trouver votre IBM Customer ID et le vôtre IBM Site ID sur votre certificat de preuve d'admissibilité auprès deIBM.

Vous pouvez également trouver votre IBM Customer ID et votre IBM Site ID dans votre compte IBM Passport Advantage Online. Une fois connecté, vous pouvez consulter les deux IDs sur la page principale ou sur la page de téléchargement du logiciel.

Ajout IBM IDs à un groupe de paramètres pour les instances de base de données RDS pour DB2

Comme vous ne pouvez pas modifier les groupes de paramètres par défaut, vous devez créer un groupe de paramètres personnalisé, puis le modifier pour inclure les valeurs de votre IBM Customer ID et de votreIBM Site ID. Pour plus d'informations sur les groupes de paramètres, veuillez consulter Groupes de paramètres de base de données pour les instances de base de données RDSAmazon.

> **Important**
>
> Vous devez créer un groupe de paramètres personnalisé avec votre IBM Customer ID et votre IBM Site ID avant de créer une instance de base de données RDS pour DB2.

Utilisez les paramètres définis dans le tableau suivant.

Paramètre	Valeur
`rds.ibm_customer_id`	`<your IBM Customer ID>`
`rds.ibm_site_id`	`<your IBM Site ID>`
`ApplyMethod`	`immediate` , `pending-reboot`

Ces paramètres sont dynamiques, ce qui signifie que toute modification apportée prend effet immédiatement et qu'il n'est pas nécessaire de redémarrer l'instance de base de données. Si vous ne souhaitez pas que les modifications prennent effet immédiatement, vous pouvez les configurer `pending-reboot` et planifier `ApplyMethod` pour qu'elles soient effectuées pendant une période de maintenance.

Vous pouvez créer et modifier un groupe de paramètres personnalisé à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API Amazon RDS.

console

Pour ajouter votre IBM Customer ID et votre IBM Site ID à un groupe de paramètres

1. Créez un nouveau groupe de paramètres de base de données. Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez le groupe de paramètres que vous avez créé. Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

AWS CLI

Pour ajouter votre IBM Customer ID et votre IBM Site ID à un groupe de paramètres

1. Créez un groupe de paramètres personnalisé en exécutant la create-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Nom du groupe de paramètres que vous créez.
   - `--db-parameter-group-family`— L'édition et la version majeure du moteur DB2. Valeurs valides : `db2-se-11.5`, `db2-ae-11.5`.
   - `--description`— Description de ce groupe de paramètres.

   Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez les paramètres du groupe de paramètres personnalisés que vous avez créé en exécutant la modify-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Le nom du groupe de paramètres que vous avez créé.
   - `--parameters`— Tableau de noms de paramètres, de valeurs et de méthodes d'application pour la mise à jour des paramètres.

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

API RDS

Pour ajouter votre IBM Customer ID et votre IBM Site ID à un groupe de paramètres

1. Créez un groupe de paramètres de base de données personnalisé à l'aide de l'CreateDBParameterGroupopération d'API Amazon RDS.

   Incluez les paramètres requis suivants :

- `DBParameterGroupName`
- `DBParameterGroupFamily`
- `Description`

Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.

2. Modifiez les paramètres du groupe de paramètres personnalisé que vous avez créé à l'aide de l'ModifyDBParameterGroupopération d'API RDS.

   Incluez les paramètres requis suivants :

   - `DBParameterGroupName`
   - `Parameters`

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Vous êtes maintenant prêt à créer une instance de base de données et à attacher le groupe de paramètres personnalisé à l'instance de base de données. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS et Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon.

Intégration avec AWS License Manager

Pour faciliter la surveillance de l'utilisation des licences RDS for Db2 dans le modèle BYOL, il AWS License Managers'intègre à RDS for Db2. License Manager prend en charge le suivi des éditions du moteur RDS pour DB2 sur la base de virtual CPUs (vCPUs). Vous pouvez également utiliser License Manager AWS Organizations pour gérer tous les comptes de votre organisation de manière centralisée.

Pour suivre l'utilisation des licences de vos instances de base de données RDS pour DB2, vous devez créer des licences autogérées. Vous pouvez créer des licences autogérées à l'aide de AWS Management Console la AWS License Manager CLI et de l' AWS License Manager API. Vous pouvez également automatiser la création de licences autogérées à l'aide AWS CloudFormation de Terraform modèles.

Les ressources RDS pour DB2 qui correspondent au filtre d'informations sur le produit sont automatiquement associées à la licence autogérée. La découverte de RDS pour les instances de base de données DB2 peut prendre jusqu'à 24 heures.

Le tableau suivant indique les valeurs disponibles pour le filtre d'informations sur le produit Engine Edition pour RDS for Db2.

Valeur	Description
`db2-se`	Édition standard DB2
`db2-ae`	Édition avancée Db2

Rubriques

- Terminologie
- Création d'une licence autogérée dans AWS License Manager
- Automatiser la création de licences autogérées à l'aide de modèles AWS License Manager
- Paramètres de création de licences autogérées

Terminologie

Cette page utilise la terminologie suivante pour aborder l'intégration d'Amazon RDS avec AWS License Manager.

Licence autogérée

> Le terme « licence autogérée » est utilisé dans. AWS License Manager La console Amazon RDS fait référence à la licence en tant que AWS License Manager configuration. Une licence autogérée contient des règles de licence basées sur les termes de vos contrats d'entreprise. Les règles que vous créez déterminent le mode de AWS traitement des commandes consommant des licences. Lors de la création d'une licence autogérée, travaillez en étroite collaboration avec l'équipe de conformité de votre organisation pour revoir les accords de votre entreprise. Pour plus d'informations, consultez la section Licences autogérées dans License Manager.

Création d'une licence autogérée dans AWS License Manager

Vous pouvez créer une licence autogérée à l'aide de la AWS Management Console AWS License Manager CLI et de l' AWS License Manager API.

> **Note**
>
> Si vous créez une instance de base de données RDS pour DB2 à l'aide du AWS Management Console, vous créerez une licence autogérée en saisissant un nom pour la licence. Amazon RDS associe ensuite l'instance de base de données à cette licence. (Dans la console Amazon RDS, cette licence est appelée AWS License Manager configuration.) Si vous souhaitez créer une instance de base de données RDS pour DB2 à l'aide de la AWS License Manager CLI ou de l' AWS License Manager API, vous devez d'abord créer une licence autogérée en suivant les étapes suivantes. La même situation s'applique à la restauration d'une instance de base de données RDS pour DB2 à un moment donné ou à partir d'un instantané.

console

Pour créer une licence autogérée afin de suivre l'utilisation des licences de votre RDS pour les instances de base de données DB2

1. Accédez à https://console.aws.amazon.com/license-manager/.
2. Créez une licence autogérée.

   Pour obtenir des instructions, voir Création d'une licence autogérée dans le guide de l'AWS License Manager utilisateur.

   Ajoutez une règle pour un RDS Product Information Filter (Filtre d'informations produit RDS) dans le panneau Product Information (Informations produit) .

   Pour plus d’informations, consultez ProductInformation dans la Référence d’API AWS License Manager .

AWS License Manager CLI

> **Note**
>
> Cette procédure utilise une commande AWS License Manager CLI.

Pour créer une licence autogérée à l'aide de AWS CLI, exécutez la AWS License Manager create-license-configurationcommande. Utilisez les `--cli-input-yaml` options `--cli-input-json` ou pour transmettre les options à la commande.

Pour de plus amples informations, veuillez consulter the section called “Paramètres de création de licences autogérées”.

La commande suivante crée une licence autogérée pour Db2 Standard Edition.

```
aws license-manager create-license-configuration --cli-input-json file://rds-db2-
se.json
```

Le JSON suivant est le contenu du `rds-db2-se.json` fichier utilisé dans la commande précédente.

```
{
    "Name": "rds-db2-se",
    "Description": "RDS Db2 Standard Edition",
    "LicenseCountingType": "vCPU",
    "LicenseCountHardLimit": false,
    "ProductInformationList": [
        {
            "ResourceType": "RDS",
            "ProductInformationFilterList": [
                {
                    "ProductInformationFilterName": "Engine Edition",
                    "ProductInformationFilterValue": ["db2-se"],
                    "ProductInformationFilterComparator": "EQUALS"
                }
            ]
        }
    ]
}
```

Pour de plus amples informations sur les informations produit, veuillez consulter Détection automatique de l'inventaire des ressources dans le Guide de l'utilisateur AWS License Manager .

Pour de plus amples informations sur le paramètre `--cli-input`, veuillez consulter Génération du squelette de l'interface de ligne de commande AWS CLI et des paramètres d'entrée à partir d'un fichier d'entrée JSON ou YAML dans le Guide de l'utilisateur de l'AWS CLI .

AWS License Manager API

> **Note**
>
> Cette procédure utilise une commande d' AWS License Manager API.

Pour créer une licence autogérée, utilisez l'opération `CreateLicenseConfiguration` AWS License Manager API avec les paramètres obligatoires suivants :

- `Name`
- `LicenseCountingType`
- `ProductInformationList`
- `ResourceType`
- `ProductInformationFilterList`
- `ProductInformationFilterName`
- `ProductInformationFilterValue`
- `ProductInformationFilterComparator`

Pour de plus amples informations sur les paramètres, veuillez consulter the section called “Paramètres de création de licences autogérées”.

Automatiser la création de licences autogérées à l'aide de modèles AWS License Manager

Vous pouvez automatiser la création de licences autogérées à l'aide AWS CloudFormation de Terraform modèles.

L'exemple de AWS CloudFormation modèle suivant crée des licences autogérées pour Db2 Standard Edition sur RDS for Db2. Pour un modèle pour Db2 Advanced Edition, mettez à jour les valeurs pour `Name`, `Description`, et`ProductInformationFilter`.

```
AWSTemplateFormatVersion: "2010-09-09"
Description: CloudFormation template to create a License Configuration for Db2 Standard
 Edition on RDS for Db2.
```

```
Resources:
  Db2LicenseConfiguration:
    Type: "AWS::LicenseManager::LicenseConfiguration"
    Properties:
      Name: "rds-db2-se"
      Description: "Db2 Standard Edition on RDS for Db2"
      LicenseCountingType: "vCPU"
      LicenseCountHardLimit: false
      ProductInformationList:
        - ResourceType: "RDS"
          ProductInformationFilterList:
            - ProductInformationFilterName: "Engine Edition"
              ProductInformationFilterValue:
                - "db2-se"
              ProductInformationFilterComparator: "EQUALS"
```

Pour plus d'informations sur l'utilisation AWS CloudFormation avec Amazon RDS, consultezCréation de ressources RDSAmazon avec AWS CloudFormation.

L'exemple de Terraform modèle suivant crée des licences autogérées pour Db2 Standard Edition sur RDS for Db2. *us-east-1*Remplacez-le par votre Région AWS. Pour un modèle pour Db2 Advanced Edition, mettez à jour les valeurs pour `name``description`, et`product_information_filter`.

```
provider "aws" {
  region = "us-east-1"
}

resource "aws_licensemanager_license_configuration" "rds_db2_license_config" {
  name                     = "rds-db2-se"
  description              = "Db2 Standard Edition on RDS for Db2
  license_counting_type    = "vCPU"
  license_count_hard_limit = false

  product_information_list {
    resource_type = "RDS"

    product_information_filter {
      name       = "Engine Edition"
      comparator = "EQUALS"
      value      = ["db2-se"]
    }
  }
```

```
}
```

Pour plus d'informations sur l'utilisation Terraform d'Amazon RDS, consultez la section Utilisation en Terraform tant qu'outil IaC AWS Cloud et les meilleures pratiques d'utilisation du AWS fournisseur Terraform dans AWS les directives prescriptives.

Paramètres de création de licences autogérées

Dans le tableau suivant, vous trouverez des informations détaillées sur les paramètres de création de licences autogérées à l'aide de la AWS License Manager CLI, de l' AWS License Manager API, d'un AWS CloudFormation modèle et d'un Terraform modèle. Le nom du paramètre indiqué dans le tableau suivant s'affiche au format du nom utilisé dans l' AWS License Manager API et dans le AWS CloudFormation modèle.

Nom du paramètre	Type de données	Obligatoire	Description
Nom	chaîne	Oui	Nom de la configuration de licence.
Description	chaîne	Non	Description de la configuration de licence.
LicenseCountingType	chaîne	Oui	Dimension utilisée pour suivre l'inventaire des licences. Valeur valide : `vCPU`.
LicenseCountHardLimit	boolean	Non	Indique si une application stricte ou souple des licences est utilisée. Le dépassement d'une limite stricte bloque le lancement de nouvelles instances.
ProductInformationList	tableau d'objets	Oui	Liste des informations sur le produit pour

Nom du paramètre	Type de données	Obligatoire	Description
			une configuration de licence.
ResourceType	chaîne	Oui	Type de ressource. Valeur valide : `RDS`.
ProductInformationFilterList	tableau d'objets	Oui	Liste des filtres d'informations sur le produit pour une configuration de licence.
ProductInformationFilterName	chaîne	Oui	Nom du type de filtre déclaré. Valeur valide : `Engine Edition`.
ProductInformationFilterValue	tableau de chaînes	Oui	La valeur sur laquelle filtrer. Vous ne devez spécifier qu'une seule valeur. Valeurs valides : `db2-se` ou `db2-ae`.
ProductInformationFilterComparator	chaîne	Oui	L'opérateur logique pour`ProductInformation FilterName` . Valeur valide : `EQUALS`.

Licence DB2 via AWS Marketplace

Dans le AWS Marketplace modèle de licence DB2, vous payez un tarif horaire pour vous abonner aux licences DB2. Ce modèle vous permet de démarrer rapidement avec RDS pour DB2 sans avoir à acheter de licences.

Pour utiliser la licence DB2 via AWS Marketplace, vous avez besoin d'un AWS Marketplace abonnement actif pour l'IBM Db2édition particulière que vous souhaitez utiliser. Si vous n'en avez pas déjà un, abonnez-vous AWS Marketplaceà cette IBM Db2 édition.

Amazon RDS prend en charge les licences DB2 AWS Marketplace pour les éditions IBM Db2 Standard et IBM Db2 Advanced.

Rubriques

- Terminologie
- Paiements et facturation
- Abonnement aux listes de DB2 Marketplace et inscription auprès de IBM
- Obtenir une offre privée

Terminologie

Cette page utilise la terminologie suivante pour aborder l'intégration d'Amazon RDS avec AWS Marketplace.

Abonnement SaaS

Dans AWS Marketplace, les produits software-as-a-service (SaaS) tels que le modèle de pay-as-you-go licence adoptent un modèle d'abonnement basé sur l'utilisation. IBM, le vendeur de logiciels pour DB2, suit votre utilisation et vous ne payez que pour ce que vous utilisez.

Offre publique

Les offres publiques vous permettent d'acheter AWS Marketplace des produits directement auprès du AWS Management Console.

Offre privée

Les offres privées sont un programme d'achat qui permet aux vendeurs et aux acheteurs de négocier des prix personnalisés et les conditions du contrat de licence utilisateur final (EULA) pour les achats en AWS Marketplace.

Frais de Db2 Marketplace

Frais facturés pour l'utilisation de la licence logicielle DB2 parIBM. Ces frais de service sont mesurés AWS Marketplace et apparaissent sur votre AWS facture dans la AWS Marketplace section correspondante.

Frais Amazon RDS

Frais AWS facturés pour les services RDS pour DB2, à l'exclusion des licences lors de l'utilisation AWS Marketplace de licences DB2. Les frais sont mesurés via le service Amazon RDS utilisé et apparaissent sur votre AWS facture.

Paiements et facturation

RDS pour Db2 s'intègre pour AWS Marketplace proposer des pay-as-you-go licences horaires pour Db2. Les frais de Db2 Marketplace couvrent les coûts de licence du logiciel Db2, et les frais Amazon RDS couvrent les coûts de votre RDS pour l'utilisation de l'instance de base de données DB2. Pour plus d'informations sur la tarification, consultez la section Tarification d'Amazon RDS pour DB2.

Pour arrêter ces frais, vous devez supprimer tous les RDS pour les instances de base de données DB2. En outre, vous pouvez supprimer vos abonnements AWS Marketplace aux licences DB2. Si vous supprimez vos abonnements sans supprimer vos instances de base de données, Amazon RDS continuera de vous facturer l'utilisation des instances de base de données. Pour de plus amples informations, veuillez consulter the section called “Suppression d'une instance DB”.

Vous pouvez consulter les factures et gérer les paiements de vos instances de base de données RDS pour DB2 qui utilisent une licence DB2 via AWS Marketplace la console.AWS Billing Vos factures incluent deux frais : un pour votre utilisation de la licence DB2 AWS Marketplace et un pour votre utilisation d'Amazon RDS. Pour plus d'informations sur la facturation, consultez la section Consulter votre facture dans le guide de AWS Billing and Cost Management l'utilisateur.

Abonnement aux listes de DB2 Marketplace et inscription auprès de IBM

Pour utiliser la licence DB2 via AWS Marketplace, vous devez utiliser le AWS Management Console pour effectuer les deux tâches suivantes. Vous ne pouvez pas effectuer ces tâches par le biais de l'API AWS CLI ou de l'API RDS.

> **Note**
>
> Si vous souhaitez créer vos instances de base de données à l'aide de l'API AWS CLI ou de l'API RDS, vous devez d'abord effectuer ces deux tâches.

Rubriques

- Tâche 1 : S'abonner à Db2 dans AWS Marketplace
- Tâche 2 : Enregistrez votre abonnement auprès IBM

Tâche 1 : S'abonner à Db2 dans AWS Marketplace

Pour utiliser la licence Db2 avec AWS Marketplace, vous devez disposer d'un AWS Marketplace abonnement actif à Db2. Les abonnements étant associés à une IBM Db2 édition spécifique, vous devez vous abonner à Db2 AWS Marketplace pour chaque édition de Db2 que vous souhaitez utiliser : édition IBM Db2 avancée, édition IBM Db2 standard. Pour plus d'informations sur les AWS Marketplace abonnements, consultez la section Abonnements basés sur l'utilisation du Saas dans le Guide de l'AWS Marketplace acheteur.

Nous vous recommandons de vous abonner à Db2 AWS Marketplace avant de commencer à créer une instance de base de données.

Tâche 2 : Enregistrez votre abonnement auprès IBM

Après vous être abonné à Db2 in AWS Marketplace, terminez l'enregistrement de votre commande IBM à partir de la AWS Marketplace page correspondant au type d'abonnement Db2 que vous avez choisi. Sur la AWS Marketplace page, choisissez Afficher les options d'achat, puis sélectionnez Configurer votre compte. Vous pouvez vous inscrire soit avec votre IBM compte existant, soit en créant un IBM compte gratuit.

Obtenir une offre privée

Vous pouvez demander une offre AWS Marketplace privée pour Db2 auprès deIBM. Pour plus d'informations, consultez la section Offres privées dans le Guide de AWS Marketplace l'acheteur.

> **Note**
>
> Si vous êtes un AWS Organizations utilisateur et que vous avez reçu une offre privée émise sur vos comptes payeur et membre, suivez la procédure ci-dessous pour vous abonner à Db2 directement sur chaque compte de votre organisation.

Pour obtenir une offre privée Db2

1. Une fois qu'une offre privée a été émise, connectez-vous à la AWS Marketplace console.

2. Ouvrez l'e-mail contenant un lien d'offre privée DB2.
3. Suivez le lien pour accéder directement à l'offre privée.

> **Note**
>
> Si vous suivez ce lien avant de vous connecter au bon compte, une erreur « Page note found » (404) s'affichera.

4. Passez en revue les termes et conditions.
5. Choisissez Accepter les conditions.

> **Note**
>
> Si une offre AWS Marketplace privée n'est pas acceptée, les frais de service Db2 AWS Marketplace continueront d'être facturés au taux horaire public.

6. Pour vérifier les détails de l'offre, sélectionnez Afficher les détails dans la liste des produits.

Une fois la procédure terminée, vous pouvez créer votre instance de base de données en suivant les étapes décrites dansthe section called “Création d'une instance de base de données”. Dans la AWS Management Console section Licence, assurez-vous de sélectionner Par AWS Marketplace.

Basculer entre les licences DB2

Vous pouvez passer d'une licence Db2 à une autre dans RDS for Db2. Par exemple, vous pouvez commencer par apporter votre propre licence (BYOL), puis passer à la licence DB2. AWS Marketplace

> **Important**
>
> Si vous souhaitez passer à la licence DB2 AWS Marketplace, assurez-vous d'avoir un AWS Marketplace abonnement actif pour l'IBM Db2édition que vous souhaitez utiliser. Si ce n'est pas le cas, abonnez-vous d'abord à Db2 AWS Marketplace pour cette édition Db2, puis terminez la procédure de restauration.

console

Pour passer d'une licence DB2 à une autre

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Automated backups (Sauvegardes automatisées).

   Les sauvegardes automatisées sont affichées dans l'onglet Current Region (Région actuelle).
3. Choisissez l'instance de base de données que vous souhaitez restaurer.
4. Sous Actions, sélectionnez Restaurer à un moment donné.

   La fenêtre Restaurer à un instant dans le passé s'affiche.
5. Choisissez Dernière heure de restauration possible pour restaurer à la dernière heure possible, ou choisissez Personnalisé pour choisir une heure.

   Si vous avez choisi Personnalisé, entrez la date et l'heure auxquelles vous souhaitez restaurer l'instance.

   > **Note**
   >
   > Les heures sont exprimées dans votre fuseau horaire local, qui est indiqué par son décalage par rapport à l'heure UTC. Par exemple, UTC-5 correspond à l' Time/Central heure avancée normale de l'Est.
6. Pour le moteur de base de données, choisissez la licence DB2 que vous souhaitez utiliser.
7. Pour Identifiant d'instance de base de données, entrez le nom de l'instance de base de données restaurée. Le nom doit être unique.
8. Choisissez d'autres options selon vos besoins, telles que la classe d'instance de base de données et le stockage, ou le fait que vous voulez utiliser la mise à l'échelle automatique du stockage.

   Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.
9. Choisissez Restaurer à un instant dans le passé.

Pour de plus amples informations, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

AWS CLI

Pour passer d'une licence DB2 à une autre, exécutez la point-in-time commande restore-db-instance-to-. L'exemple suivant restaure la dernière point-in-time version, définit le moteur de base de données sur IBM Db2 Advanced Edition et définit le modèle de licence sur la licence DB2 via AWS Marketplace.

Vous pouvez spécifier d'autres paramètres. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds restore-db-instance-to-point-in-time \
    --source-db-instance-identifier my_source_db_instance \
    --target-db-instance-identifier my_target_db_instance \
    --use-latest-restorable-time \
    --engine db2-ae \
    --license-model marketplace-license
```

Dans Windows :

```
aws rds restore-db-instance-to-point-in-time ^
    --source-db-instance-identifier my_source_db_instance ^
    --target-db-instance-identifier my_target_db_instance ^
    --use-latest-restorable-time ^
    --engine db2-ae ^
    --license-model marketplace-license
```

Pour de plus amples informations, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

API RDS

Pour passer d'une licence Db2 à une autre, appelez l'`RestoreDBInstanceToPointInTime`opération d'API Amazon RDS avec les paramètres suivants :

- `SourceDBInstanceIdentifier`
- `TargetDBInstanceIdentifier`
- `RestoreTime`
- `Engine`
- `LicenseModel`

Pour de plus amples informations, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Amazon RDS pour les classes d'instance DB2

Les capacités de calcul et de mémoire d'une instance de base de données sont déterminées par sa classe d'instance. La classe d'instance de bases de données dont vous avez besoin varie selon vos exigences en mémoire et en puissance de traitement.

RDS pris en charge pour les classes d'instance DB2

Les classes d'instance Amazon RDS pour DB2 prises en charge constituent un sous-ensemble des classes d'instances de base de données Amazon RDS. Pour obtenir la liste complète des classes d'instances Amazon RDS, consultezClasses d'instances de base de données .

Rubriques

- RDS pris en charge pour les classes d'instance DB2 pour Db2 Standard Edition
- RDS pris en charge pour les classes d'instance DB2 pour Db2 Advanced Edition

RDS pris en charge pour les classes d'instance DB2 pour Db2 Standard Edition

Le tableau suivant répertorie toutes les classes d'instances prises en charge pour la version Db2 Standard Edition 11.5.9.0 de la base de données DB2. Ces classes d'instances sont disponibles à la fois pour les licences Bring Your Own License (BYOL) et Db2. AWS Marketplace

Type de classe d'instance	Classe d'instance
Classes d'instances à usage général avec Intel Xeon Scalable processeurs de 3e génération, stockage SSD et optimisation du réseau	db.m6idn.large — db.m6idn.8xlarge

Type de classe d'instance	Classe d'instance
Classes d'instance à usage général alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.m6 en largeur — db.m6 en 8 x large
Classes d'instance à usage général	db.m7i.large — db.m7i.8xlarge db.m6i.large — db.m6i.8xlarge
Classes d'instances optimisées pour la mémoire avec une NVMe base locale SSDs, alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.x2iedn.xlarge
Classes d'instances optimisées pour la mémoire alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.r6idn.large — db.r6idn. 4xlarge db.r6inlarge—db.r6in4xlarge
Classes d'instances à mémoire optimisée	db.r7i.large — db.r7i.8xlarge db.r6i.large–db.r6i.4xlarge
Classes d'instance à capacité extensible	db.t3.small–db.t3.2xlarge

RDS pris en charge pour les classes d'instance DB2 pour Db2 Advanced Edition

Le tableau suivant répertorie toutes les classes d'instances prises en charge pour la version Db2 Advanced Edition 11.5.9.0 de la base de données DB2. Ces classes d'instances sont disponibles à la fois pour les licences Bring Your Own License (BYOL) et Db2. AWS Marketplace

Type de classe d'instance	Classe d'instance
Classes d'instances à usage général avec Intel Xeon Scalable processeurs de 3e génération, stockage SSD et optimisation du réseau	db.m6idn.large — db.m6idn. 32xlarge
Classes d'instance à usage général alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.m6 en largeur — db.m6 en 32 x large

Type de classe d'instance	Classe d'instance
Classes d'instance à usage général	db.m6i.large — db.m7i.48 x large db.m7i.large — db.m7i.48 x large
Classes d'instances optimisées pour la mémoire avec une NVMe base locale SSDs, alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.x2iedn.xlarge–db.x2iedn. 32xlarge
Classes d'instances optimisées pour la mémoire alimentées par des Intel Xeon Scalable processeurs de 3e génération	db.r6idn.large — db.r6idn. 32xlarge db.r 6 pouces 8 x large—db 6 pouces 32 x large
Classes d'instances à mémoire optimisée	db.r6i.large — db.r7i.48xlarge db.r7i.large — db.r7i.48xlarge
Classes d'instance à capacité extensible	db.t3.small–db.t3.2xlarge

Rôles par défaut RDS d'Amazon pour DB2

RDSfor Db2 ajoute les six rôles suivants et les attribue `master_user_role` à l'`ADMIN`option. Lorsque la base de données est provisionnée, RDS les autorisations Db2 sont accordées `master_user_role` à l'utilisateur principal. L'utilisateur principal peut à son tour accorder ces rôles à d'autres utilisateurs, groupes ou rôles dotés d'`GRANT`instructions natives en se connectant à la base de données.

- DBA— RDS for Db2 crée ce rôle vide avec `DATAACCESS` autorisation. L'utilisateur principal peut ajouter d'autres autorisations ou privilèges à ce rôle, puis accorder le rôle à d'autres utilisateurs, groupes ou rôles.
- DBA_ RESTRICTED — RDS for Db2 crée ce rôle vide. L'utilisateur principal peut ajouter des privilèges à ce rôle, puis l'octroyer à d'autres utilisateurs, groupes ou rôles.

- DEVELOPER— RDS for Db2 crée ce rôle vide avec `DATAACCESS` autorisation. L'utilisateur principal peut ajouter d'autres autorisations ou privilèges à ce rôle, puis accorder le rôle à d'autres utilisateurs, groupes ou rôles.
- ROLE_ NULLID _ PACKAGES — RDS car Db2 accorde des `EXECUTE` privilèges à ce rôle sur les `ALL NULLID` packages qui étaient liés par Db2 lors `CREATE DATABASE` de son exécution.
- ROLE_ PROCEDURES — RDS car Db2 accorde des `EXECUTE` privilèges à ce rôle sur toutes les `SYSIBM` procédures.
- ROLE_ TABLESPACES — RDS car DB2 accorde des `USAGE` privilèges sur les tablespaces créés par la commande. `CREATE DATABASE`

Paramètres Amazon RDS pour DB2

Amazon RDS pour Db2 utilise trois types de paramètres : les paramètres de configuration du gestionnaire de base de données, les variables de registre et les paramètres de configuration de base de données. Vous pouvez gérer les deux premiers types par le biais de groupes de paramètres et le dernier par le biais de la procédure rdsadmin.update_db_param stockée.

Par défaut, une instance de base de données RDS pour DB2 utilise un groupe de paramètres de base de données spécifique à une base de données DB2 et à une instance de base de données. Ce groupe de paramètres contient les paramètres du moteur de IBM Db2 base de données, en particulier les paramètres de configuration du gestionnaire de base de données et les variables de registre. Pour plus d'informations sur l'utilisation de groupes de paramètres, consultezGroupes de paramètres pour (Amazon RDS).

Les paramètres de configuration de base de données RDS pour DB2 sont définis sur les valeurs par défaut du moteur de stockage que vous avez sélectionné. Pour plus d'informations sur les paramètres Db2, consultez les paramètres de configuration de la base de données DB2 dans la IBM Db2 documentation.

Rubriques

- Affichage des paramètres dans des groupes de paramètres
- Affichage de tous les paramètres avec les commandes DB2
- Modification des paramètres dans les groupes de paramètres
- Modification des paramètres de configuration de la base de données à l'aide des commandes DB2

Affichage des paramètres dans des groupes de paramètres

Les paramètres de configuration du gestionnaire de base de données et les variables de registre sont définis dans des groupes de paramètres. Vous pouvez consulter les paramètres de configuration du gestionnaire de base de données et les variables de registre pour une version spécifique de DB2 à l'aide de l' AWS Management Console AWS CLI API, de ou de l'API RDS.

console

Pour afficher les valeurs de paramètres pour un groupe de paramètres DB

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Groupes de paramètres.

   Les groupes de paramètres DB s'affichent dans une liste.
3. Choisissez le nom du groupe de paramètres pour consulter la liste des paramètres associée.

AWS CLI

Vous pouvez afficher les paramètres de configuration du gestionnaire de base de données et les variables de registre pour une version DB2 en exécutant la describe-engine-default-parameterscommande. Spécifiez l'une des valeurs suivantes pour l'option `--db-parameter-group-family` :

- `db2-ae-11.5`
- `db2-se-11.5`

Par exemple, pour afficher les paramètres de la version Db2 Standard Edition 11.5, exécutez la commande suivante :

```
aws rds describe-engine-default-parameters --db-parameter-group-family db2-se-11.5
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
{
    "EngineDefaults": {
        "Parameters": [
```

```
            {
                "ParameterName": "agent_stack_sz",
                "ParameterValue": "1024",
                "Description": "You can use this parameter to determine the amount of
 memory that is allocated by Db2 for each agent thread stack.",
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "integer",
                "AllowedValues": "256-32768",
                "IsModifiable": false
            },
            {
                "ParameterName": "agentpri",
                "ParameterValue": "-1",
                "Description": "This parameter controls the priority given to all
 agents and to other database manager instance processes and threads by the operating
 system scheduler. This priority determines how CPU time is allocated to the database
 manager processes, agents, and threads relative to other processes and threads running
 on the machine.",
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "integer",
                "AllowedValues": "1-99",
                "IsModifiable": false
            },
            ...
        ]
    }
}
```

Pour répertorier uniquement les paramètres modifiables pour la version Db2 Standard Edition 11.5, exécutez la commande suivante :

Pour LinuxmacOS, ou Unix :

```
aws rds describe-engine-default-parameters \
    --db-parameter-group-family db2-se-11.5 \
    --query 'EngineDefaults.Parameters[?IsModifiable==`true`].
{ParameterName:ParameterName, DefaultValue:ParameterValue}'
```

Dans Windows :

```
aws rds describe-engine-default-parameters ^
```

```
    --db-parameter-group-family db2-se-11.5 ^
    --query 'EngineDefaults.Parameters[?IsModifiable==`true`].
{ParameterName:ParameterName, DefaultValue:ParameterValue}'
```

API RDS

Pour afficher les valeurs des paramètres d'un groupe de paramètres de base de données, utilisez l'`DescribeDBParameters`opération avec le paramètre obligatoire suivant.

- `DBParameterGroupName`

Affichage de tous les paramètres avec les commandes DB2

Vous pouvez consulter les paramètres de configuration du gestionnaire de base de données, les paramètres de configuration de base de données et les variables de registre à l'aide des commandes DB2.

Pour consulter les paramètres

1. Connectez-vous à votre base de données DB2. Dans l'exemple suivant, remplacez *database_namemaster_username*, et *master_password* par vos informations.

   ```
   db2 "connect to database_name user master_username using master_password"
   ```

2. Trouvez la version Db2 prise en charge.

   ```
   db2 "select service_level, fixpack_num from table(sysproc.env_get_inst_info()) as
    instanceinfo"
   ```

3. Affichez les paramètres d'une version Db2 spécifique.

   - Consultez les paramètres de configuration du gestionnaire de base de données en exécutant la commande suivante :

     ```
     db2 "select cast(substr(name,1,24) as varchar(24)) as name, case
         when value_flags = 'NONE' then '' else value_flags end flags,
         cast(substr(value,1,64) as varchar(64)) as current_value
         from sysibmadm.dbmcfg
         order by name asc with UR"
     ```

- Affichez tous les paramètres de configuration de votre base de données en exécutant la commande suivante :

```
db2 "select cast(substr(name,1,24) as varchar(24)) as name, case
    when value_flags = 'NONE' then '' else value_flags end flags,
    cast(substr(value,1,64) as varchar(64)) as current_value
    from table(db_get_cfg(null)) order by name asc, member asc with UR"
```

- Consultez les variables de registre actuellement définies en exécutant la commande suivante :

```
db2 "select cast(substr(reg_var_name,1,50) as varchar(50)) as reg_var_name,
    cast(substr(reg_var_value,1,50) as varchar(50)) as reg_var_value,
    level from table(env_get_reg_variables(null))
    order by reg_var_name,member with UR"
```

Modification des paramètres dans les groupes de paramètres

Vous pouvez modifier les paramètres de configuration du gestionnaire de base de données et les variables de registre dans les groupes de paramètres personnalisés à l' AWS Management Console aide de l'API AWS CLI, de ou de l'API RDS. Pour de plus amples informations, veuillez consulter Groupes de paramètres de base de données pour les instances de base de données RDSAmazon.

console

Pour modifier les paramètres de configuration et les variables de registre du gestionnaire de base de données

1. Créez un groupe de paramètres personnalisé. Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez les paramètres de ce groupe de paramètres personnalisés. Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

AWS CLI

Pour modifier les paramètres de configuration et les variables de registre du gestionnaire de base de données

1. Créez un groupe de paramètres personnalisé en exécutant la create-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Nom du groupe de paramètres que vous créez.
   - `--db-parameter-group-family`— L'édition et la version majeure du moteur DB2. Valeurs valides : `db2-se-11.5`, `db2-ae-11.5`.
   - `--description`— Description de ce groupe de paramètres.

   Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez les paramètres du groupe de paramètres personnalisés que vous avez créé en exécutant la modify-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Le nom du groupe de paramètres que vous avez créé.
   - `--parameters`— Tableau de noms de paramètres, de valeurs et de méthodes d'application pour la mise à jour des paramètres.

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

API RDS

Pour modifier les paramètres de configuration et les variables de registre du gestionnaire de base de données

1. Créez un groupe de paramètres de base de données personnalisé à l'aide de l'CreateDBParameterGroupopération.

Incluez les paramètres requis suivants :

- `DBParameterGroupName`
- `DBParameterGroupFamily`
- `Description`

Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.

2. Modifiez les paramètres du groupe de paramètres personnalisés que vous avez créé à l'aide de l'ModifyDBParameterGroupopération.

   Incluez les paramètres requis suivants :

   - `DBParameterGroupName`
   - `Parameters`

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Modification des paramètres de configuration de la base de données à l'aide des commandes DB2

Vous pouvez modifier les paramètres de configuration de la base de données à l'aide des commandes DB2.

Pour modifier les paramètres de configuration de la base de données

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Modifiez les paramètres de configuration de la base de données en appelant la procédure `rdsadmin.update_db_param` stockée. Pour de plus amples informations, veuillez consulter rdsadmin.update_db_param.

```
db2 "call rdsadmin.update_db_param(
    'database_name',
    'parameter_to_modify',
    'changed_value',
    'restart_database')"
```

EBCDICclassement pour les bases de données DB2 sur Amazon RDS

Amazon RDS pour Db2 prend en charge le EBCDIC classement pour les bases de données DB2. Vous ne pouvez spécifier une séquence de EBCDIC classement pour une base de données que lorsque vous créez la base de données à l'aide de la procédure RDS the section called “rdsadmin.create_database” stockée Amazon.

Lorsque vous créez une instance de base de données RDS pour DB2 à l'aide du AWS Management Console, ou AWS CLI RDSAPI, vous pouvez spécifier un nom de base de données. Si vous spécifiez un nom de base de données, Amazon RDS crée une base de données avec le classement par défaut de`SYSTEM`. Si vous devez créer une base de données avec EBCDIC classement, ne spécifiez pas de nom de base de données lorsque vous créez une instance de base de données.

Le classement d'une base de données dans RDS Db2 est défini au moment de la création et est immuable.

Pour créer une base de données DB2 avec classement EBCDIC

1. Si vous n'avez pas d'instance de base de données RDS pour DB2, créez-en une mais ne spécifiez pas de nom de base de données. Vous pouvez créer une instance de base de données en utilisant le AWS Management Console AWS CLI, ou RDSAPI. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données.
2. Créez une base de données DB2 et définissez l'option de classement sur une EBCDIC valeur en appelant la procédure `rdsadmin.create_database` stockée. Pour de plus amples informations, veuillez consulter rdsadmin.create_database.

> **Important**
>
> Après avoir créé une base de données à l'aide de la procédure stockée, vous ne pouvez pas modifier la séquence de classement. Si vous souhaitez qu'une base de données utilise une séquence de classement différente, supprimez-la en appelant la procédure

the section called “rdsadmin.drop_database” stockée. Créez ensuite une base de données avec la séquence de classement requise.

Fuseau horaire local pour Amazon RDS pour les instances de base de données DB2

Le fuseau horaire d'une instance de base de données Amazon RDS exécutant Db2 est défini par défaut. La valeur par défaut est UTC (temps universel coordonné). Pour correspondre au fuseau horaire de vos applications, vous pouvez définir le fuseau horaire de votre instance de base de données sur un fuseau horaire local.

Vous définissez le fuseau horaire lorsque vous créez votre instance de base de données. Vous pouvez créer votre instance de base de données à l' AWS Management Console aide de l'API RDS ou du AWS CLI. Pour plus d’informations, consultez Création d'une instance de base de données.

Si votre instance de base de données fait partie d'un déploiement multi-AZ, en cas de basculement, son fuseau horaire reste le fuseau horaire local que vous avez défini.

Vous pouvez restaurer votre instance de base de données à un moment que vous spécifiez. L'heure apparaît dans votre fuseau horaire local. Pour plus d’informations, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

La définition du fuseau horaire local sur votre instance de base de données présente les limites suivantes :

- Vous ne pouvez pas modifier le fuseau horaire d'une instance de base de données Amazon RDS pour DB2 existante.
- Vous ne pouvez pas restaurer un instantané à partir d'une instance de base de données dans un fuseau horaire dans une instance de base de données d'un autre fuseau horaire.
- Nous vous recommandons vivement de ne pas restaurer de fichier de sauvegarde d'un fuseau horaire dans un autre fuseau horaire. Si vous restaurez un fichier de sauvegarde d'un fuseau horaire à un autre, vous devez auditer vos requêtes et vos applications pour détecter les effets du changement de fuseau horaire.

Fuseaux horaires disponibles

Vous pouvez utiliser les valeurs suivantes pour le réglage du fuseau horaire.

disponibilité	Fuseau horaire
Afrique	Afrique/le Caire, Afrique/Casablanca, Afrique/Harare, Afrique/Lagos, Afrique/Luanda, Afrique/Monrovia, Afrique/Nairobi, Afrique/Tripoli, Afrique/Windhoek
Amérique	Amérique/Araguaina, Amérique/Argentine/Buenos_Aires, Amérique/Asuncion, Amérique/Bogota, Amérique/Caracas, Amérique/Chicago, Amérique/Chihuahua, Amérique/Cuiaba, Amérique/Denver, Amérique/Detroit, Amérique/Fortaleza, Amérique/Godthab, Amérique/Guatemala, Amérique/Halifax, Amérique/Lima, Amérique/Los_Angeles, Amérique/Manaus, Amérique/Matamoros, Amérique/Mexico_City, Amérique/Monterrey, Amérique/Montevideo, Amérique/New_York, Amérique/Phoenix, AmériquI/Oantiago, AmériquI/Oao_Paulo, Amérique/Tijuana, Amérique/Toronto
Asie	Asie/Amman, Asie/Achgabat, Asie/Bagdad, Asie/Bakou, Asie/Bangkok, Asie/Beyrouth, Asie/Calcutta, Asie/Damas, Asie/Dhaka, Asie/Hong_Kong, Asie/Irkoutsk, Asie/Jakarta, Asie/Jérusalem, Asie/Kaboul, Asie/Karachi, Asie/Katmandou, Asie/Kolkata, Asie/Krasnoïarsk, Asie/Magadan, Asie/Manille, Asie/Muscat, Asie/Novosibirsk, Asie/Rangoon, Asie/Riyad, AsiI/Oéoul, AsiI/Ohanghai, AsiI/Oingapour, Asie/Taipei, Asie/Téhéran, Asie/Tokyo, Asie/Oulan_Bator, Asie/Vladivostok, Asie/Iakoutsk, Asie/Yerevan
Atlantique	Atlantique/Açores, Atlantic/Cap_Vert
Australie	Australie/Adelaide, Australie/Brisbane, Australie/Darwin, Australie/Eucla, Australie/Hobart, Australie/Lord_Howe, Australie/Perth, AustraliI/Oydney
Brésil	Brésil/, Brésil/Est DeNoronha
Canada	Canada/Terre-Neuve, Canada/Saskatchewan
Etc	Etc/GMT-3
Europe	Europe/Amsterdam, Europe/Athènes, Europe/Berlin, Europe/Dublin, Europe/Helsinki, Europe/Kaliningrad, Europe/Londres, Europe/Madrid, Europe/Moscou, Europe/Paris, Europe/Prague, Europe/Rome, Europe/Sarajevo, Europe/Stockholm

disponibilité	Fuseau horaire
Pacifique	Pacifique/Apia, Pacifique/Auckland, Pacifique/Chatham, Pacifique/Fidji, Pacifique/Guam, Pacifique/Honolulu, Pacifique/Kiritimati, Pacifique/Marquises, PacifiquI/Oamoa, Pacifique/Tongatapu, Pacifique/Wake
ETATS-UNIS	États-Unis/Alaska, États-Unis/Centre, États-Unis/Indiana Est, États-Unis/Est, États-Unis/Pacifique
UTC	UTC

Conditions préalables à la création d'une instance de base de données Amazon RDS pour DB2

Les éléments suivants sont des conditions préalables à la création d'une instance de base de données.

Rubriques

- Compte administrateur
- Considérations supplémentaires

Compte administrateur

Lorsque vous créez une instance de base de données, vous devez désigner un compte administrateur pour l'instance. Amazon RDS accorde `DBADM` l'autorité à ce compte d'administrateur de base de données local.

Le compte administrateur présente les caractéristiques, capacités et limites suivantes :

- Il s'agit d'un utilisateur local et non d'un Compte AWS.
- Ne dispose pas d'autorités au niveau de l'instance DB2 telles que`SYSADM`, `SYSMAINT` ou. `SYSCTRL`
- Impossible d'arrêter ou de démarrer une instance DB2.
- Impossible de supprimer une base de données DB2 si vous avez spécifié le nom lors de la création de l'instance de base de données.
- Dispose d'un accès complet à la base de données DB2, y compris les tables de catalogue et les vues.
- Peut créer des utilisateurs et des groupes locaux à l'aide des procédures stockées Amazon RDS.
- Peut accorder et révoquer des pouvoirs et des privilèges.

Le compte administrateur peut effectuer les tâches suivantes :

- Créez, modifiez ou supprimez des instances de base de données.
- Créez des instantanés de base de données.
- Lancez les point-in-time restaurations.

- Créez des sauvegardes automatisées des instantanés de base de données.
- Créez des sauvegardes manuelles des instantanés de base de données.
- Utilisez les autres fonctionnalités d'Amazon RDS.

Considérations supplémentaires

Avant de créer une instance de base de données, prenez en compte les éléments suivants :

- Chaque instance de base de données Amazon RDS pour DB2 peut héberger jusqu'à 50 bases de données DB2. Pour de plus amples informations, veuillez consulter Plusieurs bases de données sur une instance de base de données Amazon RDS pour DB2.
- Nom de la base de données initiale
  - Si vous ne fournissez pas de nom de base de données lorsque vous créez une instance de base de données, Amazon RDS ne crée pas de base de données.
  - Ne fournissez pas de nom de base de données dans les cas suivants :
    - Vous souhaitez modifier le `db2_compatibility_vector` paramètre. Pour de plus amples informations, veuillez consulter Définition du paramètre db2_compatibility_vector.
- Dans le modèle BYOL (Bring your own license), vous devez d'abord créer un groupe de paramètres personnalisé contenant votre IBM Customer ID et votreIBM Site ID. Pour de plus amples informations, veuillez consulter Apportez votre propre licence (BYOL) pour Db2.
- Dans le AWS Marketplace modèle de licence DB2, vous avez besoin d'un AWS Marketplace abonnement actif pour l'IBM Db2édition particulière que vous souhaitez utiliser. Si vous n'en avez pas déjà un, abonnez-vous à Db2 AWS Marketplace pour l'IBM Db2édition que vous souhaitez utiliser. Pour de plus amples informations, veuillez consulter Licence DB2 via AWS Marketplace.

Plusieurs bases de données sur une instance de base de données Amazon RDS pour DB2

Vous pouvez créer plusieurs bases de données sur une seule instance de base RDS de données Db2 en appelant la procédure stockée rdsadmin.create_database. Une seule instance RDS de base de données Db2 est limitée à 50 bases de données. Ce nombre inclut les bases de données activées et désactivées.

> **Note**
>
> Si vous créez plusieurs bases de données sur une instance de base de données RDS for DB2 créée avant le 15 novembre 2024, vous devez redémarrer l'instance de base de données pour activer la prise en charge de plusieurs bases de données.

Par défaut, Amazon RDS active les bases de données lorsque vous les créez. Pour optimiser les ressources de mémoire, vous pouvez désactiver les bases de données que vous utilisez rarement, puis les activer ultérieurement en cas de besoin. Pour plus d'informations, consultez the section called “Désactivation d'une base de données” et the section called “Activation d'une base de données”.

Le nombre de bases de données activées sur une instance de base de données dépend des ressources de mémoire disponibles sur le serveur. Les ressources de mémoire varient en fonction de la classe d'instance de base de données et de la quantité de mémoire configurée pour la base de données. Pour plus d'informations sur les classes d'instances de base de données, consultez the section called “Classes d'instances de base de données”. Pour plus d'informations sur la mise à jour de la mémoire d'une base de données RDS pour DB2, consultezthe section called “rdsadmin.update_db_param”.

Nous vous recommandons de choisir une classe d'instance de base de données dotée de 2 Go de mémoire pour les tâches de base de données courantes, les exigences du système d'exploitation et les autres tâches d'RDSautomatisation Amazon telles que les sauvegardes. Pour plus d'informations sur la modification de la classe d'instance de base de données, consultezthe section called “Modification d'une instance de base de données”.

En outre, IBM recommande un minimum de 1 Go de mémoire pour chaque base de données active. Pour plus d'informations, consultez la section Exigences relatives au disque et à la mémoire dans la IBM documentation.

Vous pouvez calculer le nombre maximum de bases de données actives qu'une instance de base de données peut avoir à l'aide de la formule suivante :

```
Active database limit = (total server memory - 2 GB) / 1 GB
```

L'exemple suivant montre le nombre maximal de bases de données actives pour une instance de base de données avec une classe d'instance de base de données db.m6i.xlarge :

```
Active database limit = (total server memory - 2 GB) / 1 GB
                      = (16 GB - 2 GB) / 1 GB
                      = 14 databases
```

Lorsqu'Amazon RDS récupère une base de données après un crash, il active la base de données si elle était déjà active. Dans certains cas, par exemple lorsque vous modifiez une classe d'instance de base de données pour une configuration de mémoire inférieure, la mémoire peut être insuffisante pour activer toutes les bases de données de l'instance de base de données. Dans ces cas, Amazon RDS active les bases de données dans l'ordre dans lequel elles ont été créées.

> **Note**
>
> Toutes les bases de données qu'Amazon ne RDS peut pas activer en raison d'une mémoire insuffisante restent désactivées.

Connexion à votre instance de base de données DB2

Une fois qu'Amazon RDS a approvisionné votre instance de base de données Amazon RDS pour DB2, vous pouvez utiliser n'importe quelle application client SQL standard pour vous connecter à l'instance de base de données. Amazon RDS étant un service géré, vous ne pouvez pas vous connecter en tant que`SYSADM`, `SYSCTRLSECADM`, ou`SYSMAINT`.

Vous pouvez vous connecter à une instance de base de données qui exécute le IBM Db2 moteur de base de données en utilisant IBM Db2 CLP, IBM CLPPlus, DBeaver, ou IBM Db2 Data Management Console.

> **Note**
>
> La connexion à une base de données DB2 peut échouer si votre instance de base de données RDS pour DB2 ne dispose pas de suffisamment de mémoire. Pour de plus amples informations, veuillez consulter the section called “Erreur de connexion de base de données”.

Rubriques

- Trouver le point de terminaison de votre instance de base de données Amazon RDS pour DB2
- Connexion à votre instance de base de données Amazon RDS pour DB2 avec IBM Db2 CLP
- Connexion à votre instance de base de données Amazon RDS for Db2 avec IBM CLPPlus
- Connexion à votre instance de base de données Amazon RDS pour DB2 avec DBeaver
- Connexion à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console
- Considérations relatives aux groupes de sécurité avec Amazon RDS pour DB2

Trouver le point de terminaison de votre instance de base de données Amazon RDS pour DB2

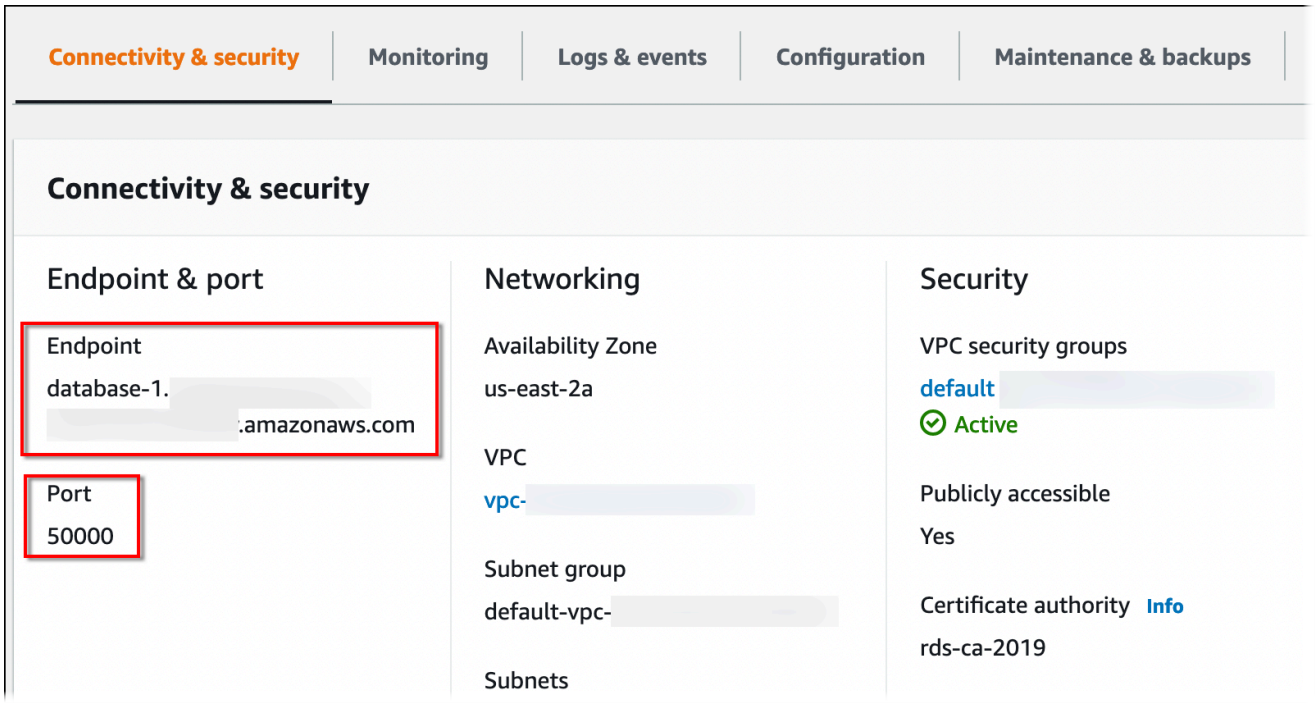
Chaque instance de base de données Amazon RDS possède un point de terminaison. Chaque point de terminaison possède le nom DNS et le numéro de port de l'instance de base de données. Pour vous connecter à votre instance de base de données Amazon RDS pour DB2 avec une application cliente SQL, vous avez besoin du nom DNS et du numéro de port de votre instance de base de données.

Vous pouvez trouver le point de terminaison d'une instance de base de données en utilisant le AWS Management Console ou le AWS CLI.

Console

Pour trouver le point de terminaison d'une instance de base de données RDS pour DB2

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console, choisissez l'instance de base de données Région AWS de votre instance.
3. Trouvez le nom DNS et le numéro de port de votre instance de base de données RDS pour DB2.
   a. Choisissez Bases de données pour afficher une liste de vos instances de bases de données.
   b. Choisissez le nom de l'instance de base de données RDS pour DB2 pour afficher les détails de l'instance.
   c. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



AWS CLI

Pour trouver le point de terminaison d'une instance de base de données RDS pour DB2, exécutez la describe-db-instancescommande. Dans l'exemple suivant, remplacez *database-1* par le nom de votre instance de base de données.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-db-instances \
    --db-instance-identifier database-1 \
    --query 'DBInstances[].
{DBInstanceIdentifier:DBInstanceIdentifier,DBName:DBName,Endpoint:Endpoint}' \
    --output json
```

Dans Windows :

```
aws rds describe-db-instances ^
    --db-instance-identifier database-1 ^
    --query 'DBInstances[].
{DBInstanceIdentifier:DBInstanceIdentifier,DBName:DBName,Endpoint:Endpoint}' ^
    --output json
```

Cette commande produit une sortie similaire à celle de l'exemple suivant. La ligne `Address` de la sortie contient le nom DNS.

```
[
    {
        "DBInstanceIdentifier": "database-1",
        "DBName": "DB2DB",
        "Endpoint": {
            "Address": "database-1.123456789012.us-east-2.amazonaws.com",
            "Port": 50000,
            "HostedZoneId": "Z2OC4A7DETW6VH"
        }
    }
]
```

Connexion à votre instance de base de données Amazon RDS pour DB2 avec IBM Db2 CLP

Vous pouvez utiliser un utilitaire de ligne de commande tel que IBM Db2 CLP pour vous connecter à Amazon RDS pour les instances de base de données DB2. Cet utilitaire fait partie deIBM Data Server Runtime Client. Pour télécharger le client depuisIBM Fix Central, voir IBMData Server Client Packages version 11.5 Mod 8 Fix Pack 0 dans IBM Support.

Rubriques

- Terminologie
- Installation du client
- Connexion à une instance de base de données
- Résolution des problèmes de connexion à votre instance de base de données RDS pour DB2

Terminologie

Les termes suivants expliquent les commandes utilisées lors de la connexion à votre instance de base de données RDS pour DB2.

nœud TCP/IP du catalogue

Cette commande enregistre un nœud de base de données distant auprès d'un client Db2 local, ce qui rend le nœud accessible à l'application cliente. Pour cataloguer un nœud, vous devez fournir des informations telles que le nom d'hôte, le numéro de port et le protocole de communication du serveur. Le nœud catalogué représente ensuite un serveur cible sur lequel résident une ou plusieurs bases de données distantes. Pour plus d'informations, consultez CATALOG TCPIP/ TCPIP4/TCPIP6 NODEla section commande dans la IBM Db2 documentation.

base de données de catalogues

Cette commande enregistre une base de données distante auprès d'un client Db2 local, ce qui rend la base de données accessible à l'application cliente. Pour cataloguer une base de données, vous devez fournir des informations telles que l'alias de la base de données, le nœud sur lequel elle réside et le type d'authentification nécessaire pour se connecter à la base de données. Pour plus d'informations, consultez CATALOG DATABASEla section commande dans la IBM Db2 documentation.

Installation du client

Ensuitedownloading the package for Linux, installez le client en utilisant les privilèges root ou administrateur.

> **Note**
>
> Pour installer le client sur AIX ouWindows, suivez la même procédure mais modifiez les commandes de votre système d'exploitation.

Pour installer le client sur Linux

1. Exécutez **`./db2_install -f sysreq`**et choisissez **`yes`**d'accepter la licence.
2. Choisissez l'emplacement où installer le client.
3. Exécutez **`clientInstallDir/instance/db2icrt -s client`***`instance_name`*`.` Remplacez *`instance_name`* par un utilisateur de système d'exploitation valide activé. Linux DansLinux, le nom de l'instance de base de données DB2 est lié au nom d'utilisateur du système d'exploitation.

   Cette commande crée un **`sqllib`**répertoire dans le répertoire personnel de l'utilisateur désigné leLinux.

Connexion à une instance de base de données

Pour vous connecter à votre instance de base de données RDS pour DB2, vous avez besoin de son nom DNS et de son numéro de port. Pour plus d'informations sur leur recherche, consultezRecherche du point de terminaison. Vous devez également connaître le nom de base de données, le nom d'utilisateur principal et le mot de passe principal que vous avez définis lors de la création de votre instance de base de données RDS pour DB2. Pour plus d'informations sur leur recherche, consultezCréation d'une instance de base de données.

Pour vous connecter à une instance de base de données RDS pour DB2 avec IBM Db2 CLP

1. Connectez-vous avec le nom d'utilisateur que vous avez spécifié lors de l'installation du IBM Db2 CLP client.
2. Cataloguez votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`node_name`*, *`dns_name`* et *`port`* par le nom du nœud dans le catalogue local, le nom DNS de votre instance de base de données et le numéro de port.

   ```
   db2 catalog TCPIP node node_name remote dns_name server port
   ```

Exemple

```
db2 catalog TCPIP node remnode remote database-1.123456789012.us-
east-1.amazonaws.com server 50000
```

3. Cataloguez la `rdsadmin` base de données et votre base de données. Cela vous permettra de vous connecter à la `rdsadmin` base de données pour effectuer certaines tâches administratives à l'aide des procédures stockées Amazon RDS. Pour plus d'informations, consultez Administration de votre instance de base de données RDS pour DB2.

   Dans l'exemple suivant, remplacez *database_alias, node_name et database_name par des alias pour cette base de données, le nom* du nœud défini à l'étape précédente et le nom de votre base de données. `server_encrypt`chiffre votre nom d'utilisateur et votre mot de passe sur le réseau.

```
db2 catalog database rdsadmin [ as database_alias ] at node node_name
 authentication server_encrypt

db2 catalog database database_name [ as database_alias ] at node node_name
 authentication server_encrypt
```

   Exemple

```
db2 catalog database rdsadmin at node remnode authentication server_encrypt

db2 catalog database testdb as rdsdb2 at node remnode authentication server_encrypt
```

4. Connectez-vous à votre base de données RDS pour DB2. Dans l'exemple suivant, remplacez *rds_database_alias, master_username et master_password* par le nom de votre base de données, le nom d'utilisateur principal et le mot de *passe* principal de votre instance de base de données RDS pour DB2.

```
db2 connect to rds_database_alias user master_username using master_password
```

   Cette commande produit une sortie similaire à l'exemple suivant :

```
Database Connection Information

    Database server        = DB2/LINUXX8664 11.5.9.0
```

```
    SQL authorization ID   = ADMIN
    Local database alias   = TESTDB
```

5. Exécutez des requêtes et visualisez les résultats. L'exemple suivant montre une instruction SQL qui sélectionne la base de données que vous avez créée.

```
db2 "select current server from sysibm.dual"
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
1
    ------------------
    TESTDB

    1 record(s) selected.
```

Résolution des problèmes de connexion à votre instance de base de données RDS pour DB2

Si le message `NULLID` d'erreur suivant s'affiche, cela indique généralement que les versions de votre client et de RDS pour le serveur DB2 ne correspondent pas. Pour les versions du client DB2 prises en charge, consultez la section Combinaisons prises en charge de clients, de pilotes et de niveaux de serveur dans la IBM Db2 documentation.

```
db2 "select * from syscat.tables"
SQL0805N Package "NULLID.SQLC2O29 0X4141414141454A69" was not found.
SQLSTATE=51002
```

Après avoir reçu cette erreur, vous devez lier les packages de votre ancien client DB2 à une version de serveur DB2 prise en charge par RDS pour DB2.

Pour lier des packages d'un ancien client DB2 à un serveur DB2 plus récent

1. Localisez les fichiers de liaison sur l'ordinateur client. Ces fichiers se trouvent généralement dans le répertoire bnd du chemin d'installation du client DB2 et portent l'extension .bnd.
2. Connectez-vous au serveur DB2. Dans l'exemple suivant, remplacez *`database_name par le nom`* de votre serveur DB2. Remplacez *`master_username`* et *`master_password`* par vos informations. Cet utilisateur a `DBADM` autorité.

```
db2 connect to database_name user master_username using master_password
```

3. Exécutez la `bind` commande pour lier les packages.

   a. Accédez au répertoire où se trouvent les fichiers de liaison sur l'ordinateur client.

   b. Exécutez la `bind` commande pour chaque fichier.

      Les options suivantes sont requises :

      - `blocking all`— Lie tous les packages du fichier de liaison dans une seule demande de base de données.
      - `grant public`— Autorise `public` l'exécution du package.
      - `sqlerror continue`— Spécifie que le `bind` processus continue même en cas d'erreur.

      Pour plus d'informations sur la `bind` commande, voir BINDcommande dans la IBM Db2 documentation.

4. Vérifiez que la liaison a réussi en interrogeant la vue du `syscat.package` catalogue ou en vérifiant le message renvoyé après la `bind` commande.

Pour plus d'informations, consultez la liste des noms de fichiers et de packages de DB2 v11.5 dans Support. IBM

Connexion à votre instance de base de données Amazon RDS for Db2 avec IBM CLPPlus

Vous pouvez utiliser un utilitaire tel que IBM CLPPlus pour vous connecter à une instance de base de données Amazon RDS pour Db2. Cet utilitaire fait partie de IBM Data Server Runtime Client. Pour télécharger le client depuis IBM Fix Central, voir IBM Packages clients de serveur de données version 11.5, mod 8, Fix Pack 0 en IBM Support.

> **Important**
>
> Nous vous recommandons de courir IBM CLPPlus sur un système d'exploitation qui prend en charge les interfaces utilisateur graphiques telles que macOS, Windows, ou Linux avec Desktop. Si vous courez sans tête Linux, utilisez switch -nw avec CLPPlus commandes.

Rubriques

- Installation du client
- Connexion à une instance de base de données

Installation du client

Après avoir téléchargé le package pour Linux, installez le client.

> **Note**
>
> Pour installer le client sur AIX or Windows, suivez la même procédure mais modifiez les commandes de votre système d'exploitation.

Pour installer le client sur Linux

1. Exécutez **`./db2_install`**.
2. Exécutez **`clientInstallDir/instance/db2icrt -s client`** ***`instance_name`***. Remplacer *`instance_name`* avec un utilisateur de système d'exploitation valide activé Linux. Dans Linux, le nom de l'instance de base de données DB2 est lié au nom d'utilisateur du système d'exploitation.

   Cette commande crée un **`sqllib`**répertoire dans le répertoire personnel de l'utilisateur désigné sur Linux.

Connexion à une instance de base de données

Pour vous connecter à votre instance de base de données RDS for Db2, vous avez besoin de son DNS nom et de son numéro de port. Pour plus d'informations sur leur recherche, consultezRecherche du point de terminaison. Vous devez également connaître le nom de la base de données, le nom d'utilisateur principal et le mot de passe principal que vous avez définis lors de la création de votre instance de base de données RDS pour DB2. Pour plus d'informations sur leur recherche, consultezCréation d'une instance de base de données.

Pour vous connecter à une instance de base de données RDS pour DB2 avec IBM CLPPlus

1. Vérifiez la syntaxe de la commande. Dans l'exemple suivant, remplacez *`clientDir`* avec l'emplacement où le client est installé.

```
cd clientDir/bin
    ./clpplus -h
```

2. Configurez votre serveur DB2. Dans l'exemple suivant, remplacez *dsn_name*, *database_name*, *endpoint*, et *port* avec le DSN nom, le nom de la base de données, le point de terminaison et le port de votre instance de base de données RDS for DB2. Pour de plus amples informations, veuillez consulter Trouver le point de terminaison de votre instance de base de données Amazon RDS pour DB2.

```
db2cli writecfg add -dsn dsn_name -database database_name -host endpoint -port port
 -parameter "Authentication=SERVER_ENCRYPT"
```

3. Connectez-vous à votre instance de base de données RDS for DB2. Dans l'exemple suivant, remplacez *master_username* and *dsn_name* avec le nom d'utilisateur et le DSN nom du maître.

```
./clpplus -nw master_username@dsn_name
```

4. A Java Shell la fenêtre s'ouvre. Entrez le mot de passe principal pour votre instance de base de données RDS for Db2.

> **Note**
>
> Si un Java Shell la fenêtre ne s'ouvre pas, exécutez **`./clpplus -nw`** pour utiliser la même fenêtre de ligne de commande.

```
Enter password: *********
```

Une connexion est établie et produit une sortie similaire à l'exemple suivant :

```
Database Connection Information :
--------------------------------
Hostname = database-1.abcdefghij.us-east-1.rds.amazonaws.com
Database server = DB2/LINUXX8664  SQL110590
SQL authorization ID = admin
Local database alias = DB2DB
```

```
Port = 50000
```

5. Exécutez des requêtes et visualisez les résultats. L'exemple suivant montre une SQL instruction qui sélectionne la base de données que vous avez créée.

```
SQL > select current server from sysibm.dual;
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
1
    --------------------
    DB2DB
    SQL>
```

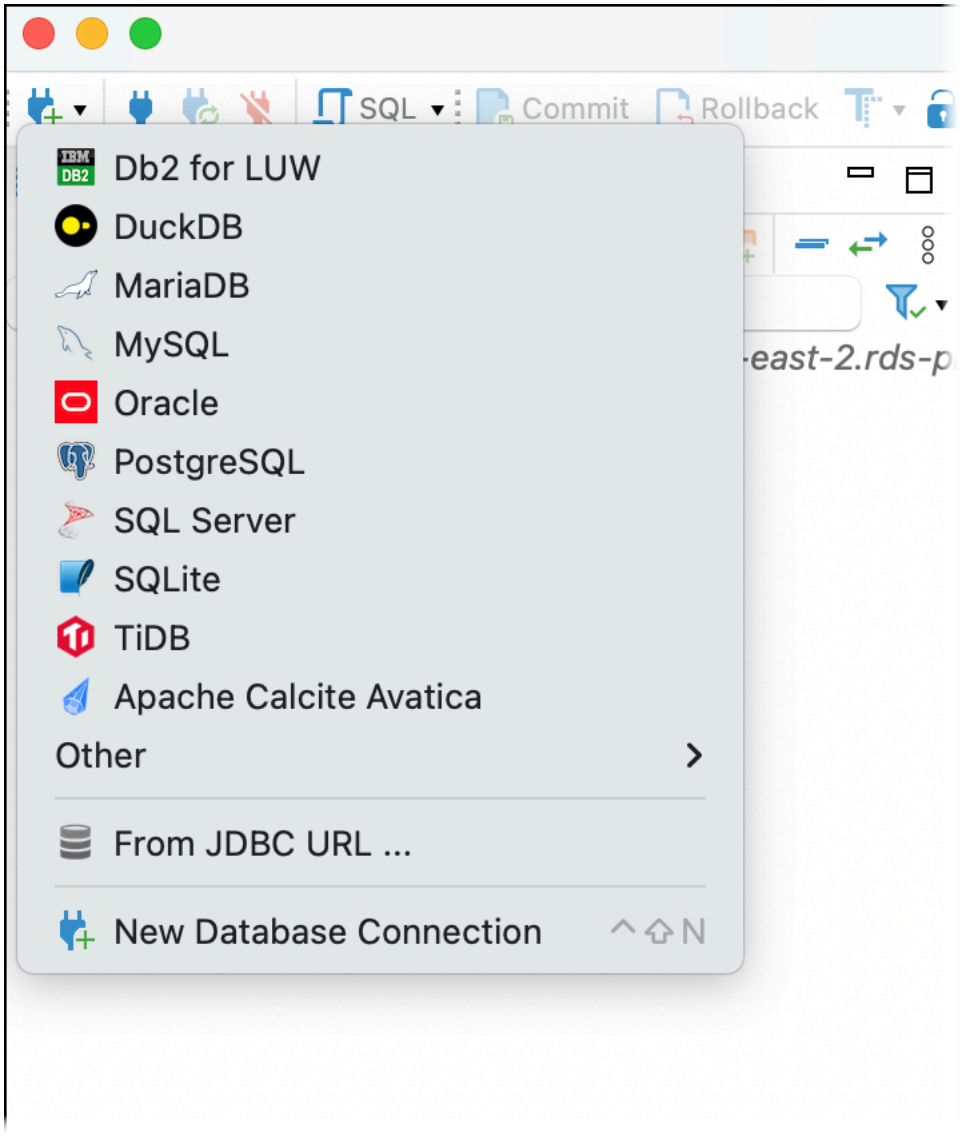
Connexion à votre instance de base de données Amazon RDS pour DB2 avec DBeaver

Vous pouvez utiliser des outils tiers tels que la connexion DBeaver à Amazon RDS pour les instances de base de données DB2. Pour télécharger cet utilitaire, consultez DBeaverla section Communauté.

Pour vous connecter à votre instance de base de données RDS pour DB2, vous avez besoin de son nom DNS et de son numéro de port. Pour plus d'informations sur leur recherche, consultezRecherche du point de terminaison. Vous devez également connaître le nom de la base de données, le nom d'utilisateur principal et le mot de passe principal que vous avez définis lors de la création de votre instance de base de données RDS pour DB2. Pour plus d'informations sur leur recherche, consultezCréation d'une instance de base de données.

Pour vous connecter à une instance de base de données RDS pour DB2 avec DBeaver

1. Démarrer DBeaver.
2. Cliquez sur l'icône Nouvelle connexion dans la barre d'outils, puis choisissez Db2 pour LUW.



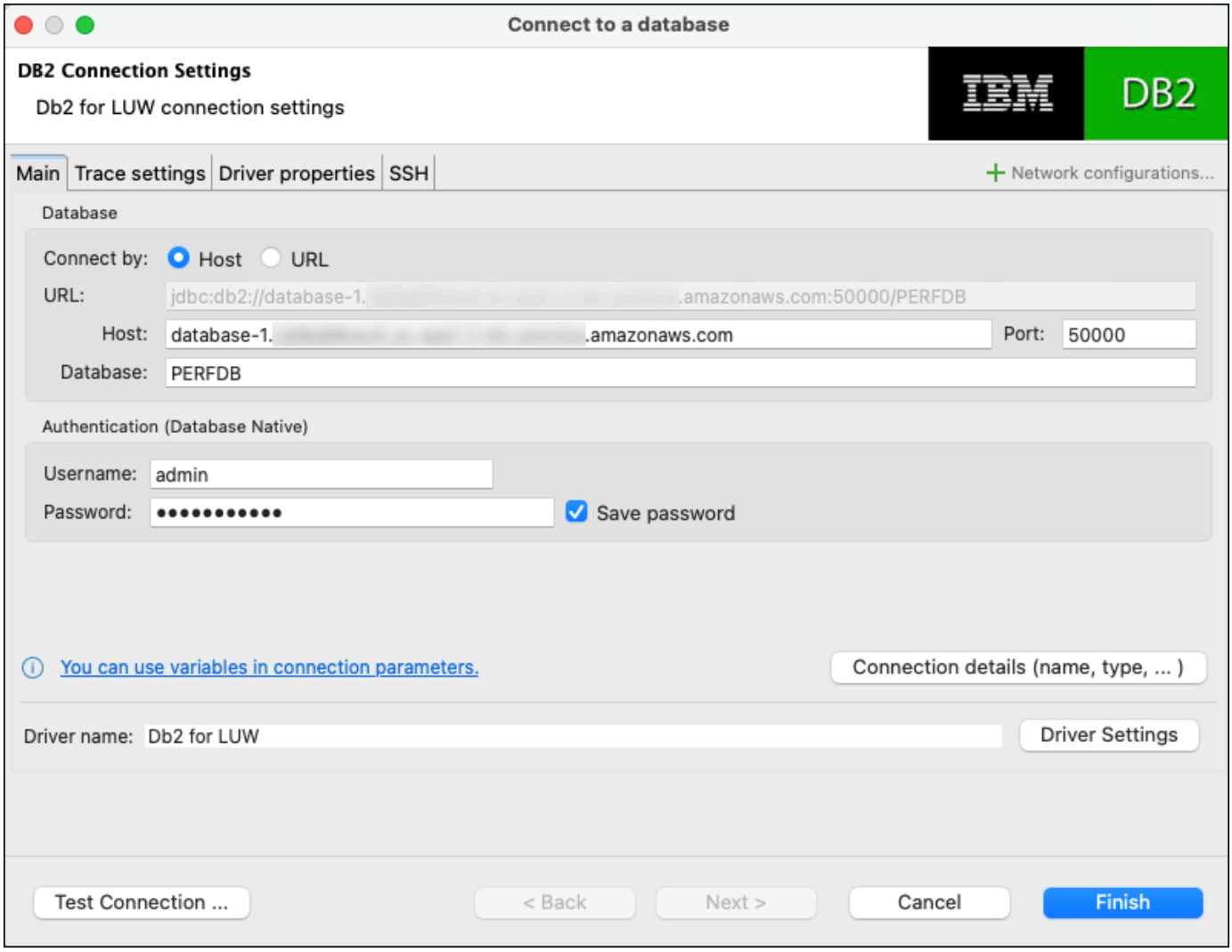
3. Dans la fenêtre Connect to a database, fournissez les informations relatives à votre instance de base de données RDS for DB2.

   a. Entrez les informations suivantes :

   - Pour Host, entrez le nom DNS de l'instance de base de données.
   - Pour Port, entrez le numéro de port de l'instance de base de données.
   - Pour Base de données, entrez le nom de la base de données.
   - Pour Nom d'utilisateur, saisissez le nom de l'administrateur de base de données pour l'instance de base de données.
   - Pour Mot de passe, entrez le mot de passe de l'administrateur de base de données pour l'instance de base de données.

b. Sélectionnez Enregistrer le mot de passe.

c. Choisissez les paramètres du pilote.



4. Dans la fenêtre Modifier le pilote, spécifiez des propriétés de sécurité supplémentaires.

a. Choisissez l'onglet Propriétés du pilote.

b. Ajoutez deux propriétés utilisateur.

i. Ouvrez le menu contextuel (clic droit), puis choisissez Ajouter une nouvelle propriété.

ii. Pour Nom de la propriété, ajoutez EncryptionAlgorithm, puis choisissez OK.

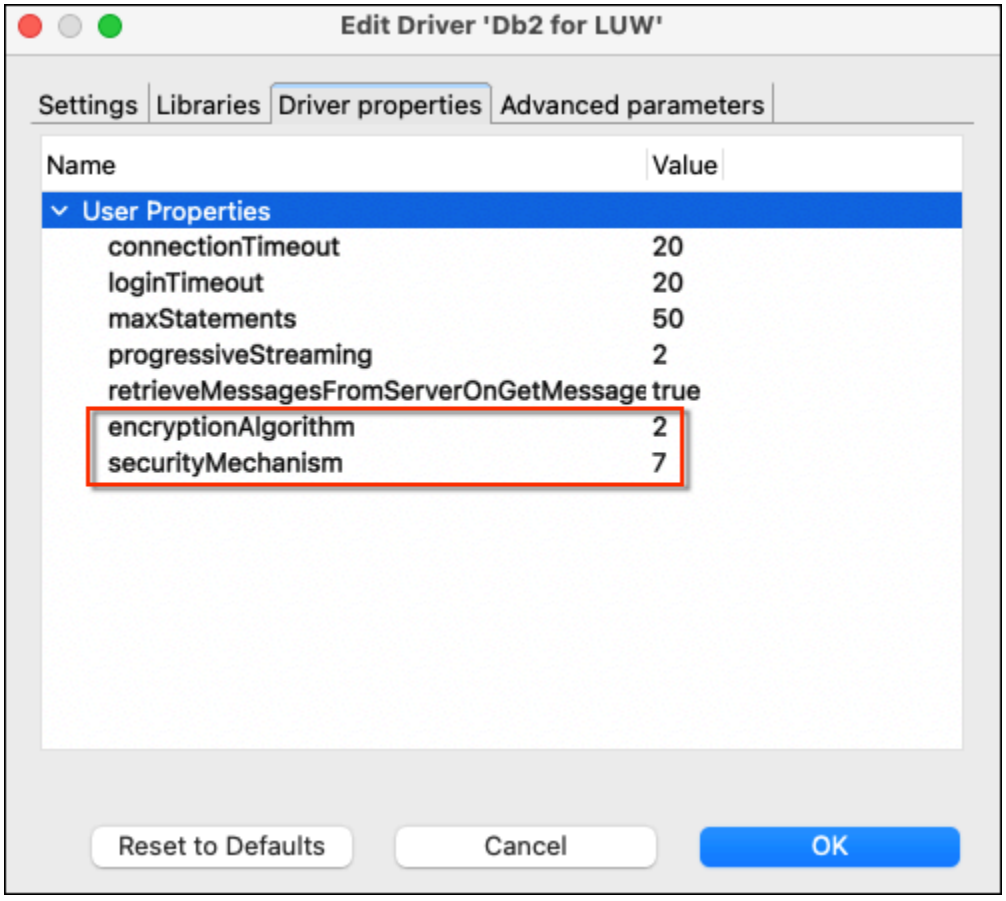
iii. La ligne EncryptionAlgorithm étant sélectionnée, choisissez la colonne Value et ajoutez 2.

iv. Ouvrez le menu contextuel (clic droit), puis choisissez Ajouter une nouvelle propriété.

v. Pour Nom de la propriété, ajoutez SecurityMechanism, puis cliquez sur OK.

vi. La ligne SecurityMechanism étant sélectionnée, choisissez la colonne Valeur et ajoutez 7.

c. Choisissez OK.

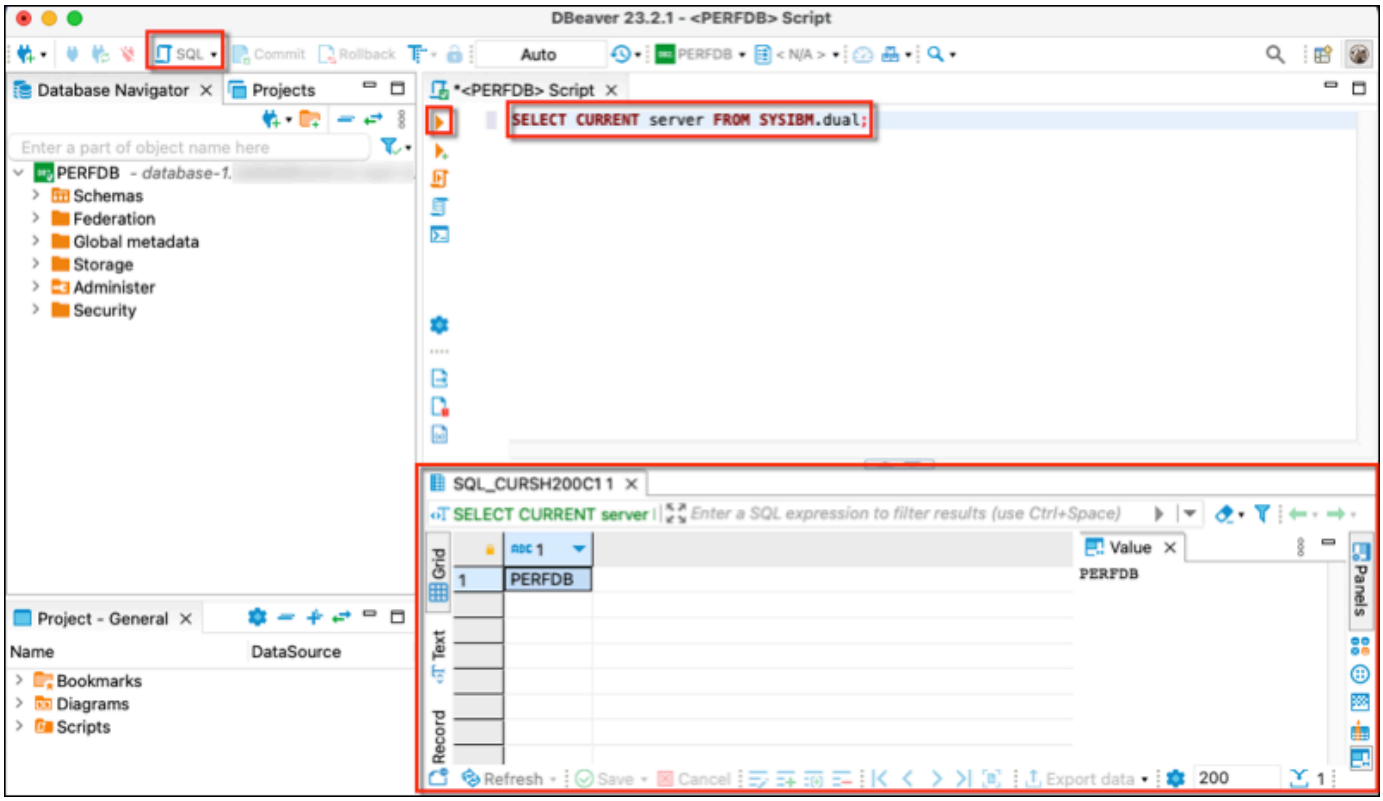


5. Dans la fenêtre Se connecter à une base de données, sélectionnez Tester la connexion.
   Si aucun pilote DB2 JBDC n'est installé sur votre ordinateur, le pilote est automatiquement téléchargé.
6. Choisissez OK.
7. Choisissez Finish (Terminer).
8. Dans l'onglet Navigation dans la base de données, choisissez le nom de la base de données.
   Vous pouvez désormais explorer des objets.

Vous êtes maintenant prêt à exécuter des commandes SQL.

Pour exécuter des commandes SQL et afficher les résultats

1. Dans le menu supérieur, choisissez SQL. Cela ouvre un panneau de script SQL.
2. Dans le panneau Script, entrez une commande SQL.
3. Pour exécuter la commande, cliquez sur le bouton Exécuter la requête SQL.
4. Dans le panneau des résultats SQL, consultez les résultats de vos requêtes SQL.



Connexion à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console

Vous pouvez vous connecter à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console. IBM Db2 Data Management Console peut administrer et surveiller plusieurs instances RDS de base de données DB2.

> **Note**
>
> Vous devez avoir un Amazon EC2 Linux ou une machine Windows qui se trouve sur le même groupe VPC de sécurité que votre instance de base de données RDS for Db2. Le groupe de

sécurité VPC and contrôle la connexion à votre instance de base de données via le réseau interne.

IBM Db2 Data Management Console nécessite une base de données de référentiel DB2 pour stocker les métadonnées et les mesures de performance, mais ne peut pas créer automatiquement une base de données de référentiel RDS pour Db2. Au lieu de cela, vous devez d'abord créer une base de données de référentiel RDS pour surveiller une ou plusieurs instances de base de données DB2. Ensuite, vous pouvez installer IBM Db2 Data Management Console et connectez-vous à votre instance de base de données RDS for Db2 avec IBM Db2 Data Management Console.

Rubriques

- Étape 1 : Création d'une base de données de référentiel pour surveiller les instances de base de données
- Étape 2 : Installation et configuration IBM Db2 Data Management Console
- Étape 3 : Configuration de la base de données du référentiel et connexion à celle-ci RDS pour les instances de base de données DB2
- Utilisation IBM Db2 Data Management Console

Étape 1 : Création d'une base de données de référentiel pour surveiller les instances de base de données

Vous pouvez utiliser une instance de base de données existante correctement dimensionnée RDS pour DB2 comme référentiel pour IBM Db2 Data Management Console pour surveiller d'autres instances RDS de base de données DB2. Cependant, étant donné que l'utilisateur administrateur n'est pas `SYSCTRL` autorisé à créer des pools de mémoire tampon et des tablespaces, en utilisant IBM Db2 Data Management Console la création d'un référentiel pour créer une base de données de référentiel échoue. Vous devez plutôt créer une base de données de référentiel. Cette base de données de référentiel surveille vos instances RDS de base de données Db2.

Vous pouvez créer une base de données de référentiel de deux manières différentes. Vous pouvez créer une base de données RDS pour DB2, puis créer manuellement un pool de mémoire tampon, un espace disque logique utilisateur et un espace disque logique temporaire du système. Vous pouvez également créer une EC2 instance Amazon distincte pour héberger un IBM Db2 Data Management Console base de données du référentiel.

Rubriques

- Création manuelle d'un pool de mémoire tampon, d'un espace disque logique utilisateur et d'un espace disque logique temporaire du système
- Création d'une EC2 instance Amazon pour héberger un IBM Db2 Data Management Console référentiels

Création manuelle d'un pool de mémoire tampon, d'un espace disque logique utilisateur et d'un espace disque logique temporaire du système

Pour créer un pool de mémoire tampon, un espace disque logique utilisateur et un espace disque logique temporaire du système

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Créez un pool de mémoire tampon pour IBM Db2 Data Management Console. Dans l'exemple suivant, remplacez *`database_name`* avec le nom du référentiel que vous avez créé pour IBM Db2 Data Management Console pour surveiller vos instances RDS de base de données Db2.

   ```
   db2 "call rdsadmin.create_bufferpool('database_name',
        'BP4CONSOLE', 1000, 'Y', 'Y', 32768)"
   ```

3. Créez un tablespace utilisateur pour IBM Db2 Data Management Console. Dans l'exemple suivant, remplacez *`database_name`* avec le nom du référentiel que vous avez créé pour IBM Db2 Data Management Console pour surveiller vos instances RDS de base de données Db2.

   ```
   db2 "call rdsadmin.create_tablespace('database_name',
        'TS4CONSOLE', 'BP4CONSOLE', 32768)"
   ```

4. Créez un tablespace temporaire système pour IBM Db2 Data Management Console. Dans l'exemple suivant, remplacez *`database_name`* avec le nom du référentiel que vous avez créé pour IBM Db2 Data Management Console pour surveiller vos instances RDS de base de données Db2.

   ```
   db2 "call rdsadmin.create_tablespace('database_name',
       'TS4CONSOLE_TEMP', 'BP4CONSOLE', 32768, 0, 0, 'S')"
   ```

Vous êtes maintenant prêt à installer IBM Db2 Data Management Console. Pour plus d'informations sur l'installation et la configuration, consultezÉtape 2 : Installation et configuration IBM Db2 Data Management Console.

Création d'une EC2 instance Amazon pour héberger un IBM Db2 Data Management Console référentiels

Vous pouvez créer une instance Amazon Elastic Compute Cloud (AmazonEC2) distincte pour héberger un IBM Db2 Data Management Console repository. Pour plus d'informations sur la création d'une EC2 instance Amazon, consultez Tutoriel : Commencer avec Amazon EC2 Linux exemples dans le guide de EC2 l'utilisateur Amazon.

Étape 2 : Installation et configuration IBM Db2 Data Management Console

Après avoir créé un pool de mémoire tampon, un espace disque logique utilisateur et un espace disque logique temporaire du système, vous êtes prêt à procéder à l'installation et à la configuration IBM Db2 Data Management Console.

> **Important**
>
> Vous devez avoir un Amazon EC2 Linux ou une machine Windows qui se trouve sur le même groupe VPC de sécurité que votre instance de base de données RDS for Db2. Le groupe de sécurité VPC and contrôle la connexion à votre instance de base de données via le réseau interne. Vous devez également avoir déjà créé une base de données de référentiel pour IBM Db2 Data Management Console.

Pour installer et configurer IBM Db2 Data Management Console

1. Téléchargement IBM Db2 Data Management Console à partir de IBM Db2 Data Management Console La version 3.1x est disponible sur le site Web de IBM support.
2. Installation IBM Db2 Data Management Console.
3. Ouvrir IBM Db2 Data Management Console et utilisez l'adresse IP de votre EC2 machine Amazon et le numéro de port que vous avez utilisé pour la HTTP HTTPS connexion à votre EC2 instance Amazon. Par exemple, utilisez `http://xx.xx.xx.xx:11080` ou`https://xx.xx.xx.xx.11081`. Remplacez *xx.xx.xx.xx* par l'adresse IP de votre EC2 machine Amazon. `11080`et `11081` sont les ports HTTP et les HTTPS connexions par défaut.

4. (Facultatif) Si vous souhaitez utiliser le port 80 ou 443 sur votre EC2 instance Amazon, vous pouvez utiliser Apache httpd ou un HTTP serveur Nginx pour créer un proxy IBM Db2 Data Management Console port vers le port 80 ou 443. Pour plus d'informations, consultez le projet Apache HTTP Server et le site Web nginx.

   Pour autoriser la connexion à IBM Db2 Data Management Console, vous devez modifier les règles de trafic entrant dans votre groupe de sécurité. Si vous utilisez un proxy, modifiez le port TCP /IP 80 ou 443 pour rediriger vers IBM Db2 Data Management Console ports. Si vous n'utilisez pas de proxy, remplacez le port TCP /IP 80 ou 443 par les ports par défaut 11080 (HTTP) ou 11081 (). HTTPS

Vous êtes maintenant prêt à vous connecter à IBM Db2 Data Management Console pour configurer la base de données du référentiel et pour vous connecter à vos instances RDS de base de données Db2. Pour de plus amples informations, veuillez consulter Configuration de la base de données du référentiel et connexion aux instances de base de données.

Étape 3 : Configuration de la base de données du référentiel et connexion à celle-ci RDS pour les instances de base de données DB2

Lorsque vous vous connectez à la base de données du référentiel pour la première fois, IBM Db2 Data Management Console configure automatiquement le référentiel. Une fois la base de données du référentiel configurée, vous pouvez ajouter des connexions de base de données à IBM Db2 Data Management Console.

Pour vous connecter à votre instance de base de données RDS for Db2, vous avez besoin de son DNS nom et de son numéro de port. Pour plus d'informations sur leur recherche, consultezRecherche du point de terminaison. Vous devez également connaître le nom de la base de données, le nom d'utilisateur principal et le mot de passe principal que vous avez définis lors de la création de votre instance de base de données RDS pour DB2. Pour plus d'informations sur leur recherche, consultezCréation d'une instance de base de données. Si vous vous connectez via Internet, autorisez le trafic vers le port de base de données. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données.

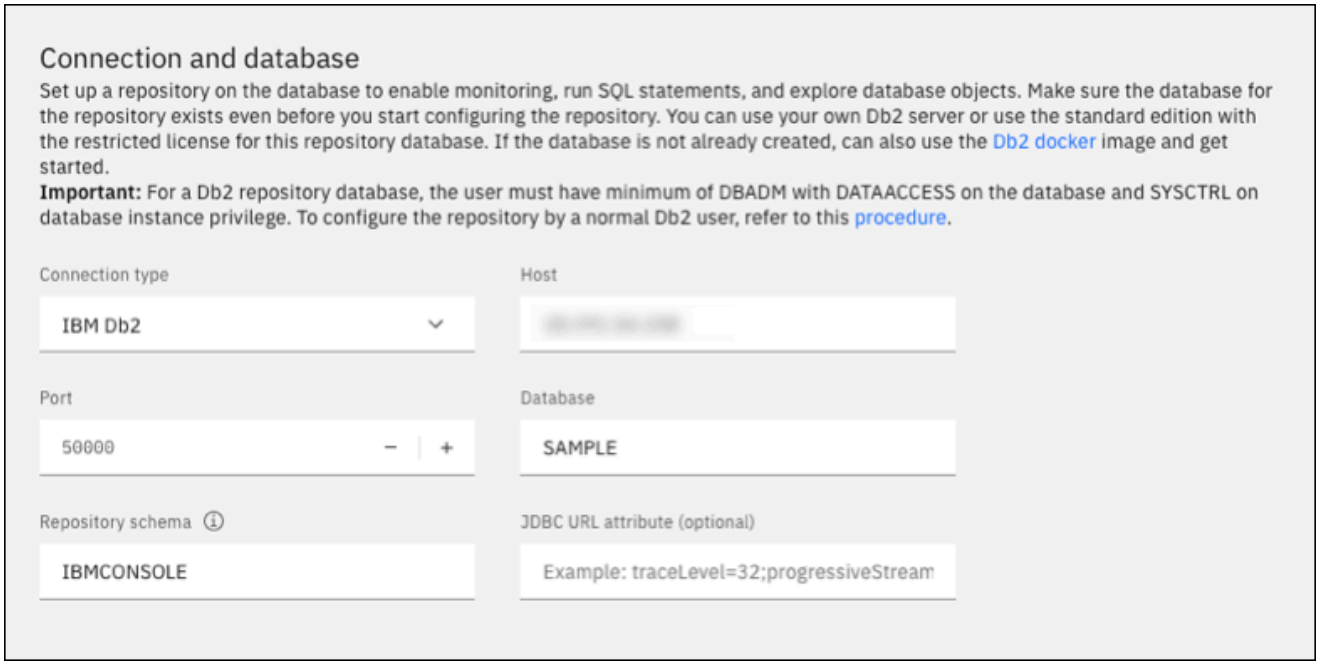
Pour vous connecter à des instances RDS de base de données DB2 avec IBM Db2 Data Management Console

1. Connectez-vous à IBM Db2 Data Management Console avec les informations d'identification que vous avez définies lors de l'installation.

2. Configurez le référentiel.

   a. Dans la section Connexion et base de données, entrez les informations suivantes pour votre instance de base de données RDS for Db2 :

      - Pour Host, entrez le DNS nom de l'instance de base de données.
      - Pour Port, entrez le numéro de port de l'instance de base de données.
      - Pour Base de données, entrez le nom de la base de données.



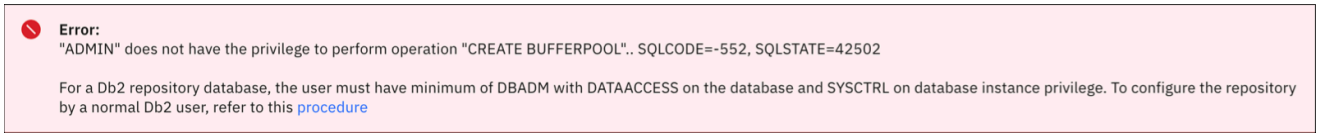
   b. Dans la section Sécurité et informations d'identification, entrez les informations suivantes pour votre instance de base de données RDS for Db2 :

      - Pour Type de sécurité, choisissez Utilisateur et mot de passe cryptés.
      - Pour Nom d'utilisateur, saisissez le nom de l'administrateur de base de données pour l'instance de base de données.
      - Pour Mot de passe, entrez le mot de passe de l'administrateur de base de données pour l'instance de base de données.

   c. Choisissez Test connection (Tester la connexion).

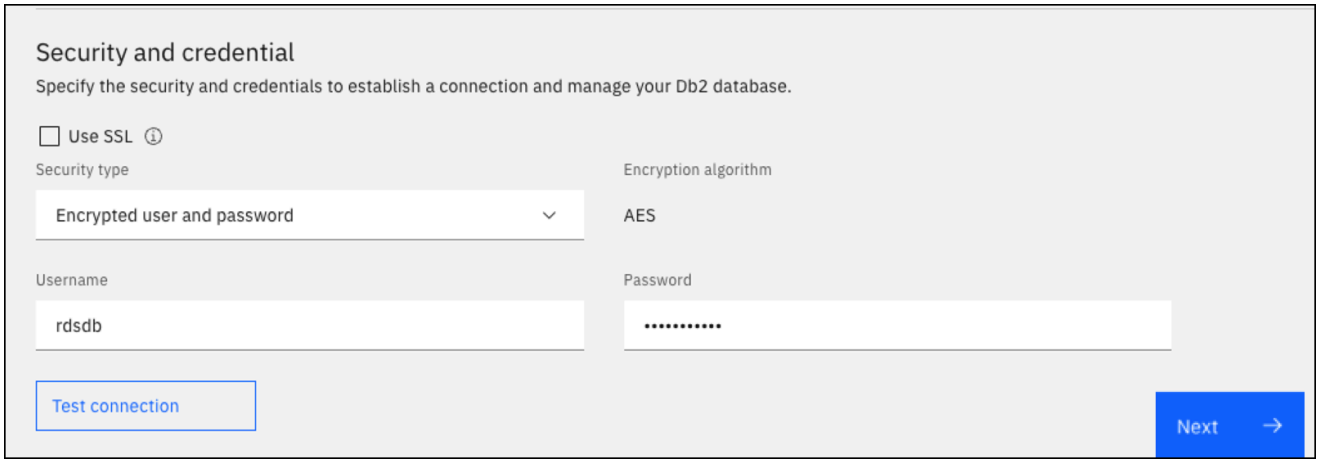
> **Note**
>
> Si la connexion échoue, vérifiez que le port de base de données est ouvert conformément aux règles de trafic entrant de votre groupe de sécurité. Pour de plus amples informations, veuillez consulter Considérations relatives aux groupes de sécurité avec Amazon RDS pour DB2.

Si vous n'avez pas créé manuellement un pool de mémoire tampon, un espace disque logique utilisateur et un espace disque logique temporaire système dans DB2, le RDS message d'erreur suivant peut s'afficher :



Assurez-vous d'avoir créé une table tampon, un tablespace et des objets pour un IBM Db2 Data Management Console référentiel pour surveiller votre instance de base de données RDS pour DB2. Vous pouvez également utiliser une instance de base de données Amazon EC2 Db2 pour héberger un IBM Db2 Data Management Console référentiel pour surveiller votre instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Étape 1 : Création d'une base de données de référentiel pour surveiller les instances de base de données.

d. Après avoir testé votre connexion avec succès, choisissez Next.

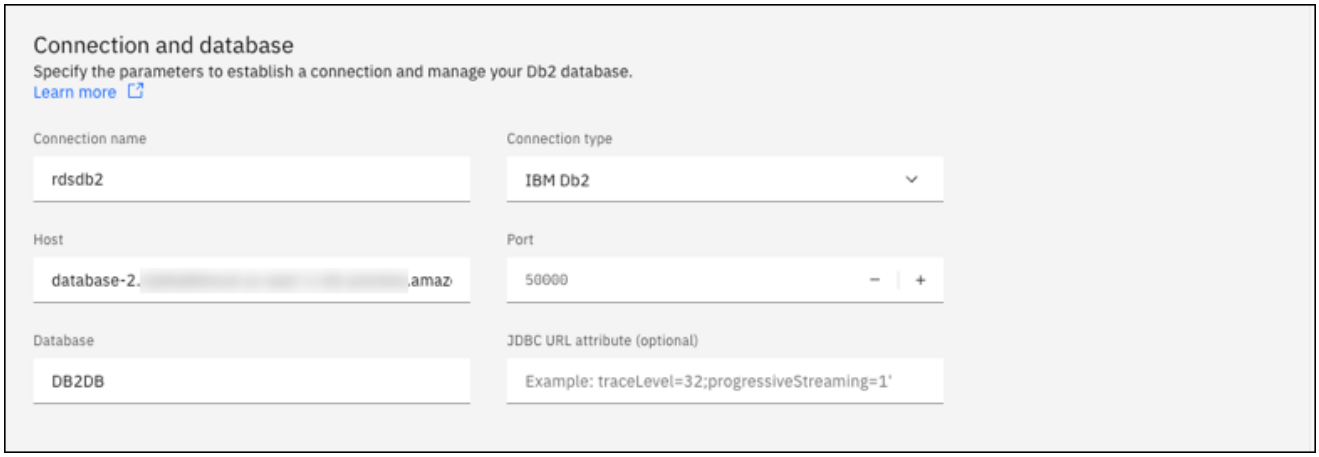


If IBM Db2 Data Management Console trouve le pool de mémoire tampon, le tablespace utilisateur et le tablespace temporaire du système dans l'instance de base de données RDS for DB2, puis IBM Db2 Data Management Console configure automatiquement la base de données du référentiel. Si vous utilisez votre instance Db2 sur votre EC2 instance Amazon comme base de données de référentiel, alors IBM Db2 Data Management Console crée automatiquement le pool de mémoire tampon et d'autres objets.

3. Dans la fenêtre d'inscription au moniteur d'événements Set statistics, choisissez Next.
4. (Facultatif) Ajoutez une nouvelle connexion. Si vous souhaitez utiliser une instance de base de données Db2 différente RDS pour l'administration et la surveillance, ajoutez une connexion à une instance de base de données autre que le référentiel RDS pour l'instance de base de données DB2.

   a. Dans la section Connexion et base de données, entrez les informations suivantes RDS pour l'instance de base de données Db2 à utiliser pour l'administration et la surveillance :

      - Dans Nom de la connexion, entrez l'identifiant de base de données DB2.
      - Pour Host, entrez le DNS nom de l'instance de base de données.
      - Pour Port, entrez le numéro de port de l'instance de base de données.
      - Pour Base de données, entrez le nom de la base de données.



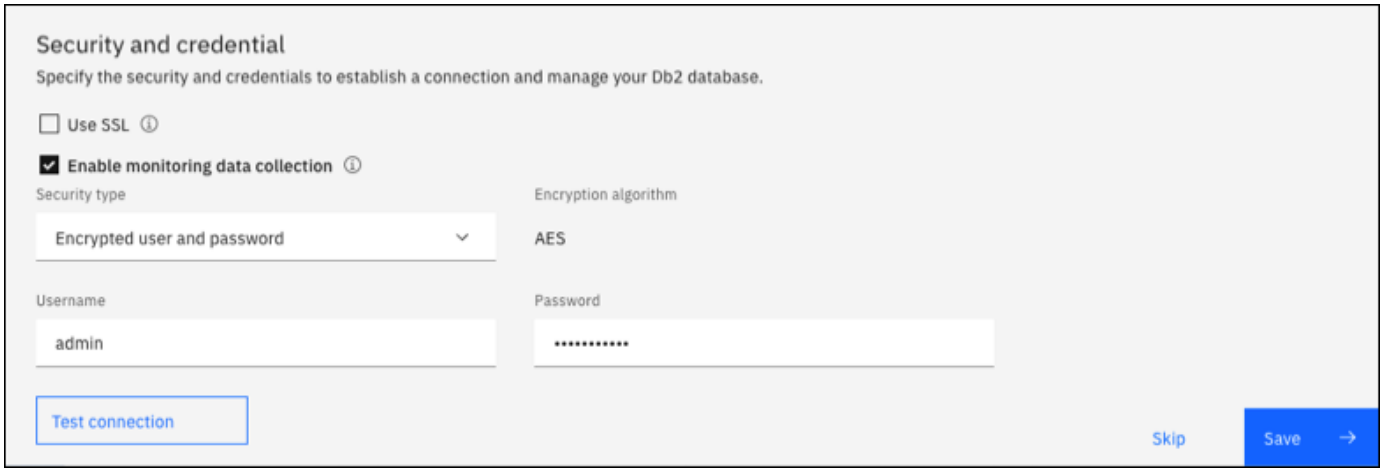
   b. Dans la section Sécurité et informations d'identification, sélectionnez Activer la collecte de données de surveillance.

   c. Entrez les informations suivantes pour votre instance de base de données RDS for Db2 :

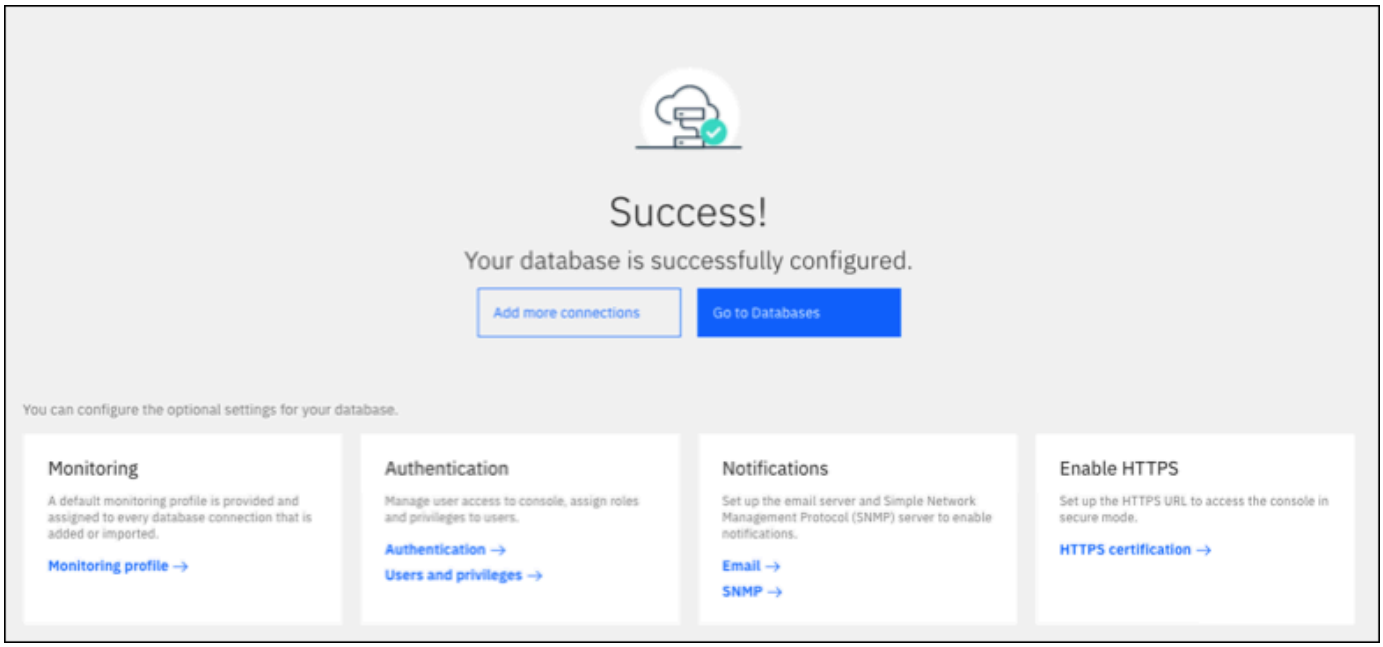
- Pour Nom d'utilisateur, saisissez le nom de l'administrateur de base de données pour l'instance de base de données.
- Pour Mot de passe, entrez le mot de passe de l'administrateur de base de données pour l'instance de base de données.

d. Choisissez Test connection (Tester la connexion).

e. Après avoir testé votre connexion avec succès, choisissez Enregistrer.



Une fois la connexion ajoutée, une fenêtre semblable à la suivante apparaît. Cette fenêtre indique que votre base de données a été correctement configurée.



5. Choisissez Accéder aux bases de données. Une fenêtre de bases de données similaire à la suivante s'affiche. Cette fenêtre est un tableau de bord qui affiche les métriques, les statuts et les connexions.



Vous pouvez maintenant commencer à utiliser IBM Db2 Data Management Console.

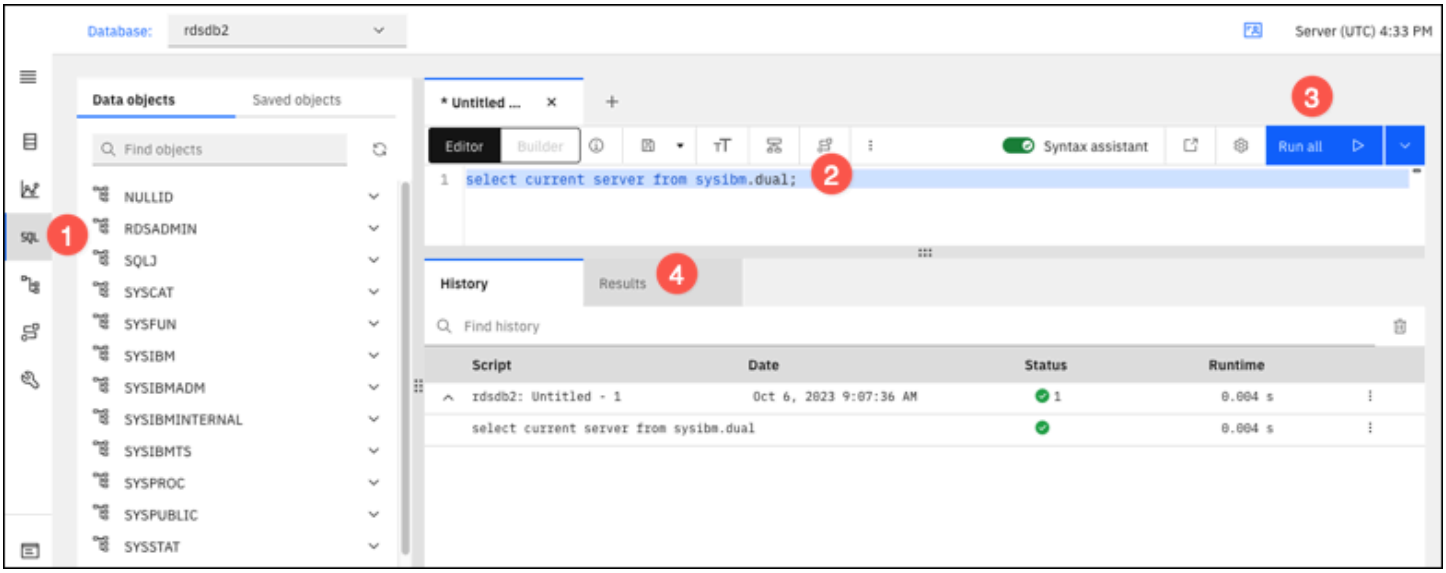
Utilisation IBM Db2 Data Management Console

Vous pouvez utiliser … IBM Db2 Data Management Console pour effectuer les types de tâches suivants :

- Gérez plusieurs instances RDS de base de données DB2.
- Exécutez SQL des commandes.
- Explorez, créez ou modifiez des données et des objets de base de données.
- Créez `EXPLAIN PLAN` des déclarations dansSQL.
- Réglez les requêtes.

Pour exécuter SQL des commandes et afficher les résultats

1. Dans la barre de navigation de gauche, choisissez SQL.
2. Entrez une SQL commande.
3. Choisissez Tout exécuter.
4. Pour afficher les résultats, cliquez sur l'onglet Résultats.



Considérations relatives aux groupes de sécurité avec Amazon RDS pour DB2

Pour que vous puissiez vous connecter à votre instance de base de données Amazon RDS pour DB2, celle-ci doit être associée à un groupe de sécurité contenant les adresses IP et la configuration réseau nécessaires. Votre instance de base de données RDS pour DB2 peut utiliser le groupe de sécurité par défaut. Si vous avez attribué un groupe de sécurité non configuré par défaut lors de la création de l'instance de base de données RDS pour DB2, le pare-feu empêche les connexions Internet. Pour obtenir des informations sur la création d'un groupe de sécurité, consultez Contrôle d'accès par groupe de sécurité.

Une fois le groupe de sécurité créé, vous devez modifier votre instance de base de données pour l'associer au groupe de sécurité. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Vous pouvez améliorer la sécurité en utilisant SSL pour chiffrer les connexions à votre instance de base de données. Pour plus d'informations, voir Utilisation SSL/TLS avec une instance de base de données Amazon RDS pour DB2.

Sécurisation des connexions aux instances de base de données Amazon RDS pour DB2

Amazon RDS pour DB2 permet d'améliorer la sécurité de votre instance de base de données RDS pour DB2.

Rubriques

- Utilisation SSL/TLS avec une instance de base de données Amazon RDS pour DB2
- Utilisation de Kerberos l'authentification pour Amazon RDS pour Db2

Utilisation SSL/TLS avec une instance de base de données Amazon RDS pour DB2

Le protocole SSL est un protocole standard pour sécuriser les connexions réseau entre le client et le serveur. Après la version 3.0 de SSL, le nom a été changé en TLS, mais nous appelons encore souvent le protocole SSL. Amazon RDS prend en charge le chiffrement SSL pour Amazon RDS pour les instances de base de données DB2. L'utilisation SSL/TLS, you can encrypt a connection between your application client and your RDS for Db2 DB instance. SSL/TLS du support est disponible dans tous les cas Régions AWS pour RDS pour Db2.

Pour activer SSL/TLS le chiffrement d'une instance de base de données RDS pour DB2, ajoutez l'option SSL DB2 au groupe de paramètres associé à l'instance de base de données. Amazon RDS utilise un deuxième port, comme l'exige Db2, pour les SSL/TLS connexions. Cela permet à la fois d'établir une communication en texte clair et cryptée SSL entre une instance de base de données et un client DB2. Par exemple, vous pouvez utiliser le port avec une communication en texte clair pour communiquer avec d'autres ressources à l'intérieur d'un VPC, tout en utilisant le port avec une communication à chiffrement SSL pour communiquer avec des ressources extérieures au VPC.

Rubriques

- Création d'une SSL/TLS connexion
- Connectez-vous à votre serveur de base de données DB2

Création d'une SSL/TLS connexion

Pour créer une SSL/TLS connexion, choisissez une autorité de certification (CA), téléchargez un ensemble de certificats pour tous Régions AWS et ajoutez des paramètres à un groupe de paramètres personnalisé.

Étape 1 : Choisissez une autorité de certification et téléchargez un certificat

Choisissez une autorité de certification (CA) et téléchargez un ensemble de certificats pour tous Régions AWS. Pour de plus amples informations, veuillez consulter .

Étape 2 : Mettre à jour les paramètres d'un groupe de paramètres personnalisé

> **Important**
>
> Si vous utilisez le modèle BYOL (Bring your own license) pour RDS for DB2, modifiez le groupe de paramètres personnalisé que vous avez créé pour votre et votreIBM Customer ID. IBM Site ID Si vous utilisez un modèle de licence différent pour RDS pour DB2, suivez la procédure pour ajouter des paramètres à un groupe de paramètres personnalisé. Pour de plus amples informations, veuillez consulter Options de licence Amazon RDS pour DB2.

Vous ne pouvez pas modifier les groupes de paramètres par défaut pour les instances de base de données RDS pour DB2. Par conséquent, vous devez créer un groupe de paramètres personnalisé, le modifier, puis l'associer à vos instances de base de données RDS pour DB2. Pour plus d'informations sur les groupes de paramètres, veuillez consulter Groupes de paramètres de base de données pour les instances de base de données RDSAmazon.

Utilisez les paramètres définis dans le tableau suivant.

Paramètre	Valeur
`DB2COMM`	`TCPIP,SSL` ou `SSL`
`SSL_SVCENAME`	`<any port number except the number used for the non-SSL port>`

Pour mettre à jour les paramètres d'un groupe de paramètres personnalisé

1. Créez un groupe de paramètres personnalisé en exécutant la create-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Nom du groupe de paramètres que vous créez.
   - `--db-parameter-group-family`— L'édition et la version majeure du moteur DB2. Valeurs valides : `db2-se-11-5`, `db2-ae-11.5`.
   - `--description`— Description de ce groupe de paramètres.

   Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez les paramètres du groupe de paramètres personnalisés que vous avez créé en exécutant la modify-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Le nom du groupe de paramètres que vous avez créé.
   - `--parameters`— Tableau de noms de paramètres, de valeurs et de méthodes d'application pour la mise à jour des paramètres.

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).
3. Associez le groupe de paramètres à votre instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon.

Connectez-vous à votre serveur de base de données DB2

Les instructions de connexion à votre serveur de base de données DB2 sont spécifiques à la langue.

Java

Pour vous connecter à votre serveur de base de données DB2 à l'aide de Java

1. Téléchargez le pilote JDBC. Pour plus d'informations, consultez la section Versions et téléchargements du pilote DB2 JDBC dans la documentation de IBM support.
2. Créez un fichier de script shell avec le contenu suivant. Ce script ajoute tous les certificats du bundle à unJava KeyStore.

> **Important**
>
> Vérifiez qu'`keytool`il existe sur le chemin indiqué dans le script afin que celui-ci puisse le localiser. Si vous utilisez un client DB2, vous pouvez le trouver ci-dessous`keytool`. `~sqlib/java/jdk64/jre/bin`

```
#!/bin/bash
PEM_FILE=$1
PASSWORD=$2
KEYSTORE=$3
# number of certs in the PEM file
CERTS=$(grep 'END CERTIFICATE' $PEM_FILE| wc -l)
for N in $(seq 0 $(($CERTS - 1))); do
    ALIAS="${PEM_FILE%.*}-$N"
    cat $PEM_FILE |
    awk "n==$N { print }; /END CERTIFICATE/ { n++ }" |
    keytool -noprompt -import -trustcacerts -alias $ALIAS -keystore $KEYSTORE -
storepass $PASSWORD
done
```

3. Pour exécuter le script shell et importer le PEM fichier contenant le bundle de certificats dans unJava KeyStore, exécutez la commande suivante. *`shell_file_name.sh`*Remplacez-le par le nom de votre fichier de script shell et *`password`* par le mot de passe de votreJava KeyStore.

```
./shell_file_name.sh global-bundle.pem password truststore.jks
```

4. Pour vous connecter à votre serveur DB2, exécutez la commande suivante. Remplacez les espaces réservés suivants dans l'exemple par votre RDS pour les informations d'instance de base de données DB2.

- *ip_address*— L'adresse IP du point de terminaison de votre instance de base de données.
- *port*— Le numéro de port de la connexion SSL. Il peut s'agir de n'importe quel numéro de port à l'exception du numéro utilisé pour le port non SSL.
- *database_name*— Le nom de votre base de données dans votre instance de base de données.
- *master_username*— Le nom d'utilisateur principal de votre instance de base de données.
- *master_password*— Le mot de passe principal de votre instance de base de données.

```
export trustStorePassword=MyPassword
java -cp ~/dsdriver/jdbc_sqlj_driver/linuxamd64/db2jcc4.jar \
com.ibm.db2.jcc.DB2Jcc -url \
"jdbc:db2://ip_address:port/database_name:\
sslConnection=true;sslTrustStoreLocation=\
~/truststore.jks;\
sslTrustStorePassword=${trustStorePassword};\
sslVersion=TLSv1.2;\
encryptionAlgorithm=2;\
securityMechanism=7;" \
-user master_username -password master_password
```

Node.js

Pour vous connecter à votre serveur de base de données DB2 à l'aide de Node.js

1. Installez le node-ibm_dbpilote. Pour plus d'informations, consultez la section Installation du pilote node-ibm_db sur les systèmes Linux et UNIX dans la documentation. IBM Db2
2. Créez un JavaScript fichier basé sur le contenu suivant. Remplacez les espaces réservés suivants dans l'exemple par votre RDS pour les informations d'instance de base de données DB2.

   - *ip_address*— L'adresse IP du point de terminaison de votre instance de base de données.
   - *master_username*— Le nom d'utilisateur principal de votre instance de base de données.
   - *master_password*— Le mot de passe principal de votre instance de base de données.

- *database_name*— Le nom de votre base de données dans votre instance de base de données.
- *port*— Le numéro de port de la connexion SSL. Il peut s'agir de n'importe quel numéro de port à l'exception du numéro utilisé pour le port non SSL.

```
var ibmdb = require("ibm_db");
const hostname = "ip_address";
const username = "master_username";
const password = "master_password";
const database = "database_name";
const port = "port";
const certPath = "/root/qa-bundle.pem";
ibmdb.open("DRIVER={DB2};DATABASE=" + database + ";HOSTNAME=" +
 hostname + ";UID=" + username + ";PWD=" + password + ";PORT=" + port +
 ";PROTOCOL=TCPIP;SECURITY=SSL;SSLServerCertificate=" + certPath + ";", function
 (err, conn){
 if (err) return console.log(err);
 conn.close(function () {
 console.log('done');
 });
});
```

3. Pour exécuter le JavaScript fichier, exécutez la commande suivante.

```
node ssl-test.js
```

Python

Pour vous connecter à votre serveur de base de données DB2 à l'aide de Python

1. Créez un Python fichier avec le contenu suivant. Remplacez les espaces réservés suivants dans l'exemple par votre RDS pour les informations d'instance de base de données DB2.

   - *port*— Le numéro de port de la connexion SSL. Il peut s'agir de n'importe quel numéro de port à l'exception du numéro utilisé pour le port non SSL.
   - *master_username*— Le nom d'utilisateur principal de votre instance de base de données.
   - *master_password*— Le mot de passe principal de votre instance de base de données.

- *database_name*— Le nom de votre base de données dans votre instance de base de données.
- *ip_address*— L'adresse IP du point de terminaison de votre instance de base de données.

```
import click
import ibm_db
import sys

port = port;
master_user_id = "master_username" # Master id used to create your DB instance
master_password = "master_password" # Master password used to create your DB
 instance
db_name = "database_name" # If not given "db-name'
vpc_customer_private_ip = "ip_address" # Hosts end points - Customer private IP
 Addressicert_path = "/root/ssl/global-bundle.pem" # cert path

@click.command()
@click.option("--path", help="certificate path")
def db2_connect(path):

    try:
        conn =
 ibm_db.connect(f"DATABASE={db_name};HOSTNAME={vpc_customer_private_ip};PORT={port};

 PROTOCOL=TCPIP;UID={master_user_id};PWD={master_password};SECURITY=ssl;SSLServerCertifi
 "", "")
        try:
            ibm_db.exec_immediate(conn, 'create table tablename (a int);')
            print("Query executed successfully")
        except Exception as e:
            print(e)
        finally:
            ibm_db.close(conn)
            sys.exit(1)
    except Exception as ex:
        print("Trying to connect...")

if __name__ == "__main__":
    db2_connect()
```

2. Créez le script shell suivant, qui exécute le Python fichier que vous avez créé. *python_file_name.py*Remplacez-le par le nom de votre fichier de Python script.

```
#!/bin/bash
PEM_FILE=$1
# number of certs in the PEM file
CERTS=$(grep 'END CERTIFICATE' $PEM_FILE| wc -l)

for N in $(seq 0 $(($CERTS - 1))); do
    ALIAS="${PEM_FILE%.*}-$N"
    cert=`cat $PEM_FILE | awk "n==$N { print }; /END CERTIFICATE/ { n++ }"`
    cat $PEM_FILE | awk "n==$N { print }; /END CERTIFICATE/ { n++ }" >
 $ALIAS.pem
    python3 <python_file_name.py> --path $ALIAS.pem
    output=`echo $?`
    if [ $output == 1 ]; then
        break
    fi
done
```

3. Pour importer le PEM fichier avec le bundle de certificats et exécuter le script shell, exécutez la commande suivante. *shell_file_name.sh*Remplacez-le par le nom de votre fichier de script shell.

```
./shell_file_name.sh global-bundle.pem
```

DB2 CLP

Pour vous connecter à votre serveur de base de données DB2 à l'aide de Db2 CLP

1. Pour vous connecter à votre instance Db2 à l'aide deDb2 CLP, vous avez besoin GSKit de ce que vous pouvez télécharger sur IBM Fix Central. Pour l'utiliserDb2 CLP, vous avez également besoin du IBM Db2 client, que vous pouvez télécharger à partir de la page Télécharger les clients et les pilotes de la version 11.5 initiale dans IBM Support.
2. Créez un keystore.

```
gsk8capicmd_64 -keydb -create -db "directory/keystore-filename" -pw
 "changeThisPassword" -type pkcs12 -stash
```

3. Importez les ensembles de certificats dans le keystore.

```
gsk8capicmd_64 -cert -import -file global-bundle.pem -target directory/keystore-filename> -target_stashed
```

4. Mettez à jour la configuration de l'instance Db2.

```
db2 update dbm cfg using SSL_CLNT_KEYDB keystore-filename
 SSL_CLNT_STASH keystore stash file immediate
```

5. Cataloguez le nœud et la base de données.

```
db2 catalog tcpip node ssluse1 REMOTE endpoint SERVER ssl_svcename security ssl

db2 catalog database testdb as ssltest at node ssluse1
```

6. Etablit une connexion à la base de données

```
db2 connect to ssltest user username using password
```

Utilisation de Kerberos l'authentification pour Amazon RDS pour Db2

Vous pouvez utiliser l'Kerberosauthentification pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de base de données Amazon RDS pour DB2. Dans cette configuration, votre instance de base de données fonctionne avec AWS Directory Service for Microsoft Active Directory, également appelée AWS Managed Microsoft AD. Vous ajoutez le domaine et les autres informations de votre AWS Managed Microsoft AD répertoire à votre instance de base de données RDS pour DB2. Lorsque les utilisateurs s'authentifient auprès d'une instance de base de données RDS pour DB2 jointe au domaine de confiance, les demandes d'authentification sont transmises au AWS Managed Microsoft AD répertoire que vous avez créé avec. AWS Directory Service

Vous pouvez gagner du temps et de l'argent en conservant toutes les informations d'identification dans le même annuaire. Cette approche vous permet d'avoir un endroit centralisé de stockage et de gestion des informations d'identification pour plusieurs instances de base de données. L'utilisation d'un annuaire peut également améliorer votre profil de sécurité global.

Vous pouvez également accéder aux informations d'identification à partir de votre propre annuaire Microsoft Active Directory sur site. Pour ce faire, vous créez une relation de domaine d'approbation afin que l'annuaire AWS Managed Microsoft AD approuve votre annuaire Microsoft Active Directory

sur site. Ainsi, vos utilisateurs peuvent accéder à vos instances de base de données RDS pour DB2 avec la même expérience d'authentification unique (SSO) Windows que lorsqu'ils accèdent aux charges de travail de votre réseau sur site.

Pour plus d'informations, voir Qu'est-ce que c'est AWS Directory Service ? dans le Guide AWS Directory Service d'administration.

Pour plus d'informations sur Kerberos l'authentification, consultez les rubriques suivantes :

Rubriques

- Configuration de Kerberos l'authentification pour Amazon RDS pour les instances de base de données DB2
- Connexion à Amazon RDS pour DB2 avec Kerberos authentification

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour plus d'informations sur la disponibilité des versions et des régions de RDS pour DB2 avec Kerberos authentification, consultez. Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS

> **Note**
>
> Kerberosl'authentification n'est pas prise en charge pour les classes d'instances de base de données déconseillées pour RDS pour les instances de base de données DB2. Pour de plus amples informations, veuillez consulter Amazon RDS pour les classes d'instance DB2.

Vue d'ensemble de l'Kerberosauthentification pour les instances de base de données RDS pour DB2

Pour configurer Kerberos l'authentification pour une instance de base de données RDS pour DB2, effectuez les étapes générales suivantes, décrites plus en détail ultérieurement :

1. AWS Managed Microsoft AD À utiliser pour créer un AWS Managed Microsoft AD répertoire. Vous pouvez utiliser le AWS Management Console, le AWS Command Line Interface (AWS CLI) ou

AWS Directory Service pour créer le répertoire. Pour plus d'informations, consultez la section Création de votre AWS Managed Microsoft AD répertoire dans le Guide d'AWS Directory Service administration.

2. Créez un rôle AWS Identity and Access Management (IAM) qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess` Le rôle IAM permet à Amazon RDS de passer des appels vers votre annuaire.

   Pour que le rôle IAM autorise l'accès, le point de terminaison AWS Security Token Service (AWS STS) doit être activé correctement Région AWS pour votre Compte AWS. AWS STS les points de terminaison sont actifs par défaut dans tous les cas Régions AWS, et vous pouvez les utiliser sans autre action. Pour plus d'informations, consultez la section Activation et désactivation AWS STS dans et Région AWS dans le guide de l'utilisateur IAM.

3. Créez ou modifiez une instance de base de données RDS pour DB2 à l'aide de l' AWS Management Console API RDS ou de l'API RDS selon l'une des méthodes suivantes : AWS CLI

   - Créez une nouvelle instance de base de données RDS pour DB2 à l'aide de la console, de la create-db-instancecommande ou de l'opération Create DBInstance API. Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
   - Modifiez une instance de base de données RDS pour DB2 existante à l'aide de la console, de la modify-db-instancecommande ou de l'opération ModifyDBInstanceAPI. Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
   - Restaurez une instance de base de données RDS pour DB2 à partir d'un instantané de base de données à l'aide de la console, de la restore-db-instance-from-db-snapshotcommande ou de l'opération RestoreDBInstanceFromDBSnapshotAPI. Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.
   - Restaurez une instance de base de données RDS pour DB2 à point-in-time l'aide de la console, de la restore-db-instance-to-point-in-timecommande ou de l'opération RestoreDBInstanceToPointInTimeAPI. Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

   Vous pouvez localiser l'instance de base de données dans le même Amazon Virtual Private Cloud (VPC) que le répertoire ou dans un autre VPC. Compte AWS Lorsque vous créez ou modifiez l'instance de base de données RDS pour DB2, effectuez les tâches suivantes :

   - Fournissez l'identifiant du domaine (identifiant `d-*`) qui a été généré lors de la création de votre annuaire.
   - Fournissez le nom du rôle IAM que vous avez créé.

- Vérifiez que le groupe de sécurité de l'instance de base de données peut recevoir du trafic entrant en provenance du groupe de sécurité de l'annuaire.

4. Configurez votre client DB2 et vérifiez que le trafic peut circuler entre l'hôte du client et AWS Directory Service pour les ports suivants :
   - Port TCP/UDP 53 — DNS
   - TCP 88 — authentification Kerberos
   - TCP 389 — LDAP
   - TCP 464 — authentification Kerberos

Gestion d'une instance de base de données dans un domaine

Vous pouvez utiliser l'API AWS Management Console AWS CLI, la ou l'API RDS pour gérer votre instance de base de données et sa relation avec votreMicrosoft Active Directory. Par exemple, vous pouvez associer un Active Directory pour activer Kerberos l'authentification. Vous pouvez également supprimer l'association pour désactiver ou Active Directory désactiver Kerberos l'authentification. Vous pouvez également déplacer une instance de base de données pour qu'elle soit authentifiée de manière externe par l'une Microsoft Active Directory vers l'autre.

Par exemple, en exécutant la commande modify-db-instanceCLI, vous pouvez effectuer les actions suivantes :

- Réessayez Kerberos d'activer l'authentification en cas d'échec d'une adhésion en spécifiant l'ID de répertoire de l'adhésion actuelle pour l'`--domain`option.
- Désactivez Kerberos l'authentification sur une instance de base de données `none` en spécifiant l'`--domain`option.
- Déplacez une instance de base de données d'un domaine à un autre en spécifiant l'identifiant de domaine du nouveau domaine pour l'`--domain`option.

Présentation de l'appartenance au domaine

Après la création ou la modification de votre instance de base de données, elle devient un membre du domaine. Vous pouvez consulter l'état de l'appartenance au domaine dans la console ou en exécutant la describe-db-instancescommande. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- `kerberos-enabled`— L'Kerberosauthentification est activée sur l'instance de base de données.

- `enabling-kerberos`— AWS est en train d'activer l'Kerberosauthentification sur cette instance de base de données.
- `pending-enable-kerberos`— Kerberos L'activation de l'authentification est en attente sur cette instance de base de données.
- `pending-maintenance-enable-kerberos`— AWS tentera d'activer Kerberos l'authentification sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `pending-disable-kerberos`— La désactivation de Kerberos l'authentification est en attente sur cette instance de base de données.
- `pending-maintenance-disable-kerberos`— AWS tentera de désactiver Kerberos l'authentification sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `enable-kerberos-failed`— Un problème de configuration a AWS empêché l'activation de Kerberos l'authentification sur l'instance de base de données. Corrigez le problème de configuration avant de réémettre la commande pour modifier l'instance de base de données.
- `disabling-kerberos`— AWS est en train de désactiver l'Kerberosauthentification sur cette instance de base de données.

Une demande d'activation de Kerberos l'authentification peut échouer en raison d'un problème de connectivité réseau ou d'un rôle IAM incorrect. Dans certains cas, la tentative d'activation de Kerberos l'authentification peut échouer lorsque vous créez ou modifiez une instance de base de données. Dans ce cas, vérifiez que vous utilisez le rôle IAM approprié, puis modifiez l'instance de base de données pour rejoindre le domaine.

Configuration de Kerberos l'authentification pour Amazon RDS pour les instances de base de données DB2

Vous utilisez AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD) pour configurer l'Kerberosauthentification pour une instance de base de données RDS pour DB2. Pour configurer Kerberos l'authentification, procédez comme suit :

Rubriques

- Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD
- Étape 2 : Créer une approbation
- Étape 3 : créer un rôle IAM auquel Amazon RDS pourra accéder AWS Directory Service

- Étape 4 : Créer et configurer des utilisateurs
- Étape 5 : créer un groupe d'administrateurs RDS pour DB2 dans AWS Managed Microsoft AD
- Étape 6 : Modifier le paramètre de base de données
- Étape 7 : créer ou modifier une instance de base de données RDS pour DB2
- Étape 8 : récupérer le SID du groupe Active Directory dans PowerShell
- Étape 9 : ajouter un SID aux GroupName mappages vers votre instance de base de données RDS pour DB2
- Étape 10 : Configuration d'un client DB2

Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD

AWS Directory Service crée un système entièrement géré Active Directory dans le AWS Cloud. Lorsque vous créez un AWS Managed Microsoft AD annuaire, il AWS Directory Service crée deux contrôleurs de domaine et deux serveurs DNS pour vous. Les serveurs de répertoire sont créés dans des sous-réseaux différents d'un VPC. Cette redondance permet de s'assurer que votre répertoire reste accessible y compris en cas de défaillance.

Lorsque vous créez un AWS Managed Microsoft AD répertoire, AWS Directory Service exécute les tâches suivantes en votre nom :

- Configure un Active Directory au sein de votre VPC.
- Création d'un compte d'administrateur d'annuaire avec le nom d'utilisateur `Admin` et le mot de passe spécifié. Ce compte est utilisé pour gérer votre annuaire.

  > **Important**
  >
  > Assurez-vous d'enregistrer ce mot de passe. AWS Directory Service ne stocke pas ce mot de passe et il ne peut pas être récupéré ou réinitialisé.

- Création d'un groupe de sécurité pour les contrôleurs de l'annuaire. Le groupe de sécurité doit autoriser la communication avec le RDS pour l'instance de base de données DB2.

Lorsque vous lancez AWS Directory Service for Microsoft Active Directory, AWS crée une unité organisationnelle (UO) contenant tous les objets de votre répertoire. Cette unité d'organisation, qui porte le nom NetBIOS que vous avez entré lorsque vous avez créé votre annuaire, est située dans la racine du domaine. La racine du domaine est détenue et gérée par AWS.

Le `Admin` compte créé avec votre AWS Managed Microsoft AD annuaire dispose d'autorisations pour les activités administratives les plus courantes de votre unité d'organisation :

- Créez, mettez à jour ou supprimez des utilisateurs.
- Ajoutez des ressources à votre domaine, telles que des serveurs de fichiers ou d'impression, puis attribuez des autorisations pour ces ressources aux utilisateurs de votre unité d'organisation.
- Créez OUs des conteneurs supplémentaires.
- Déléguer des autorités.
- Restaurez les objets supprimés de la Active Directory corbeille.
- Exécutez Active Directory et les modules DNS (Domain Name Service) pour Windows PowerShell le AWS Directory Service.

Le compte `Admin` dispose également de droits pour exécuter les activités suivantes au niveau du domaine :

- Gérer les configurations DNS (ajouter, supprimer ou mettre à jour des enregistrements, des zones et des redirecteurs)
- Afficher les journaux d'événements DNS.
- Afficher les journaux d'événements de sécurité.

Pour créer un répertoire avec AWS Managed Microsoft AD

1. Connectez-vous à la AWS Directory Service console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/directoryservicev2/.
2. Choisissez Configurer le répertoire.
3. Choisissez AWS Managed Microsoft AD. AWS Managed Microsoft AD est la seule option actuellement prise en charge pour une utilisation avec Amazon RDS.
4. Choisissez Suivant.
5. Sur la page Enter directory information (Saisir les détails du répertoire), renseignez les informations suivantes :
   - Édition — Choisissez l'édition qui répond à vos besoins.
   - Nom DNS du répertoire : nom complet du répertoire, tel que`corp.example.com`.
   - Nom NetBIOS du répertoire : nom abrégé facultatif pour le répertoire, tel que. `CORP`

- Description du répertoire : description facultative du répertoire.
- Mot de passe administrateur : mot de passe de l'administrateur du répertoire. Le processus de création du répertoire crée un compte administrateur avec le nom d'utilisateur `Admin` et ce mot de passe.

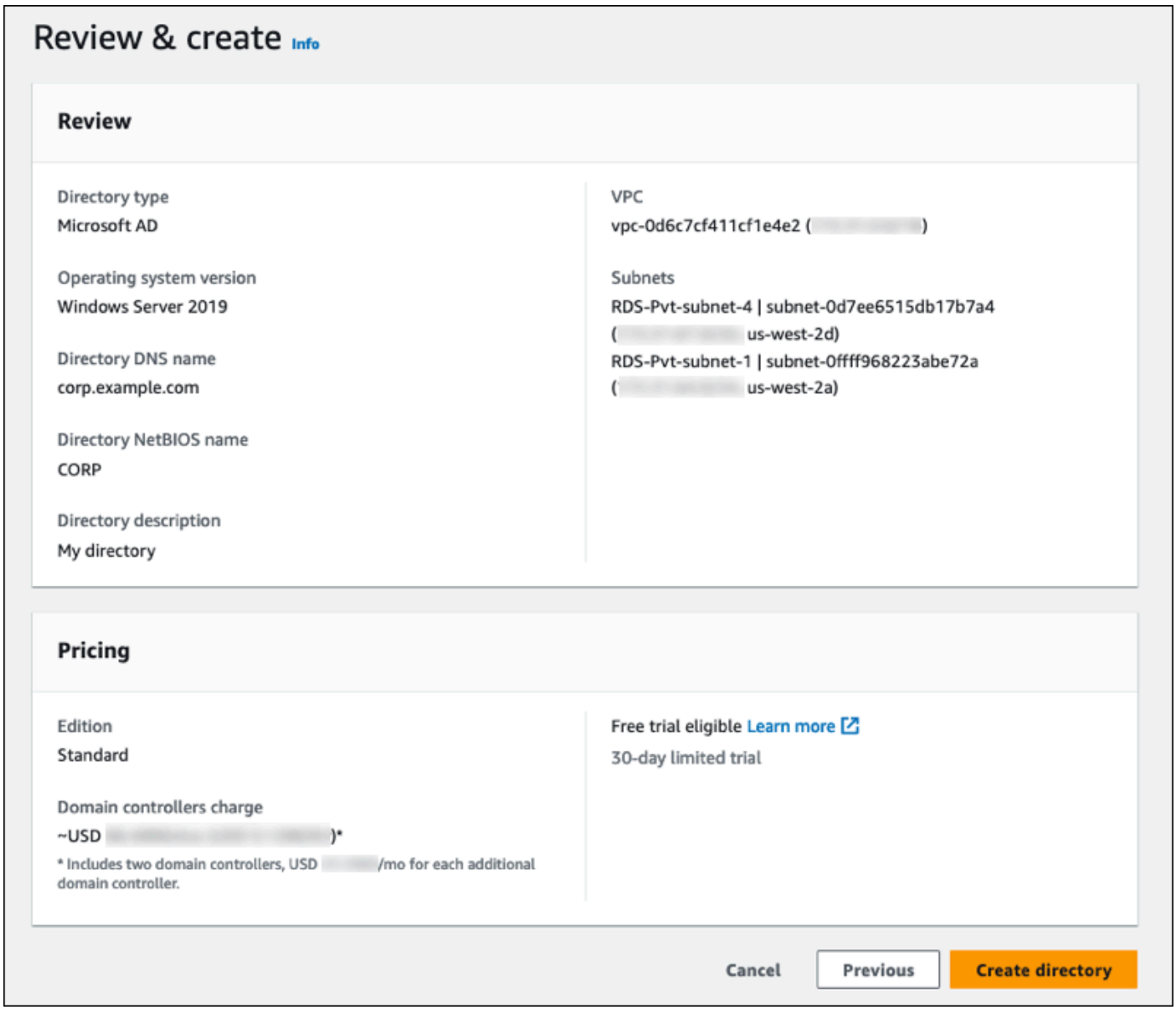
  Le mot de passe de l'administrateur de l'annuaire ne peut pas contenir le terme « admin ». Le mot de passe est sensible à la casse et doit comporter entre 8 et 64 caractères. Il doit également contenir au moins un caractère de trois des quatre catégories suivantes :

  - Lettres minuscules (a–z)
  - Lettres majuscules (A–Z)
  - Chiffres (0–9)
  - Caractères non alphanumériques (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)
  - Confirmer le mot de passe — Entrez à nouveau le mot de passe administrateur.

  > **Important**
  >
  > Assurez-vous d'enregistrer ce mot de passe. AWS Directory Service ne stocke pas ce mot de passe et il ne peut pas être récupéré ou réinitialisé.

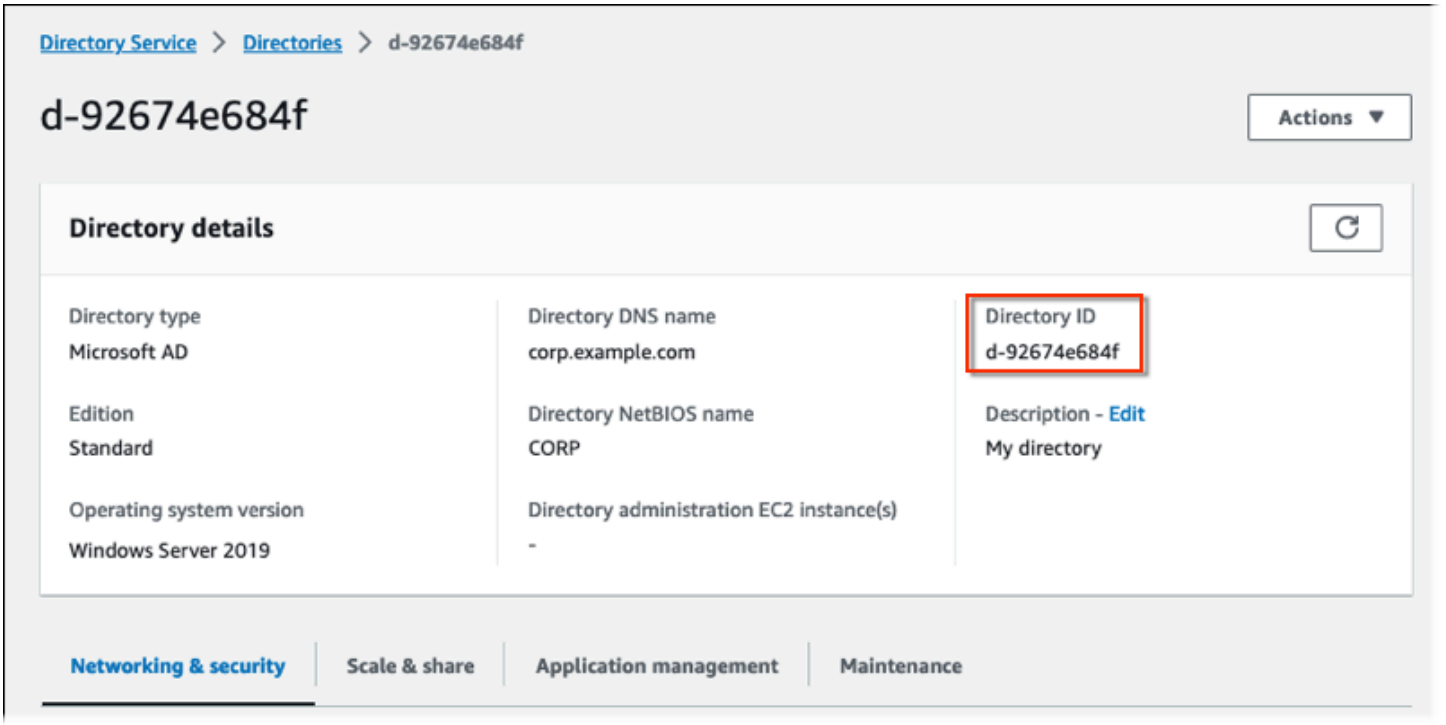
6. Choisissez Suivant.
7. Sur la page Choose VPC and subnets (Choisir un VPC et des sous-réseaux), indiquez les informations suivantes :

   - VPC — Choisissez le VPC pour le répertoire. Vous pouvez créer l'instance de base de données RDS pour Db2 dans ce même VPC ou dans un autre VPC.
   - Sous-réseaux : choisissez les sous-réseaux pour les serveurs d'annuaire. Les deux sous-réseaux doivent être dans des zones de disponibilité différentes.
8. Choisissez Suivant.
9. Vérifiez les informations du répertoire. Si vous devez apporter des modifications, choisissez Previous (Précédent) et entrez ces modifications. Lorsque les informations sont correctes, choisissez Create directory (Créer l'annuaire).



La création de l'annuaire prend plusieurs minutes. Lorsqu'il est créé, la valeur du champ Status (Statut) devient Active (Actif).

Pour consulter les informations relatives à votre répertoire, choisissez l'ID du répertoire sous ID du répertoire. Notez la valeur de Directory ID (ID du répertoire). Vous avez besoin de cette valeur lorsque vous créez ou modifiez votre instance de base de données RDS pour DB2.



Étape 2 : Créer une approbation

Si vous prévoyez d'utiliser AWS Managed Microsoft AD uniquement, passez àÉtape 3 : créer un rôle IAM auquel Amazon RDS pourra accéder AWS Directory Service.

Pour activer l'authentification Kerberos à l'aide de votre Active Directory autogéré, vous devez créer une relation de confiance forestière entre votre Active Directory autogéré et le. Une approbation forestière est une relation de confiance entre un Microsoft AD et un Active Directory autogéré et AWS Managed Microsoft AD créé à l'étape précédente. L'approbation peut également être bidirectionnelle. Dans ce cas, les deux Active Directory s'approuvent mutuellement. Pour plus d'informations sur la configuration des approbations forestières à l'aide AWS Directory Service de la section Quand créer une relation d'approbation dans le Guide d'administration du AWS Directory Service.

Étape 3 : créer un rôle IAM auquel Amazon RDS pourra accéder AWS Directory Service

Pour qu'Amazon RDS puisse vous appeler AWS Directory Service , vous avez Compte AWS besoin d'un rôle IAM qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess` Ce rôle permet à Amazon RDS de passer des appels à AWS Directory Service.

Lorsque vous créez une instance de base de données à l'aide de AWS Management Console et que votre compte utilisateur de console dispose de l'`iam:CreateRole`autorisation, la console crée automatiquement le rôle IAM nécessaire. Dans ce cas, le nom du rôle est `rds-`

`directoryservice-kerberos-access-role`. Sinon, vous devez créer le rôle IAM manuellement. Lorsque vous créez ce rôle IAM`Directory Service`, choisissez et associez la politique AWS gérée `AmazonRDSDirectoryServiceAccess` à celui-ci.

Pour plus d'informations sur la création de rôles IAM pour un service, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'utilisateur IAM.

> **Note**
>
> Le rôle IAM utilisé pour Windows l'authentification pour RDS pour ne Microsoft SQL Server peut pas être utilisé pour RDS pour DB2.

Vous pouvez également créer des stratégies avec les autorisations obligatoires au lieu d'utiliser la politique gérée `AmazonRDSDirectoryServiceAccess`. Dans ce cas, le rôle IAM doit respecter la politique de confiance IAM suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.amazonaws.com",
          "rds.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Le rôle doit également respecter la politique de rôle IAM suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ds:DescribeDirectories",
        "ds:AuthorizeApplication",
        "ds:UnauthorizeApplication",
        "ds:GetAuthorizedApplicationDetails"
      ],
    "Effect": "Allow",
    "Resource": "*"
    }
  ]
}
```

Étape 4 : Créer et configurer des utilisateurs

Vous pouvez créer des utilisateurs à l'aide de l'Active Directory Users and Computersoutil. C'est l'un Active Directory Domain Services des Active Directory Lightweight Directory Services outils. Pour plus d'informations, consultez la section Ajouter des utilisateurs et des ordinateurs au Active Directory domaine dans la Microsoft documentation. Dans ce cas, les utilisateurs sont des individus ou d'autres entités, tels que leurs ordinateurs, qui font partie du domaine et dont l'identité est conservée dans l'annuaire.

Pour créer des utilisateurs dans un AWS Directory Service annuaire, vous devez être connecté à une EC2 instance Amazon Windows basée qui est membre de l' AWS Directory Service annuaire. Dans le même temps, vous devez être connecté en tant qu'utilisateur autorisé à créer des utilisateurs. Pour plus d'informations, consultez Créer un utilisateur dans le Guide d'administration AWS Directory Service .

Étape 5 : créer un groupe d'administrateurs RDS pour DB2 dans AWS Managed Microsoft AD

RDS pour Db2 ne prend pas en charge Kerberos l'authentification de l'utilisateur principal ou des deux utilisateurs réservés Amazon RDS et. `rdsdb` `rdsadmin` Vous devez plutôt créer un nouveau groupe appelé `masterdba` in AWS Managed Microsoft AD. Pour plus d'informations, voir Créer un

compte de groupe Active Directory dans la Microsoft documentation. Tous les utilisateurs que vous ajoutez à ce groupe auront des privilèges d'utilisateur principal.

Une fois Kerberos l'authentification activée, l'utilisateur principal perd son `masterdba` rôle. Par conséquent, l'utilisateur principal ne pourra pas accéder à l'appartenance au groupe d'utilisateurs local de l'instance à moins que vous ne désactiviez Kerberos l'authentification. Pour continuer à utiliser l'utilisateur principal connecté par mot de passe, créez un utilisateur AWS Managed Microsoft AD portant le même nom que l'utilisateur principal. Ajoutez ensuite cet utilisateur au groupe`masterdba`.

Étape 6 : Modifier le paramètre de base de données

Si vous prévoyez d'utiliser AWS Managed Microsoft AD uniquement, passez àÉtape 7 : créer ou modifier une instance de base de données RDS pour DB2.

Pour activer l'authentification Kerberos à l'aide de votre Active Directory autogéré, vous devez définir le paramètre sur `AWS_MANAGED_AD_WITH_TRUST` dans votre groupe `rds.active_directory_configuration` de paramètres. Par défaut, ce paramètre est défini sur `AWS_MANAGED_AD` pour l'utilisation de AWS Managed Microsoft AD uniquement.

Pour plus d'informations sur la modification des paramètres de base de données, consultezModification des paramètres dans les groupes de paramètres.

Étape 7 : créer ou modifier une instance de base de données RDS pour DB2

Créez ou modifiez une instance de base de données RDS pour DB2 à utiliser avec votre annuaire. Vous pouvez utiliser l'API AWS Management Console AWS CLI, le ou l'API RDS pour associer une instance de base de données à un annuaire. Vous pouvez effectuer cette opération de différentes manières :

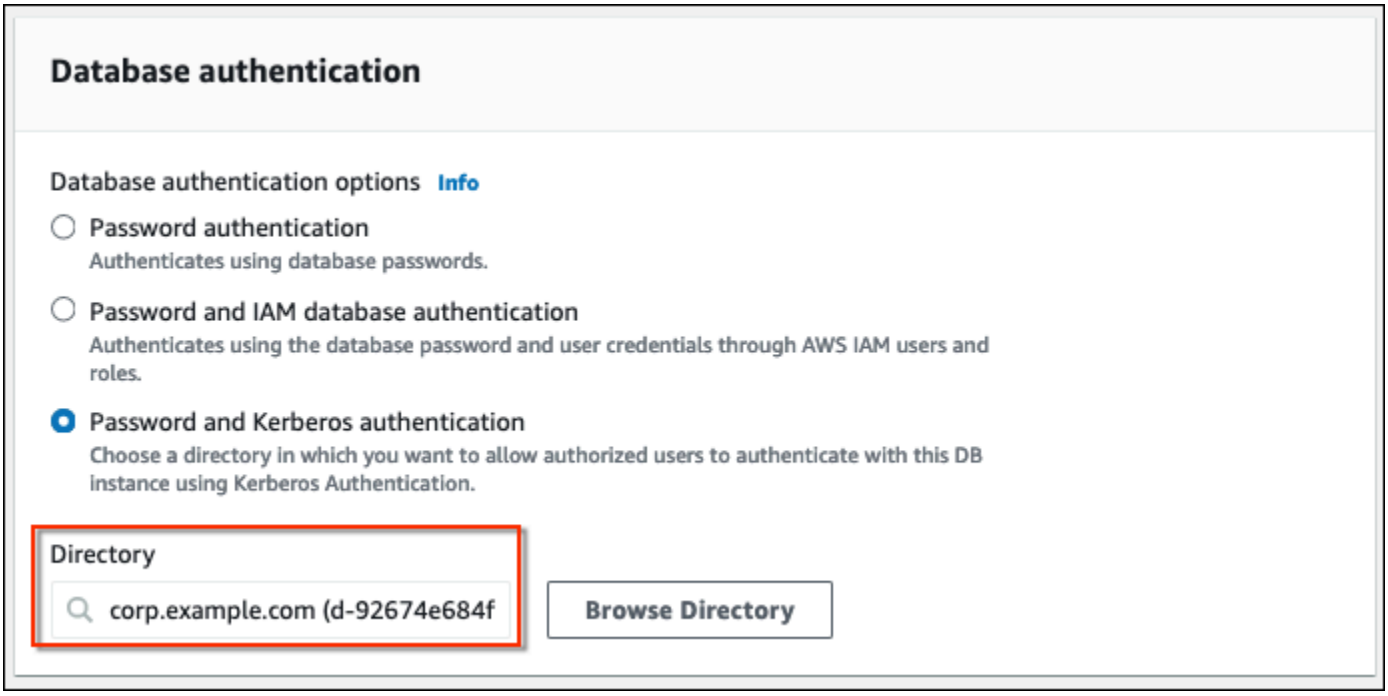
- Créez une nouvelle instance de base de données RDS pour DB2 à l'aide de la console, de la create-db-instancecommande ou de l'opération CreateDBInstanceAPI. Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données RDS pour DB2 existante à l'aide de la console, de la modify-db-instancecommande ou de l'opération Modify DBInstance API. Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Restaurez une instance de base de données RDS pour DB2 à partir d'un instantané de base de données à l'aide de la console, de la restore-db-instance-from-db-snapshotcommande ou de l'opération RestoreDBInstanceFromDBSnapshotAPI. Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.

- Restaurez une instance de base de données RDS pour DB2 à point-in-time l'aide de la console, de la restore-db-instance-to-point-in-timecommande ou de l'opération RestoreDBInstanceToPointInTimeAPI. Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Kerberosl'authentification n'est prise en charge que pour les instances de base de données RDS pour DB2 dans un VPC. L'instance de base de données peut être dans le même VPC que l'annuaire ou dans un VPC différent. L'instance de base de données doit utiliser un groupe de sécurité qui autorise l'entrée et la sortie au sein du VPC du répertoire afin que l'instance de base de données puisse communiquer avec l'annuaire.

console

Lorsque vous utilisez la console pour créer, modifier ou restaurer une instance de base de données, choisissez Mot de passe et Kerberos authentification dans la section Authentification de base de données. Ensuite, choisissez Browse Directory (Parcourir le répertoire). Sélectionnez le répertoire ou choisissez Create directory pour utiliser le Directory Service.



AWS CLI

Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de base de données puisse utiliser le répertoire que vous avez créé :

- Pour le `--domain` paramètre, utilisez l'identifiant de domaine (`d-*`« identifiant ») généré lors de la création du répertoire.
- Pour le paramètre `--domain-iam-role-name`, utilisez le rôle que vous avez créé qui utilise la politique IAM gérée `AmazonRDSDirectoryServiceAccess`.

L'exemple suivant modifie une instance de base de données pour utiliser un répertoire. Remplacez les espaces réservés suivants dans l'exemple par vos propres valeurs :

- *`db_instance_name`*— Le nom de votre instance de base de données RDS pour DB2.
- *`directory_id`*— L'ID du AWS Directory Service for Microsoft Active Directory répertoire que vous avez créé.
- *`role_name`*— Le nom du rôle IAM que vous avez créé.

```
aws rds modify-db-instance --db-instance-identifier db_instance_name --domain
 d-directory_id --domain-iam-role-name role_name
```

> **Important**
>
> Si vous modifiez une instance de base de données pour activer Kerberos l'authentification, redémarrez-la après avoir effectué la modification.

Étape 8 : récupérer le SID du groupe Active Directory dans PowerShell

Un ID de sécurité (SID) identifie de manière unique un principal de sécurité ou un groupe de sécurité. Lorsqu'un groupe ou un compte de sécurité est créé dans Active Directory, Active Directory attribue un SID au groupe. Pour récupérer le SID du groupe de sécurité AD depuis Active Directory, utilisez l'`Get-ADGroup`applet de commande sur un ordinateur client Windows qui fait partie du domaine Active Directory. Le `Identity` paramètre indique le nom du groupe Active Directory pour lequel vous souhaitez obtenir le SID.

L'exemple suivant renvoie le SID du groupe Active Directory`adgroup1`.

```
C:\Users\Admin> Get-ADGroup -Identity adgroup1 | select SID

             SID
-----------------------------------------------
```

```
S-1-5-21-3168537779-1985441202-1799118680-1612
```

Vous devez générer ce mappage pour tous les groupes concernés par la base de données.

Étape 9 : ajouter un SID aux GroupName mappages vers votre instance de base de données RDS pour DB2

Vous devez ajouter le SID aux GroupName mappages créés à l'étape précédente à votre instance de base de données RDS pour DB2. Pour chaque mappage, appelez la procédure stockée suivante. Remplacez le *SID* et *group_name* par vos propres informations.

```
db2 connect to rdsadmin
db2 "call rdsadmin.set_sid_group_mapping(?, 'SID','group_name')"
```

Pour de plus amples informations, veuillez consulter rdsadmin.set_sid_group_mapping.

Pour plus d'informations sur la vérification de l'état de la tâche, consultezrdsadmin.get_task_status.

Étape 10 : Configuration d'un client DB2

Pour configurer un client DB2

1. Créez un fichier /etc/krb5.conf (ou équivalent) pour pointer vers le domaine.

   > **Note**
   >
   > Pour les systèmes d'exploitation Windows, créez un fichier C:\windows\krb5.ini.

2. Vérifiez que le trafic peut circuler entre l'hôte client et AWS Directory Service. Utilisez un utilitaire réseau tel que Netcat pour les tâches suivantes :

   a. Vérifiez le trafic via DNS pour le port 53.

   b. Vérifiez que le trafic est dépassé TCP/UDP pour le port 53 et pourKerberos, y compris les ports 88 et 464 pour AWS Directory Service.

3. Vérifiez que le trafic peut circuler entre l'hôte du client et l'instance de base de données via le port de la base de données. Vous pouvez utiliser la commande db2 pour vous connecter et accéder à la base de données.

L'exemple suivant est le contenu du fichier /etc/krb5.conf pour : AWS Managed Microsoft AD

```
[libdefaults]
default_realm = EXAMPLE.COM
[realms]
EXAMPLE.COM = {
kdc = example.com
admin_server = example.com
}
[domain_realm]
.example.com = EXAMPLE.COM
example.com = EXAMPLE.COM
```

Connexion à Amazon RDS pour DB2 avec Kerberos authentification

Utilisez la procédure suivante pour vous connecter à votre instance de base de données Amazon RDS pour DB2 avec Kerberos authentification.

Pour vous connecter à RDS for DB2 avec Kerberos authentification

1. À partir d'une invite de commande, exécutez la commande suivante. Dans l'exemple suivant, remplacez *username* avec votre Microsoft Active Directory nom d'utilisateur.

   ```
   kinit username
   ```

2. Si l'instance de base de données RDS for DB2 utilise un accès publicVPC, ajoutez l'adresse IP du point de terminaison de votre instance de base de données à votre `/etc/hosts` fichier sur le EC2 client Amazon. L'exemple suivant obtient l'adresse IP, puis l'ajoute au `/etc/hosts` fichier.

   ```
   % dig +short Db2-endpoint.AWS-Region.rds.amazonaws.com
   ;; Truncated, retrying in TCP mode.
   ec2-34-210-197-118.AWS-Region.compute.amazonaws.com.
   34.210.197.118

   % echo "34.210.197.118  Db2-endpoint.AWS-Region.rds.amazonaws.com" >> /etc/hosts
   ```

3. Utilisez la commande suivante pour vous connecter à une instance de base de données RDS for DB2 associée à Active Directory. Remplacer *database_name* avec le nom de votre base de données RDS for DB2.

   ```
   db2 connect to database_name
   ```

Administration de votre instance de base de données Amazon RDS pour DB2

Cette rubrique couvre les tâches de gestion courantes que vous effectuez avec une instance de base de données Amazon RDS pour DB2. Certaines tâches sont identiques pour toutes les instances de base de données Amazon RDS. D'autres tâches sont spécifiques à RDS pour DB2.

Les tâches suivantes sont communes à toutes les bases de données RDS. Il existe également des tâches spécifiques à RDS pour DB2, telles que la connexion à une base de données RDS pour DB2 avec un client SQL standard.

Type de tâche	Documentation
Classes d'instance, stockage et PIOPS Si vous créez une instance de production, découvrez comment fonctionnent les classes d'instance, les types de stockage et les IOPS provisionnées dans Amazon RDS.	Classes d'instances de base de données Types de stockage Amazon RDS
Déploiements multi-AZ Une instance de base de données de production doit utiliser des déploiements multi-AZ. Les déploiements Multi-AZ améliorent la disponibilité, la durabilité des donnés et la tolérance aux pannes pour les instances de bases de données.	Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS
Amazon VPC Si vous disposez Compte AWS d'un cloud privé virtuel (VPC) par défaut, votre instance de base de données est automatiq uement créée dans le VPC par défaut. Si votre compte n'a pas de VPC par défaut et que vous voulez que l'instance de base de données soit dans un VPC, créez le VPC et les groupes de sous-réseaux avant de créer l'instance de base de données.	Utilisation d'un(e) instance de base de données dans un VPC
Groupes de sécurité Par défaut, les instances de base de données utilisent un pare-feu qui empêche l'accès. Veillez à créer un groupe de sécurité	Contrôle d'accès par groupe de sécurité

Type de tâche	Documentation
avec les adresses IP et la configuration réseau voulues pour accéder à l'instance de base de données.	
Groupes de paramètres Étant donné que votre instance de base de données RDS pour DB2 nécessite que vous ajoutiez les `rds.ibm_site_id` paramètres `rds.ibm_customer_id` et, créez un groupe de paramètres avant de créer l'instance de base de données. Si votre instance de base de données nécessite d'autres paramètre s de base de données spécifiques, ajoutez-les également à ce groupe de paramètres avant de créer l'instance de base de données.	Ajout IBM IDs à un groupe de paramètres pour les instances de base de données RDS pour DB2 Groupes de paramètres pour (Amazon RDS)
Groupes d'options Si votre instance de base de données nécessite des options spécifiques, créez un groupe d'options avant de créer l'instance de base de données.	Options pour Amazon RDS pour les instances de base de données DB2
Connexion à votre instance de base de données Après avoir créé un groupe de sécurité et l'avoir associé à une instance de base de données, vous pouvez vous connecter à l'instance de base de données avec n'importe quelle application cliente SQL standard telle queIBM Db2 CLP.	Connexion à votre instance de base de données DB2
Sauvegarde et restauration Vous pouvez configurer votre instance de base de données pour effectuer des sauvegardes de stockage automatisées ou prendre des instantanés de stockage manuels, puis restaurer des instances à partir des sauvegardes ou des instantanés.	Sauvegarde, restauration et exportation de données

Type de tâche	Documentation
Surveillance Vous pouvez surveiller une instance de base de données RDS pour DB2 avec. IBM Db2 Data Management Console Vous pouvez également surveiller une instance de base de données RDS pour DB2 à l'aide des métriques, des événements et de la surveillance améliorée d' CloudWatch Amazon RDS.	Connexion à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console Afficher les métriques dans la RDS console Amazon Consulter les RDS événement s Amazon Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée
Les fichiers journaux Vous pouvez accéder aux fichiers journaux de votre instance de base de données RDS pour DB2.	Surveillance des fichiers journaux RDSAmazon

Rubriques

- Exécution de tâches système courantes pour les instances de base de données Amazon RDS pour Db2
- Exécution de tâches de base de données courantes pour les instances RDS de base de données Amazon pour Db2

Exécution de tâches système courantes pour les instances de base de données Amazon RDS pour Db2

Vous pouvez effectuer certaines tâches courantes d'administrateur de base de données liées au système sur vos instances Amazon RDS DB exécutant Db2. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données et restreint l'accès à certaines procédures et tables du système qui nécessitent des privilèges avancés.

Pour plus d'informations sur l'octroi et la révocation de privilèges et sur l'attachement à la base de données distante RDS pour Db2, consultez les rubriques suivantes.

Rubriques

- Octroi et révocation de privilèges pour RDS pour Db2
- Connexion à la télécommande RDS pour une instance de base de données DB2

Créer un point de terminaison de base de données personnalisé

Lorsque vous migrez vers Amazon RDS pour Db2, vous pouvez utiliser un point de terminaison de base de données personnalisé afin URLs de minimiser les modifications apportées à votre application. Par exemple, si vous l'utilisez `db2.example.com` comme DNS enregistrement actuel, vous pouvez l'ajouter à Amazon Route 53. Dans Route 53, vous pouvez utiliser des zones hébergées privées pour mapper le point de terminaison de votre DNS base de données actuel à un point de terminaison RDS de base de données DB2. Pour ajouter une personnalisation `A` ou un `CNAME` enregistrement pour un point de terminaison RDS de base de données Amazon, consultez la section Enregistrement et gestion de domaines à l'aide d'Amazon Route 53 dans le guide du développeur Amazon Route 53.

> **Note**
>
> Si vous ne pouvez pas transférer votre domaine vers Route 53, vous pouvez utiliser votre DNS fournisseur pour créer un `CNAME` enregistrement pour le point de terminaison URL de la base de données RDS for Db2. Consultez la documentation DNS de votre fournisseur.

Octroi et révocation de privilèges pour RDS pour Db2

Les utilisateurs accèdent aux bases de données par le biais de l'appartenance à des groupes attachés aux bases de données.

Utilisez les procédures suivantes pour accorder et révoquer des privilèges afin de contrôler l'accès à votre base de données.

Ces procédures utilisent l'IBM Db2 CLPexécution sur une machine locale pour se connecter à une instance de base de données RDS pour DB2. Assurez-vous de cataloguer le nœud TCP/IP et la base de données pour vous connecter à votre instance de base de données RDS pour DB2 exécutée

sur votre machine locale. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données Amazon RDS pour DB2 avec IBM Db2 CLP.

Rubriques

- Accorder à un utilisateur l'accès à votre base de données
- Modifier le mot de passe d'un utilisateur
- Ajouter des groupes à un utilisateur
- Supprimer des groupes d'un utilisateur
- Supprimer un utilisateur
- Lister les utilisateurs
- Création d'un rôle
- Octroi d'un rôle
- Révocation d'un rôle
- Supprimer un rôle
- Octroi des autorisations de base
- Révocation de l'autorisation de base de données

Accorder à un utilisateur l'accès à votre base de données

Pour autoriser un utilisateur à accéder à votre base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

   Cette commande produit une sortie similaire à l'exemple suivant :

   ```
   Database Connection Information

   Database server        = DB2/LINUXX8664 11.5.8.0
   SQL authorization ID   = ADMIN
   Local database alias   = RDSADMIN
   ```

2. Ajoutez un utilisateur à votre liste d'autorisation en appelant`rdsadmin.add_user`. Pour de plus amples informations, veuillez consulter rdsadmin.add_user.

```
db2 "call rdsadmin.add_user(
    'username',
    'password',
    'group_name,group_name')"
```

3. (Facultatif) Ajoutez des groupes supplémentaires à l'utilisateur en appelant`rdsadmin.add_groups`. Pour de plus amples informations, veuillez consulter rdsadmin.add_groups.

```
db2 "call rdsadmin.add_groups(
    'username',
    'group_name,group_name')"
```

4. Confirmez les autorisations mises à la disposition de l'utilisateur. Dans l'exemple suivant, remplacez *rds_database_aliasmaster_user*, et *master_password* par vos propres informations. Remplacez également *username* par le nom d'utilisateur de l'utilisateur.

```
db2 terminate
db2 connect to rds_database_alias user master_user using master_password
db2 "SELECT SUBSTR(AUTHORITY,1,20) AUTHORITY, D_USER, D_GROUP, D_PUBLIC
       FROM TABLE (SYSPROC.AUTH_LIST_AUTHORITIES_FOR_AUTHID ('username', 'U') ) AS
 T
       ORDER BY AUTHORITY"
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
AUTHORITY            D_USER D_GROUP D_PUBLIC
-------------------- ------ ------- --------
ACCESSCTRL           N      N       N
BINDADD              N      N       N
CONNECT              N      N       N
CREATETAB            N      N       N
CREATE_EXTERNAL_ROUT N      N       N
CREATE_NOT_FENCED_RO N      N       N
CREATE_SECURE_OBJECT N      N       N
DATAACCESS           N      N       N
DBADM                N      N       N
EXPLAIN              N      N       N
```

```
IMPLICIT_SCHEMA      N      N      N
LOAD                 N      N      N
QUIESCE_CONNECT      N      N      N
SECADM               N      N      N
SQLADM               N      N      N
SYSADM               *      N      *
SYSCTRL              *      N      *
SYSMAINT             *      N      *
SYSMON               *      N      *
WLMADM               N      N      N
```

5. Accordez le RDS pour les rôles ROLE_NULLID_PACKAGES DB2 et ROLE_PROCEDURES au groupe auquel vous avez ajouté l'utilisateur. ROLE_TABLESPACES Pour de plus amples informations, veuillez consulter Rôles par défaut RDS d'Amazon pour DB2.

> **Note**
>
> Nous créons des RDS pour les instances de base de données DB2 en RESTRICTIVE mode. Par conséquent, le RDS pour les rôles ROLE_NULLID_PACKAGES DB2 et ROLE_PROCEDURES accorde ROLE_TABLESPACES des privilèges d'exécution sur les NULLID packages pour IBM Db2 CLP et. Dynamic SQL Ces rôles accordent également des privilèges aux utilisateurs sur les tablespaces.

a. Connectez-vous à votre base de données DB2. Dans l'exemple suivant, remplacez *database_namemaster_user*, et *master_password* par vos propres informations.

```
db2 connect to database_name user master_user using master_password
```

b. Accordez le rôle ROLE_NULLED_PACKAGES à un groupe. Dans l'exemple suivant, remplacez *group_name* par le nom du groupe auquel vous souhaitez ajouter le rôle.

```
db2 "grant role ROLE_NULLID_PACKAGES to group group_name"
```

c. Accordez le rôle ROLE_TABLESPACES au même groupe. Dans l'exemple suivant, remplacez *group_name* par le nom du groupe auquel vous souhaitez ajouter le rôle.

```
db2 "grant role ROLE_TABLESPACES to group group_name"
```

d. Accordez le rôle `ROLE_PROCEDURES` au même groupe. Dans l'exemple suivant, remplacez *`group_name`* par le nom du groupe auquel vous souhaitez ajouter le rôle.

```
db2 "grant role ROLE_PROCEDURES to group group_name"
```

6. Accordez `connect` `bindadd``createtab`,, et des `IMPLICIT_SCHEMA` autorisations au groupe auquel vous avez ajouté l'utilisateur. Dans l'exemple suivant, remplacez *`group_name`* par le nom du deuxième groupe auquel vous avez ajouté l'utilisateur.

```
db2 "grant usage on workload SYSDEFAULTUSERWORKLOAD to public"
db2 "grant connect, bindadd, createtab, implicit_schema on database to
 group group_name"
```

7. Répétez les étapes 4 à 6 pour chaque groupe supplémentaire auquel vous avez ajouté l'utilisateur.
8. Testez l'accès de l'utilisateur en se connectant en tant qu'utilisateur, en créant une table, en insérant des valeurs dans la table et en renvoyant les données de la table. Dans l'exemple suivant, remplacez *`rds_database_aliasusername`*, et *`password`* par le nom de la base de données ainsi que le nom d'utilisateur et le mot de passe de l'utilisateur.

```
db2 connect to rds_database_alias user username using password
db2 "create table t1(c1 int not null)"
db2 "insert into t1 values (1),(2),(3),(4)"
db2 "select * from t1"
```

Modifier le mot de passe d'un utilisateur

Pour modifier le mot de passe d'un utilisateur

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

```
db2 connect to rdsadmin user master_username using master_password
```

2. Changez le mot de passe en appelant`rdsadmin.change_password`. Pour de plus amples informations, veuillez consulter rdsadmin.change_password.

```
db2 "call rdsadmin.change_password(
    'username',
    'new_password')"
```

Ajouter des groupes à un utilisateur

Pour ajouter des groupes à un utilisateur

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Ajoutez des groupes à un utilisateur en appelant`rdsadmin.add_groups`. Pour de plus amples informations, veuillez consulter rdsadmin.add_groups.

   ```
   db2 "call rdsadmin.add_groups(
       'username',
       'group_name,group_name')"
   ```

Supprimer des groupes d'un utilisateur

Pour supprimer des groupes d'un utilisateur

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Supprimez des groupes en appelant`rdsadmin.remove_groups`. Pour de plus amples informations, veuillez consulter rdsadmin.remove_groups.

   ```
   db2 "call rdsadmin.remove_groups(
       'username',
       'group_name,group_name')"
   ```

Supprimer un utilisateur

Pour supprimer un utilisateur de la liste d'autorisations

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Supprimez un utilisateur de votre liste d'autorisation en appelant`rdsadmin.remove_user`. Pour de plus amples informations, veuillez consulter rdsadmin.remove_user.

   ```
   db2 "call rdsadmin.remove_user('username')"
   ```

Lister les utilisateurs

Pour répertorier les utilisateurs sur une liste d'autorisation, appelez la procédure `rdsadmin.list_users` stockée. Pour de plus amples informations, veuillez consulter rdsadmin.list_users.

```
db2 "call rdsadmin.list_users()"
```

Création d'un rôle

Vous pouvez utiliser la procédure rdsadmin.create_role stockée pour créer un rôle.

Pour créer un rôle

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Réglez Db2 pour qu'il affiche le contenu.

   ```
   db2 set serveroutput on
   ```

3. Créez un rôle. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.create_role”.

```
db2 "call rdsadmin.create_role(
    'database_name',
    'role_name')"
```

4. Configurez Db2 pour qu'il ne produise pas de contenu.

```
db2 set serveroutput off
```

Octroi d'un rôle

Vous pouvez utiliser la procédure rdsadmin.grant_role stockée pour attribuer un rôle à un rôle, à un utilisateur ou à un groupe.

Pour attribuer un rôle

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *master_username* et *master_password* par vos informations.

```
db2 connect to rdsadmin user master_username using master_password
```

2. Réglez Db2 pour qu'il affiche le contenu.

```
db2 set serveroutput on
```

3. Attribuez un rôle. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.grant_role”.

```
db2 "call rdsadmin.grant_role(
    'database_name',
    'role_name',
    'grantee',
    'admin_option')"
```

4. Configurez Db2 pour qu'il ne produise pas de contenu.

```
db2 set serveroutput off
```

Révocation d'un rôle

Vous pouvez utiliser la procédure rdsadmin.revoke_role stockée pour révoquer le rôle d'un rôle, d'un utilisateur ou d'un groupe.

Pour révoquer un rôle

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Révoquer un rôle. Pour de plus amples informations, veuillez consulter the section called "rdsadmin.revoke_role".

   ```
   db2 "call rdsadmin.revoke_role(
       ?,
       'database_name',
       'role_name',
       'grantee')"
   ```

Supprimer un rôle

Vous pouvez utiliser la procédure rdsadmin.drop_role stockée pour supprimer un rôle.

Pour supprimer un rôle

1. Connectez-vous à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Supprimez un rôle. Pour de plus amples informations, veuillez consulter the section called "rdsadmin.drop_role".

   ```
   db2 "call rdsadmin.drop_role(
       ?,
       'database_name',
       'role_name')"
   ```

Octroi des autorisations de base

L'utilisateur principal, qui dispose `DBADM` d'une autorisation `DBADMACCESSCTRL`, peut accorder ou `DATAACCESS` autoriser un rôle, un utilisateur ou un groupe.

Pour accorder l'autorisation de base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Accordez l'accès à un utilisateur en appelant`rdsadmin.dbadm_grant`. Pour de plus amples informations, veuillez consulter rdsadmin.dbadm_grant.

   ```
   db2 "call rdsadmin.dbadm_grant(
       ?,
       'database_name,
       'authorization',
       'grantee')"
   ```

Exemple de cas d'utilisation

La procédure suivante vous explique comment créer un rôle, accorder une `DBADM` autorisation au rôle, attribuer le rôle à un utilisateur et octroyer le rôle à un groupe.

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Créez un rôle appelé `PROD_ROLE` pour une base de données appeléeTESTDB. Pour de plus amples informations, veuillez consulter rdsadmin.create_role.

   ```
   db2 "call rdsadmin.create_role(
       'TESTDB',
       'PROD_ROLE')"
   ```

3. Attribuez le rôle à un utilisateur appeléPROD_USER. L'administrateur PROD_USER est autorisé à attribuer des rôles. Pour de plus amples informations, veuillez consulter rdsadmin.grant_role.

```
db2 "call rdsadmin.grant_role(
    ?,
    'TESTDB',
    'PROD_ROLE',
    'USER PROD_USER',
    'Y')"
```

4. (Facultatif) Fournissez des autorisations ou des privilèges supplémentaires. L'exemple suivant accorde DBADM l'autorisation à un rôle nommé PROD_ROLE d'après une base de données appeléeFUNDPROD. Pour de plus amples informations, veuillez consulter rdsadmin.dbadm_grant.

```
db2 "call rdsadmin.dbadm_grant(
    ?,
    'FUNDPROD',
    'DBADM',
    'ROLE PROD_ROLE')"
```

5. Mettez fin à votre session.

```
db2 terminate
```

6. Connectez-vous à la TESTDB base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *master_username* et *master_password* par vos propres informations.

```
db2 connect to TESTDB user master_username using master_password
```

7. Ajoutez d'autres autorisations au rôle.

```
db2 "grant connect, implicit_schema on database to role PROD_ROLE"
```

8. Accordez le rôle PROD_ROLE à un groupe.

```
db2 "grant role PROD_ROLE to group PRODGRP"
```

Les utilisateurs appartenant au groupe PRODGRP peuvent désormais effectuer des actions telles que la connexion à la TESTDB base de données, la création de tables ou la création de schémas.

Révocation de l'autorisation de base de données

L'utilisateur principal, qui dispose `DBADM` d'une autorisation, peut révoquer `DBADM` l'`DATAACCESS`autorisation d'un rôle, d'un utilisateur ou d'un groupe. `ACCESSCTRL`

Pour révoquer l'autorisation de base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Révoquez l'accès de l'utilisateur en appelant`rdsadmin.dbadm_revoke`. Pour de plus amples informations, veuillez consulter rdsadmin.dbadm_revoke.

   ```
   db2 "call rdsadmin.dbadm_revoke(
       ?,
       'database_name,
       'authorization',
       'grantee')"
   ```

Connexion à la télécommande RDS pour une instance de base de données DB2

Suivez les étapes suivantes pour vous connecter à votre télécommande RDS pour une instance de base de données DB2 et exécuter `get snapshot` des opérations.

Pour se connecter à la télécommande RDS pour une instance de base de données DB2

1. Exécuter une activité côté client IBM Db2 CLP séance. Pour plus d'informations sur le catalogage de votre instance de base de données et de votre base de données RDS pour DB2, consultez. Connexion à votre instance de base de données Amazon RDS pour DB2 avec IBM Db2 CLP Notez le nom d'utilisateur principal et le mot de passe principal de votre instance de base de données RDS pour DB2.
2. Attachez-vous à l'instance de base de données RDS for Db2. Dans l'exemple suivant, remplacez *`node_name`*, *`master_username`*, et *`master_password`* avec le nom du TCPIP nœud que vous avez catalogué ainsi que le nom d'utilisateur principal et le mot de passe principal RDS pour votre instance de base de données Db2.

```
db2 attach to node_name user master_username using master_password
```

Après la connexion à la télécommande RDS pour l'instance de base de données DB2, vous pouvez exécuter les commandes suivantes et d'autres `get snapshot` commandes. Pour plus d'informations, consultez .GET SNAPSHOT commande dans le IBM Db2 .

```
db2 list applications
db2 get snapshot for all databases
db2 get snapshot for database manager
db2 get snapshot for all applications
```

Exécution de tâches de base de données courantes pour les instances RDS de base de données Amazon pour Db2

Vous pouvez effectuer certaines DBA tâches courantes liées aux bases de données sur vos instances de base de données Amazon RDS for Db2. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données. De plus, l'utilisateur principal ne peut pas exécuter de commandes ou `SYSADM` d'`SYSMAINT`utilitaires nécessitant des `SYSCTRL` autorisations.

Pour plus d'informations sur les tâches courantes relatives aux pools de mémoire tampon, aux bases de données et aux tablespaces, consultez les rubriques suivantes.

Rubriques

- Tâches courantes pour les pools de mémoire tampon
- Tâches courantes pour les bases de données
- Tâches courantes pour les tablespaces

Tâches courantes pour les pools de mémoire tampon

Vous pouvez créer, modifier ou supprimer des pools de mémoire tampon pour une base de données RDS pour DB2. La création, la modification ou la suppression de pools de tampons nécessitent un niveau `SYSADM` ou une `SYSCTRL` autorité supérieur, auxquels l'utilisateur principal n'a pas accès. Utilisez plutôt les procédures stockées Amazon RDS.

Vous pouvez également vider les piscines tampons.

Rubriques

- Création d'un pool de mémoire tampon
- Modification d'un pool de mémoire tampon
- Supprimer un pool de mémoire tampon
- Rinçage des piscines tampons

Création d'un pool de mémoire tampon

Pour créer un pool de mémoire tampon pour votre base de données RDS pour DB2, appelez la procédure `rdsadmin.create_bufferpool` stockée. Pour plus d'informations, voir la CREATE BUFFERPOOLdéclaration dans la IBM Db2 documentation.

Pour créer un pool de mémoire tampon

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Créez un pool de mémoire tampon en appelant`rdsadmin.create_bufferpool`. Pour de plus amples informations, veuillez consulter rdsadmin.create_bufferpool.

   ```
   db2 "call rdsadmin.create_bufferpool(
       'database_name',
       'buffer_pool_name',
       buffer_pool_size,
       'immediate',
       'automatic',
       page_size,
       number_block_pages,
       block_size)"
   ```

Modification d'un pool de mémoire tampon

Pour modifier un pool de mémoire tampon pour votre base de données RDS pour DB2, appelez la procédure `rdsadmin.alter_bufferpool` stockée. Pour plus d'informations, voir la ALTER BUFFERPOOLdéclaration dans la IBM Db2 documentation.

Pour modifier un pool de mémoire tampon

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Modifiez un pool de mémoire tampon en appelant`rdsadmin.alter_bufferpool`. Pour de plus amples informations, veuillez consulter rdsadmin.alter_bufferpool.

   ```
   db2 "call rdsadmin.alter_bufferpool(
       'database_name',
       'buffer_pool_name',
       buffer_pool_size,
       'immediate',
       'automatic',
       change_number_blocks,
       number_block_pages,
       block_size)"
   ```

Supprimer un pool de mémoire tampon

Pour supprimer un pool de mémoire tampon pour votre base de données RDS pour DB2, appelez la procédure `rdsadmin.drop_bufferpool` stockée. Pour plus d'informations, consultez la section Suppression de pools de mémoire tampon dans la IBM Db2 documentation.

> **Important**
>
> Assurez-vous qu'aucun tablespace n'est attribué au pool de mémoire tampon que vous souhaitez supprimer.

Pour supprimer un pool de mémoire tampon

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Supprimez un pool de mémoire tampon en appelant`rdsadmin.drop_bufferpool`. Pour de plus amples informations, veuillez consulter rdsadmin.drop_bufferpool.

```
db2 "call rdsadmin.drop_bufferpool(
    'database_name',
    'buffer_pool_name')"
```

Rinçage des piscines tampons

Vous pouvez vider les pools de mémoire tampon pour forcer un point de contrôle afin que RDS pour DB2 enregistre des pages de la mémoire vers le stockage.

> **Note**
>
> Il n'est pas nécessaire de vider les piscines tampons. DB2 écrit les journaux de manière synchrone avant de valider les transactions. Les pages sales se trouvent peut-être toujours dans un pool de mémoire tampon, mais Db2 les écrit dans le stockage de manière asynchrone. Même si le système s'arrête de façon inattendue, lorsque vous redémarrez la base de données, DB2 effectue automatiquement une restauration après incident. Lors de la reprise après incident, Db2 écrit les modifications validées dans la base de données ou annule les modifications pour les transactions non validées.

Pour vider les pools de mémoire tampon

1. Connectez-vous à votre base de données DB2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *rds_database_aliasmaster_username*, et *master_password* par vos propres informations.

```
db2 connect to rds_database_alias user master_username using master_password
```

2. Videz les piscines tampons.

```
db2 flush bufferpools all
```

Tâches courantes pour les bases de données

Vous pouvez créer, supprimer ou restaurer des bases de données sur votre instance de base de données RDS pour DB2. La création, la suppression ou la restauration de bases de données nécessitent une `SYSADM` autorité de niveau supérieur, qui n'est pas accessible à l'utilisateur principal. Utilisez plutôt les procédures stockées Amazon RDS.

Vous pouvez également effectuer des tâches de gestion courantes telles que la surveillance, la maintenance et la collecte d'informations sur vos bases de données.

Rubriques

- Création d’une base de données
- Configuration des paramètres d'une base de données
- Modification des paramètres de base de données
- Configuration de la conservation des journaux
- Informations du journal des annonces
- Désactivation d'une base de données
- Activation d'une base de données
- Réactivation d'une base de données
- Supprimer une base de données
- Restauration d'une base de données
- Liste des bases de données
- Collecte d'informations sur les bases de données
- Forcer les applications à quitter les bases de données
- Génération de rapports de performance

Création d’une base de données

Pour créer une base de données sur votre instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.create_database` stockée. Pour plus d'informations, consultez CREATE DATABASEla section commande dans la IBM Db2 documentation.

> **Note**
>
> Si vous envisagez de modifier le `db2_compatibility_vector` paramètre, modifiez-le avant de créer une base de données. Pour de plus amples informations, veuillez consulter Définition du paramètre db2_compatibility_vector.

Pour créer une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Créez une base de données en appelant`rdsadmin.create_database`. Pour de plus amples informations, veuillez consulter rdsadmin.create_database.

   ```
   db2 "call rdsadmin.create_database('database_name')"
   ```

3. (Facultatif) Créez des bases de données supplémentaires en `rdsadmin.create_database` appelant chaque base de données que vous souhaitez créer. Chaque instance de base de données DB2 peut contenir jusqu'à 50 bases de données. Pour de plus amples informations, veuillez consulter rdsadmin.create_database.

   ```
   db2 "call rdsadmin.create_database('database_name')"
   ```

4. (Facultatif) Vérifiez que votre base de données a été créée à l'aide de l'une des méthodes suivantes :

   - Appelez `rdsadmin.list_databases`. Pour de plus amples informations, veuillez consulter rdsadmin.list_databases.
   - Exécutez la commande SQL suivante :

     ```
     db2 "select varchar(r.task_type,25) as task_type, r.database_name,
         varchar(r.lifecycle,15) as lifecycle, r.created_at, r.database_name,
         varchar(bson_to_json(task_input_params),256) as input_params,
         varchar(r.task_output,1024) as task_output
         from table(rdsadmin.get_task_status(null,null,'create_database'))
     ```

```
    as r order by created_at desc"
```

Configuration des paramètres d'une base de données

Pour configurer les paramètres d'une base de données sur votre instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.set_configuration` stockée. Par exemple, vous pouvez configurer le nombre de tampons ou de manipulateurs de tampons à créer lors d'une opération de restauration.

Pour configurer les paramètres d'une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. (Facultatif) Vérifiez vos paramètres de configuration actuels en appelant`rdsadmin.show_configuration`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.show_configuration”.

   ```
   db2 "call rdsadmin.show_configuration('name')"
   ```

3. Configurez les paramètres de la base de données en appelant`rdsadmin.set_configuration`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.set_configuration”.

   ```
   db2 "call rdsadmin.set_configuration(
       'name',
       'value')"
   ```

Modification des paramètres de base de données

Amazon RDS pour Db2 utilise trois types de paramètres : les paramètres de configuration du gestionnaire de base de données, les variables de registre et les paramètres de configuration de base de données. Vous pouvez mettre à jour les deux premiers types par le biais de groupes de paramètres et le dernier par le biais de la procédure rdsadmin.update_db_param stockée.

> **Note**
>
> Vous ne pouvez modifier que les valeurs des paramètres existants. Vous ne pouvez pas ajouter de nouveaux paramètres que RDS pour DB2 ne prend pas en charge.

Pour plus d'informations sur ces paramètres et sur la façon de modifier leurs valeurs, consultezthe section called “Paramètres DB2”.

Configuration de la conservation des journaux

Pour configurer la durée pendant laquelle Amazon RDS conserve les fichiers journaux de votre base de données RDS pour DB2, appelez la `rdsadmin.set_archive_log_retention` procédure stockée.

Pour configurer la conservation des journaux pour une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. (Facultatif) Vérifiez votre configuration actuelle pour la conservation des journaux en appelant`rdsadmin.show_archive_log_retention`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.show_archive_log_retention”.

   ```
   db2 "call rdsadmin.show_archive_log_retention(
       ?,
       'database_name')"
   ```

3. Configurez la conservation des journaux pour la base de données en appelant`rdsadmin.set_archive_log_retention`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.set_archive_log_retention”.

   ```
   db2 "call rdsadmin.set_archive_log_retention(
       ?,
       'database_name',
       'archive_log_retention_hours')"
   ```

Informations du journal des annonces

Pour répertorier les informations relatives aux fichiers journaux d'archivage, y compris des informations telles que la taille totale de stockage utilisée, appelez la procédure `rdsadmin.list_archive_log_information` stockée.

Pour répertorier les informations de journal d'une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Renvoie une liste des informations du fichier journal en appelant`rdsadmin.list_archive_log_information`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.list_archive_log_information”.

   ```
   db2 "call rdsadmin.list_archive_log_information(
       ?,
       'database_name')"
   ```

Désactivation d'une base de données

Pour désactiver une base de données sur votre instance de base de données RDS pour DB2, appelez la `rdsadmin.deactivate_database` procédure stockée.

Par défaut, Amazon RDS active une base de données lorsque vous créez une base de données sur votre instance de base de données RDS pour DB2. Vous pouvez désactiver les bases de données peu utilisées pour économiser les ressources de mémoire.

Pour désactiver une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Désactivez une base de données en appelant`rdsadmin.deactivate_database`. Pour de plus amples informations, veuillez consulter rdsadmin.deactivate_database.

```
db2 "call rdsadmin.deactivate_database(
    ?,
    'database_name')"
```

Activation d'une base de données

Pour activer une base de données sur une instance de base de données RDS autonome pour DB2, appelez la `rdsadmin.activate_database` procédure stockée.

Par défaut, Amazon RDS active une base de données lorsque vous créez une base de données sur votre instance de base de données RDS pour DB2. Vous pouvez désactiver les bases de données peu utilisées pour économiser les ressources de mémoire, puis activer ultérieurement une base de données désactivée.

Pour activer une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Activez une base de données en appelant`rdsadmin.activate_database`. Pour de plus amples informations, veuillez consulter rdsadmin.activate_database.

```
db2 "call rdsadmin.activate_database(
    ?,
    'database_name')"
```

Réactivation d'une base de données

Pour réactiver une base de données, appelez la procédure `rdsadmin.reactivate_database` stockée. Après avoir modifié les configurations de base de données, vous devrez peut-être réactiver une base de données sur une instance de base de données RDS pour DB2. Pour déterminer si vous

devez réactiver une base de données, connectez-vous à la base de données et `db2 get db cfg show detail` exécutez-la.

Vous pouvez également appeler cette procédure stockée pour réactiver une base de données sur une instance de base de données autonome RDS pour DB2 après avoir modifié les configurations de base de données. Vous pouvez également réactiver une base de données sur une instance de base de données RDS pour DB2 autonome en appelant d'abord la procédure `rdsadmin.deactivate_database` stockée, puis la procédure stockée. `rdsadmin.activate_database` Pour plus d'informations, consultez Désactivation d'une base de données et Activation d'une base de données.

Pour réactiver une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Réactivez une base de données en appelant`rdsadmin.reactivate_database`. Pour de plus amples informations, veuillez consulter rdsadmin.reactivate_database.

   ```
   db2 "call rdsadmin.reactivate_database(
       ?,
       'database_name')"
   ```

Supprimer une base de données

Pour supprimer une base de données de votre instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.drop_database` stockée. Pour plus d'informations, consultez la section Suppression de bases de données dans la IBM Db2 documentation.

> **Note**
>
> Vous pouvez supprimer une base de données en appelant la procédure stockée uniquement si certaines conditions sont remplies. Pour plus d'informations, reportez-vous the section called “Notes d’utilisation” à`rdsadmin.drop_database`.

Pour supprimer une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Supprimez une base de données en appelant`rdsadmin.drop_database`. Pour de plus amples informations, veuillez consulter rdsadmin.drop_database.

   ```
   db2 "call rdsadmin.drop_database('database_name')"
   ```

Restauration d'une base de données

Pour déplacer une base de données d'un compartiment Amazon S3 vers votre instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.restore_database` stockée. Pour plus d'informations, consultez RESTORE DATABASEla section commande dans la IBM Db2 documentation.

Pour restaurer une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. (Facultatif) Vérifiez vos paramètres de configuration actuels pour optimiser l'opération de restauration en appelant`rdsadmin.show_configuration`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.show_configuration”.

   ```
   db2 "call rdsadmin.show_configuration('name')"
   ```

3. Configurez les paramètres pour optimiser l'opération de restauration en appelant`rdsadmin.set_configuration`. La définition explicite de ces valeurs peut améliorer les performances lors de la restauration de bases de données contenant de gros volumes de données. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.set_configuration”.

```
db2 "call rdsadmin.set_configuration(
    'name',
    'value')"
```

4. Restaurez la base de données en appelant`rdsadmin.restore_database`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.restore_database”.

```
db2 "call rdsadmin.restore_database(
    ?,
    'database_name',
    's3_bucket_name',
    's3_prefix',
    restore_timestamp,
    'backup_type')"
```

5. (Facultatif) Vérifiez que votre base de données a été restaurée en appelant `rdsadmin.list_databases` et en vérifiant que la base de données restaurée est répertoriée. Pour de plus amples informations, veuillez consulter rdsadmin.list_databases.

6. Remettez la base de données en ligne et appliquez des journaux de transactions supplémentaires en appelant`rdsadmin.rollforward_database`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.rollforward_database”.

```
db2 "call rdsadmin.rollforward_database(
    ?,
    'database_name',
    's3_bucket_name',
    s3_prefix,
    'rollforward_to_option',
    'complete_rollforward')"
```

7. (Facultatif) Vérifiez l'état de la procédure `rdsadmin.rollforward_database` stockée en appelant la procédure the section called “rdsadmin.rollforward_status” stockée.

8. Si vous l'avez défini `complete_rollforward` `FALSE` à l'étape précédente, vous devez terminer de remettre la base de données en ligne en appelant`rdsadmin.complete_rollforward`. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.complete_rollforward”.

```
db2 "call rdsadmin.complete_rollforward(
    ?,
```

```
    'database_name')"
```

9. (Facultatif) Vérifiez l'état de la procédure `rdsadmin.complete_rollforward` stockée en appelant la procédure the section called “rdsadmin.rollforward_status” stockée.

Liste des bases de données

Vous pouvez répertorier toutes vos bases de données exécutées sur Amazon RDS pour DB2 en appelant la fonction définie par l'`rdsadmin.list_databases`utilisateur.

Pour répertorier vos bases de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Répertoriez vos bases de données en appelant`rdsadmin.list_databases`. Pour de plus amples informations, veuillez consulter rdsadmin.list_databases.

```
db2 "select * from table(rdsadmin.list_databases())"
```

Collecte d'informations sur les bases de données

Pour collecter des informations sur une base de données sur une instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.db2pd_command` stockée. Ces informations peuvent vous aider à surveiller vos bases de données ou à résoudre les problèmes.

Pour collecter des informations sur une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Collectez des informations sur la base de données en appelant`rdsadmin.db2pd_command`. Pour de plus amples informations, veuillez consulter rdsadmin.db2pd_command.

```
db2 "call rdsadmin.db2pd_command('db2pd_cmd')"
```

Forcer les applications à quitter les bases de données

Pour forcer les applications à quitter une base de données sur votre instance de base de données RDS pour DB2, appelez la procédure `rdsadmin.force_application` stockée. Avant de procéder à la maintenance de vos bases de données, forcez les applications à quitter vos bases de données.

Pour forcer les applications à quitter une base de données

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Forcez les applications à quitter une base de données en appelant`rdsadmin.force_application`. Pour de plus amples informations, veuillez consulter rdsadmin.force_application.

   ```
   db2 "call rdsadmin.force_application(
       ?,
       'applications')"
   ```

Génération de rapports de performance

Vous pouvez générer des rapports de performance à l'aide d'une procédure ou d'un script. Pour plus d'informations sur l'utilisation d'une procédure, voir DBSUMMARYprocédure - Générer un rapport récapitulatif des mesures de performance du système et de l'application dans la IBM Db2 documentation.

Db2 inclut un `db2mon.sh` fichier dans son `~sqllib/sample/perf` répertoire. L'exécution du script produit un rapport détaillé et peu coûteux sur les métriques SQL. Pour télécharger le `db2mon.sh` fichier et les fichiers de script associés, consultez le perfrépertoire dans le référentiel IBM db2-samplesGitHub.

Pour générer des rapports de performance à l'aide du script

1. Connectez-vous à votre base de données DB2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *master_username* et *master_password* par vos propres informations.

```
db2 connect to rdsadmin user master_username using master_password
```

2. Créez un pool de mémoire tampon nommé db2monbp avec une taille de page de 4096 en appelant`rdsadmin.create_bufferpool`. Pour de plus amples informations, veuillez consulter rdsadmin.create_bufferpool.

```
db2 "call rdsadmin.create_bufferpool('database_name','db2monbp',4096)"
```

3. Créez un tablespace temporaire nommé db2montmptbsp qui utilise le pool de db2monbp mémoire tampon en appelant. `rdsadmin.create_tablespace` Pour de plus amples informations, veuillez consulter rdsadmin.create_tablespace.

```
db2 "call rdsadmin.create_tablespace('database_name',\
    'db2montmptbsp','db2monbp',4096,1000,100,'T')"
```

4. Ouvrez le `db2mon.sh` script et modifiez la ligne concernant la connexion à une base de données.

   a. Supprimez la ligne suivante.

```
db2 -v connect to $dbName
```

   b. Remplacez la ligne de l'étape précédente par la ligne suivante. Dans l'exemple suivant, remplacez *master_username* et *master_password* par le nom d'utilisateur principal et le mot de passe principal de votre instance de base de données RDS pour DB2.

```
db2 -v connect to $dbName user master_username using master_password
```

   c. Supprimez les lignes suivantes.

```
db2 -v create bufferpool db2monbp

db2 -v create user temporary tablespace db2montmptbsp bufferpool db2monbp

db2 -v drop tablespace db2montmptbsp
```

```
db2 -v drop bufferpool db2monbp
```

5. Exécutez le `db2mon.sh` script pour générer un rapport à des intervalles spécifiés. Dans l'exemple suivant, remplacez *`absolute_path`* par le chemin complet du fichier de script, *`rds_database_alias`* par le nom de votre base de données et *`seconds`* par le nombre de secondes (0 à 3 600) entre la génération du rapport.

   ```
   absolute_path/db2mon.sh rds_database_alias seconds | tee -a db2mon.out
   ```

   Exemples

   L'exemple suivant montre que le fichier de script se trouve dans le `perf` répertoire situé sous le `home` répertoire.

   ```
   /home/db2inst1/sqllib/samples/perf/db2mon.sh rds_database_alias seconds | tee -a
    db2mon.out
   ```

6. Supprimez le pool de mémoire tampon et le tablespace créés pour le `db2mon.sh` fichier. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par le nom d'utilisateur principal et le mot de passe principal de votre instance de base de données RDS pour DB2. Remplacez *`database_name`* par le nom de votre base de données. Pour plus d'informations, consultez rdsadmin.drop_tablespace et rdsadmin.drop_bufferpool.

   ```
   db2 connect to rdsadmin user master_username using master_password

   db2 "call rdsadmin.drop_tablespace('database_name','db2montmptbsp')"

   db2 "call rdsadmin.drop_bufferpool('database_name','db2monbp')"
   ```

Gestion du stockage

Db2 utilise le stockage automatique pour gérer le stockage physique des objets de base de données tels que les tables, les index et les fichiers temporaires. Au lieu d'allouer manuellement de l'espace de stockage et de suivre les chemins de stockage utilisés, le stockage automatique permet au système Db2 de créer et de gérer les chemins de stockage selon les besoins. Cela peut simplifier l'administration des bases de données DB2 et réduire le risque d'erreurs dues à des erreurs humaines. Pour plus d'informations, voir Stockage automatique dans le IBM Db2 .

Avec RDS Db2, vous pouvez augmenter dynamiquement la taille de stockage grâce à l'extension automatique des volumes logiques et du système de fichiers. Pour de plus amples informations, veuillez consulter Utilisation du stockage pour les RDS instances de base de données Amazon.

Tâches courantes pour les tablespaces

Vous pouvez créer, modifier, renommer ou supprimer des tablespaces pour une base de données RDS pour DB2. La création, la modification, le renommage ou la suppression de tablespaces nécessitent une `SYSADM` autorité de niveau supérieur, qui n'est pas accessible à l'utilisateur principal. Utilisez plutôt les procédures RDS stockées Amazon.

Rubriques

- Création d'un tablespace
- Modification d'un tablespace
- Modification du nom d'un tablespace
- Supprimer un tablespace
- Vérifier l'état d'un tablespace
- Renvoyer des informations détaillées sur les tablespaces
- Répertorier l'état et le groupe de stockage d'un tablespace
- Lister les espaces disque logiques d'une table
- Répertorier les conteneurs de tablespaces

Création d'un tablespace

Pour créer un tablespace pour votre base de données RDS for DB2, appelez la `rdsadmin.create_tablespace` procédure stockée. Pour plus d'informations, consultez .CREATE TABLESPACE déclaration dans le IBM Db2 .

> **Important**
>
> Pour créer un espace disque logique, vous devez disposer d'un pool de mémoire tampon de même taille de page à associer au tablespace. Pour de plus amples informations, veuillez consulter Tâches courantes pour les pools de mémoire tampon.

Pour créer un tablespace

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Créez un tablespace en appelant. `rdsadmin.create_tablespace` Pour de plus amples informations, veuillez consulter rdsadmin.create_tablespace.

```
db2 "call rdsadmin.create_tablespace(
    'database_name',
    'tablespace_name',
    'buffer_pool_name',
    tablespace_initial_size,
    tablespace_increase_size,
    'tablespace_type')"
```

Modification d'un tablespace

Pour modifier un tablespace pour votre base de données RDS for DB2, appelez la `rdsadmin.alter_tablespace` procédure stockée. Vous pouvez utiliser cette procédure stockée pour modifier le pool de mémoire tampon d'un espace disque logique, abaisser le seuil maximal ou mettre un espace disque logique en ligne. Pour plus d'informations, consultez .ALTER TABLESPACE déclaration dans le IBM Db2 .

Pour modifier un tablespace

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

```
db2 "connect to rdsadmin user master_username using master_password"
```

2. Modifiez un tablespace en appelant. `rdsadmin.alter_tablespace` Pour de plus amples informations, veuillez consulter rdsadmin.alter_tablespace.

```
db2 "call rdsadmin.alter_tablespace(
    'database_name',
```

```
    'tablespace_name',
    'buffer_pool_name',
    buffer_pool_size,
    tablespace_increase_size,
    'max_size', 'reduce_max',
    'reduce_stop',
    'reduce_value',
    'lower_high_water',
    'lower_high_water_stop',
    'switch_online')"
```

Modification du nom d'un tablespace

Pour modifier le nom d'un tablespace pour votre base de données RDS for DB2, appelez la `rdsadmin.rename_tablespace` procédure stockée. Pour plus d'informations, consultez .RENAME TABLESPACE déclaration dans le IBM Db2 .

Pour renommer un tablespace

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *master_username* and *master_password* avec vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Renommez un tablespace en appelant. `rdsadmin.rename_tablespace` Pour plus d'informations, notamment les restrictions relatives au nom que vous pouvez donner à un espace disque logique, consultez. rdsadmin.rename_tablespace

   ```
   db2 "call rdsadmin.rename_tablespace(
       'database_name',
       'source_tablespace_name',
       'target_tablespace_name')"
   ```

Supprimer un tablespace

Pour supprimer un tablespace pour votre base de données RDS for DB2, appelez la `rdsadmin.drop_tablespace` procédure stockée. Avant de supprimer un espace disque logique,

supprimez d'abord tous les objets qu'il contient, tels que des tables, des index ou des objets de grande taille (). LOBs Pour plus d'informations, voir Supprimer les espaces de table dans IBM Db2 .

Pour supprimer un tablespace

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

   ```
   db2 "connect to rdsadmin user master_username using master_password"
   ```

2. Supprimez un tablespace en appelant. `rdsadmin.drop_tablespace` Pour de plus amples informations, veuillez consulter rdsadmin.drop_tablespace.

   ```
   db2 "call rdsadmin.drop_tablespace(
       'database_name',
       'tablespace_name')"
   ```

Vérifier l'état d'un tablespace

Vous pouvez vérifier l'état d'un tablespace à l'aide de cette fonction. `cast`

Pour vérifier l'état d'un tablespace

1. Connectez-vous à votre base de données DB2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance RDS de base de données Db2. Dans l'exemple suivant, remplacez *`rds_database_alias`*, *`master_username`*, et *`master_password`* avec vos propres informations.

   ```
   db2 connect to rds_database_alias user master_username using master_password
   ```

2. Renvoie un résultat récapitulatif.

   Pour un résultat récapitulatif :

   ```
   db2 "select cast(tbsp_id as smallint) as tbsp_id,
   cast(tbsp_name as varchar(35)) as tbsp_name,
   cast(tbsp_type as varchar(3)) as tbsp_type,
   cast(tbsp_state as varchar(10)) as state,
   ```

```
cast(tbsp_content_type as varchar(8)) as contents from
 table(mon_get_tablespace(null,-1)) order by tbsp_id"
```

Renvoyer des informations détaillées sur les tablespaces

Vous pouvez renvoyer des informations sur un tablespace pour un membre ou pour tous les membres à l'`cast`aide de cette fonction.

Pour renvoyer des informations détaillées sur les tablespaces

1. Connectez-vous à votre base de données DB2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance RDS de base de données Db2. Dans l'exemple suivant, remplacez *`rds_database_alias`*, *`master_username`*, et *`master_password`* avec vos propres informations.

   ```
   db2 connect to rds_database_alias user master_username using master_password
   ```

2. Renvoie des informations sur tous les tablespaces de la base de données pour un membre ou pour tous les membres.

   Pour un membre :

   ```
   db2 "select cast(member as smallint) as member,
   cast(tbsp_id as smallint) as tbsp_id,
   cast(tbsp_name as varchar(35)) as tbsp_name,
   cast(tbsp_type as varchar(3)) as tbsp_type,
   cast(tbsp_state as varchar(10)) as state,
   cast(tbsp_content_type as varchar(8)) as contents,
   cast(tbsp_total_pages as integer) as total_pages,
   cast(tbsp_used_pages as integer) as used_pages,
   cast(tbsp_free_pages as integer) as free_pages,
   cast(tbsp_page_top as integer) as page_hwm,
   cast(tbsp_page_size as integer) as page_sz,
   cast(tbsp_extent_size as smallint) as extent_sz,
   cast(tbsp_prefetch_size as smallint) as prefetch_sz,
   cast(tbsp_initial_size as integer) as initial_size,
   cast(tbsp_increase_size_percent as smallint) as increase_pct,
   cast(storage_group_name as varchar(12)) as stogroup from
    table(mon_get_tablespace(null,-1)) order by member, tbsp_id "
   ```

   Pour tous les membres :

```
db2 "select cast(member as smallint) as member
cast(tbsp_id as smallint) as tbsp_id,
cast(tbsp_name as varchar(35)) as tbsp_name,
cast(tbsp_type as varchar(3)) as tbsp_type,
cast(tbsp_state as varchar(10)) as state,
cast(tbsp_content_type as varchar(8)) as contents,
cast(tbsp_total_pages as integer) as total_pages,
cast(tbsp_used_pages as integer) as used_pages,
cast(tbsp_free_pages as integer) as free_pages,
cast(tbsp_page_top as integer) as page_hwm,
cast(tbsp_page_size as integer) as page_sz,
cast(tbsp_extent_size as smallint) as extent_sz,
cast(tbsp_prefetch_size as smallint) as prefetch_sz,
cast(tbsp_initial_size as integer) as initial_size,
cast(tbsp_increase_size_percent as smallint) as increase_pct,
cast(storage_group_name as varchar(12)) as stogroup from
 table(mon_get_tablespace(null,-2)) order by member, tbsp_id "
```

Répertorier l'état et le groupe de stockage d'un tablespace

Vous pouvez répertorier l'état et le groupe de stockage d'un tablespace en exécutant une SQL instruction.

Pour répertorier l'état et le groupe de stockage d'un tablespace, exécutez l'instruction suivante : SQL

```
db2 "SELECT varchar(tbsp_name, 30) as tbsp_name,
                  varchar(TBSP_STATE, 30) state,
                  tbsp_type,
                  varchar(storage_group_name,30) storage_group
FROM TABLE(MON_GET_TABLESPACE('',-2)) AS t"
```

Lister les espaces disque logiques d'une table

Vous pouvez répertorier les espaces disque logiques d'une table en exécutant une SQL instruction.

Pour répertorier les espaces disque logiques d'une table, exécutez l'instruction suivanteSQL. Dans l'exemple suivant, remplacez *SCHEMA_NAME* and *TABLE_NAME* avec les noms de votre schéma et de votre table :

```
db2 "SELECT
```

```
  VARCHAR(SD.TBSPACE,30) AS DATA_SPACE,
  VARCHAR(SL.TBSPACE,30) AS LONG_SPACE,
  VARCHAR(SI.TBSPACE,30) AS INDEX_SPACE
FROM
  SYSCAT.DATAPARTITIONS P
  JOIN SYSCAT.TABLESPACES SD ON SD.TBSPACEID = P.TBSPACEID
  LEFT JOIN SYSCAT.TABLESPACES SL ON SL.TBSPACEID = P.LONG_TBSPACEID
  LEFT JOIN SYSCAT.TABLESPACES SI ON SI.TBSPACEID = P.INDEX_TBSPACEID
WHERE
   TABSCHEMA = 'SCHEMA_NAME'
AND TABNAME   = 'TABLE_NAME'"
```

Répertorier les conteneurs de tablespaces

Vous pouvez répertorier tous les conteneurs d'espaces disque logiques ou des conteneurs d'espaces disque logiques spécifiques à l'aide de la commande. `cast`

Pour répertorier les conteneurs d'un espace disque logique

1. Connectez-vous à votre base de données DB2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance RDS de base de données Db2. Dans l'exemple suivant, remplacez *`rds_database_alias`*, *`master_username`*, et *`master_password`* avec vos propres informations :

   ```
   db2 connect to rds_database_alias user master_username using master_password
   ```

2. Renvoie une liste de tous les conteneurs d'espaces disque logiques de la base de données ou de conteneurs de tablespaces spécifiques.

   Pour tous les conteneurs de tablespaces :

   ```
   db2 "select cast(member as smallint) as member,
   cast(tbsp_name as varchar(35)) as tbsp_name,
   cast(container_id as smallint) as id,
   cast(container_name as varchar(60)) as container_path, container_type as type from
    table(mon_get_container(null,-2)) order by member,tbsp_id,container_id"
   ```

   Pour des conteneurs de tablespaces spécifiques :

   ```
   db2 "select cast(member as smallint) as member,
   cast(tbsp_name as varchar(35)) as tbsp_name,
   ```

```
cast(container_id as smallint) as id,
cast(container_name as varchar(60)) as container_path, container_type as type from
 table(mon_get_container('TBSP_1',-2)) order by member, tbsp_id,container_id"
```

Intégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3

Vous pouvez transférer des fichiers entre votre instance de base de données Amazon RDS for Db2 et un bucket Amazon Simple Storage Service (Amazon S3) contenant des procédures stockées par AmazonRDS. Pour de plus amples informations, veuillez consulter Référence de procédure stockée Amazon RDS pour DB2.

> **Note**
>
> Votre instance de base de données et votre compartiment Amazon S3 doivent se trouver dans la même Région AWS.

RDSPour que Db2 puisse s'intégrer à Amazon S3, votre instance de base de données doit avoir accès à un compartiment Amazon S3 dans lequel réside votre RDS for Db2. Si vous ne disposez pas actuellement d'un compartiment S3, créez-en un.

Rubriques

- Étape 1 : Créer une stratégie IAM
- Étape 2 : Créez un IAM rôle et associez votre IAM politique
- Étape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2

Étape 1 : Créer une stratégie IAM

Au cours de cette étape, vous créez une politique AWS Identity and Access Management (IAM) avec les autorisations requises pour transférer des fichiers de votre compartiment Amazon S3 vers votre RDS instance de base de données. Cette étape suppose également que vous avez déjà créé un compartiment S3. Pour plus d'informations, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.

Avant de créer la politique, notez les informations suivantes :

- Le nom de ressource Amazon (ARN) de votre compartiment
- La clé ARN for your AWS Key Management Service (AWS KMS), si votre bucket utilise SSE-KMS or SSE-S3 chiffrement.

Créez une IAM politique qui inclut les autorisations suivantes :

```
"kms:GenerateDataKey",
"kms:Decrypt",
"s3:PutObject",
"s3:GetObject",
"s3:AbortMultipartUpload",
"s3:ListBucket",
"s3:DeleteObject",
"s3:GetObjectVersion",
"s3:ListMultipartUploadParts"
```

Vous pouvez créer une IAM politique en utilisant le AWS Management Console ou le AWS Command Line Interface (AWS CLI).

Console

Pour créer une IAM politique permettant à Amazon RDS d'accéder à votre compartiment Amazon S3

1. Connectez-vous à la IAM console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Politiques.
3. Choisissez Créer une politique, puis choisissez JSON.
4. Ajoutez des actions par service. Pour transférer des fichiers d'un compartiment Amazon S3 vers AmazonRDS, vous devez sélectionner des autorisations de compartiment et des autorisations d'objet.
5. Développer les Ressources. Vous devez spécifier les ressources de votre bucket et de votre objet.
6. Choisissez Suivant.
7. Dans Nom de la stratégie, entrez le nom de cette stratégie.
8. (Facultatif) Pour Description, saisissez une description pour cette stratégie.
9. Choisissez Create Policy (Créer une politique).

AWS CLI

Pour créer une IAM politique permettant à Amazon RDS d'accéder à votre compartiment Amazon S3

1. Exécutez le create-policycommande. Dans l'exemple suivant, remplacez *iam_policy_name* et *s3_bucket_name* par le nom de votre IAM politique et le nom du compartiment Amazon S3 dans lequel réside votre base de données RDS for Db2.

   Dans Linux, macOS, ou Unix:

```
aws iam create-policy \
    --policy-name iam_policy_name \
    --policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Action": [
            "kms:GenerateDataKey",
            "kms:Decrypt",
            "s3:PutObject",
            "s3:GetObject",
            "s3:AbortMultipartUpload",
            "s3:ListBucket",
            "s3:DeleteObject",
            "s3:GetObjectVersion",
            "s3:ListMultipartUploadParts"
          ],
          "Resource": [
            "arn:aws:s3:::s3_bucket_name/*",
            "arn:aws:s3:::s3_bucket_name"
          ]
        }
      ]
    }'
```

   Dans Windows:

```
aws iam create-policy ^
    --policy-name iam_policy_name ^
    --policy-document '{
      "Version": "2012-10-17",
```

```
    "Statement": [
      {
        "Effect": "Allow",
          "Action": [
            "s3:PutObject",
            "s3:GetObject",
            "s3:AbortMultipartUpload",
            "s3:ListBucket",
            "s3:DeleteObject",
            "s3:GetObjectVersion",
            "s3:ListMultipartUploadParts"
          ],
          "Resource": [
            "arn:aws:s3:::s3_bucket_name/*",
            "arn:aws:s3:::s3_bucket_name"
          ]
      }
    ]
  }'
```

2. Une fois la stratégie créée, notez celle ARN de la stratégie. Il te faut le ARN formulaireÉtape 2 : Créez un IAM rôle et associez votre IAM politique.

Pour plus d'informations sur la création d'une IAM politique, voir Création de IAM politiques dans le Guide de IAM l'utilisateur.

Étape 2 : Créez un IAM rôle et associez votre IAM politique

Cette étape suppose que vous avez créé la IAM politique dansÉtape 1 : Créer une stratégie IAM. Au cours de cette étape, vous créez un IAM rôle pour votre instance de base de données RDS for Db2, puis vous attachez votre IAM politique au rôle.

Vous pouvez créer un IAM rôle pour votre instance de base de données en utilisant le AWS Management Console ou le AWS CLI.

Console

Pour créer un IAM rôle et y associer votre IAM politique

1. Connectez-vous à la IAM console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/iam/.
2. Dans le panneau de navigation, choisissez Roles (Rôles).

3. Sélectionnez Create role (Créer un rôle).
4. Pour Type d'entité de confiance, sélectionnez Service AWS.
5. Pour Service ou cas d'utilisation, sélectionnez RDS, puis sélectionnez RDS— Ajouter un rôle à la base de données.
6. Choisissez Suivant.
7. Pour les politiques d'autorisations, recherchez et sélectionnez le nom de la IAM politique que vous avez créée.
8. Choisissez Suivant.
9. Pour Role name (Nom du rôle), saisissez un nom de rôle.
10. (Facultatif) Pour Description, saisissez une description pour le nouveau rôle.
11. Sélectionnez Créer un rôle.

AWS CLI

Pour créer un IAM rôle et y associer votre IAM politique

1. Exécutez le create-rolecommande. Dans l'exemple suivant, remplacez-le *iam_role_name* par le nom de votre IAM rôle.

   Dans Linux, macOS, ou Unix:

```
aws iam create-role \
    --role-name iam_role_name \
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "rds.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

   Dans Windows:

```
aws iam create-role ^
    --role-name iam_role_name ^
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "rds.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

2. Une fois le rôle créé, notez le ARN nom du rôle. Il te faut le ARN formulaireÉtape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2.
3. Exécutez le attach-role-policycommande. Dans l'exemple suivant, remplacez *iam_policy_arn* par le ARN de la IAM politique que vous avez créée dansÉtape 1 : Créer une stratégie IAM. *iam_role_name*Remplacez-le par le nom du IAM rôle que vous venez de créer.

   Dans Linux, macOS, ou Unix:

   ```
   aws iam attach-role-policy \
      --policy-arn iam_policy_arn \
      --role-name iam_role_name
   ```

   Dans Windows:

   ```
   aws iam attach-role-policy ^
      --policy-arn iam_policy_arn ^
      --role-name iam_role_name
   ```

Pour plus d'informations, consultez la section Création d'un rôle pour déléguer des autorisations à un IAM utilisateur dans le Guide de IAM l'utilisateur.

Étape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2

Au cours de cette étape, vous ajoutez votre IAM rôle à votre instance de base de données RDS for Db2. Notez les critères suivants :

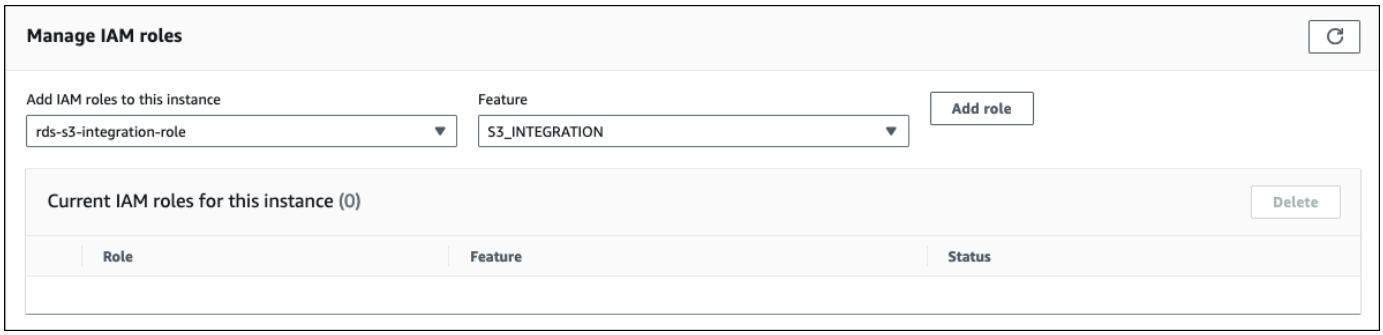
- Vous devez avoir accès à un IAM rôle associé à la politique d'autorisation Amazon S3 requise.
- Vous ne pouvez associer qu'un seul IAM rôle à votre instance de base de données RDS for Db2 à la fois.
- Votre instance de base de données RDS for DB2 doit être à l'état Disponible.

Vous pouvez ajouter un IAM rôle à votre instance de base de données en utilisant le AWS Management Console ou le AWS CLI.

Console

Pour ajouter un IAM rôle à votre instance de base de données RDS for DB2

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom RDS de votre instance de base de données for DB2.
4. Dans l'onglet Connectivité et sécurité, faites défiler l'écran jusqu'à la section Gérer les IAM rôles au bas de la page.
5. Pour Ajouter IAM des rôles à cette instance, choisissez le rôle que vous avez créé dansÉtape 2 : Créez un IAM rôle et associez votre IAM politique.
6. Dans Feature, choisissez S3_ INTEGRATION.
7. Choisissez Ajouter un rôle.



AWS CLI

Pour ajouter un IAM rôle à votre instance de base de données RDS for Db2, exécutez add-role-to-db-instancecommande. Dans l'exemple suivant, remplacez *db_instance_name* et *iam_role_arn* par le nom de votre instance de base ARN de données et le IAM rôle que vous avez créé dansÉtape 2 : Créez un IAM rôle et associez votre IAM politique.

Dans Linux, macOS, ou Unix:

```
aws rds add-role-to-db-instance \
    --db-instance-identifier db_instance_name \
    --feature-name S3_INTEGRATION \
    --role-arn iam_role_arn \
```

Dans Windows:

```
aws rds add-role-to-db-instance ^
    --db-instance-identifier db_instance_name ^
    --feature-name S3_INTEGRATION ^
    --role-arn iam_role_arn ^
```

Pour confirmer que le rôle a été ajouté avec succès à votre instance de base de données RDS for DB2, exécutez describe-db-instancescommande. Dans l'exemple suivant, remplacez *db_instance_name* par le nom de votre instance de base de données.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-instances \
    --filters "Name=db-instance-id,Values=db_instance_name" \
    --query 'DBInstances[].AssociatedRoles'
```

Dans Windows:

```
aws rds describe-db-instances ^
    --filters "Name=db-instance-id,Values=db_instance_name" ^
    --query 'DBInstances[].AssociatedRoles'
```

Cette commande produit une sortie similaire à l'exemple suivant :

```
[
```

```
    [
        {
            "RoleArn": "arn:aws:iam::0123456789012:role/rds-db2-s3-role",
            "FeatureName": "S3_INTEGRATION",
            "Status": "ACTIVE"
        }
    ]
]
```

Migration de données vers Amazon RDS pour DB2

Vous pouvez migrer des bases de données DB2 autogérées vers Amazon RDS pour DB2 en utilisant des outils Db2 AWS ou natifs.

Pour plus d'informations sur la migration de votre base de données Db2 vers Amazon RDS for Db2 à l'aide de AWS services, consultez. Utilisation de AWS services pour migrer des données de Db2 vers Amazon RDS pour Db2

Pour plus d'informations sur la migration de votre base de données Db2 vers Amazon RDS for Db2 à l'aide des outils Db2 natifs, consultez. Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2

Utilisation de AWS services pour migrer des données de Db2 vers Amazon RDS pour Db2

Amazon propose RDS plusieurs méthodes pour migrer des données d'une base de données Db2 vers Amazon RDS for Db2. Vous pouvez effectuer une migration unique de votre base de données DB2 depuis Linux, AIX, ou Windows environnements pour Amazon RDS pour Db2. Pour minimiser les temps d'arrêt, vous pouvez effectuer une migration quasiment nulle. Vous pouvez migrer vos données en les enregistrant sur Amazon S3 et en les chargeant table par table dans votre base de données DB2. Vous pouvez également effectuer une migration synchrone par réplication ou utilisation AWS Database Migration Service.

Pour les migrations ponctuelles pour Linuxbasée sur des bases de données DB2, Amazon RDS ne prend en charge que les sauvegardes hors ligne et en ligne. Amazon RDS ne prend pas en charge les fonctionnalités incrémentielles et Delta sauvegardes. Pour des migrations quasiment nulles Linuxbasées sur des bases de données DB2, Amazon a RDS besoin de sauvegardes en ligne. Nous vous recommandons d'utiliser des sauvegardes en ligne pour des migrations à temps d'arrêt quasi nul et des sauvegardes hors ligne pour des migrations capables de gérer les temps d'arrêt.

Rubriques

- Migration à partir de Linux to Linux pour Amazon RDS pour Db2
- Migration à partir de Linux to Linux avec un temps d'arrêt quasi nul pour Amazon RDS pour DB2
- Migration synchrone depuis Linux to Linux pour Amazon RDS pour Db2
- Migration à partir de AIX or Windows to Linux pour Amazon RDS pour Db2
- Migration de données DB2 via Amazon S3 vers Amazon RDS for Db2

- Migration vers Amazon RDS pour DB2 avec () AWS Database Migration ServiceAWS DMS

Migration à partir de Linux to Linux pour Amazon RDS pour Db2

Avec cette approche de migration, vous sauvegardez votre base de données DB2 autogérée dans un compartiment Amazon S3. Ensuite, vous utilisez les procédures RDS stockées Amazon pour restaurer votre base de données Db2 sur une instance de base de données Amazon RDS for Db2. Pour plus d'informations sur l'utilisation d'Amazon S3, consultezIntégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3.

Backup and Restore RDS for Db2 suit le IBM Db2 chemins de mise à niveau et restrictions pris en charge. Pour plus d'informations, consultez les sections Chemins de mise à niveau pris en charge pour les serveurs DB2 et Restrictions de mise à niveau pour les serveurs DB2 dans le IBM Db2 .

Rubriques

- Limitations et recommandations relatives à l'utilisation de la restauration native
- Sauvegarde de votre base de données sur Amazon S3
- Création d'un groupe de stockage automatique par défaut
- Restauration de votre base de données DB2

Limitations et recommandations relatives à l'utilisation de la restauration native

Les limites et recommandations suivantes s'appliquent à l'utilisation de la restauration native :

- Amazon prend RDS uniquement en charge la migration des versions sur site de Db2 qui correspondent aux versions prises en charge RDS pour les versions Db2. Pour plus d'informations sur les versions prises en charge, consultezVersions mineures de DB2 prises en charge sur Amazon RDS.
- Amazon prend RDS uniquement en charge les sauvegardes hors ligne et en ligne pour la restauration native. Amazon RDS ne prend pas en charge la technologie incrémentielle ou Delta sauvegardes.
- Vous ne pouvez pas effectuer de restauration à partir d'un compartiment Amazon S3 situé dans une Région AWS région différente de la région dans laquelle se trouve votre instance de base de données RDS for Db2.
- Amazon S3 limite la taille des fichiers chargés dans un compartiment Amazon S3 à 5 To. Si le fichier de sauvegarde de votre base de données dépasse 5 To, divisez-le en fichiers plus petits.

- Amazon RDS ne prend pas en charge les routines externes non clôturées, les restaurations incrémentielles ou Delta restaure.
- Vous ne pouvez pas effectuer de restauration à partir d'une base de données source chiffrée, mais vous pouvez effectuer une restauration sur une RDS instance de base de données Amazon cryptée.

Lorsque vous restaurez votre base de données, la sauvegarde est copiée puis extraite sur votre instance de base de données RDS pour DB2. Nous vous recommandons de prévoir pour votre instance de base de données RDS for DB2 un espace de stockage égal ou supérieur à la somme de la taille de sauvegarde et de la taille de la base de données d'origine sur le disque.

La taille maximale de la base de données restaurée est la taille maximale de base de données prise en charge moins la taille de la sauvegarde. Par exemple, si la taille maximale de base de données prise en charge est de 64 TiB et que la taille de la sauvegarde est de 30 TiB, la taille maximale de la base de données restaurée est de 34 TiB.

```
64 TiB - 30 TiB = 34 TiB
```

Sauvegarde de votre base de données sur Amazon S3

Pour sauvegarder votre base de données sur Amazon S3, vous avez besoin des AWS composants suivants :

- Un compartiment Amazon S3 pour stocker vos fichiers de sauvegarde : chargez tous les fichiers de sauvegarde que vous souhaitez migrer vers AmazonRDS. Nous vous recommandons d'utiliser des sauvegardes hors ligne pour les migrations susceptibles de gérer les temps d'arrêt. Si vous possédez déjà un compartiment S3, vous pouvez l'utiliser. Si vous n'avez pas de compartiment S3, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.

> **Note**
>
> Si votre base de données est volumineuse et que son transfert vers un compartiment S3 prend du temps, vous pouvez commander un AWS Snow Family appareil et demander AWS à effectuer la sauvegarde. Une fois que vous avez copié vos fichiers sur l'appareil et que vous les avez renvoyés à l'équipe Snow Family, celle-ci transfère vos images sauvegardées dans votre compartiment S3. Pour en savoir plus, consultez la documentation AWS Snow Family.

- Un IAM rôle pour accéder au compartiment S3 : si vous avez déjà un IAM rôle, vous pouvez l'utiliser. Si vous n'avez pas de rôle, consultezÉtape 2 : Créez un IAM rôle et associez votre IAM politique.
- Une IAM politique assortie de relations de confiance et d'autorisations associées à votre IAM rôle : pour plus d'informations, consultezÉtape 1 : Créer une stratégie IAM.
- Le IAM rôle ajouté à votre instance de base de données RDS for Db2 : pour plus d'informations, consultezÉtape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2.

Création d'un groupe de stockage automatique par défaut

Votre base de données source doit disposer d'un groupe de stockage automatique par défaut. Si votre base de données ne possède pas de groupe de stockage automatique par défaut, vous devez en créer un.

Pour créer un groupe de stockage automatique par défaut

1. Connectez-vous à votre base de données source. Dans l'exemple suivant, remplacez *source_database* par le nom de votre base de données.

```
db2 connect to source_database
```

2. Créez un groupe de stockage automatique et définissez-le comme groupe par défaut. Dans l'exemple suivant, remplacez *storage_path* par le chemin absolu vers l'emplacement du groupe de stockage.

```
db2 "create stogroup IBMSTOGROUP ON storage_path set as default"
```

3. Mettez fin aux processus principaux.

```
db2 terminate
```

4. Désactivez la base de données et arrêtez tous les services de base de données. Dans l'exemple suivant, remplacez *source_database* par le nom de la base de données pour laquelle vous avez créé le groupe de stockage.

```
db2 deactivate db source_database
```

5. Sauvegardez la base de données. Dans l'exemple suivant, remplacez *source_database* par le nom de la base de données pour laquelle vous avez créé le groupe de stockage. Remplacez

file_system_path par le chemin absolu vers lequel vous souhaitez sauvegarder la base de données.

```
db2 backup database source_database to file_system_path
```

Restauration de votre base de données DB2

Après avoir sauvegardé votre base de données sur Amazon S3 et créé un groupe de stockage automatique, vous êtes prêt à restaurer votre base de données DB2 sur votre instance de base de données RDS for Db2.

Pour restaurer votre base de données DB2 sur votre instance de base RDS de données Db2

1. Connectez-vous à votre instance de base de données RDS for DB2. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données DB2.
2. (Facultatif) Pour vous assurer que votre base de données est configurée avec les paramètres optimaux pour l'opération de restauration, vous pouvez appeler the section called “rdsadmin.show_configuration” pour vérifier les valeurs de `RESTORE_DATABASE_PARALLELISM` et`RESTORE_DATABASE_NUM_BUFFERS`. Appelez the section called “rdsadmin.set_configuration” pour modifier ces valeurs, le cas échéant. La définition explicite de ces valeurs peut améliorer les performances lors de la restauration de bases de données contenant de gros volumes de données.
3. Restaurez votre base de données en appelant`rdsadmin.restore_database`. Pour de plus amples informations, veuillez consulter rdsadmin.restore_database.

Migration à partir de Linux to Linux avec un temps d'arrêt quasi nul pour Amazon RDS pour DB2

Avec cette approche de migration, vous faites migrer un Linuxbase de données DB2 basée sur une base de données DB2 autogérée (source) vers Amazon RDS pour Db2. Cette approche entraîne des interruptions ou des interruptions minimes, voire nulles, pour l'application ou les utilisateurs. Cette approche sauvegarde votre base de données et la restaure grâce à la réexécution des journaux, ce qui permet d'éviter toute interruption des opérations en cours et d'assurer une haute disponibilité de votre base de données.

Pour obtenir une migration quasiment sans interruption de service, RDS car Db2 implémente la restauration avec réexécution des journaux. Cette approche prend une sauvegarde de votre système

autogéré Linuxbase de données DB2 basée sur le serveur Db2 et la restaure sur le RDS serveur for DB2. Avec les procédures RDS stockées Amazon, vous appliquez ensuite les journaux de transactions suivants pour mettre à jour la base de données.

Rubriques

- Limites et recommandations pour une migration proche de zéro temps d'arrêt
- Sauvegarde de votre base de données sur Amazon S3
- Création d'un groupe de stockage automatique par défaut
- Migration de votre base de données DB2

Limites et recommandations pour une migration proche de zéro temps d'arrêt

Les limites et recommandations suivantes s'appliquent à l'utilisation d'une migration à temps d'arrêt quasi nul :

- Amazon a RDS besoin d'une sauvegarde en ligne pour une migration quasiment sans interruption de service. Cela est dû au fait qu'Amazon RDS maintient votre base de données en attente de reconduction lorsque vous chargez vos journaux de transactions archivés. Pour de plus amples informations, veuillez consulter the section called “Migration de votre base de données DB2”.
- Vous ne pouvez pas effectuer de restauration à partir d'un compartiment Amazon S3 situé dans une Région AWS région différente de la région dans laquelle se trouve votre instance de base de données RDS for Db2.
- Amazon S3 limite la taille des fichiers chargés dans un compartiment S3 à 5 To. Si le fichier de sauvegarde de votre base de données dépasse 5 To, divisez-le en fichiers plus petits.
- Amazon RDS ne prend pas en charge les routines externes non clôturées, les restaurations incrémentielles ou Delta restaure.
- Vous ne pouvez pas effectuer de restauration à partir d'une base de données source chiffrée, mais vous pouvez effectuer une restauration sur une RDS instance de base de données Amazon cryptée.

Lorsque vous restaurez votre base de données, Amazon RDS copie votre sauvegarde puis l'extrait sur votre instance de base de données RDS pour DB2. Nous vous recommandons de prévoir pour votre instance de base de données RDS for DB2 un espace de stockage égal ou supérieur à la somme de la taille de sauvegarde et de la taille de la base de données d'origine sur le disque.

La taille maximale de la base de données restaurée est la taille maximale de base de données prise en charge moins la taille de la sauvegarde. Par exemple, si la taille maximale de base de données prise en charge est de 64 TiB et que la taille de la sauvegarde est de 30 TiB, la taille maximale de la base de données restaurée est de 34 TiB.

```
64 TiB - 30 TiB = 34 TiB
```

Sauvegarde de votre base de données sur Amazon S3

Pour sauvegarder votre base de données sur Amazon S3, vous avez besoin des AWS composants suivants :

- Un compartiment Amazon S3 pour stocker vos fichiers de sauvegarde : chargez tous les fichiers de sauvegarde que vous souhaitez migrer vers AmazonRDS. Amazon a RDS besoin d'une sauvegarde en ligne pour une migration quasiment sans interruption de service. Si vous possédez déjà un compartiment S3, vous pouvez l'utiliser. Si vous n'avez pas de compartiment S3, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.

  > **Note**
  >
  > Si votre base de données est volumineuse et que son transfert vers un compartiment S3 prend du temps, vous pouvez commander un AWS Snow Family appareil et demander AWS à effectuer la sauvegarde. Une fois que vous avez copié vos fichiers sur l'appareil et que vous les avez renvoyés à l'équipe Snow Family, celle-ci transfère vos images sauvegardées dans votre compartiment S3. Pour en savoir plus, consultez la documentation AWS Snow Family.

- Un IAM rôle pour accéder au compartiment S3 : si vous avez déjà un rôle AWS Identity and Access Management (IAM), vous pouvez utiliser ce rôle. Si vous n'avez pas de rôle, consultezÉtape 2 : Créez un IAM rôle et associez votre IAM politique.
- Une IAM politique assortie de relations de confiance et d'autorisations associées à votre IAM rôle : pour plus d'informations, consultezÉtape 1 : Créer une stratégie IAM.
- Le IAM rôle ajouté à votre instance de base de données RDS for Db2 : pour plus d'informations, consultezÉtape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2.

Création d'un groupe de stockage automatique par défaut

Votre base de données source doit disposer d'un groupe de stockage automatique par défaut. Si votre base de données ne possède pas de groupe de stockage automatique par défaut, vous devez en créer un.

Pour créer un groupe de stockage automatique par défaut

1. Connectez-vous à votre base de données source. Dans l'exemple suivant, remplacez *source_database* par le nom de votre base de données.

   ```
   db2 connect to source_database
   ```

2. Créez un groupe de stockage automatique et définissez-le comme groupe par défaut. Dans l'exemple suivant, remplacez *storage_path* par le chemin absolu vers l'emplacement du groupe de stockage.

   ```
   db2 "create stogroup IBMSTOGROUP ON storage_path set as default"
   ```

3. Mettez fin aux processus de backend.

   ```
   db2 terminate
   ```

Migration de votre base de données DB2

Après avoir sauvegardé votre base de données sur Amazon S3 et créé un groupe de stockage automatique, vous êtes prêt à migrer votre base de données DB2 vers votre instance de base de données RDS for Db2.

Pour effectuer une migration avec un temps d'arrêt proche de zéro

1. Effectuez une sauvegarde en ligne de votre base de données source. Pour plus d'informations, consultez .BACKUP DATABASE commande dans le IBM Db2 .
2. Copiez la sauvegarde de votre base de données dans un compartiment Amazon S3. Pour plus d'informations sur l'utilisation d'Amazon S3, consultez le guide de l'utilisateur d'Amazon Simple Storage Service.
3. Connectez-vous au `rdsadmin` serveur avec *master_username* et *master_password* pour votre instance de base de données RDS for DB2.

```
db2 connect to rdsadmin user master_username using master_password
```

4. (Facultatif) Pour vous assurer que votre base de données est configurée avec les paramètres optimaux pour l'opération de restauration, vous pouvez appeler the section called “rdsadmin.show_configuration” pour vérifier les valeurs de `RESTORE_DATABASE_PARALLELISM` et`RESTORE_DATABASE_NUM_BUFFERS`. Appelez the section called “rdsadmin.set_configuration” pour modifier ces valeurs, le cas échéant. La définition explicite de ces valeurs peut améliorer les performances lors de la restauration de bases de données contenant de gros volumes de données.
5. Restaurez la sauvegarde sur le serveur RDS for Db2 en appelant`rdsadmin.restore_database`. Définissez `backup_type` sur `ONLINE`. Pour de plus amples informations, veuillez consulter rdsadmin.restore_database.
6. Copiez vos journaux d'archives depuis votre serveur source vers votre compartiment S3. Pour plus d'informations, consultez la section Enregistrement des archives dans le IBM Db2 .
7. Appliquez les journaux d'archivage autant de fois que nécessaire en appelant`rdsadmin.rollforward_database`. Définissez `complete_rollforward` ce paramètre `FALSE` sur pour maintenir la base de données dans un `ROLL-FORWARD PENDING` état normal. Pour de plus amples informations, veuillez consulter rdsadmin.rollforward_database.
8. Après avoir appliqué tous les journaux d'archivage, mettez la base de données en ligne en appelant`rdsadmin.complete_rollforward`. Pour de plus amples informations, veuillez consulter rdsadmin.complete_rollforward.
9. Basculez les connexions des applications vers le serveur RDS pour DB2 en mettant à jour les points de terminaison de votre application pour la base de données ou en mettant à jour les DNS points de terminaison pour rediriger le trafic vers le serveur RDS pour DB2. Vous pouvez également utiliser la fonctionnalité de redirection automatique du client Db2 sur votre base de données Db2 autogérée avec le point de terminaison de base de données Db2. RDS Pour plus d'informations, consultez la section Description et configuration du reroutage automatique du client dans le IBM Db2 .
10. (Facultatif) Arrêtez votre base de données source.

Migration synchrone depuis Linux to Linux pour Amazon RDS pour Db2

Avec cette approche de migration, vous configurez la réplication entre votre base de données DB2 autogérée et votre instance de base de données Amazon RDS for Db2. Les modifications apportées à la base de données autogérée sont répliquées sur l'instance de base de données RDS for Db2

en temps quasi réel. Cette approche permet d'assurer une disponibilité continue et de minimiser les temps d'arrêt pendant le processus de migration.

Migration à partir de AIX or Windows to Linux pour Amazon RDS pour Db2

Avec cette approche de migration, vous utilisez des outils Db2 natifs pour sauvegarder votre base de données Db2 autogérée dans un compartiment Amazon S3. Les outils Db2 natifs incluent l'`export`utilitaire, la commande `db2move` système ou la commande `db2look` système. Votre base de données DB2 peut être autogérée ou dans Amazon Elastic Compute Cloud (AmazonEC2). Vous pouvez déplacer des données depuis votre AIX or Windows système vers votre compartiment Amazon S3. Utilisez ensuite un client Db2 pour charger les données directement depuis le compartiment S3 dans votre base de données Amazon RDS for Db2. Les temps d'arrêt dépendent de la taille de votre base de données. Pour plus d'informations sur l'utilisation d'Amazon S3, consultezIntégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3.

Pour migrer votre base de données DB2 vers RDS Db2

1. Préparez-vous à sauvegarder votre base de données. Configurez une quantité de stockage suffisante pour conserver la sauvegarde sur votre système DB2 autogéré.
2. Sauvegardez votre base de données.
   a. Exécutez le db2look commande système pour extraire le fichier de langage de définition des données (DDL) pour tous les objets.
   b. Exécutez soit l'utilitaire d'exportation DB2, soit db2move une commande système, ou une CREATE EXTERNAL TABLE instruction pour décharger les données de la table DB2 vers le stockage sur votre système DB2.
3. Déplacez votre sauvegarde vers un compartiment Amazon S3. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3.

> **Note**
>
> Si votre base de données est volumineuse et que son transfert vers un compartiment S3 prend du temps, vous pouvez commander un AWS Snow Family appareil et demander AWS à effectuer la sauvegarde. Une fois que vous avez copié vos fichiers sur l'appareil et que vous les avez renvoyés à l'équipe Snow Family, celle-ci transfère vos

images sauvegardées dans votre compartiment S3. Pour en savoir plus, consultez la documentation AWS Snow Family.

4. Utilisez un client Db2 pour charger des données directement depuis votre compartiment S3 vers votre base de données RDS for Db2. Pour de plus amples informations, veuillez consulter Migration avec Amazon S3.

Migration de données DB2 via Amazon S3 vers Amazon RDS for Db2

Avec cette approche de migration, vous enregistrez d'abord les données d'une seule table dans un fichier de données que vous placez dans un compartiment Amazon S3. Vous utilisez ensuite la LOADcommande pour charger les données de ce fichier de données dans une table de votre base de données Amazon RDS for Db2. Pour plus d'informations sur l'utilisation d'Amazon S3, consultezIntégration d'une instance de base de données Amazon RDS for Db2 à Amazon S3.

Rubriques

- Sauvegarde de vos données sur Amazon S3
- Chargement de vos données dans RDS quatre tables DB2

Sauvegarde de vos données sur Amazon S3

Pour enregistrer les données d'une seule table sur Amazon S3, utilisez un utilitaire de base de données pour extraire les données de votre système de gestion de base de données (DBMS) dans un CSV fichier. Chargez ensuite le fichier de données sur Amazon S3.

Pour stocker des fichiers de données sur Amazon S3, vous avez besoin des AWS composants suivants :

- Un compartiment Amazon S3 pour stocker vos fichiers de sauvegarde : si vous possédez déjà un compartiment S3, vous pouvez l'utiliser. Si vous n'avez pas de compartiment S3, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.
- Un IAM rôle pour accéder au compartiment S3 : si vous avez déjà un IAM rôle, vous pouvez l'utiliser. Si vous n'avez pas de rôle, consultezÉtape 2 : Créez un IAM rôle et associez votre IAM politique.
- Une IAM politique assortie de relations de confiance et d'autorisations associées à votre IAM rôle : pour plus d'informations, consultezÉtape 1 : Créer une stratégie IAM.

- Le IAM rôle ajouté à votre instance de base de données RDS for Db2 : pour plus d'informations, consultezÉtape 3 : Ajoutez votre IAM rôle à votre instance de base de données RDS for DB2.

Chargement de vos données dans RDS quatre tables DB2

Après avoir enregistré vos fichiers de données sur Amazon S3, vous pouvez charger les données de ces fichiers dans des tables individuelles de votre instance de base de données RDS for Db2.

Pour charger les données de votre table DB2 dans votre table de base de données RDS pour DB2

1. Connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

2. Cataloguez un alias d'accès au stockage qui pointe vers le compartiment Amazon S3 dans lequel sont stockés vos fichiers enregistrés. Prenez note du nom de cet alias à utiliser à l'étape suivante. Vous ne devez effectuer cette étape qu'une seule fois si vous prévoyez de charger plusieurs tables à partir de fichiers de données stockés dans le même compartiment Amazon S3.

   L'exemple suivant répertorie un alias nommé *`my_s3_alias`* qui accorde à un utilisateur nommé *`jorge_souza`* accès à un bucket nommé *`amzn-s3-demo-bucket`*.

   ```
   db2 "call rdsadmin.catalog_storage_access(?, 'my_s3_alias', 'amzn-s3-demo-bucket',
    'USER', 'jorge_souza')"
   ```

   Pour plus d'informations sur cette procédure stockée, voirthe section called “rdsadmin.catalog_storage_access”.

3. Exécutez la `LOAD` commande à l'aide de l'alias d'accès au stockage qui pointe vers votre compartiment Amazon S3.

   > **Note**
   >
   > Si la `LOAD` commande renvoie une erreur, vous devrez peut-être créer un point de terminaison de VPC passerelle pour Amazon S3 et ajouter des règles de sortie au groupe de sécurité. Pour de plus amples informations, veuillez consulter the section called “ I/O Erreur de fichier”.

L'exemple suivant charge des données à partir d'un fichier de données nommé *my_s3_datafile.csv* dans une table nommée *my_db2_table*. L'exemple suppose que le fichier de données se trouve dans le compartiment Amazon S3 que l'alias a nommé *my_s3_alias* pointe vers.

```
db2 "load from db2remote://my_s3_alias//my_s3_datafile.csv of DEL insert
 into my_db2_table";
```

L'exemple suivant charge LOBs à partir d'un fichier de données nommé *my_table1_export.ixf* dans une table nommée *my_db2_table*. L'exemple suppose que le fichier de données se trouve dans le compartiment Amazon S3 que l'alias a nommé *my_s3_alias* pointe vers.

```
db2 "call sysproc.admin_cmd('load from
 "db2remote://my_s3_alias//my_table1_export.ixf" of ixf
        lobs from "db2remote://my_s3_alias//" xml from "db2remote://my_s3_alias//"
        modified by lobsinfile implicitlyhiddeninclude identityoverride
 generatedoverride periodoverride transactionidoverride
        messages on server
        replace into "my_schema"."my_db2_table"
                                nonrecoverable
        indexing mode incremental allow no access')"
```

Répétez cette étape pour chaque fichier de données du compartiment Amazon S3 que vous souhaitez charger dans une table de votre instance de base de données RDS for Db2.

Pour plus d'informations sur la `LOAD` commande, voir LOADcommande.

Migration vers Amazon RDS pour DB2 avec () AWS Database Migration ServiceAWS DMS

Vous pouvez l'utiliser AWS DMS pour des migrations ponctuelles, puis effectuer une synchronisation entre Db2 sous Linux, Unix (comme AIX) et Windows vers Amazon RDS pour DB2. Pour plus d'informations, veuillez consulter Présentation de AWS Database Migration Service.

Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2

Vous pouvez utiliser plusieurs outils, utilitaires et commandes Db2 natifs pour déplacer des données directement d'une base de données DB2 vers une base de données Amazon RDS pour DB2. Pour utiliser ces outils Db2 natifs, vous devez être en mesure de connecter votre machine cliente à une instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Connexion d'une machine cliente à une instance de base de données Amazon RDS pour DB2.

> **Note**
>
> Une autre méthode pour déplacer vos données consiste à les enregistrer d'abord dans un compartiment Amazon S3, puis à utiliser la `LOAD` commande pour transférer ces données dans une table de votre base de données RDS for Db2. Cette méthode fournit les meilleures performances lors de la migration d'une grande quantité de données grâce à une bonne connectivité réseau entre RDS Db2 et S3. Pour de plus amples informations, veuillez consulter the section called “Migration avec Amazon S3”.

Nom de l'outil	Cas d’utilisation	Limites
look db2	Copie des métadonnées d'une base de données DB2 autogérée vers une base de données Db2 RDS pour DB2.	• Vous devez modifier la syntaxe de création de pools de mémoire tampon, de création de tablespac es et de création de rôles pour qu'elle corresponde à la syntaxe utilisée par le. Procédures stockées RDS pour DB2
IMPORT commande	Migration de petites tables et de tables contenant de gros objets (LOBs) d'une machine cliente vers l'instance de base de données RDS for DB2.	• Plus lent que l'`LOAD`utilitair e en raison `INSERT` des opérations de `DELETE` journalisation.

Nom de l'outil	Cas d'utilisation	Limites
		• Performances médiocres avec une bande passante réseau limitée.
INGEST utilité	Diffusion continue de données à partir de fichiers et de canaux sans objets volumineux (LOBs) sur la machine cliente vers l'instance de base de données RDS for Db2. Supports `INSERT` et `MERGE` opérations.	• Impossible de diffuser les fichiers de données contenantLOBs. Utilisez plutôt la `IMPORT` commande. • Connectivité requise entre la base de données DB2 autogérée et RDS la base de données DB2.
INSERT commande	Copier des données dans de petites tables depuis une base de données DB2 autogérée vers une base de données RDS pour DB2.	• Connectivité requise entre la base de données DB2 autogérée et RDS la base de données DB2. • Performances médiocres avec une bande passante réseau limitée.
LOAD CLIENT commande	Migration de petites tables sans objets volumineux (LOBs) d'une machine cliente vers l'instance de base de données RDS for DB2.	• Impossible de migrer les fichiers de données qui contiennentLOBs. Utilisez plutôt la `IMPORT` commande. • Performances médiocres avec une bande passante réseau limitée.

Connexion d'une machine cliente à une instance de base de données Amazon RDS pour DB2

Pour utiliser l'un des outils Db2 natifs pour déplacer des données d'une base de données DB2 vers une base de données Amazon RDS pour DB2, vous devez d'abord connecter votre machine cliente à une instance de base de données RDS pour DB2.

La machine cliente peut être l'une des suivantes :

- Une instance Amazon Elastic Compute Cloud (AmazonEC2) sur Linux, Windows, ou macOS. Cette instance doit se trouver dans le même cloud privé virtuel (VPC) que votre instance de base de données RDS for DB2 AWS Cloud9, ou AWS CloudShell.
- Une instance Db2 autogérée dans une instance AmazonEC2. Les instances doivent être identiquesVPC.
- Une instance Db2 autogérée dans une instance AmazonEC2. Les instances peuvent être différentes VPCs si vous avez activé le VPC peering. Pour plus d'informations, consultez la section Créer une connexion de VPC peering dans le guide de VPCpeering d'Amazon Virtual Private Cloud.
- Une machine locale en cours d'exécution Linux, Windows, ou macOS dans un environnement autogéré. Vous devez disposer d'une connectivité publique RDS pour Db2 ou activer la VPN connectivité entre les instances Db2 autogérées et. AWS

Pour connecter votre machine cliente à votre instance de base de données RDS for DB2, connectez-vous à votre machine cliente avec IBM Db2 Data Management Console. Pour plus d'informations, reportez-vous Création d'une instance de base de données Amazon RDS aux sections etIBM Db2 Data Management Console.

Vous pouvez utiliser AWS Database Migration Service (AWS DMS) pour exécuter des requêtes sur la base de données, exécuter un plan SQL d'exécution et surveiller la base de données. Pour plus d'informations, consultez Qu'est-ce que AWS Database Migration Service ? dans le guide de AWS Database Migration Service l'utilisateur.

Une fois que vous avez connecté avec succès votre machine cliente à votre instance de base de données RDS for Db2, vous êtes prêt à utiliser n'importe quel outil Db2 natif pour copier des données. Pour de plus amples informations, veuillez consulter Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2.

Copier les métadonnées de base de données de Db2 vers Amazon RDS pour Db2 avec db2look

`db2look`est un outil Db2 natif qui extrait les fichiers du langage de définition des données (DDL), les objets, les autorisations, les configurations et les mises en page WLM de base de données. Vous pouvez l'utiliser `db2look` pour copier les métadonnées d'une base de données DB2 autogérée vers une base de données Amazon RDS for DB2. Pour plus d'informations, consultez la section Imitation de bases de données à l'aide de db2look dans le IBM Db2 .

Pour copier les métadonnées de la base de données

1. Exécutez l'`db2look`outil sur votre système DB2 autogéré pour extraire le DDL fichier. Dans l'exemple suivant, remplacez *`database_name`* avec le nom de votre base de données DB2.

   ```
   db2look -d database_name -e -l -a -f -wlm -cor -createdb -printdbcfg -o db2look.sql
   ```

2. Si votre machine cliente a accès à la base de données source (DB2 autogérée) et à l'instance de base RDS de données Db2, vous pouvez créer le `db2look.sql` fichier sur la machine cliente en l'attachant directement à l'instance distante. Cataloguez ensuite l'instance Db2 autogérée distante.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *`dns_ip_address`* and *`port`* avec le DNS nom ou l'adresse IP et le numéro de port de la base de données DB2 autogérée.

      ```
      db2 catalog tcpip node srcnode REMOTE dns_ip_address server port
      ```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *`source_database_name`* and *`source_database_alias`* avec le nom de la base de données DB2 autogérée et l'alias que vous souhaitez utiliser pour cette base de données.

      ```
      db2 catalog database source_database_name as source_database_alias at node
       srcnode \
          authentication server_encrypt
      ```

   c. Joignez-le à la base de données source. Dans l'exemple suivant, remplacez *`source_database_alias`*, *`user_id`*, et *`user_password`* avec l'alias que vous avez créé à l'étape précédente, ainsi que l'ID utilisateur et le mot de passe de la base de données DB2 autogérée.

      ```
      db2look -d source_database_alias -i user_id -w user_password -e -l -a -f -wlm \
      ```

```
    -cor -createdb -printdbcfg -o db2look.sql
```

3. Si vous ne pouvez pas accéder à la base de données Db2 autogérée distante depuis la machine cliente, copiez le `db2look.sql` fichier sur la machine cliente. Cataloguez ensuite l'instance de base de données RDS for Db2.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *`dns_ip_address`* and *`port`* avec le DNS nom ou l'adresse IP et le numéro de port de l'instance de base de données RDS for Db2.

   ```
   db2 catalog tcpip node remnode REMOTE dns_ip_address server port
   ```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *`rds_database_name`* and *`rds_database_alias`* avec le nom de la base de données RDS for DB2 et l'alias que vous souhaitez utiliser pour cette base de données.

   ```
   db2 catalog database rds_database_name as rds_database_alias at node remnode \
       authentication server_encrypt
   ```

   c. Cataloguez la base de données d'administration qui gère RDS Db2. Vous ne pouvez pas utiliser cette base de données pour stocker des données.

   ```
   db2 catalog database rdsadmin as rdsadmin at node remnode authentication
    server_encrypt
   ```

4. Créez des pools de mémoire tampon et des tablespaces. L'administrateur n'a pas les privilèges nécessaires pour créer des pools de mémoire tampon ou des tablespaces. Toutefois, vous pouvez utiliser les procédures RDS stockées d'Amazon pour les créer.

   a. Recherchez les noms et les définitions des pools de mémoire tampon et des tablespaces dans le `db2look.sql` fichier.

   b. Connectez-vous à Amazon à RDS l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* and *`master_password`* avec vos propres informations.

   ```
   db2 connect to rdsadmin user master_username using master_password
   ```

   c. Créez un pool de mémoire tampon en appelant`rdsadmin.create_bufferpool`. Pour de plus amples informations, veuillez consulter rdsadmin.create_bufferpool.

```
db2 "call rdsadmin.create_bufferpool(
    'database_name',
    'buffer_pool_name',
    buffer_pool_size,
    'immediate',
    'automatic',
    page_size,
    number_block_pages,
    block_size)"
```

d. Créez un tablespace en appelant. `rdsadmin.create_tablespace` Pour de plus amples informations, veuillez consulter rdsadmin.create_tablespace.

```
db2 "call rdsadmin.create_tablespace(
    'database_name',
    'tablespace_name',
    'buffer_pool_name',
    tablespace_initial_size,
    tablespace_increase_size,
    'tablespace_type')"
```

e. Répétez les étapes c ou d pour chaque pool de mémoire tampon ou tablespace supplémentaire que vous souhaitez ajouter.

f. Mettez fin à votre connexion.

```
db2 terminate
```

5. Créez des tables et des objets.

a. Connectez-vous à votre base de données RDS for Db2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS for DB2. Dans l'exemple suivant, remplacez *rds_database_name*, *master_username*, et *master_password* avec vos propres informations.

```
db2 connect to rds_database_name user master_username using master_password
```

b. Exécutez le fichier `db2look.sql`.

```
db2 -tvf db2look.sql
```

c. Mettez fin à votre connexion.

```
db2 terminate
```

Importation de données depuis une machine cliente vers Amazon RDS pour Db2 à l'aide du IMPORT command

Vous pouvez utiliser la `IMPORT` commande depuis un ordinateur client pour importer vos données dans le serveur Amazon RDS for Db2.

> **Important**
>
> La méthode de `IMPORT` commande est utile pour migrer de petites tables et des tables contenant de gros objets (LOBs). La `IMPORT` commande est plus lente que l'`LOAD`utilitaire en raison des opérations de `DELETE` journalisation `INSERT` et. Si votre bande passante réseau entre la machine cliente et RDS pour Db2 est limitée, nous vous recommandons d'utiliser une autre approche de migration. Pour de plus amples informations, veuillez consulter Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2.

Pour importer des données dans le RDS serveur for DB2

1. Connectez-vous à votre machine cliente avec IBM Db2 Data Management Console. Pour plus d'informations, consultezConnexion à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console.
2. Cataloguez la base de données RDS for DB2 sur la machine cliente.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *`dns_ip_address`* and *`port`* avec le DNS nom ou l'adresse IP et le numéro de port de la base de données DB2 autogérée.

   ```
   db2 catalog tcpip node srcnode REMOTE dns_ip_address server port
   ```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *`source_database_name`* and *`source_database_alias`* avec le nom de la base de données DB2 autogérée et l'alias que vous souhaitez utiliser pour cette base de données.

   ```
   db2 catalog database source_database_name as source_database_alias at node
    srcnode \
   ```

```
    authentication server_encrypt
```

3. Joignez-le à la base de données source. Dans l'exemple suivant, remplacez *source_database_alias*, *user_id*, et *user_password* avec l'alias que vous avez créé à l'étape précédente, ainsi que l'ID utilisateur et le mot de passe de la base de données DB2 autogérée.

```
db2look -d source_database_alias -i user_id -w user_password -e -l -a -f -wlm \
    -cor -createdb -printdbcfg -o db2look.sql
```

4. Générez le fichier de données à l'aide de la EXPORT commande sur votre système DB2 autogéré. Dans l'exemple suivant, remplacez *directory* avec le répertoire de votre machine cliente où se trouve votre fichier de données. Remplacez *file_name* and *table_name* avec le nom du fichier de données et le nom de la table.

```
db2 "export to /directory/file_name.txt of del lobs to /directory/lobs/ \
    modified by coldel\| select * from table_name"
```

5. Connectez-vous à votre base de données RDS for Db2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS for DB2. Dans l'exemple suivant, remplacez *rds_database_alias*, *master_username,* and *master_password* avec vos propres informations.

```
db2 connect to rds_database_alias user master_username using master_password
```

6. Utilisez la IMPORT commande pour importer des données depuis un fichier de la machine cliente vers la base de données distante RDS pour DB2. Pour plus d'informations, consultez .IMPORT commande dans le IBM Db2 . Dans l'exemple suivant, remplacez *directory* and *file_name* avec le répertoire de votre machine cliente où se trouve votre fichier de données et le nom du fichier de données. Remplacez *SCHEMA_NAME* and *TABLE_NAME* avec le nom de votre schéma et de votre table.

```
db2 "IMPORT from /directory/file_name.tbl OF DEL LOBS FROM /directory/lobs/ \
    modified by coldel\| replace into SCHEMA_NAME.TABLE_NAME"
```

7. Mettez fin à votre connexion.

```
db2 terminate
```

Importation de données depuis une machine cliente vers Amazon RDS pour Db2 à l'aide du LOAD command

Vous pouvez utiliser la `LOAD CLIENT` commande pour charger des données depuis un fichier sur une machine cliente vers le RDS serveur Db2. Comme il n'existe aucune SSH connectivité avec le serveur RDS for Db2, vous pouvez utiliser la `LOAD CLIENT` commande sur votre serveur DB2 autogéré ou sur votre machine cliente DB2.

> **Important**
>
> La méthode de `LOAD CLIENT` commande est utile pour migrer de petites tables. Si votre bande passante réseau entre le client et RDS pour Db2 est limitée, nous vous recommandons d'utiliser une autre approche de migration. Pour plus d'informations, consultez le Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2.
>
> Si votre fichier de données inclut des références à des noms de fichiers d'objets volumineux, la `LOAD` commande ne fonctionnera pas car les objets volumineux (LOBs) doivent résider sur le serveur DB2. Si vous essayez LOBs de charger depuis la machine cliente vers RDS le serveur Db2, vous recevrez un SQL3025N erreur. Utilisation de la IMPORT commande à la place.

Pour charger des données sur le RDS serveur for DB2

1. Connectez-vous à votre machine cliente avec IBM Db2 Data Management Console. Pour plus d'informations, consultezConnexion à votre instance de base de données Amazon RDS for Db2 avec IBM Db2 Data Management Console.
2. Cataloguez la base de données RDS for DB2 sur la machine cliente.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *`dns_ip_address`* and *`port`* avec le DNS nom ou l'adresse IP et le numéro de port de la base de données DB2 autogérée.

   ```
   db2 catalog tcpip node srcnode REMOTE dns_ip_address server port
   ```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *`source_database_name`* and *`source_database_alias`* avec le nom de la base de données DB2 autogérée et l'alias que vous souhaitez utiliser pour cette base de données.

```
db2 catalog database source_database_name as source_database_alias at node
 srcnode \
    authentication server_encrypt
```

3. Joignez-le à la base de données source. Dans l'exemple suivant, remplacez *source_database_alias*, *user_id*, et *user_password* avec l'alias que vous avez créé à l'étape précédente, ainsi que l'ID utilisateur et le mot de passe de la base de données DB2 autogérée.

```
db2look -d source_database_alias -i user_id -w user_password -e -l -a -f -wlm \
    -cor -createdb -printdbcfg -o db2look.sql
```

4. Générez le fichier de données à l'aide de la EXPORT commande sur votre système DB2 autogéré. Dans l'exemple suivant, remplacez *directory* avec le répertoire de votre machine cliente où se trouve votre fichier de données. Remplacez *file_name* and *TABLE_NAME* avec le nom du fichier de données et le nom de la table.

```
db2 "export to /directory/file_name.txt of del modified by coldel\| \
    select * from TPCH.TABLE_NAME"
```

5. Connectez-vous à votre base de données RDS for Db2 à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS for DB2. Dans l'exemple suivant, remplacez *rds_database_alias*, *master_username*, et *master_password* avec vos propres informations.

```
db2 connect to rds_database_alias user master_username using master_password
```

6. Utilisez la LOAD commande pour charger les données d'un fichier de la machine cliente vers la base de données distante RDS pour DB2. Pour plus d'informations, consultez .LOAD commande dans le IBM Db2 . Dans l'exemple suivant, remplacez *directory* avec le répertoire de votre machine cliente où se trouve votre fichier de données. Remplacez *file_name* and *TABLE_NAME* avec le nom du fichier de données et le nom de la table.

```
db2 "LOAD CLIENT from /directory/file_name.txt \
    modified by coldel\| replace into TPCH.TABLE_NAME \
    nonrecoverable without prompting"
```

7. Mettez fin à votre connexion.

```
db2 terminate
```

Importation de données de Db2 vers Amazon RDS pour Db2 à l'aide du INSERT command

Vous pouvez utiliser la `INSERT` commande depuis un serveur DB2 autogéré pour insérer vos données dans une base de données Amazon RDS for DB2. Avec cette approche de migration, vous utilisez un surnom pour la télécommande RDS pour l'instance de base de données DB2. Votre base de données DB2 autogérée (source) doit être en mesure de se connecter à la base RDS de données Db2 (cible).

> **Important**
>
> La méthode de `INSERT` commande est utile pour migrer de petites tables. Si la bande passante réseau entre votre base de données DB2 autogérée et la base RDS de données DB2 est limitée, nous vous recommandons d'utiliser une autre approche de migration. Pour de plus amples informations, veuillez consulter Utilisation d'outils Db2 natifs pour migrer des données de Db2 vers Amazon RDS for Db2.

Pour copier des données d'une base de données DB2 autogérée vers une base de données RDS pour DB2

1. Cataloguez l'instance de base de données RDS for Db2 sur l'instance Db2 autogérée.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *dns_ip_address* and *port* avec le DNS nom ou l'adresse IP et le numéro de port de la base de données DB2 autogérée.

   ```
   db2 catalog tcpip node remnode REMOTE dns_ip_address SERVER port
   ```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *rds_database_name* avec le nom de la base de données figurant sur votre instance de base de données RDS for DB2.

   ```
   db2 catalog database rds_database_name as remdb at node remnode \
       authentication server_encrypt
   ```

2. Activez la fédération sur l'instance Db2 autogérée. Dans l'exemple suivant, remplacez *source_database_name* avec le nom de votre base de données sur l'instance Db2 autogérée.

```
db2 update dbm cfg using FEDERATED YES source_database_name
```

3. Créez des tables sur l'instance de base de données RDS for Db2.

   a. Cataloguez le nœud. Dans l'exemple suivant, remplacez *dns_ip_address* and *port* avec le DNS nom ou l'adresse IP et le numéro de port de la base de données DB2 autogérée.

```
db2 catalog tcpip node srcnode REMOTE dns_ip_address server port
```

   b. Cataloguez la base de données. Dans l'exemple suivant, remplacez *source_database_name* and *source_database_alias* avec le nom de la base de données DB2 autogérée et l'alias que vous souhaitez utiliser pour cette base de données.

```
db2 catalog database source_database_name as source_database_alias at node
 srcnode \
    authentication server_encrypt
```

4. Joignez-le à la base de données source. Dans l'exemple suivant, remplacez *source_database_alias*, *user_id*, et *user_password* avec l'alias que vous avez créé à l'étape précédente, ainsi que l'ID utilisateur et le mot de passe de la base de données DB2 autogérée.

```
db2look -d source_database_alias -i user_id -w user_password -e -l -a -f -wlm \
    -cor -createdb -printdbcfg -o db2look.sql
```

5. Configurez la fédération et créez un surnom pour la table de base de données RDS for Db2 sur l'instance Db2 autogérée.

   a. Connectez-vous à votre base de données locale. Dans l'exemple suivant, remplacez *source_database_name* avec le nom de la base de données sur votre instance Db2 autogérée.

```
db2 connect to source_database_name
```

   b. Créez un wrapper pour accéder aux sources de données DB2.

```
db2 create wrapper drda
```

c. Définissez une source de données sur une base de données fédérée. Dans l'exemple suivant, remplacez *admin* and *admin_password* avec vos informations d'identification pour votre instance Db2 autogérée. Remplacez *rds_database_name* avec le nom de la base de données figurant sur votre instance de base de données RDS for DB2.

```
db2 "create server rdsdb2 type DB2/LUW version '11.5.9.0' \
    wrapper drda authorization "admin" password "admin_password" \
    options( dbname 'rds_database_name', node 'remnode')"
```

d. Cartographiez les utilisateurs des deux bases de données. Dans l'exemple suivant, remplacez *master_username* and *master_password* avec vos informations d'identification pour votre instance de base de données RDS for DB2.

```
db2 "create user mapping for user server rdsdb2 \
    options (REMOTE_AUTHID 'master_username', REMOTE_PASSWORD
 'master_password')"
```

e. Vérifiez la connexion au serveur RDS for Db2.

```
db2 set passthru rdsdb2
```

f. Créez un surnom pour la table dans la base de données distante RDS pour DB2. Dans l'exemple suivant, remplacez *NICKNAME* and *TABLE_NAME* avec un surnom pour la table et le nom de la table.

```
db2 create nickname REMOTE.NICKNAME for RDSDB2.TABLE_NAME.NICKNAME
```

6. Insérez des données dans la table de la base de données distante RDS pour DB2. Utilisez le surnom dans une `select` instruction sur la table locale de l'instance Db2 autogérée. Dans l'exemple suivant, remplacez *NICKNAME* and *TABLE_NAME* avec un surnom pour la table et le nom de la table.

```
db2 "INSERT into REMOTE.NICKNAME select * from RDS2DB2.TABLE_NAME.NICKNAME"
```

Importation de données de Db2 vers Amazon RDS pour Db2 à l'aide du INGEST utilité

Vous pouvez utiliser cet `INGEST` utilitaire pour diffuser en continu des données provenant de fichiers et de canaux d'une machine cliente vers une instance de base de données Amazon RDS pour DB2

cible. L'`INGEST`utilitaire prend en charge `INSERT` et `MERGE` fonctionne. Pour plus d'informations, consultez la section Utilitaire d'ingestion dans le IBM Db2 .

Comme l'`INGEST`utilitaire prend en charge les pseudonymes, vous pouvez l'utiliser pour transférer des données de votre base de données DB2 autogérée vers une base de données RDS pour DB2. Cette approche fonctionne tant que la connectivité réseau existe entre les deux bases de données.

> **Important**
>
> L'`INGEST`utilitaire ne prend pas en charge les objets de grande taille (LOBs). Utilisez plutôt la `IMPORT`commande.

Pour utiliser la `RESTARTABLE` fonctionnalité de l'`INGEST`utilitaire, exécutez la commande suivante sur la base de données RDS for DB2.

```
db2 "call sysproc.sysinstallobjects('INGEST','C',NULL,NULL)"
```

Amazon RDS pour la fédération DB2

Vous pouvez utiliser votre base de données Amazon RDS for Db2 en tant que base de données fédérée. Après avoir configuré la fédération RDS pour Db2, vous pourrez accéder aux données de plusieurs bases de données et les interroger à partir de votre base de données RDS pour DB2. La fédération vous évite d'avoir à migrer des données vers votre base de données RDS for DB2 ou à consolider les données dans une seule base de données.

En utilisant votre base de données RDS for Db2 en tant que base de données fédérée, vous pouvez continuer à accéder à toutes les fonctionnalités de RDS for DB2 et tirer parti de diverses Services AWS, tout en conservant vos données dans différentes bases de données. Vous pouvez configurer à la fois une fédération homogène qui connecte différentes bases de données du même type ou une fédération hétérogène qui connecte différentes bases de données de différents types.

Vous devez d'abord connecter votre base de données DB2 RDS pour Db2 à des bases de données distantes. Vous pouvez ensuite exécuter des requêtes sur toutes vos bases de données connectées. Par exemple, vous pouvez exécuter une SQL JOIN instruction qui joint des tables de votre base de données RDS for DB2 à des tables d'une base de données Db2 on z/OS distante.

Rubriques

- Fédération homogène
- Fédération hétérogène

Fédération homogène

Vous pouvez configurer une fédération homogène entre votre base de données RDS for Db2 et la famille de produits Db2 suivante :

- DB2 pour LinuxUNIX, Windows () LUW
- DB2 iSeries
- Db2 pour z/OS

RDSpour DB2, la fédération homogène ne prend pas en charge les actions suivantes :

- Exécution de `CATALOG` commandes pour configurer un répertoire de nœuds et une base de données distante sur une base de données hôte RDS pour DB2
- Configuration de l'équilibrage de charge de travail (WLB) lors de la fédération vers DB2 sous z/OS

- Configuration du fichier de configuration du pilote du serveur de IBM données (`db2dsdriver.cfg`)

RDSpour DB2, la fédération homogène répond aux exigences suivantes :

- Vous devez créer le DRDA wrapper en `UNFENCED` mode. Si ce n'est pas le cas, la fédération ne fonctionnera pas RDS pour Db2.
- Vous devez autoriser le trafic entrant et sortant de votre RDS base de données hôte Db2 vers vos bases de données hôtes distantes. Pour de plus amples informations, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Rubriques

- Étape 1 : créer un DRDA wrapper et un serveur fédéré
- Étape 2 : Création d'un mappage utilisateur
- Étape 3 : vérifier la connexion

Étape 1 : créer un DRDA wrapper et un serveur fédéré

Pour une fédération homogène, créez un DRDA wrapper et un serveur fédéré. La connexion à l'hôte distant utilise `HOSTPORT`, et`DBNAME`.

Choisissez l'une des méthodes suivantes en fonction du type de votre base de données Db2 distante :

- Base de données DB2 pour Linux et Windows (LUX) : exécutez les SQL commandes suivantes. UNIX Dans l'exemple suivant, remplacez *`server_name`* avec le nom du serveur que vous utiliserez pour la fédération. Remplacez *`db2_version`* avec la version de votre base de données Db2 distante. Remplacez *`username`* and *`password`* avec vos informations d'identification pour la base de données Db2 distante à laquelle vous souhaitez vous connecter. Remplacez *`db_name`*, *`dns_name`*, et *`port`* avec les valeurs appropriées pour la base de données Db2 distante à laquelle vous souhaitez vous connecter.

```
create wrapper drda options(DB2_FENCED 'N');
create server server_name type DB2/LUW wrapper drda version 'db2_version'
 authorization "master_username" password "master_password" options (add DBNAME
 'db_name',add HOST 'dns_name',add PORT 'port');
```

Exemple

```
create wrapper drda options(DB2_FENCED 'N');
create server SERVER1 type DB2/LUW wrapper drda version '11.5' authorization
 "sysuser" password "******" options (add DBNAME 'TESTDB2',add HOST
 'ip-123-45-67-899.us-west-1.compute.internal',add PORT '25010');
```

- DB2 iSeries — Exécutez les SQL commandes suivantes. Dans l'exemple suivant, remplacez *wrapper_name* and *library_name* avec un nom pour votre DRDA wrapper et le fichier de bibliothèque de wrapper. Remplacez *server_name* avec le nom du serveur que vous utiliserez pour la fédération. Remplacez *db2_version* avec la version de votre base de données Db2 distante. Remplacez *username* and *password* avec vos informations d'identification pour la base de données Db2 distante à laquelle vous souhaitez vous connecter. Remplacez *dns_name*, *port*, et *db_name* avec les valeurs appropriées pour la base de données Db2 distante à laquelle vous souhaitez vous connecter.

```
create wrapper wrapper_name library 'library name' options(DB2_FENCED 'N');
create server server_name type db2/mvs version db2_version wrapper wrapper_name
 authorization "sername" password "password" options (HOST 'dns_name', PORT 'port',
 DBNAME 'db_name');
```

Exemple

```
create wrapper WRAPPER1 library 'libdb2drda.so' options(DB2_FENCED 'N');
create server SERVER1 type db2/mvs version 11 wrapper WRAPPER1 authorization
 "sysuser" password "******" options (HOST 'test1.123.com', PORT '446', DBNAME
 'STLEC1');
```

- Db2 for z/OS — Exécutez les commandes suivantesSQL. Dans l'exemple suivant, remplacez *wrapper_name* and *library_name* avec un nom pour votre DRDA wrapper et le fichier de bibliothèque de wrapper. Remplacez *server_name* avec le nom du serveur que vous utiliserez pour la fédération. Remplacez *db2_version* avec la version de votre base de données Db2 distante. Remplacez *username* and *password* avec vos informations d'identification pour la base de données Db2 distante à laquelle vous souhaitez vous connecter. Remplacez *dns_name*, *port*, et *db_name* avec les valeurs appropriées pour la base de données Db2 distante à laquelle vous souhaitez vous connecter.

```
create wrapper wrapper_name library 'library_name' options(DB2_FENCED 'N');
```

```
create server server_name type db2/mvs version db2_version wrapper wrapper_name
 authorization "username" password "password" options (HOST 'dns_name', PORT 'port',
 DBNAME 'db_name');
```

Exemple

```
create wrapper WRAPPER1 library 'libdb2drda.so' OPTIONS(DB2_FENCED 'N');
create server SERVER1 type db2/mvs version 11 wrapper WRAPPER1 authorization
 "sysuser" password "******" options (HOST 'test1.123.com', PORT '446', DBNAME
 'STLEC1');
```

Étape 2 : Création d'un mappage utilisateur

Créez un mappage utilisateur pour associer votre serveur fédéré à votre serveur de source de données en exécutant la SQL commande suivante. Dans l'exemple suivant, remplacez *server_name* avec le nom du serveur distant sur lequel vous souhaitez effectuer des opérations. Il s'agit du serveur que vous avez créé à l'étape 1. Remplacez *username* and *password* avec vos informations d'identification pour ce serveur distant.

```
create user mapping for user server server_name options (REMOTE_AUTHID 'username',
 REMOTE_PASSWORD 'password');
```

Pour plus d'informations, consultez la section Mappages d'utilisateurs dans le IBM Db2 .

Étape 3 : vérifier la connexion

Vérifiez que la configuration de votre fédération s'est bien déroulée en vérifiant la connexion. Ouvrez une session pour envoyer des SQL commandes natives à votre source de données distante à l'aide de la SET PASSTHRU commande, puis créez une table sur le serveur de données distant.

1. Ouvrez et fermez une session pour la soumettre SQL à une source de données. Dans l'exemple suivant, remplacez *server_name* avec le nom du serveur que vous avez créé pour la fédération à l'étape 1.

   ```
   set passthru server_name;
   ```

2. Créez une nouvelle table. Dans l'exemple suivant, remplacez *column_name*, *data_type*, et *value* avec les éléments appropriés pour votre table.

```
create table table_name
 ( column_name data_type(value), column_name data_type(value);
```

Pour plus d'informations, voir la CREATETABLEdéclaration contenue dans le IBM Db2 .

3. Créez un index, insérez des valeurs pour les lignes dans le tableau et réinitialisez la connexion. La réinitialisation de la connexion interrompt la connexion mais conserve les processus principaux. Dans l'exemple suivant, remplacez *index_name*, *table_name*, *column_name*, et *columnx_value* avec vos informations.

```
create index index_name on table_name(column_name);
insert into table_name values(column1_value,column2_value,column3_value);
insert into table_name values(column1_value,column2_value,column3_value);
set passthru reset;

connect reset;
```

4. Connectez-vous à votre base de données Db2 distante, créez un surnom pour votre serveur distant et effectuez des opérations. Lorsque vous avez terminé d'accéder aux données de la base de données Db2 distante, réinitialisez puis mettez fin à la connexion. Dans l'exemple suivant, remplacez *database_name* avec le nom de votre base de données Db2 distante. Remplacez *nickname* avec un nom. Remplacez *server_name* and *table_name* avec le nom du serveur distant et la table de ce serveur sur lequel vous souhaitez effectuer des opérations. Remplacez *username* avec les informations relatives à votre serveur distant. Remplacez *sql_command* avec l'opération à effectuer sur le serveur distant.

```
connect to database_name;
create nickname nickname for server_name."username"."table_name";
select sql_command from nickname;
connect reset;
terminate;
```

Exemple

L'exemple suivant crée une session directe pour autoriser les opérations sur le serveur fédéré. `testdb10`

Ensuite, il crée la table `t1` avec trois colonnes avec différents types de données.

Ensuite, l'exemple crée l'index `i1_t1` sur trois colonnes de la table`t1`. Ensuite, il insère deux lignes contenant des valeurs pour ces trois colonnes, puis se déconnecte.

Enfin, l'exemple se connecte à la base de données Db2 distante `testdb2` et crée un surnom pour la table sur `t1` le serveur fédéré. `testdb10` Il crée le surnom avec le nom d'utilisateur `TESTUSER` pour cette source de données. Une SQL commande génère toutes les données de la table`t1`. L'exemple déconnecte et met fin à la session.

```
set passthru testdbl0;

create table t1 ( c1 decimal(13,0), c2 char(200), c3 int);

create index i1_t1 on t1(c3);
insert into t1 values(1,'Test',1);
insert into t1 values(2,'Test 2',2);
connect reset;

connect to testdb2;
create nickname remote_t1 for testdbl0."TESTUSER"."T1";
select * from remote_t1;
connect reset;
terminate;
```

Fédération hétérogène

Vous pouvez configurer une fédération hétérogène entre votre base de données RDS for Db2 et d'autres sources de données telles qu'Oracle et Microsoft SQL Server. Pour une liste complète des sources de données prises LUW en charge par Db2, consultez la Matrice de support des sources de données de la fédération fournie dans Db2 LUW V11.5 sur le site de support. IBM

RDSpour DB2, la fédération hétérogène ne prend pas en charge les éléments suivants :

- Wrappers natifs pour les autres sources de données
- JDBCenveloppes pour les autres sources de données
- Fédération avec les sources de données Sybase, Informix et Teradata, car ces sources de données nécessitent l'installation d'un logiciel client sur Db2 RDS

RDSpour DB2, la fédération hétérogène répond aux exigences suivantes :

- RDScar Db2 ne prend en charge que la méthode ODBC wrapper.

- Si vous créez une définition explicite d'un wrapper, vous devez définir l'option `DB2_FENCED` sur. `'N'` Pour obtenir la liste des options d'encapsulation valides pourODBC, consultez ODBCles options du IBM Db2 .
- Vous devez autoriser le trafic entrant et sortant de votre RDS base de données hôte Db2 vers votre base de données hôte distante. Pour de plus amples informations, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Pour plus d'informations sur la fédération avec Oracle, voir Comment interroger Oracle à l'aide de Db2 Federation et du ODBC pilote ? sur le site de IBM Support.

Pour plus d'informations sur les sources de données qui prennent en charge la fédération, consultez la matrice de support des sources de données de la fédération fournie dans DB2 LUW V11.5 sur le site de support. IBM

Rubriques

- Étape 1 : Création d'un ODBC wrapper
- Étape 2 : Création d'un serveur fédéré
- Étape 3 : Création d'un mappage utilisateur
- Étape 4 : Vérifiez la connexion

Étape 1 : Création d'un ODBC wrapper

Créez un wrapper en exécutant la commande suivante :

```
db2 "create wrapper odbc options( module '/home/rdsdb/sqllib/federation/odbc/lib/
libodbc.so')"
```

Étape 2 : Création d'un serveur fédéré

Créez un serveur fédéré en exécutant la commande suivante. Dans l'exemple suivant, remplacez *`server_name`* avec le nom du serveur que vous utiliserez pour la fédération. Remplacez *`wrapper_type`* avec l'emballage approprié. Remplacez *`db_version`* avec la version de votre base de données distante. Remplacez *`dns_name`*, *`port`*, et *`service_name`* avec les valeurs appropriées pour la base de données distante à laquelle vous souhaitez vous connecter.

```
db2 "create server server_name type wrapper_type version db_version options (HOST
 'dns_name', PORT 'port', SERVICE_NAME 'service_name')“
```

Pour plus d'informations sur les types de wrapper, consultez la Matrice de support des sources de données de la fédération intégrée dans DB2 LUW V11.5 sur le site de support. IBM

Exemple

L'exemple suivant crée un serveur fédéré pour une base de données Oracle distante.

```
db2 "create server server1 type oracle_odbc version 12.1 options (HOST
 'test1.amazon.com', PORT '1521', SERVICE_NAME 'pdborcl.amazon.com')“
```

Étape 3 : Création d'un mappage utilisateur

Créez un mappage utilisateur pour associer votre serveur fédéré à votre serveur de source de données en exécutant la SQL commande suivante. Dans l'exemple suivant, remplacez *server_name* avec le nom du serveur distant sur lequel vous souhaitez effectuer des opérations. Il s'agit du serveur que vous avez créé à l'étape 2. Remplacez *username* and *password* avec vos informations d'identification pour ce serveur distant.

```
create user mapping for user server server_name options (REMOTE_AUTHID 'username',
 REMOTE_PASSWORD 'password');
```

Pour plus d'informations, consultez la section Mappages d'utilisateurs dans le IBM Db2 .

Étape 4 : Vérifiez la connexion

Vérifiez que la configuration de votre fédération s'est bien déroulée en vérifiant la connexion. Ouvrez une session pour envoyer des SQL commandes natives à votre source de données distante à l'aide de la SET PASSTHRU commande, puis créez une table sur le serveur de données distant.

1. Ouvrez et fermez une session pour la soumettre SQL à une source de données. Dans l'exemple suivant, remplacez *server_name* avec le nom du serveur que vous avez créé pour la fédération à l'étape 2.

   ```
   set passthru server_name;
   ```

2. Créez une nouvelle table. Dans l'exemple suivant, remplacez *column_name*, *data_type*, et *value* avec les éléments appropriés pour votre table.

   ```
   create table table_name
    ( column_name data_type(value), column_name data_type(value);
   ```

Pour plus d'informations, voir la CREATETABLEdéclaration contenue dans le IBM Db2 .

3. Créez un index, insérez des valeurs pour les lignes dans le tableau et réinitialisez la connexion. La réinitialisation de la connexion interrompt la connexion mais conserve les processus principaux. Dans l'exemple suivant, remplacez *index_name*, *table_name*, *column_name*, et *columnx_value* avec vos informations.

```
create index index_name on table_name(column_name);
insert into table_name values(column1_value,column2_value,column3_value);
insert into table_name values(column1_value,column2_value,column3_value);
set passthru reset;

connect reset;
```

4. Connectez-vous à votre base de données Db2 distante, créez un surnom pour votre serveur distant et effectuez des opérations. Lorsque vous avez terminé d'accéder aux données de la base de données Db2 distante, réinitialisez puis mettez fin à la connexion. Dans l'exemple suivant, remplacez *database_name* avec le nom de votre base de données Db2 distante. Remplacez *nickname* avec un nom. Remplacez *server_name* and *table_name* avec le nom du serveur distant et la table de ce serveur sur lequel vous souhaitez effectuer des opérations. Remplacez *username* avec les informations relatives à votre serveur distant. Remplacez *sql_command* avec l'opération à effectuer sur le serveur distant.

```
connect to database_name;
create nickname nickname for server_name."username"."table_name";
select sql_command from nickname;
connect reset;
terminate;
```

Exemple

L'exemple suivant crée une session directe pour autoriser les opérations sur le serveur fédéré. `testdb10`

Ensuite, il crée la table `t1` avec trois colonnes avec différents types de données.

Ensuite, l'exemple crée l'index `i1_t1` sur trois colonnes de la table`t1`. Ensuite, il insère deux lignes contenant des valeurs pour ces trois colonnes, puis se déconnecte.

Enfin, l'exemple se connecte à la base de données Db2 distante `testdb2` et crée un surnom pour la table sur `t1` le serveur fédéré. `testdb10` Il crée le surnom avec le nom d'utilisateur `TESTUSER` pour cette source de données. Une SQL commande génère toutes les données de la table`t1`. L'exemple déconnecte et met fin à la session.

```
set passthru testdbl0;

create table t1 ( c1 decimal(13,0), c2 char(200), c3 int);

create index i1_t1 on t1(c3);
insert into t1 values(1,'Test',1);
insert into t1 values(2,'Test 2',2);
connect reset;

connect to testdb2;
create nickname remote_t1 for testdbl0."TESTUSER"."T1";
select * from remote_t1;
connect reset;
terminate;
```

Utilisation de répliques pour Amazon RDS pour DB2

Pour configurer la réplication entre les instances de base de données RDS pour DB2, vous pouvez créer des répliques de bases de données. Pour un résumé des différences entre les répliques RDS pour DB2 et les autres moteurs de base de données, consultez. Différences entre les répliques de lecture pour les moteurs de base de données

Une base de données répliquée DB2 est une copie physique de votre base de données principale. Une réplique Db2 en mode veille est appelée réplique en veille. DB2 n'autorise pas la lecture ou l'écriture dans une réplique, mais vous pouvez promouvoir une réplique pour la rendre inscriptible. La réplique promue contient les données répliquées au moment où la demande de promotion a été faite. RDS pour Db2 utilise Db2 High Availability Disaster Recover (HADR) pour la réplication. Pour plus d'informations, consultez la section High Availability Disaster Recovery (HADR) dans la IBM Db2 documentation.

Répliques de secours

Lorsque vous créez ou modifiez une réplique Db2, vous pouvez la placer en mode veille. Pour les répliques de secours, la variable d'environnement Db2 `DB2_HADR_ROS` est définie de `OFF` telle sorte que la base de données de répliques n'accepte pas les connexions utilisateur. Les répliques de secours sont principalement utilisées pour la reprise après sinistre entre régions.

Une réplique de secours ne peut pas servir une charge de travail en lecture seule. La réplique de secours ne contient aucun journal d'archivage.

Vous pouvez créer jusqu'à trois répliques à partir d'une instance de base de données source.

Activations de bases de données

Db2 HADR est configuré au niveau de la base de données. Une fois que vous avez créé des répliques, le HADR est défini pour toutes les bases de données DB2`rdsadmin`, y compris celles que RDS gère entièrement. Avant de créer des répliques DB2, vous devez activer explicitement toutes les bases de données. Dans le cas contraire, la création de répliques échoue et Amazon RDS émet un événement. Une fois qu'une instance de base de données possède une ou plusieurs répliques, vous ne pouvez activer ou désactiver aucune base de données sur l'instance de base de données à l'aide des procédures `rdsadmin.deactivate_database` stockées `rdsadmin.activate_database` ou. Vous pouvez toutefois appeler la procédure `rdsadmin.reactivate_database` stockée pour réactiver les bases de données. Pour de plus amples informations, veuillez consulter Procédures stockées pour les bases de données pour RDS pour Db2.

Configurations HADR

Vous pouvez voir toutes les configurations HADR d'une base de données en vous connectant à la base de données, puis en l'exécutant`db2 get db cfg`.

Conservation des journaux d'archivage

Amazon RDS purge les journaux d'une instance de base de données principale une fois que les conditions suivantes sont remplies :

- Les journaux datent d'au moins deux heures.
- Le paramètre relatif aux heures de conservation des journaux d'archives est dépassé.
- Les journaux d'archivage ont été correctement répliqués sur toutes les instances de base de données de réplication. Cette condition s'applique à la fois aux instances de base de données situées dans la même région Région AWS et aux instances de base de données interrégionales.

Pour plus d'informations sur la définition des heures de conservation des journaux d'archivage, veuillez consulter rdsadmin.set_archive_log_retention.

Amazon RDS vérifie et nettoie chaque base de données individuellement. Si une base de données perd la connexion HADR ou si les informations relatives à cette connexion ne sont pas disponibles, Amazon RDS ignore la base de données et ne purge pas les journaux d'archivage.

Pannes lors de la réplication DB2

Lorsque vous créez une réplique, Amazon RDS prend un instantané de base de données de votre instance de base de données source et commence la réplication. Lorsque l'opération de capture instantanée de base de données commence, l'instance de base de données source subit une très brève I/O suspension. La I/O suspension dure généralement une seconde environ. Toutefois, si l'instance de base de données source est un déploiement multi-AZ, l'instance de base de données source ne subit aucune I/O suspension. En effet, dans le cas des déploiements multi-AZ, le snapshot est pris à partir de l'instance de base de données secondaire.

Le snapshot de base de données devient la réplique DB2. Amazon RDS définit les paramètres et les autorisations nécessaires pour la base de données source et la réplique sans interruption de service. De même, si vous supprimez un réplica, aucune panne ne se produit.

Exigences et considérations relatives au RDS pour les répliques DB2

Avant de créer une réplique DB2, familiarisez-vous avec les exigences et considérations suivantes.

Exigences de version et de licence pour les répliques RDS pour Db2

Avant de créer une réplique RDS pour DB2, consultez les informations suivantes :

- Db2 Advanced Edition (AE) et Standard Edition (SE) peuvent créer des répliques en mode veille à la fois pour le modèle Bring your own license (BYOL) et pour le modèle de licence Db2 Through. AWS Marketplace
- Vous devez spécifier un groupe de paramètres personnalisé pour la réplique lue. Pour les répliques qui utilisent le modèle BYOL, ce groupe de paramètres personnalisés doit inclure votre IBM Site ID et. IBM Customer ID Vous pouvez spécifier ce groupe de paramètres personnalisé pour la réplique en lecture à l'aide de l'API AWS Management Console, de AWS CLI , ou de l'API RDS. Pour de plus amples informations, veuillez consulter IBM IDs pour apporter votre propre licence (BYOL) pour Db2.
- Pour les répliques de secours, le nombre de vCPU est de deux, quelle que soit la taille de l'instance de base de données. Les éléments suivants décrivent les différences entre les types de licence RDS pour DB2 :
  - Pour le modèle BYOL, les AWS License Manager configurations montrent que RDS pour les instances de base de données DB2 utilise deux v. CPUs
  - Pour le AWS Marketplace modèle de licence DB2 via, les factures reflètent les coûts de licence pour deux v. CPUs
- Toutes les versions de DB2 11.5 prennent en charge les instances de base de données répliquées. Les répliques DB2 prennent en charge les mises à niveau de versions mineures.

Considérations relatives à la sauvegarde et à la restauration pour les répliques RDS pour DB2

Avant de créer une réplique RDS pour DB2, tenez compte des points suivants :

- Pour créer des instantanés de répliques RDS pour Db2 ou activer les sauvegardes automatiques, veillez à définir manuellement la période de conservation des sauvegardes. Les sauvegardes automatiques ne sont pas activées par défaut.
- Lorsque vous restaurez une sauvegarde de réplica, vous rétablissez l'heure de la base de données, et non l'heure à laquelle la sauvegarde a été effectuée. L'heure de la base de données

désigne la dernière heure de transaction appliquée des données dans la sauvegarde. La différence est significative car une réplique peut prendre des minutes, voire des heures, de retard par rapport à la base de données principale. Lorsqu'il existe plusieurs bases de données, RDS pour DB2 utilise l'heure de base de données la plus ancienne.

Pour trouver la différence, exécutez la AWS CLI describe-db-snapshotscommande ou appelez l'DBSnapshotsopération RDS API Describe. Comparez la `SnapshotDatabaseTime` valeur à la `OriginalSnapshotCreateTime` valeur. La `SnapshotDatabaseTime` valeur est l'heure de la sauvegarde de la réplique dans la base de données. La `OriginalSnapshotCreateTime` valeur est la dernière transaction appliquée sur la base de données principale.

Considérations relatives à la réplication pour les répliques RDS pour DB2

Avant de créer une réplique RDS pour DB2, tenez compte des points suivants :

- La réplication utilise Db2 HADR pour toutes les bases de données de l'instance de base de données RDS for Db2.
- La réplication ne prend pas en charge la `LOAD` commande. Si vous exécutez la `LOAD` commande depuis l'instance de base de données source, vous recevrez des données incohérentes.
- RDS pour DB2 ne reproduit pas les éléments suivants :
  - Accès au stockage. Soyez conscient des données, telles que les tables externes, qui dépendent de l'accès au stockage.
  - Non en ligne. LOBs
  - Binaires de procédures stockées externes (en C ou Java).
- Pour les répliques de secours, RDS pour Db2 réplique les éléments suivants :
  - Utilisateurs locaux, à l'exception des utilisateurs principaux
  - Paramètres de configuration de base de données

Considérations diverses relatives aux répliques RDS pour Db2

Avant de créer une réplique RDS pour DB2, tenez compte des points suivants :

- RDS for Db2 réplique les configurations de base de données sur les répliques. Lorsque RDS for Db2 fait la promotion d'une réplique, il désactive et active chaque base de données.

- RDS pour Db2 réplique les utilisateurs locaux, mais pas l'utilisateur principal, vers les répliques. Vous pouvez modifier l'utilisateur principal sur la réplique. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Toutes les bases de données doivent être actives. Pour plus d'informations sur l'activation des bases de données, consultezProcédures stockées pour les bases de données pour RDS pour Db2.
- Toutes les procédures stockées de création, de suppression, de restauration ou de transfert de bases de données doivent être effectuées avant de créer une réplique. Pour plus d'informations sur ces procédures stockées, consultezProcédures stockées pour les bases de données pour RDS pour Db2.
- Lorsque la réplique est créée, Amazon RDS définit le paramètre au niveau de la base de données `blocknonlogged` pour toutes les bases de données de l'instance de base de données source sur. `YES` Lorsque la réplique source redevient une instance autonome, Amazon RDS redéfinit la valeur à. `NO` Pour plus d'informations, voir blocknonlogged - Création de blocs de tables autorisant le paramètre de configuration des activités non enregistrées dans la documentation. IBM Db2
- Lorsque la réplique est créée, Amazon RDS définit le paramètre au niveau de la base de données `logindexbuild` pour toutes les bases de données de l'instance de base de données source sur. `YES` Lorsque la réplique source redevient une instance autonome, Amazon RDS redéfinit la valeur à. `NO` Pour plus d'informations, voir logindexbuild - Paramètre de configuration créé par les pages d'index de log dans la IBM Db2 documentation.

Préparation à la création d'une réplique RDS pour DB2

Avant de commencer à utiliser votre réplique, effectuez les tâches suivantes :

Tâches

- Activer les sauvegardes automatiques
- Planifier les ressources de calcul et de stockage
- Préparation des bases de données

Activer les sauvegardes automatiques

Avant qu'une instance de base de données puisse être utilisée comme instance de bases de données source, vous devez activer les sauvegardes automatiques sur l'instance de base de données source. Pour savoir comment effectuer cette procédure, veuillez consulter Activation des sauvegardes automatiques.

Planifier les ressources de calcul et de stockage

Assurez-vous que l'instance de base de données source et ses réplicas sont dimensionnés correctement, en termes de calcul et de stockage, en fonction de leur charge opérationnelle. Si un réplica atteint sa capacité en termes de ressources de calcul, de réseau ou de stockage, il arrête de recevoir ou d'appliquer les modifications provenant de sa source. RDS for Db2 n'intervient pas pour atténuer le décalage de réplication élevé entre une instance de base de données source et ses répliques. Vous pouvez modifier les ressources de stockage et d'UC d'un réplica indépendamment de sa source et d'autres réplicas.

Préparation des bases de données

Avant de créer une réplique, assurez-vous que l'instance de base de données contient toutes les bases de données que vous souhaitez voir présentes sur l'instance de base de données. Après la création de la réplique, vous ne pouvez pas créer, supprimer ou restaurer de manière native une base de données sur l'instance de base de données. Tous les appels vers rdsadmin.create_databaserdsadmin.drop_database, ou les procédures rdsadmin.restore_database stockées échouent.

Assurez-vous également que toutes les bases de données de l'instance de base de données sont actives. Dans le cas contraire, la création de répliques échoue. Pour plus d'informations sur l'activation des bases de données, consultezProcédures stockées pour les bases de données pour RDS pour Db2.

Création d'une réplique RDS pour DB2 en mode veille

Vous pouvez créer une réplique en mode veille à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS.

console

Pour créer une réplique de secours à partir d'une instance de base de données RDS source pour DB2

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données RDS pour DB2 que vous souhaitez utiliser comme source pour une réplique de secours.

4. Sous Actions, choisissez Créer un réplica.
5. Pour le mode Replica, choisissez Standby.
6. Choisissez les paramètres que vous souhaitez utiliser. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture. Ajustez les autres paramètres selon les besoins.
7. Pour les régions, choisissez l' Région AWS endroit où la réplique de secours sera lancée.
8. Choisissez la taille de votre instance et le type de stockage. Nous vous recommandons d'utiliser le même type de stockage et la même classe d'instance de base de données que l'instance de base de données source pour le réplica en lecture.
9. Pour le déploiement multi-AZ, choisissez Créer une instance de secours pour créer une instance de secours de votre réplique dans une autre zone de disponibilité afin de prendre en charge le basculement de la réplique de secours.
10. Choisissez les autres paramètres que vous voulez utiliser.
11. Choisissez Créer un réplica.

Sur la page Bases de données, le réplica de secours a le rôle Replica.

AWS CLI

Pour créer une réplique Db2 en mode veille, définissez `--replica-mode` la valeur `mounted` dans la AWS CLI commande create-db-instance-read-replica.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier my_read_replica \
    --source-db-instance-identifier my_db_instance \
    --replica-mode mounted
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier my_read_replica ^
    --source-db-instance-identifier my_db_instance ^
    --replica-mode mounted
```

API RDS

Pour créer une réplique Db2 en mode veille, spécifiez l'opération Create `ReplicaMode=mounted` dans l'API RDS. DBInstance ReadReplica

Utilisation de RDS pour les sauvegardes de répliques DB2

Vous pouvez créer et restaurer des sauvegardes d'une réplique RDS pour DB2. RDS pour DB2 prend en charge à la fois les sauvegardes automatiques et les instantanés manuels. Pour de plus amples informations, veuillez consulter Sauvegarde, restauration et exportation de données. Les sections suivantes décrivent les principales différences entre la gestion des sauvegardes d'une base de données principale et celle d'une réplique RDS pour DB2.

Rubriques

- Activation de RDS pour les sauvegardes de répliques DB2
- Restauration d'une sauvegarde de répliques RDS pour DB2

Activation de RDS pour les sauvegardes de répliques DB2

Les sauvegardes automatiques ne sont pas activées par défaut sur une réplique RDS pour DB2. Activez les sauvegardes automatisées en définissant la période de conservation des sauvegardes sur une valeur positive différente de zéro.

console

Pour activer immédiatement les sauvegardes automatisées

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez modifier.
3. Sélectionnez Modify (Modifier).
4. Pour la période de conservation des sauvegardes, choisissez une valeur positive différente de zéro, par exemple trois jours.
5. Choisissez Continuer.
6. Choisissez Apply immediately (Appliquer immédiatement).

7. Choisissez Modifier l'instance de base de données pour enregistrer vos modifications et activer les sauvegardes automatisées.

AWS CLI

Pour activer les sauvegardes automatiques, utilisez la AWS CLI modify-db-instancecommande.

Incluez les paramètres suivants :

- `--db-instance-identifier`
- `--backup-retention-period`
- `--apply-immediately` ou `--no-apply-immediately`

L'exemple suivant active les sauvegardes automatisées en définissant la période de conservation des sauvegardes à trois jours. Les modifications sont appliquées immédiatement.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my_db_instance  \
    --backup-retention-period 3 \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my_db_instance  ^
    --backup-retention-period 3 ^
    --apply-immediately
```

API RDS

Pour activer les sauvegardes automatisées, utilisez l'DBInstanceopération de modification de l'API RDS avec les paramètres obligatoires suivants :

- `DBInstanceIdentifier`
- `BackupRetentionPeriod`

Restauration d'une sauvegarde de répliques RDS pour DB2

Vous pouvez restaurer une sauvegarde de réplique RDS pour DB2 de la même manière que vous pouvez restaurer une sauvegarde de la base de données principale. Pour plus d'informations, consultez les rubriques suivantes :

- Restauration vers une instance de base de données
- Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

Le principal élément à prendre en compte lorsque vous restaurez une sauvegarde de réplica est de déterminer l'instant dans le passé auquel vous effectuez la restauration. L'heure de la base de données désigne la dernière heure de transaction appliquée des données dans la sauvegarde. Lorsque vous restaurez une sauvegarde de réplica, vous rétablissez l'heure de la base de données, et non l'heure à laquelle la sauvegarde s'est terminée. La différence est significative car une réplique peut prendre des minutes ou des heures de retard par rapport à la base de données principale. Ainsi, l'heure de la sauvegarde de la base de données d'une réplique et le moment auquel vous la restaurez peuvent être bien antérieurs à l'heure de création de la sauvegarde.

Pour connaître la différence entre l'heure de la base de données et l'heure de création, exécutez la AWS CLI describe-db-snapshotscommande ou appelez l'DBSnapshotsopération RDS API Describe. Comparez la `SnapshotDatabaseTime` valeur et la `OriginalSnapshotCreateTime` valeur. La `SnapshotDatabaseTime` valeur correspond à l'heure de base de données la plus ancienne parmi toutes les bases de données de la copie de sauvegarde. La `OriginalSnapshotCreateTime` valeur est la dernière transaction appliquée sur la base de données principale. Notez que les délais de réplication peuvent être différents pour plusieurs bases de données et que le temps de la base de données peut se situer entre ces deux temps.

L' AWS CLI exemple suivant montre la différence entre les deux temps :

```
aws rds describe-db-snapshots \
    --db-instance-identifier my_db2_replica
    --db-snapshot-identifier my_replica_snapshot

{
    "DBSnapshots": [
        {
            "DBSnapshotIdentifier": "my_replica_snapshot",
            "DBInstanceIdentifier": "my_db2_replica",
            "SnapshotDatabaseTime": "2022-07-26T17:49:44Z",
```

```
            ...
            "OriginalSnapshotCreateTime": "2021-07-26T19:49:44Z"
        }
    ]
}
```

Résolution des problèmes de réplication de RDS pour DB2

Ce contenu décrit les problèmes et solutions possibles liés à la réplication RDS pour DB2.

Rubriques

- Surveillance du décalage de réplication DB2
- Erreurs de réplication DB2

Surveillance du décalage de réplication DB2

Pour surveiller le délai de réplication dans Amazon CloudWatch, consultez la `ReplicaLag` métrique Amazon RDS. Pour obtenir plus d'informations sur la durée du retard de réplication, consultez Supervision de la réplication en lecture et CloudWatch Métriques Amazon pour Amazon RDS.

Pour une réplique de secours, si le délai est trop long, interrogez la `MON_GET_HADR` table pour connaître l'état de l'instance de base de données source. N'interrogez pas l'instance de base de données de réplique car les instances de base de données de réplique n'acceptent pas les connexions utilisateur.

Erreurs de réplication DB2

La réplication DB2 peut présenter un état d'erreur pour plusieurs raisons. Surveillez les événements et l'état de l'instance de base de données pour vous assurer que l'instance de base de données est en train de se répliquer. Pour confirmer que l'instance de base de données est en train de se répliquer, consultez les journaux de diagnostic de la réplique dans la console Amazon RDS. Dans les journaux, recherchez les erreurs dans les messages HADR. Comparez le numéro de séquence du journal au numéro de séquence principal. Pour de plus amples informations, veuillez consulter Fichiers journaux RDS de base de données Amazon pour DB2.

Options pour Amazon RDS pour les instances de base de données DB2

Vous trouverez ci-dessous les options, ou fonctionnalités supplémentaires, disponibles pour les RDS instances Amazon exécutant le moteur de base de données Db2. Pour activer ces options, vous pouvez les ajouter à un groupe d'options personnalisé, puis associer ce dernier à votre instance de base de données. Pour plus d'informations sur l'utilisation de groupes d'options, consultez Utilisation de groupes d'options.

Amazon RDS prend en charge les options suivantes pour DB2 :

Option	ID d'option
Journalisation des audits DB2	`DB2_AUDIT`

Journalisation des audits DB2

Avec la journalisation des audits DB2, Amazon RDS enregistre l'activité de la base de données, y compris les utilisateurs qui se connectent à la base de données et les requêtes exécutées sur la base de données. RDS télécharge les journaux d'audit complets dans votre compartiment Amazon S3, en utilisant le rôle AWS Identity and Access Management (IAM) que vous fournissez.

Rubriques

- Configuration de la journalisation des audits DB2
- Gestion de la journalisation des audits DB2
- Consultation des journaux d'audit
- Résolution des problèmes liés à l'enregistrement des audits DB2

Configuration de la journalisation des audits DB2

Pour activer la journalisation des audits pour une base de données Amazon RDS pour DB2, vous devez activer l'`DB2_AUDIT`option sur l'instance de base de données RDS pour DB2. Configurez ensuite une politique d'audit pour activer la fonctionnalité pour la base de données spécifique. Pour activer l'option sur l'instance de base de données RDS pour DB2, vous configurez les paramètres de l'`DB2_AUDIT`option. Pour ce faire, vous devez fournir les noms de ressources Amazon (ARNs) pour votre compartiment Amazon S3 et le rôle IAM avec les autorisations d'accès à votre compartiment.

Pour configurer la journalisation d'audit DB2 pour une base de données RDS pour DB2, procédez comme suit.

Rubriques

- Étape 1 : Créer un compartiment Amazon S3
- Étape 2 : Création d'une politique IAM
- Étape 3 : créer un rôle IAM et associer votre politique IAM
- Étape 4 : Configuration d'un groupe d'options pour la journalisation des audits DB2
- Étape 5 : Configuration de la politique d'audit
- Étape 6 : Vérifiez la configuration de l'audit

Étape 1 : Créer un compartiment Amazon S3

Si ce n'est pas déjà fait, créez un compartiment Amazon S3 dans lequel Amazon RDS peut télécharger les fichiers journaux d'audit de votre base de données RDS for DB2. Les restrictions suivantes s'appliquent au compartiment S3 que vous utilisez comme cible pour vos fichiers d'audit :

- Il doit se trouver dans le même emplacement Région AWS que votre instance de base de données RDS pour DB2.
- Il ne doit pas être ouvert au public.
- Le propriétaire du compartiment doit également être le propriétaire du rôle IAM.

Pour savoir comment créer un compartiment Amazon S3, consultez la section Création d'un compartiment dans le guide de l'utilisateur Amazon S3.

Une fois que vous avez activé la journalisation des audits, Amazon RDS envoie automatiquement les journaux depuis votre instance de base de données vers les emplacements suivants :

- Journaux au niveau des instances de base de données — *bucket_name*`/db2-audit-logs/`*dbi_resource_id*`/`*date_time_utc*`/`
- Journaux au niveau des bases de données : *bucket_name*`/db2-audit-logs/`*dbi_resource_id*`/`*date_time_utc*`/`*db_name*`/`

Prenez note de l'Amazon Resource Name (ARN) de votre compartiment. Ces informations sont nécessaires pour effectuer les étapes suivantes.

Étape 2 : Création d'une politique IAM

Créez une politique IAM avec les autorisations requises pour transférer les fichiers journaux d'audit de votre instance de base de données vers votre compartiment Amazon S3. Cette étape suppose que vous disposez d'un compartiment S3.

Avant de créer la politique, collectez les informations suivantes :

- L'ARN de votre bucket.
- L'ARN de votre clé AWS Key Management Service (AWS KMS), si votre compartiment utilise SSE-KMS le chiffrement.

Créez une politique IAM qui inclut les autorisations suivantes :

```
"s3:ListBucket",
 "s3:GetBucketAcl",
 "s3:GetBucketLocation",
 "s3:PutObject",
 "s3:ListMultipartUploadParts",
 "s3:AbortMultipartUpload",
 "s3:ListAllMyBuckets"
```

> **Note**
>
> Amazon RDS a besoin d'une `s3:ListAllMyBuckets` action interne pour vérifier qu'il Compte AWS est propriétaire à la fois du compartiment S3 et de l'instance de base de données RDS pour DB2.

Si votre compartiment utilise SSE-KMS le chiffrement, incluez également les autorisations suivantes pour votre rôle et votre AWS KMS clé IAM.

Incluez les autorisations suivantes dans la politique de votre rôle IAM.

```
"kms:GenerateDataKey",
 "kms:Decrypt"
```

Incluez les autorisations suivantes dans la politique de clé pour votre AWS KMS clé. *111122223333*Remplacez-le par votre numéro de compte et *AR0A123456789EXAMPLE* par le nom de votre rôle IAM.

```
{
  "Sid": "Allow RDS role to use the key",
  "Effect": "Allow",
  "Principal": {
    "AWS": [
      "arn:aws:sts::111122223333:assumed-role/AR0A123456789EXAMPLE/RDS-Db2Audit",
      "arn:aws:iam::111122223333:role/AR0A123456789EXAMPLE"
    ]
  },
  "Action": [
    "kms:GenerateDataKey",
    "kms:Decrypt"
  ],
  "Resource": "*"
```

```
}
```

Vous pouvez créer une politique IAM en utilisant le AWS Management Console ou le AWS Command Line Interface (AWS CLI).

console

Pour créer une politique IAM afin d'autoriser Amazon RDS à accéder à votre compartiment Amazon S3

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le volet de navigation, choisissez Politiques.
3. Choisissez Create policy, puis choisissez JSON.
4. Dans Ajouter des actions, filtrez par S3. Ajoutez l'accès ListBucketGetBucketAcl, et GetBucketLocation.
5. Pour Ajouter une ressource, choisissez Ajouter. Pour Type de ressource, choisissez bucket, puis entrez le nom de votre bucket. Choisissez ensuite Ajouter une ressource.
6. Choisissez Ajouter un nouveau relevé.
7. Dans Ajouter des actions, filtrez par S3. Ajoutez l'accès PutObjectListMultipartUploadParts, et AbortMultipartUpload.
8. Pour Ajouter une ressource, choisissez Ajouter. Pour Type de ressource, choisissez un objet, puis entrez*your bucket name/**. Choisissez ensuite Ajouter une ressource.
9. Choisissez Ajouter un nouveau relevé.
10. Dans Ajouter des actions, filtrez par S3. Ajoutez un accès ListAllMyBuckets.
11. Pour Ajouter une ressource, choisissez Ajouter. Pour Type de ressource, sélectionnez Toutes les ressources. Choisissez ensuite Ajouter une ressource.
12. Si vous utilisez vos propres clés KMS pour chiffrer les données :
    1. Choisissez Ajouter un nouveau relevé.
    2. Dans Ajouter des actions, filtrez par KMS. Ajoutez l'accès GenerateDataKeyet déchiffrez.
    3. Pour Ajouter une ressource, choisissez Ajouter. Pour Type de ressource, sélectionnez Toutes les ressources. Choisissez ensuite Ajouter une ressource.
13. Choisissez Suivant.
14. Dans Nom de la stratégie, entrez le nom de cette stratégie.

15. (Facultatif) Pour Description, saisissez une description pour cette stratégie.
16. Choisissez Create Policy (Créer une politique).

AWS CLI

Pour créer une politique IAM afin d'autoriser Amazon RDS à accéder à votre compartiment Amazon S3

1. Exécutez la commande create-policy. Dans l'exemple suivant, remplacez *iam_policy_name* et *amzn-s3-demo-bucket* par le nom de votre politique IAM et le nom de votre compartiment Amazon S3 cible.

   Pour LinuxmacOS, ou Unix :

```
aws iam create-policy \
    --policy-name iam_policy_name \
    --policy-document '{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "Statement1",
                "Effect": "Allow",
                "Action": [
                    "s3:ListBucket",
                    "s3:GetBucketAcl",
                    "s3:GetBucketLocation"
                ],
                "Resource": [
                    "arn:aws:s3:::amzn-s3-demo-bucket"
                ]
            },
            {
                "Sid": "Statement2",
                "Effect": "Allow",
                "Action": [
                    "s3:PutObject",
                    "s3:ListMultipartUploadParts",
                    "s3:AbortMultipartUpload"
                ],
                "Resource": [
                    "arn:aws:s3:::amzn-s3-demo-bucket/*"
                ]
```

```
            },
            {
                "Sid": "Statement3",
                "Effect": "Allow",
                "Action": [
                    "s3:ListAllMyBuckets"
                ],
                "Resource": [
                    "*"
                ]
            },
            {
                "Sid": "Statement4",
                "Effect": "Allow",
                "Action": [
                    "kms:GenerateDataKey",
                    "kms:Decrypt"
                ],
                "Resource": [
                    "*"
                ]
            }
        ]
  }'
```

Dans Windows :

```
aws iam create-policy ^
    --policy-name iam_policy_name ^
    --policy-document '{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Sid": "Statement1",
                "Effect": "Allow",
                "Action": [
                    "s3:ListBucket",
                    "s3:GetBucketAcl",
                    "s3:GetBucketLocation"
                ],
                "Resource": [
                    "arn:aws:s3:::amzn-s3-demo-bucket"
                ]
```

```
            },
            {
                "Sid": "Statement2",
                "Effect": "Allow",
                "Action": [
                    "s3:PutObject",
                    "s3:ListMultipartUploadParts",
                    "s3:AbortMultipartUpload"
                ],
                "Resource": [
                    "arn:aws:s3:::amzn-s3-demo-bucket/*"
                ]
            },
            {
                "Sid": "Statement3",
                "Effect": "Allow",
                "Action": [
                    "s3:ListAllMyBuckets"
                ],
                "Resource": [
                    "*"
                ]
            },
            {
                "Sid": "Statement4",
                "Effect": "Allow",
                "Action": [
                    "kms:GenerateDataKey",
                    "kms:Decrypt"
                ],
                "Resource": [
                    "*"
                ]
            }
        ]
}'
```

2. Une fois la stratégie créée, notez son ARN. Vous avez besoin de l'ARN pour[Étape 3 : créer un rôle IAM et associer votre politique IAM](#).

Pour plus d'informations sur la création d'une stratégie IAM, consultez la section [Création de politiques IAM](#) dans le Guide de l'utilisateur IAM.

Étape 3 : créer un rôle IAM et associer votre politique IAM

Cette étape suppose que vous avez créé la politique IAM dansÉtape 2 : Création d'une politique IAM. Au cours de cette étape, vous créez un rôle IAM pour votre instance de base de données RDS pour DB2, puis vous attachez votre politique IAM au rôle.

Vous pouvez créer un rôle IAM pour votre instance de base de données à l'aide de la console ou du AWS CLI.

console

Pour créer un rôle IAM et y associer votre politique IAM

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Sélectionnez Create role (Créer un rôle).
4. Pour Type d'entité de confiance, sélectionnez Service AWS.
5. Pour Service ou cas d'utilisation, sélectionnez RDS, puis sélectionnez RDS — Ajouter un rôle à la base de données.
6. Choisissez Suivant.
7. Pour les politiques d'autorisations, recherchez et sélectionnez le nom de la politique IAM que vous avez créée.
8. Choisissez Suivant.
9. Pour Role name (Nom du rôle), saisissez un nom de rôle.
10. (Facultatif) Pour Description, saisissez une description pour le nouveau rôle.
11. Choisissez Créer un rôle.

AWS CLI

Pour créer un rôle IAM et y associer votre politique IAM

1. Exécutez la commande create-role. Dans l'exemple suivant, remplacez-le *iam_role_name* par le nom de votre rôle IAM.

   Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
    --role-name iam_role_name \
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "rds.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

Dans Windows :

```
aws iam create-role ^
    --role-name iam_role_name ^
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
            "Service": "rds.amazonaws.com"
          },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

2. Une fois le rôle créé, notez l'ARN de ce rôle. Vous avez besoin de cet ARN pour l'étape suivante,Étape 4 : Configuration d'un groupe d'options pour la journalisation des audits DB2.
3. Exécutez la commande attach-role-policy. Dans l'exemple suivant, remplacez *iam_policy_arn* par l'ARN de la politique IAM que vous avez créée dansÉtape 2 : Création d'une politique IAM. *iam_role_name*Remplacez-le par le nom du rôle IAM que vous venez de créer.

   Pour LinuxmacOS, ou Unix :

```
aws iam attach-role-policy \
```

```
    --policy-arn iam_policy_arn \
    --role-name iam_role_name
```

Dans Windows :

```
aws iam attach-role-policy ^
    --policy-arn iam_policy_arn ^
    --role-name iam_role_name
```

Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

Étape 4 : Configuration d'un groupe d'options pour la journalisation des audits DB2

Le processus d'ajout de l'option de journalisation d'audit Db2 à une instance de base de données RDS pour DB2 est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajouter et configurer toutes les options requises.
3. Associez le groupe d'options à l'instance de base de données.

Une fois que vous avez ajouté l'option de journalisation d'audit Db2, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, vous pouvez créer des audits et stocker les journaux d'audit dans votre compartiment S3.

Pour ajouter et configurer la journalisation d'audit DB2 sur le groupe d'options d'une instance de base de données

1. Choisissez l'une des méthodes suivantes :
   - Utiliser un groupe d'options existant.
   - Créez un groupe d'options de base de données personnalisé et utilisez-le. Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'option DB2_AUDIT au groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
   - Pour IAM_ROLE_ARN, entrez l'ARN du rôle IAM que vous avez créé dans. the section called "Créez un rôle IAM et associez votre politique IAM"

- Pour S3_BUCKET_ARN, entrez l'ARN du compartiment S3 à utiliser pour vos journaux d'audit DB2. Le bucket doit se trouver dans la même région que votre instance de base de données RDS pour DB2. La politique associée au rôle IAM que vous avez saisi doit autoriser les opérations requises sur cette ressource.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante. Choisissez l'une des méthodes suivantes :

   - Si vous créez une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance.
   - Sur une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Étape 5 : Configuration de la politique d'audit

Pour configurer la politique d'audit de votre base de données RDS pour DB2, connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour DB2. Appelez ensuite la procédure `rdsadmin.configure_db_audit` stockée avec le nom de base de données de votre base de données et les valeurs de paramètres applicables.

L'exemple suivant se connecte à la base de données et configure une politique d'audit pour les `testdb` catégories AUDIT, CHECKING, OBJMAINT, SECMAINT, SYSADMIN et VALIDATE. La valeur d'état `BOTH` enregistre les réussites et les échecs, et `ERROR TYPE` c'est le cas `NORMAL` par défaut. Pour plus d'informations sur l'utilisation de cette procédure stockée, consultezthe section called “rdsadmin.configure_db_audit”.

```
db2 "connect to rdsadmin user master_user using master_password"
db2 "call rdsadmin.configure_db_audit('testdb', 'ALL', 'BOTH', ?)"
```

Étape 6 : Vérifiez la configuration de l'audit

Pour vous assurer que votre politique d'audit est correctement configurée, vérifiez l'état de votre configuration d'audit.

Pour vérifier la configuration, connectez-vous à la `rdsadmin` base de données à l'aide du nom d'utilisateur principal et du mot de passe principal de votre instance de base de données RDS pour

DB2. Exécutez ensuite l'instruction SQL suivante avec le nom de base de données de votre base de données. Dans l'exemple suivant, le nom de la base de données est*testdb*.

```
db2 "select task_id, task_type, database_name, lifecycle,
    varchar(bson_to_json(task_input_params), 500) as task_params,
    cast(task_output as varchar(500)) as task_output
    from table(rdsadmin.get_task_status(null,'testdb','CONFIGURE_DB_AUDIT'))"

Sample Output

TASK_ID              TASK_TYPE            DATABASE_NAME       LIFECYCLE
-------------------- -------------------- --------------- -------------
                   2  CONFIGURE_DB_AUDIT            DB2DB       SUCCESS

... continued ...
TASK_PARAMS
------------------------------------------------------------
{ "AUDIT_CATEGORY" : "ALL", "CATEGORY_SETTING" : "BOTH" }

... continued ...
                                          TASK_OUTPUT
-------------------------------------------------------
2023-12-22T20:27:03.029Z Task execution has started.

2023-12-22T20:27:04.285Z Task execution has completed successfully.
```

Gestion de la journalisation des audits DB2

Après avoir configuré la journalisation d'audit DB2, vous pouvez modifier la politique d'audit pour une base de données spécifique ou désactiver la journalisation d'audit au niveau de la base de données ou pour l'ensemble de l'instance de base de données. Vous pouvez également modifier le compartiment Amazon S3 dans lequel vos fichiers journaux sont chargés.

Rubriques

- Modification d'une politique d'audit DB2
- Modification de l'emplacement de vos fichiers journaux
- Désactivation de la journalisation des audits DB2

Modification d'une politique d'audit DB2

Pour modifier la politique d'audit d'une base de données RDS pour DB2 spécifique, exécutez la procédure `rdsadmin.configure_db_audit` stockée. Avec cette procédure stockée, vous pouvez modifier les catégories, les paramètres des catégories et la configuration du type d'erreur de la politique d'audit. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.configure_db_audit”.

Modification de l'emplacement de vos fichiers journaux

Pour modifier le compartiment Amazon S3 dans lequel vos fichiers journaux sont chargés, effectuez l'une des opérations suivantes :

- Modifiez le groupe d'options actuel attaché à votre instance de base de données RDS pour DB2 : mettez à jour le `S3_BUCKET_ARN` paramètre de l'`DB2_AUDIT`option afin qu'elle pointe vers le nouveau compartiment. Assurez-vous également de mettre à jour la politique IAM attachée au rôle IAM spécifié par le `IAM_ROLE_ARN` paramètre du groupe d'options attaché. Cette politique IAM doit fournir à votre nouveau compartiment les autorisations d'accès requises. Pour plus d'informations sur les autorisations requises dans la politique IAM, consultezCréer une politique IAM.
- Attachez votre instance de base de données RDS pour DB2 à un autre groupe d'options : modifiez votre instance de base de données pour changer le groupe d'options qui y est attaché. Assurez-vous que le nouveau groupe d'options est configuré avec `S3_BUCKET_ARN` les bons `IAM_ROLE_ARN` paramètres. Pour plus d'informations sur la configuration de ces paramètres pour l'`DB2_AUDIT`option, consultezConfiguration d'un groupe d'options.

Lorsque vous modifiez le groupe d'options, assurez-vous d'appliquer les modifications immédiatement. Pour de plus amples informations, veuillez consulter the section called “Modification d'une instance de base de données”.

Désactivation de la journalisation des audits DB2

Pour désactiver la journalisation des audits DB2, effectuez l'une des opérations suivantes :

- Désactivez la journalisation des audits pour l'instance de base de données RDS pour DB2 : modifiez votre instance de base de données et supprimez le groupe d'options contenant l'`DB2_AUDIT`option. Pour de plus amples informations, veuillez consulter the section called “Modification d'une instance de base de données”.
- Désactiver la journalisation des audits pour une base de données spécifique : arrêtez la journalisation des audits et supprimez la politique d'audit en appelant

`rdsadmin.disable_db_audit` avec le nom de base de données de votre base de données. Pour de plus amples informations, veuillez consulter the section called “rdsadmin.disable_db_audit”.

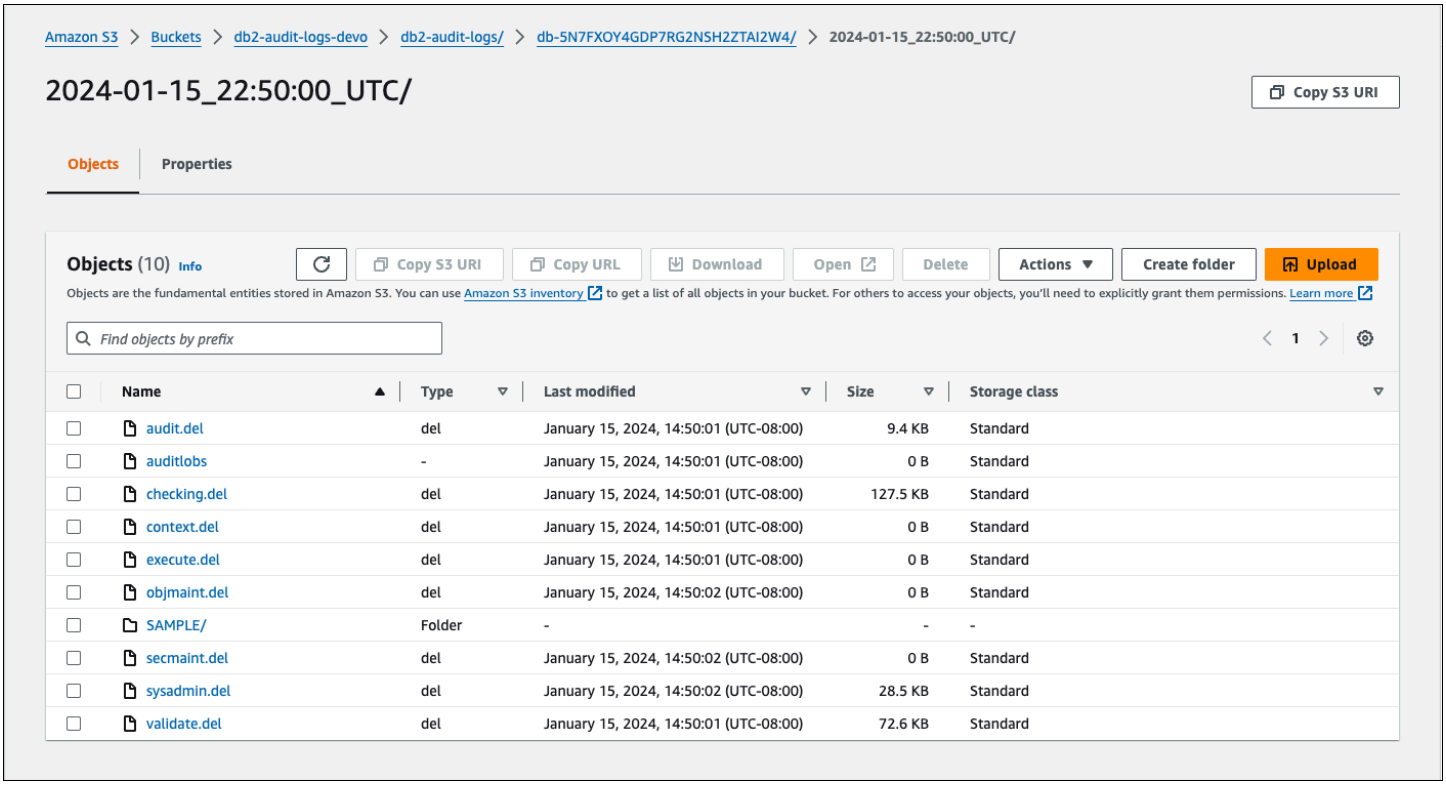
```
db2 "call rdsadmin.disable_db_audit(
    'db_name',
    ?)"
```

Consultation des journaux d'audit

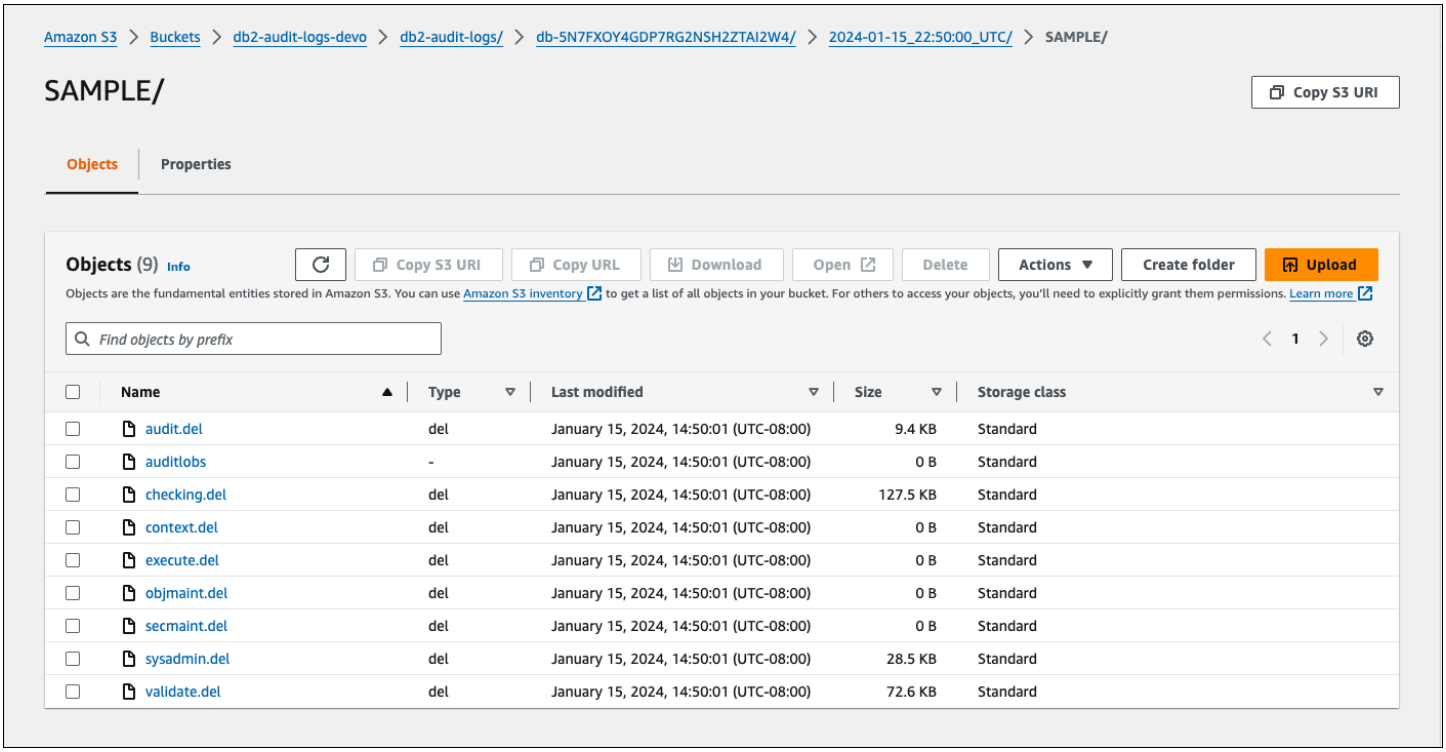
Après avoir activé la journalisation des audits DB2, attendez au moins une heure avant de consulter les données d'audit dans votre compartiment Amazon S3. Amazon RDS envoie automatiquement les journaux de votre instance de base de données RDS pour DB2 aux emplacements suivants :

- Journaux au niveau des instances de base de données — *bucket_name*/db2-audit-logs/*dbi_resource_id*/*date_time_utc*/
- Journaux au niveau des bases de données : *bucket_name*/db2-audit-logs/*dbi_resource_id*/*date_time_utc*/*db_name*/

L'exemple de capture d'écran suivant de la console Amazon S3 montre une liste de dossiers pour les fichiers journaux au niveau de l'instance de base de données RDS for DB2.



L'exemple de capture d'écran suivant de la console Amazon S3 montre les fichiers journaux au niveau de la base de données pour l'instance de base de données RDS pour DB2.



Résolution des problèmes liés à l'enregistrement des audits DB2

Utilisez les informations suivantes pour résoudre les problèmes courants liés à la journalisation des audits DB2.

Impossible de configurer la politique d'audit

Si l'appel de la procédure stockée `rdsadmin.configure_db_audit` renvoie une erreur, il se peut que le groupe d'options contenant l'`DB2_AUDIT`option ne soit pas associé à l'instance de base de données RDS pour DB2. Modifiez l'instance de base de données pour ajouter le groupe d'options, puis réessayez d'appeler la procédure stockée. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Aucune donnée dans le compartiment Amazon S3

Si les données de journalisation sont absentes du compartiment Amazon S3, vérifiez les points suivants :

- Le compartiment Amazon S3 se trouve dans la même région que votre instance de base de données RDS pour DB2.
- Le rôle que vous avez spécifié dans le paramètre d'`IAM_ROLE_ARN`option est configuré avec les autorisations requises pour télécharger des journaux dans votre compartiment Amazon S3. Pour de plus amples informations, veuillez consulter Créer une politique IAM.
- Les paramètres ARNs pour `IAM_ROLE_ARN` et `S3_BUCKET_ARN` option sont corrects dans le groupe d'options associé à votre instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter Configuration d'un groupe d'options.

Vous pouvez vérifier l'état des tâches de votre configuration de journalisation d'audit en vous connectant à la base de données et en exécutant une instruction SQL. Pour de plus amples informations, veuillez consulter Vérifiez la configuration de l'audit.

Vous pouvez également consulter les événements pour en savoir plus sur les raisons pour lesquelles les journaux peuvent être absents. Pour plus d'informations sur la façon d'afficher les événements, consultezthe section called “Afficher les journaux, les événements et les flux dans la RDS console Amazon”.

Procédures stockées externes pour Amazon RDS pour DB2

Vous pouvez créer des routines externes et les enregistrer dans vos bases de données Amazon RDS pour DB2 en tant que procédures stockées externes. Actuellement, RDS pour Db2 ne prend en charge que les routines basées sur Java pour les procédures stockées externes.

Procédures stockées externes basées sur Java

Les procédures stockées externes basées sur Java sont des routines Java externes que vous enregistrez dans votre base de données RDS pour DB2 en tant que procédures stockées externes.

Rubriques

- Limitations des procédures stockées externes basées sur Java
- Configuration de procédures stockées externes basées sur Java

Limitations des procédures stockées externes basées sur Java

Avant de développer votre routine externe, tenez compte des limites et restrictions suivantes.

Pour créer votre routine externe, assurez-vous d'utiliser le kit de développement Java (JDK) fourni par Db2. Pour plus d'informations, consultez la section Support logiciel Java pour les produits de base de données DB2.

Votre programme Java peut créer des fichiers uniquement dans le `/tmp` répertoire, et Amazon RDS ne prend pas en charge l'activation des autorisations d'exécution ou de définition d'un identifiant utilisateur (SUID) sur ces fichiers. Votre programme Java ne peut pas non plus utiliser les appels système socket ou les appels système suivants :

- _sysctl
- acct
- afs_syscall
- bpf
- capset
- chown
- chroot
- create_module

- delete_module
- fanotify_init
- fanotify_mark
- finit_module
- fsconfig
- fsopen
- fspick
- get_kernel_syms
- getpmsg
- init_module
- mount
- move_mount
- nfsservctl
- open_by_handle_at
- open_tree
- pivot_root
- putpmsg
- query_module
- quotactl
- reboot
- security
- setdomainname
- setfsuid
- sethostname
- sysfs
- tuxcall
- umount2
- uselib
- ustat

- vhangup
- vserver

Pour des restrictions supplémentaires sur les routines externes pour DB2, consultez la section Restrictions sur les routines externes dans la IBM Db2 documentation.

Configuration de procédures stockées externes basées sur Java

Pour configurer une procédure stockée externe, créez un fichier .jar avec votre routine externe, installez-le sur votre base de données RDS pour DB2, puis enregistrez-le en tant que procédure stockée externe.

Rubriques

- Étape 1 : activer les procédures stockées externes
- Étape 2 : installez le fichier .jar avec votre routine externe
- Étape 3 : enregistrement de la procédure stockée externe
- Étape 4 : Valider la procédure stockée externe

Étape 1 : activer les procédures stockées externes

Pour activer les procédures stockées externes, dans un groupe de paramètres personnalisé associé à votre instance de base de données, définissez le paramètre `db2_alternate_authz_behaviour` sur l'une des valeurs suivantes :

- `EXTERNAL_ROUTINE_DBADM`— Accorde implicitement l'`CREATE_EXTERNAL_ROUTINE`autorisation à tout utilisateur, groupe ou rôle ayant `DBADM` autorité.
- `EXTERNAL_ROUTINE_DBAUTH`— Permet à un utilisateur `DBADM` autorisé d'accorder des `CREATE_EXTERNAL_ROUTINE` autorisations à n'importe quel utilisateur, groupe ou rôle. Dans ce cas, aucun utilisateur, groupe ou rôle ne reçoit implicitement cette autorisation, pas même un utilisateur `DBADM` autorisé.

Pour plus d'informations sur ce paramètre, consultez l'instruction GRANT (autorités de base de données) dans la IBM Db2 documentation.

Vous pouvez créer et modifier un groupe de paramètres personnalisé à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API Amazon RDS.

console

Pour configurer le paramètre db2_alternate_authz_behaviour dans un groupe de paramètres personnalisé

1. Si vous souhaitez utiliser un groupe de paramètres de base de données personnalisé différent de celui utilisé par votre instance de base de données, créez un nouveau groupe de paramètres de base de données. Si vous utilisez le modèle BYOL (Bring your own license), assurez-vous que le nouveau groupe de paramètres personnalisés inclut le IBM IDs. Pour plus d'informations à ce sujet IDs, consultezthe section called “IBM IDs pour apporter votre propre licence (BYOL) pour Db2”. Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Définissez la valeur du `db2_alternate_authz_behaviour` paramètre dans votre groupe de paramètres personnalisé. Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

AWS CLI

Pour configurer le paramètre db2_alternate_authz_behaviour dans un groupe de paramètres personnalisé

1. Si vous souhaitez utiliser un groupe de paramètres de base de données personnalisé différent de celui utilisé par votre instance de base de données, créez un groupe de paramètres personnalisé en exécutant la create-db-parameter-groupcommande. Si vous utilisez le modèle BYOL (Bring your own license), assurez-vous que le nouveau groupe de paramètres personnalisés inclut le IBM IDs. Pour plus d'informations à ce sujet IDs, consultezthe section called “IBM IDs pour apporter votre propre licence (BYOL) pour Db2”.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Nom du groupe de paramètres que vous créez.
   - `--db-parameter-group-family`— L'édition et la version majeure du moteur DB2. Les valeurs valides sont `db2-se-11.5` et `db2-ae-11.5`.
   - `--description`— Description de ce groupe de paramètres.

Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.

L'exemple suivant montre comment créer un groupe de paramètres personnalisé nommé d'après `MY_EXT_SP_PARAM_GROUP` la famille de groupes de paramètres`db2-se-11.5`.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-parameter-group \
--region us-east-1 \
--db-parameter-group-name MY_EXT_SP_PARAM_GROUP \
--db-parameter-group-family db2-se-11.5 \
--description "test db2 external routines"
```

Dans Windows :

```
aws rds create-db-parameter-group ^
--region us-east-1 ^
--db-parameter-group-name MY_EXT_SP_PARAM_GROUP ^
--db-parameter-group-family db2-se-11.5 ^
--description "test db2 external routines"
```

2. Modifiez le `db2_alternate_authz_behaviour` paramètre dans votre groupe de paramètres personnalisé en exécutant la modify-db-parameter-groupcommande.

   Inclure les options requises suivantes :

   - `--db-parameter-group-name`— Le nom du groupe de paramètres que vous avez créé.
   - `--parameters`— Tableau de noms de paramètres, de valeurs et de méthodes d'application pour la mise à jour des paramètres.

   Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

   L'exemple suivant montre comment modifier le groupe de paramètres `MY_EXT_SP_PARAM_GROUP` en définissant la valeur de `db2_alternate_authz_behaviour` to`EXTERNAL_ROUTINE_DBADM`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name MY_EXT_SP_PARAM_GROUP \
    --parameters
 "ParameterName='db2_alternate_authz_behaviour',ParameterValue='EXTERNAL_ROUTINE_DBADM',App
```

Dans Windows :

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name MY_EXT_SP_PARAM_GROUP ^
    --parameters
 "ParameterName='db2_alternate_authz_behaviour',ParameterValue='EXTERNAL_ROUTINE_DBADM',App
```

API RDS

Pour configurer le paramètre db2_alternate_authz_behaviour dans un groupe de paramètres personnalisé

1. Si vous souhaitez utiliser un groupe de paramètres de base de données personnalisé différent de celui utilisé par votre instance de base de données, créez un nouveau groupe de paramètres de base de données à l'aide de l'CreateDBParameterGroupopération d'API Amazon RDS. Si vous utilisez le modèle BYOL (Bring your own license), assurez-vous que le nouveau groupe de paramètres personnalisés inclut le IBM Db2 IDs. Pour plus d'informations à ce sujet IDs, consultezthe section called “IBM IDs pour apporter votre propre licence (BYOL) pour Db2”.

   Incluez les paramètres requis suivants :

   - `DBParameterGroupName`
   - `DBParameterGroupFamily`
   - `Description`

   Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon.
2. Modifiez le `db2_alternate_authz_behaviour` paramètre dans votre groupe de paramètres personnalisé que vous avez créé à l'aide de l'ModifyDBParameterGroupopération d'API RDS.

Incluez les paramètres requis suivants :

- `DBParameterGroupName`
- `Parameters`

Pour plus d'informations sur la modification d'un groupe de paramètres, consultezModification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Étape 2 : installez le fichier .jar avec votre routine externe

Après avoir créé votre routine Java, créez le fichier .jar, puis exécutez-le `db2 "call sqlj.install_jar('file:file_path',jar_ID)"` pour l'installer sur votre base de données RDS pour DB2.

L'exemple suivant montre comment créer une routine Java et l'installer sur une base de données RDS pour DB2. L'exemple inclut un exemple de code pour une routine simple que vous pouvez utiliser pour tester le processus. Cet exemple repose sur les hypothèses suivantes :

- Le code Java est compilé sur un serveur sur lequel Db2 est installé. Il s'agit d'une bonne pratique, car le fait de ne pas compiler avec le JDK fourni par IBM peut entraîner des erreurs inexpliquées.
- Le serveur dispose de la base de données RDS pour DB2 cataloguée localement.

Si vous souhaitez essayer le processus avec l'exemple de code suivant, copiez-le puis enregistrez-le dans un fichier nomméMYJAVASP.java.

```
import java.sql.*;
public class MYJAVASP
{
public static void my_JAVASP (String inparam) throws SQLException, Exception
{
try
{
// Obtain the calling context's connection details.
Connection myConn = DriverManager.getConnection("jdbc:default:connection");
String myQuery = "INSERT INTO TEST.TEST_TABLE VALUES (?, CURRENT DATE)";
PreparedStatement myStmt = myConn.prepareStatement(myQuery);
myStmt.setString(1, inparam);
myStmt.executeUpdate();
```

```
}
catch (SQLException sql_ex)
{
throw sql_ex;
}
catch (Exception ex)
{
throw ex;
}
}
```

La commande suivante compile la routine Java.

```
~/sqllib/java/jdk64/bin/javac MYJAVASP.java
```

La commande suivante crée le fichier .jar.

```
~/sqllib/java/jdk64/bin/jar cvf MYJAVASP.jar MYJAVASP.class
```

Les commandes suivantes se connectent à la base de données nommée `MY_DB2_DATABASE` et installent le fichier .jar.

```
db2 "connect to MY_DB2_DATABASE user master_username using master_password"

db2 "call sqlj.install_jar('file:/tmp/MYJAVASP.jar','MYJAVASP')"
db2 "call sqlj.refresh_classes()"
```

Étape 3 : enregistrement de la procédure stockée externe

Après avoir installé le fichier .jar sur votre base de données RDS pour DB2, enregistrez-le en tant que procédure stockée en exécutant la `db2 CREATE PROCEDURE` commande or. `db2 REPLACE PROCEDURE`

L'exemple suivant montre comment se connecter à la base de données et enregistrer la routine Java créée à l'étape précédente en tant que procédure stockée.

```
db2 "connect to MY_DB2_DATABASE user master_username using master_password"

create procedure TESTSP.MYJAVASP (in input char(6))
specific myjavasp
```

```
dynamic result sets 0
deterministic
language java
parameter style java
no dbinfo
fenced
threadsafe
modifies sql data
program type sub
external name 'MYJAVASP!my_JAVASP';
```

Étape 4 : Valider la procédure stockée externe

Procédez comme suit pour tester l'exemple de procédure stockée externe enregistré à l'étape précédente.

Pour valider la procédure stockée externe

1. Créez un tableau comme `TEST.TEST_TABLE` dans l'exemple suivant.

   ```
   db2 "create table TEST.TEST_TABLE(C1 char(6), C2 date)"
   ```

2. Appelez la nouvelle procédure stockée externe. L'appel renvoie un statut de`0`.

   ```
   db2 "call TESTSP.MYJAVASP('test')"
   Return Status = 0
   ```

3. Interrogez la table que vous avez créée à l'étape 1 pour vérifier les résultats de l'appel de procédure stockée.

   ```
   db2 "SELECT * from TEST.TEST_TABLE"
   ```

   La requête produit une sortie similaire à l'exemple suivant :

   ```
   C1     C2
   ------ ----------
   test   02/05/2024
   ```

Problèmes connus et limites d'Amazon RDS pour DB2

Les problèmes et limites connus liés à l'utilisation d'Amazon RDS pour DB2 sont les suivants :

Rubriques

- Limitation de l'authentification
- Routines non clôturées
- Tablespaces de stockage non automatiques pendant la migration
- Définition du paramètre db2_compatibility_vector
- Migration de bases de données contenant des packages INVALIDES

Limitation de l'authentification

Amazon RDS est défini `db2auth` sur `JCC_ENFORCE_SECMEC` par défaut. Toutefois, si vous ne souhaitez pas appliquer le chiffrement de l'identifiant utilisateur et du mot de passe par câble, vous pouvez annuler ce paramètre en le remplaçant par un `db2auth` paramètre `CLEAR_TEXT` dans le groupe de paramètres. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Routines non clôturées

RDS pour DB2 ne prend pas en charge la création de routines non clôturées ni la migration de ces routines en sauvegardant et en restaurant des données. Pour vérifier si votre base de données contient des routines non clôturées, exécutez la commande SQL suivante :

```
SELECT 'COUNT:' || count(*) FROM SYSCAT.ROUTINES where fenced='N' and routineschema not
 in ('SQLJ','SYSCAT','SYSFUN','SYSIBM','SYSIBMADM','SYSPROC','SYSTOOLS')
```

Tablespaces de stockage non automatiques pendant la migration

RDS pour DB2 ne prend pas en charge la création de nouveaux tablespaces de stockage non automatiques. Lorsque vous utilisez la restauration native pour une migration unique de votre base de données, RDS pour DB2 convertit automatiquement vos tablespaces de stockage non automatiques en espaces automatiques, puis restaure votre base de données sur RDS pour DB2. Pour plus d'informations sur les migrations ponctuelles, reportez-vous Migration à partir de Linux to

Linux pour Amazon RDS pour Db2 aux sections etMigration à partir de AIX or Windows to Linux pour Amazon RDS pour Db2.

Définition du paramètre db2_compatibility_vector

Avec Amazon RDS, vous pouvez créer une base de données initiale lorsque vous créez l'instance de base de données, puis que vous modifiez les paramètres dans un groupe de paramètres associé. Toutefois, pour Db2, si vous souhaitez définir le `db2_compatibility_vector` paramètre dans un groupe de paramètres, vous devez d'abord modifier le paramètre dans un groupe de paramètres personnalisé, créer l'instance de base de données sans base de données, puis créer une base de données à l'aide de la procédure `rdsadmin.create_database` stockée.

Pour définir le **`db2_compatibility_vector`** paramètre

1. Créez un groupe de paramètres personnalisé. (Vous ne pouvez pas modifier les paramètres des groupes de paramètres par défaut.)
2. Modifiez le paramètre.
3. Créez une instance de base de données.
4. Créez une base de données à l'aide de la procédure `rdsadmin.create_database` stockée.
5. Associez le groupe de paramètres à l'instance de base de données qui contient la base de données.

Migration de bases de données contenant des packages INVALIDES

Si vous migrez des bases de données DB2 contenant des `INVALID` packages vers RDS pour DB2 à l'aide de la `RESTORE` commande, vous risquez de rencontrer des problèmes lorsque vous commencez à utiliser les bases de données. `INVALID`les packages peuvent provoquer des problèmes en raison de la configuration des autorisations pour l'utilisateur de l'instance de base de données `rdsdb` et de la suppression de l'autorisation de`PUBLIC`. `INVALID`les packages entraînent l'échec des commandes suivantes :

- `db2updv115`
- `db2 "call SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS()"`

Avant de migrer votre base de données à l'aide de la `RESTORE` commande, assurez-vous qu'elle ne contient pas de `INVALID` packages en exécutant la commande suivante :

```
db2 "SELECT 'COUNT:' || count(*) FROM SYSCAT.INVALIDOBJECTS"
```

Si la commande renvoie un nombre supérieur à zéro, appelez la commande suivante :

```
db2 "call SYSPROC.ADMIN_REVALIDATE_DB_OBJECTS()"
```

Appelez ensuite la commande précédente pour confirmer que votre base de données ne contient plus de `INVALID` packages.

```
db2 "SELECT 'COUNT:' || count(*) FROM SYSCAT.INVALIDOBJECTS"
```

Vous êtes maintenant prêt à effectuer une sauvegarde de votre base de données et à la restaurer sur votre instance de base de données RDS pour DB2.

Référence de procédure stockée Amazon RDS pour DB2

Vous pouvez gérer vos instances de base de données Amazon RDS for Db2 exécutant le moteur Db2 en appelant des procédures stockées intégrées.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.activate_database”	Bases de données	Utilisez la procédure `rdsadmin.activate_database` stockée pour activer une base de données sur une instance de base de données RDS autonome pour DB2.
the section called “rdsadmin.add_groups”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.add_groups` stockée pour ajouter un ou plusieurs groupes à un utilisateur pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.add_user”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.add_user` stockée pour ajouter un utilisateur à une liste d'autorisations pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.alter_bufferpool”	Piscines tampons	Utilisez la procédure `rdsadmin.alter_bufferpool` stockée pour modifier un pool de mémoire tampon pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.alter_tablespace”	Espaces de table	Utilisez la procédure `rdsadmin.alter_tablespace` stockée pour modifier un tablespace pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.catalog_storage_access”	Accès au stockage	Utilisez la procédure `rdsadmin.catalog_storage_access` stockée pour catalogue r un alias de stockage permettant d'accéder à

Procédure stockée	Catégorie	Description
		un compartiment Amazon S3 contenant des fichiers de données DB2 pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.change_password”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.change_password` stockée pour modifier le mot de passe d'un utilisateur pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.complete_rollforward”	Bases de données	Utilisez la procédure `rdsadmin.complete_rollforward` stockée pour mettre en ligne une base de données sur une instance de base de données RDS pour DB2 à partir d'un `ROLL-FORWARD PENDING` état. Un `ROLL-FORWARD PENDING` état se produit lorsque vous avez appelé the section called “rdsadmin.rollforward_database” mais que vous avez défini le `complete_rollforward` paramètre sur`FALSE`.
the section called “rdsadmin.configure_db_audit”	Politiques d'audit	Utilisez la procédure `rdsadmin.configure_db_audit` stockée pour modifier une politique d'audit pour une base de données sur une instance de base de données RDS pour DB2. S'il n'existe aucune stratégie d'audit, l'exécution de cette procédure stockée crée une stratégie d'audit.
the section called “rdsadmin.create_bufferpool”	Piscines tampons	Utilisez la procédure `rdsadmin.create_bufferpool` stockée pour créer un pool de mémoire tampon pour une base de données sur une instance de base de données RDS pour DB2.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.create_database”	Bases de données	Utilisez la procédure `rdsadmin.create_database` stockée pour créer une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.create_role”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.create_role` stockée pour créer un rôle à associer à une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.create_tablespace”	Espaces de table	Utilisez la procédure `rdsadmin.create_tablespace` stockée pour créer un tablespace pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.db2pd_command”	Bases de données	Utilisez la procédure `rdsadmin.db2pd_command` stockée pour collecter des informations sur une base de données sur une instance de base de données RDS pour DB2. Ces informations peuvent aider à surveiller et à dépanner les bases de données dans RDS for Db2.
the section called “rdsadmin.dbadm_grant”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.dbadm_grant` stockée pour accorder un ou plusieurs types d'autorisation (`DBADM`,`ACCESSCTRL` , ou`DATAACCESS`) à un ou plusieurs rôles, utilisateurs ou groupes pour une base de données sur une instance de base de données RDS pour DB2.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.dbadm_revoke”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.dbadm_revoke` stockée pour révoquer un ou plusieurs types d'autorisation (`DBADMACCESSCTRL` , ou`DATAACCESS`) d'un ou plusieurs rôles, utilisateurs ou groupes pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.deactivate_database”	Bases de données	Utilisez la procédure `rdsadmin.deactivate_database` stockée pour désactiver une base de données sur une instance de base de données RDS pour DB2. Vous pouvez désactiver les bases de données pour économiser les ressources de mémoire.
the section called “rdsadmin.disable_db_audit”	Politiques d'audit	Utilisez la procédure `rdsadmin.disable_db_audit` stockée pour arrêter la journalisation des audits et supprimer une politique d'audit d'une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.drop_bufferpool”	Piscines tampons	Utilisez la procédure `rdsadmin.drop_bufferpool` stockée pour supprimer un pool de mémoire tampon d'une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.drop_database”	Bases de données	Utilisez la procédure `rdsadmin.drop_database` stockée pour supprimer une base de données d'une instance de base de données RDS pour DB2.
the section called “rdsadmin.drop_role”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.drop_role` stockée pour supprimer un rôle d'une base de données sur une instance de base de données RDS pour DB2.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.drop_tablespace”	Espaces de table	Utilisez la procédure `rdsadmin.drop_tablespace` stockée pour supprimer un tablespace d'une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.force_application”	Bases de données	Utilisez la procédure `rdsadmin.force_application` stockée pour forcer les applications à quitter une base de données sur une instance de base de données RDS for DB2 afin d'effectuer la maintenance.
the section called “rdsadmin.grant_role”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.grant_role` stockée pour attribuer un rôle à un rôle, à un utilisateur ou à un groupe bénéficiaire dans une base de données sur une instance de base de données RDS pour DB2. Vous pouvez également utiliser cette procédure stockée pour donner au bénéficiaire l'`DBADM`autorisation d'attribuer des rôles.
the section called “rdsadmin.list_archive_log_information”	Bases de données	Utilisez la procédure `rdsadmin.list_archive_log_information` stockée pour renvoyer des informations sur les journaux d'archivage d'une base de données sur une instance de base de données RDS pour DB2. Ces informations incluent des détails tels que la taille et la date de création des fichiers journaux individuels, ainsi que le stockage total utilisé par les fichiers journaux d'archive.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.list_sid_group_mapping”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.list_sid_group_mapping` stockée pour renvoyer une liste de tous les mappages d'ID de sécurité (SID) et de groupes Active Directory configurés sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.list_users”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.list_users` stockée pour renvoyer une liste d'utilisateurs figurant sur une liste d'autorisations pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.reactivate_database”	Bases de données	Utilisez la procédure `rdsadmin.reactivate_database` stockée pour réactiver une base de données sur une instance de base de données RDS pour DB2 après avoir modifié la configuration de la base de données. Pour une base de données sur une instance de base de données autonome, vous pouvez utiliser cette procédure stockée ou la procédure rdsadmin.activate_database stockée.
the section called “rdsadmin.remove_groups”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.remove_groups` stockée pour supprimer un ou plusieurs groupes d'un utilisateur pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.remove_sid_group_mapping”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.remove_sid_group_mapping` stockée pour supprimer un ID de sécurité (SID) et le mappage de groupe Active Directory correspondant d'une instance de base de données RDS pour DB2.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.remove_user”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.remove_user` stockée pour supprimer un utilisateur d'une liste d'autorisations pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.rename_tablespace”	Espaces de table	Utilisez la procédure `rdsadmin.rename_tablespace` stockée pour renommer un tablespace pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.restore_database”	Bases de données	Utilisez la procédure `rdsadmin.restore_database` stockée pour restaurer une base de données sur une instance de base de données RDS pour DB2 à partir d'un compartiment Amazon S3.
the section called “rdsadmin.revoke_role”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.revoke_role` stockée pour révoquer un rôle attribué à un rôle, à un utilisateur ou à un groupe bénéficiaire dans une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.rollforward_database”	Bases de données	Utilisez la procédure `rdsadmin.rollforward_database` stockée pour mettre en ligne une base de données sur une instance de base de données RDS pour DB2 et pour appliquer des journaux de transactions après avoir restauré une base de données sur une instance de base de données RDS pour DB2 en appelant. the section called “rdsadmin.restore_database”

Procédure stockée	Catégorie	Description
the section called “rdsadmin.rollforward_status”	Bases de données	Utilisez la procédure `rdsadmin.rollforward_status` stockée pour demander l'état d'avancement de l'appel de la procédure the section called “rdsadmin.complete_rollforward” stockée the section called “rdsadmin.rollforward_database” ou sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.set_archive_log_retention”	Bases de données	Utilisez la procédure `rdsadmin.set_archive_log_retention` stockée pour configurer la durée de conservation des fichiers journaux d'archivage pour une base de données sur une instance de base de données RDS pour DB2. Vous pouvez également utiliser cette procédure stockée pour désactiver la conservation des journaux d'archivage.
the section called “rdsadmin.set_configuration”	Bases de données	Utilisez la procédure `rdsadmin.set_configuration` stockée pour configurer certains paramètres d'une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.set_sid_group_mapping”	Octroi et révocation de privilèges	Utilisez la procédure `rdsadmin.set_sid_group_mapping` stockée pour créer un mappage entre un ID de sécurité (SID) et le groupe Active Directory correspondant sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.show_archive_log_retention”	Bases de données	Utilisez la procédure `rdsadmin.show_archive_log_retention` stockée pour renvoyer le paramètre actuel de conservation du journal d'archivage pour une base de données sur une instance de base de données RDS pour DB2.

Procédure stockée	Catégorie	Description
the section called “rdsadmin.show_configuration”	Bases de données	Utilisez la procédure `rdsadmin.show_configuration` stockée pour renvoyer un ou plusieurs paramètres modifiables pour une base de données sur une instance de base de données RDS pour DB2.
the section called “rdsadmin.uncatalog_storage_access”	Accès au stockage	Utilisez la procédure `rdsadmin.uncatalog_storage_access` stockée pour supprimer un alias de stockage permettant d'accéder à un compartiment Amazon S3 contenant des fichiers de données DB2.
the section called “rdsadmin.update_db_param”	Bases de données	Utilisez la procédure `rdsadmin.update_db_param` stockée pour mettre à jour les paramètres de base de données d'une base de données sur une instance de base de données RDS pour DB2.

Rubriques

- Considérations relatives aux procédures stockées Amazon RDS pour DB2
- Procédures stockées pour l'octroi et la révocation de privilèges pour RDS pour Db2
- Procédures stockées pour les politiques d'audit pour RDS pour Db2
- Procédures stockées pour les pools de mémoire tampon RDS pour Db2
- Procédures stockées pour les bases de données pour RDS pour Db2
- Procédures stockées pour l'accès au stockage RDS pour Db2
- Procédures stockées pour les espaces de table pour RDS pour Db2

Considérations relatives aux procédures stockées Amazon RDS pour DB2

Avant d'utiliser les procédures stockées du système Amazon RDS pour les instances de base de données RDS pour DB2 exécutant le moteur Db2, consultez les informations suivantes :

- Avant d'exécuter les procédures stockées, vous devez d'abord vous connecter à la `rdsadmin` base de données en tant qu'utilisateur principal de votre instance de base de données RDS pour DB2. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par vos propres informations.

  ```
  db2 "connect to rdsadmin user master_username using master_password"
  ```

- Les procédures stockées renvoient le `ERR_MESSAGE` paramètre, qui indique si la procédure stockée s'est exécutée correctement ou non et pourquoi elle n'a pas été exécutée correctement.

  Exemples

  L'exemple suivant indique que la procédure stockée s'est exécutée correctement.

  ```
  Parameter Name : ERR_MESSAGE
  Parameter Value : -
  Return Status = 0
  ```

  L'exemple suivant indique que la procédure stockée n'a pas été exécutée correctement car le nom du compartiment Amazon S3 utilisé dans la procédure stockée n'était pas valide.

  ```
  Parameter Name : ERR_MESSAGE
  Parameter Value : Invalid S3 bucket name
  Return Status = -1006
  ```

Pour les messages d'erreur renvoyés lors de l'appel de procédures stockées, consultezthe section called “Erreurs de procédure stockée”.

Pour plus d'informations sur la vérification de l'état d'une procédure stockée, consultezrdsadmin.get_task_status.

Procédures stockées pour l'octroi et la révocation de privilèges pour RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent les utilisateurs, les rôles, les groupes et les autorisations pour les bases de données Amazon RDS for Db2. Pour exécuter ces procédures, l'utilisateur principal doit d'abord se connecter à la `rdsadmin` base de données.

Pour les tâches utilisant ces procédures stockées, reportez-vous aux sections Octroi et révocation de privilèges etConfiguration de Kerberos l'authentification.

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.create_role
- rdsadmin.grant_role
- rdsadmin.revoke_role
- rdsadmin.drop_role
- rdsadmin.add_user
- rdsadmin.change_password
- rdsadmin.list_users
- rdsadmin.remove_user
- rdsadmin.add_groups
- rdsadmin.remove_groups
- rdsadmin.dbadm_grant
- rdsadmin.dbadm_revoke
- rdsadmin.set_sid_group_mapping
- rdsadmin.list_sid_group_mapping
- rdsadmin.remove_sid_group_mapping

rdsadmin.create_role

Crée un rôle.

Syntaxe

```
db2 "call rdsadmin.create_role(
    'database_name',
    'role_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

role_name

Nom du rôle que vous souhaitez créer. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de création d'un rôle, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant crée un rôle appelé base `MY_ROLE` de donnéesDB2DB.

```
db2 "call rdsadmin.create_role(
    'DB2DB',
    'MY_ROLE')"
```

rdsadmin.grant_role

Attribue un rôle à un rôle, à un utilisateur ou à un groupe.

Syntaxe

```
db2 "call rdsadmin.grant_role(
    ?,
    'database_name',
    'role_name',
    'grantee',
```

```
    'admin_option')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètres qui génère l'identifiant unique de la tâche. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

role_name

Nom du rôle que vous souhaitez créer. Le type de données est`varchar`.

grantee

Rôle, utilisateur ou groupe devant recevoir l'autorisation. Le type de données est`varchar`. Valeurs valides: `ROLE`, `USER`, `GROUP`, `PUBLIC`.

Le format doit être une valeur suivie d'un nom. Séparez les valeurs et les noms multiples par des virgules. Exemple : « `USER` *`user1`*`, `*`user2`*`, GROUP` *`group1`*`, `*`group2`* ». Remplacez les noms par vos propres informations.

Le paramètre d'entrée suivant est facultatif :

admin_option

Spécifie si le bénéficiaire `ROLE` est `DBADM` autorisé à attribuer des rôles. Le type de données est`char`. L'argument par défaut est `N`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de l'attribution d'un rôle, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : attribution d'un rôle à un rôle, à un utilisateur et à un groupe, et octroi d'une autorisation

L'exemple suivant attribue un rôle appelé base `ROLE_TEST` de données `TESTDB` au rôle appelé`role1`, à l'utilisateur appelé `user1` et au groupe appelé`group1`. `ROLE_TEST`reçoit l'autorisation d'administrateur pour attribuer des rôles.

```
db2 "call rdsadmin.grant_role(
    ?,
    'TESTDB',
    'ROLE_TEST',
    'ROLE role1, USER user1, GROUP group1',
    'Y')"
```

Exemple 2 : Attribuer un rôle à **`PUBLIC`** et ne pas accorder d'autorisation

L'exemple suivant attribue un rôle appelé base `ROLE_TEST` de données `TESTDB` à`PUBLIC`. `ROLE_TEST`n'a pas l'autorisation d'administrateur d'attribuer des rôles.

```
db2 "call rdsadmin.grant_role(
    ?,
    'TESTDB',
    'ROLE_TEST',
    'PUBLIC')"
```

rdsadmin.revoke_role

Révoque un rôle d'un rôle, d'un utilisateur ou d'un groupe.

Syntaxe

```
db2 "call rdsadmin.revoke_role(
    ?,
    'database_name',
    'role_name',
    'grantee')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètres qui génère l'identifiant unique de la tâche. Ce paramètre accepte uniquement ?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

role_name

Nom du rôle que vous souhaitez révoquer. Le type de données est`varchar`.

grantee

Le rôle, l'utilisateur ou le groupe à qui l'autorisation doit être perdue. Le type de données est`varchar`. Valeurs valides: `ROLE`, `USER`, `GROUP`, `PUBLIC`.

Le format doit être une valeur suivie d'un nom. Séparez les valeurs et les noms multiples par des virgules. Exemple : « `USER` *`user1`*`, `*`user2`*`, GROUP` *`group1`*`, `*`group2`* ». Remplacez les noms par vos propres informations.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de révocation d'un rôle, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : Révocation du rôle d'un rôle, d'un utilisateur ou d'un groupe

L'exemple suivant révoque un rôle appelé base `ROLE_TEST` de données `TESTDB` du rôle appelé`role1`, de l'utilisateur appelé `user1` et du groupe appelé`group1`.

```
db2 "call rdsadmin.revoke_role(
    ?,
    'TESTDB',
    'ROLE_TEST',
```

```
    'ROLE role1, USER user1, GROUP group1')"
```

Exemple 2 : Révocation du rôle de **`PUBLIC`**

L'exemple suivant révoque un rôle appelé `ROLE_TEST` database `TESTDB` from`PUBLIC`.

```
db2 "call rdsadmin.revoke_role(
    ?,
    'TESTDB',
    'ROLE_TEST',
    'PUBLIC')"
```

rdsadmin.drop_role

Supprime un rôle.

Syntaxe

```
db2 "call rdsadmin.drop_role(
    ?,
    'database_name',
    'role_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètres qui génère l'identifiant unique de la tâche. Ce paramètre accepte uniquement ?.

Les paramètres d'entrée suivants sont requis :

`database_name`

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

`role_name`

Nom du rôle que vous souhaitez supprimer. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de suppression d'un rôle, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime un rôle appelé base ROLE_TEST de donnéesTESTDB.

```
db2 "call rdsadmin.drop_role(
    ?,
    'TESTDB',
    'ROLE_TEST')"
```

rdsadmin.add_user

Ajoute un utilisateur à une liste d'autorisations.

Syntaxe

```
db2 "call rdsadmin.add_user(
    'username',
    'password',
    'group_name,group_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

username

Le nom d'utilisateur d'un utilisateur. Le type de données estvarchar.

password

Le mot de passe d'un utilisateur. Le type de données estvarchar.

Le paramètre suivant est facultatif :

group_name

Le nom du groupe auquel vous souhaitez ajouter l'utilisateur. Le type de données estvarchar. La valeur par défaut est une chaîne vide ou nulle.

Notes d'utilisation

Vous pouvez ajouter un utilisateur à un ou plusieurs groupes en séparant les noms des groupes par des virgules.

Vous pouvez créer un groupe lorsque vous créez un nouvel utilisateur ou lorsque vous ajoutez un groupe à un utilisateur existant. Vous ne pouvez pas créer un groupe tout seul.

> **Note**
>
> Le nombre maximum d'utilisateurs que vous pouvez ajouter en appelant `rdsadmin.add_user` est de 5 000.

Pour plus d'informations sur la vérification de l'état de l'ajout d'un utilisateur, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant crée un utilisateur appelé `jorge_souza` et l'affecte aux groupes appelés `sales` et`inside_sales`.

```
db2 "call rdsadmin.add_user(
    'jorge_souza',
    '*******',
    'sales,inside_sales')"
```

rdsadmin.change_password

Modifie le mot de passe d'un utilisateur.

Syntaxe

```
db2 "call rdsadmin.change_password(
    'username',
    'new_password')"
```

Paramètres

Les paramètres suivants sont obligatoires :

username

Le nom d'utilisateur d'un utilisateur. Le type de données est`varchar`.

new_password

Nouveau mot de passe pour l'utilisateur. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de modification d'un mot de passe, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant modifie le mot de passe de`jorge_souza`.

```
db2 "call rdsadmin.change_password(
    'jorge_souza',
    '*******')"
```

rdsadmin.list_users

Répertorie les utilisateurs sur une liste d'autorisation.

Syntaxe

```
db2 "call rdsadmin.list_users()"
```

Notes d'utilisation

Pour plus d'informations sur la vérification du statut des utilisateurs de la liste, consultezrdsadmin.get_task_status.

rdsadmin.remove_user

Supprime l'utilisateur de la liste d'autorisation.

Syntaxe

```
db2 "call rdsadmin.remove_user('username')"
```

Paramètres

Les paramètres suivants sont obligatoires :

username

Le nom d'utilisateur d'un utilisateur. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de suppression d'un utilisateur, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime l'accès aux bases `jorge_souza` de données dans RDS pour les instances de base de données DB2.

```
db2 "call rdsadmin.remove_user('jorge_souza')"
```

rdsadmin.add_groups

Ajoute des groupes à un utilisateur.

Syntaxe

```
db2 "call rdsadmin.add_groups(
    'username',
    'group_name,group_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

username

Le nom d'utilisateur d'un utilisateur. Le type de données est`varchar`.

group_name

Le nom du groupe auquel vous souhaitez ajouter l'utilisateur. Le type de données est`varchar`. La valeur par défaut est une chaîne vide.

Notes d’utilisation

Vous pouvez ajouter un ou plusieurs groupes à un utilisateur en séparant les noms des groupes par des virgules. Pour plus d'informations sur la vérification de l'état de l'ajout de groupes, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant ajoute les `b2b_sales` groupes `direct_sales` et à l'utilisateur`jorge_souza`.

```
db2 "call rdsadmin.add_groups(
    'jorge_souza',
    'direct_sales,b2b_sales')"
```

rdsadmin.remove_groups

Supprime des groupes d'un utilisateur.

Syntaxe

```
db2 "call rdsadmin.remove_groups(
    'username',
    'group_name,group_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

`username`

Le nom d'utilisateur d'un utilisateur. Le type de données est`varchar`.

`group_name`

Nom du groupe dont vous souhaitez supprimer l'utilisateur. Le type de données est`varchar`.

Notes d’utilisation

Vous pouvez supprimer un ou plusieurs groupes d'un utilisateur en séparant les noms des groupes par des virgules.

Pour plus d'informations sur la vérification de l'état de suppression de groupes, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime les `b2b_sales` groupes `direct_sales` et de l'utilisateur`jorge_souza`.

```
db2 "call rdsadmin.remove_groups(
    'jorge_souza',
    'direct_sales,b2b_sales')"
```

rdsadmin.dbadm_grant

`DBADM`Accorde `ACCESSCTRL` ou `DATAACCESS` autorisation à un rôle, à un utilisateur ou à un groupe.

Syntaxe

```
db2 "call rdsadmin.dbadm_grant(
    ?,
    'database_name',
    'authorization',
    'grantee')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètres qui génère l'identifiant unique de la tâche. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

authorization

Type d'autorisation à accorder. Le type de données est`varchar`. Valeurs valides : `DBADM`, `ACCESSCTRL`, `DATAACCESS`.

Séparez les différents types par des virgules.

grantee

Rôle, utilisateur ou groupe devant recevoir l'autorisation. Le type de données est`varchar`. Valeurs valides : `ROLE`, `USER`, `GROUP`.

Le format doit être une valeur suivie d'un nom. Séparez les valeurs et les noms multiples par des virgules. Exemple : « `USER` *`user1`*`, `*`user2`*`, GROUP` *`group1`*`, `*`group2`* ». Remplacez les noms par vos propres informations.

Notes d'utilisation

Le rôle pour bénéficier de l'accès doit exister.

Pour plus d'informations sur la vérification de l'état de l'octroi de l'accès administrateur à la base de données, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : Octroi à un administrateur de base de données d'un accès à un rôle

L'exemple suivant accorde à l'administrateur de base de données l'accès à la base de données nommée `TESTDB` pour le rôle`ROLE_DBA`.

```
db2 "call rdsadmin.dbadm_grant(
    ?,
    'TESTDB',
    'DBADM',
    'ROLE ROLE_DBA')"
```

Exemple 2 : Octroi d'un accès administrateur de base de données à un utilisateur et à un groupe

L'exemple suivant accorde à l'administrateur de base de données l'accès à la base de données nommée `TESTDB` pour `user1` et`group1`.

```
db2 "call rdsadmin.dbadm_grant(
    ?,
    'TESTDB',
    'DBADM',
```

```
    'USER user1, GROUP group1')"
```

Exemple 3 : Octroi d'un accès d'administrateur de base de données à plusieurs utilisateurs et groupes

L'exemple suivant accorde à l'administrateur de base de données l'accès à la base de données nommée TESTDB pour `user1` `user2``group1`,, et`group2`.

```
db2 "call rdsadmin.dbadm_grant(
    ?,
    'TESTDB',
    'DBADM',
    'USER user1, user2, GROUP group1, group2')"
```

rdsadmin.dbadm_revoke

Révoque DBADM ou DATAACCESS autorise un rôle, un utilisateur ou un groupe. ACCESSCTRL

Syntaxe

```
db2 "call rdsadmin.dbadm_revoke(
    ?,
    'database_name',
    'authorization',
    'grantee')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Identifiant unique de la tâche. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données sur laquelle la commande sera exécutée. Le type de données est`varchar`.

authorization

Type d'autorisation à révoquer. Le type de données est`varchar`. Valeurs valides : `DBADM`, `ACCESSCTRL`, `DATAACCESS`.

Séparez les différents types par des virgules.

grantee

Le rôle, l'utilisateur ou le groupe dont l'autorisation doit être révoquée. Le type de données est`varchar`. Valeurs valides : `ROLE`, `USER`, `GROUP`.

Le format doit être une valeur suivie d'un nom. Séparez les valeurs et les noms multiples par des virgules. Exemple : « `USER` *`user1`*`, `*`user2`*`, GROUP` *`group1`*`, `*`group2`* ». Remplacez les noms par vos propres informations.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de la révocation de l'accès administrateur de base de données, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : Révocation de l'accès à un rôle d'administrateur de base de données

L'exemple suivant révoque l'accès de l'administrateur de base de données à la base de données nommée `TESTDB` pour le rôle`ROLE_DBA`.

```
db2 "call rdsadmin.dbadm_revoke(
    ?,
    'TESTDB',
    'DBADM',
    'ROLE ROLE_DBA')"
```

Exemple 2 : Révocation de l'accès administrateur de base de données à un utilisateur ou à un groupe

L'exemple suivant révoque l'accès de l'administrateur de base de données à la base de données nommée `TESTDB` pour `user1` et`group1`.

```
db2 "call rdsadmin.dbadm_revoke(
    ?,
```

```
    'TESTDB',
    'DBADM',
    'USER user1, GROUP group1')"
```

Exemple 3 : Révocation de l'accès administrateur de base de données à plusieurs utilisateurs et groupes

L'exemple suivant révoque l'accès de l'administrateur de base de données à la base de données nommée TESTDB pour `user1``user2`,`group1`, et`group2`.

```
db2 "call rdsadmin.dbadm_revoke(
    ?,
    'TESTDB',
    'DBADM',
    'USER user1, user2, GROUP group1, group2')"
```

rdsadmin.set_sid_group_mapping

Crée un mappage entre un ID de sécurité (SID) et le groupe Active Directory correspondant.

Syntaxe

```
db2 "call rdsadmin.set_sid_group_mapping(
    ?,
    'SID',
    'group_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

SID

L'ID de sécurité (SID). Le type de données est`varchar`.

group_name

Nom du groupe Active Directory à mapper au SID. Le type de données est`varchar`.

Notes d'utilisation

Utilisez cette procédure stockée pour activer l'authentification Kerberos auprès des groupes Active Directory. Si le `SID` ou existe `group_name` déjà dans le mappage, cette procédure stockée échoue.

Pour plus d'informations sur la manière de trouver le SID d'un groupe, consultezÉtape 8 : récupérer le SID du groupe Active Directory dans PowerShell.

Pour plus d'informations sur la vérification de l'état de création d'un mappage, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant associe un SID à un groupe appelé`my_group`.

```
db2 "call rdsadmin.set_sid_group_mapping(
    ?,
    'S-1-5-21-9146495592-531070549-834388463-513',
    'my_group')"
```

rdsadmin.list_sid_group_mapping

Répertorie tous les mappages d'ID de sécurité (SID) et de groupes Active Directory configurés sur l'instance de base de données.

Syntaxe

```
db2 "call rdsadmin.list_sid_group_mapping()"
```

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état des mappages de listes, consultezrdsadmin.get_task_status.

rdsadmin.remove_sid_group_mapping

Supprime un ID de sécurité (SID) et le mappage de groupe Active Directory correspondant d'une instance de base de données.

Syntaxe

```
db2 "call rdsadmin.remove_sid_group_mapping(
    ?,
    'SID')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

SID

L'ID de sécurité (SID). Le type de données est`varchar`.

Notes d’utilisation

Pour plus d'informations sur la manière de trouver le SID d'un groupe, consultezÉtape 8 : récupérer le SID du groupe Active Directory dans PowerShell.

Pour plus d'informations sur la vérification de l'état de suppression des mappages, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime un mappage SID du groupe auquel il a été mappé.

```
db2 "call rdsadmin.remove_sid_group_mapping(
    ?,
    'S-1-5-21-9146495592-531070549-834388463-513')"
```

Procédures stockées pour les politiques d'audit pour RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent les politiques d'audit pour Amazon RDS pour les bases de données DB2 qui utilisent la journalisation des audits. Pour de plus amples informations, veuillez consulter the section called “Journalisation des audits DB2”. Pour exécuter ces procédures, l'utilisateur principal doit d'abord se connecter à la `rdsadmin` base de données.

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.configure_db_audit
- rdsadmin.disable_db_audit

rdsadmin.configure_db_audit

Configure la politique d'audit pour la base de données RDS pour DB2 spécifiée par. *db_name* Si la politique que vous configurez n'existe pas, l'appel de cette procédure stockée la crée. Si cette règle existe, l'appel de cette procédure stockée la modifie avec les valeurs de paramètres que vous fournissez.

Syntaxe

```
db2 "call rdsadmin.configure_db_audit(
    'db_name',
    'category',
    'category_setting',
    '?')"
```

Paramètres

Les paramètres suivants sont obligatoires.

db_name

Nom de base de données de la base de données RDS pour DB2 pour laquelle configurer la politique d'audit. Le type de données est`varchar`.

category

Nom de la catégorie pour laquelle configurer cette politique d'audit. Le type de données est`varchar`. Les valeurs suivantes sont valides pour ce paramètre :

- `ALL`— Avec`ALL`, Amazon RDS n'inclut pas les `ERROR` catégories `CONTEXTEXECUTE`, ou.
- `AUDIT`
- `CHECKING`
- `CONTEXT`
- `ERROR`
- `EXECUTE`— Vous pouvez configurer cette catégorie avec ou sans données. Avec des moyens de données pour enregistrer également les valeurs de données d'entrée fournies pour toutes les variables hôtes et marqueurs de paramètres. La valeur par défaut est sans données. Pour plus d'informations, consultez la description du *category_setting* paramètre et lethe section called “Exemples”.
- `OBJMAINT`
- `SECMAINT`
- `SYSADMIN`
- `VALIDATE`

Pour plus d'informations sur ces catégories, consultez le IBM Db2 documentation.

category_setting

Paramètre pour la catégorie d'audit spécifiée. Le type de données est`varchar`.

Le tableau suivant indique les valeurs de paramètres de catégorie valides pour chaque catégorie.

Catégorie	Paramètres de catégorie valides
`ALL`	`BOTH`|`FAILURE`|`SUCCESS`|`NONE`
`AUDIT`	
`CHECKING`	
`CONTEXT`	
`OBJMAINT`	

Catégorie	Paramètres de catégorie valides
SECMAINT SYSADMIN VALIDATE	
ERROR	AUDIT|NORMAL . La valeur par défaut estNORMAL.
EXECUTE	BOTH,WITH|BOTH,WITHOUT|FAILURE,WITH| FAILURE,WITHOUT|SUCCESS,WITH|SUCCESS ,WITHOUT|NONE

Notes d'utilisation

Avant d'appelerrdsadmin.configure_db_audit, assurez-vous que l'instance de base de données RDS pour DB2 avec la base de données pour laquelle vous configurez la politique d'audit est associée à un groupe d'options doté de cette option. DB2_AUDIT Pour de plus amples informations, veuillez consulter the section called "Configuration de la journalisation des audits DB2".

Après avoir configuré la stratégie d'audit, vous pouvez vérifier l'état de la configuration d'audit pour la base de données en suivant les étapes décrites dansVérifiez la configuration de l'audit.

La spécification ALL du category paramètre n'inclut pas les ERROR catégories CONTEXTEXECUTE, ou. Pour ajouter ces catégories à votre politique d'audit, appelez rdsadmin.configure_db_audit séparément pour chaque catégorie que vous souhaitez ajouter. Pour de plus amples informations, veuillez consulter the section called "Exemples".

Exemples

Les exemples suivants créent ou modifient la politique d'audit pour une base de données nomméeTESTDB. Dans les exemples 1 à 5, si la ERROR catégorie n'a pas été configurée auparavant, elle est définie sur NORMAL (valeur par défaut). Pour modifier ce paramètre surAUDIT, suivezExample 6: Specifying the ERROR category.

Exemple 1 : Spécification de la **ALL** catégorie

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'ALL', 'BOTH', ?)"
```

Dans l'exemple, l'appel configure les `VALIDATE` catégories`AUDIT`,`CHECKING`,`OBJMAINT`, `SECMAINTSYSADMIN`, et dans la politique d'audit. Spécifier `BOTH` signifie que les événements réussis et les échecs seront audités pour chacune de ces catégories.

Exemple 2 : Spécification de la **`EXECUTE`** catégorie avec des données

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'EXECUTE', 'SUCCESS,WITH', ?)"
```

Dans l'exemple, l'appel configure la `EXECUTE` catégorie dans la politique d'audit. La spécification `SUCCESS,WITH` signifie que les journaux de cette catégorie n'incluront que les événements réussis et incluront les valeurs de données d'entrée fournies pour les variables hôtes et les marqueurs de paramètres.

Exemple 3 : Spécification de la **`EXECUTE`** catégorie sans données

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'EXECUTE', 'FAILURE,WITHOUT', ?)"
```

Dans l'exemple, l'appel configure la `EXECUTE` catégorie dans la politique d'audit. Cette spécification `FAILURE,WITHOUT` signifie que les journaux de cette catégorie n'incluront que les événements défaillants et n'incluront pas les valeurs de données d'entrée fournies pour les variables hôtes et les marqueurs de paramètres.

Exemple 4 : Spécification de la **`EXECUTE`** catégorie sans événements de statut

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'EXECUTE', 'NONE', ?)"
```

Dans l'exemple, l'appel configure la `EXECUTE` catégorie dans la politique d'audit. Spécifier `NONE` signifie qu'aucun événement de cette catégorie ne sera audité.

Exemple 5 : Spécification de la **`OBJMAINT`** catégorie

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'OBJMAINT', 'NONE', ?)"
```

Dans l'exemple, l'appel configure la `OBJMAINT` catégorie dans la politique d'audit. Spécifier `NONE` signifie qu'aucun événement de cette catégorie ne sera audité.

Exemple 6 : Spécification de la **`ERROR`** catégorie

```
db2 "call rdsadmin.configure_db_audit('TESTDB', 'ERROR', 'AUDIT', ?)"
```

Dans l'exemple, l'appel configure la `ERROR` catégorie dans la politique d'audit. La spécification `AUDIT` signifie que toutes les erreurs, y compris les erreurs survenant dans le journal d'audit lui-même, sont enregistrées dans les journaux. Le type d'erreur par défaut est`NORMAL`. Avec`NORMAL`, les erreurs générées par l'audit sont ignorées et seuls les `SQLCODE` s correspondant aux erreurs associées à l'opération en cours sont capturés.

rdsadmin.disable_db_audit

Arrête la journalisation d'audit pour la base de données RDS pour DB2 spécifiée par *db_name* et supprime la politique d'audit configurée pour cette base de données.

> **Note**
>
> Cette procédure stockée supprime uniquement les politiques d'audit configurées lors d'un appelthe section called “rdsadmin.configure_db_audit”.

Syntaxe

```
db2 "call rdsadmin.disable_db_audit('db_name', ?)"
```

Paramètres

Les paramètres suivants sont obligatoires.

db_name

Nom de base de données de la base de données RDS pour DB2 pour laquelle la journalisation des audits doit être désactivée. Le type de données est`varchar`.

Notes d’utilisation

L'appel `rdsadmin.disable_db_audit` ne désactive pas la journalisation des audits pour l'instance de base de données RDS pour DB2. Pour désactiver la journalisation des audits au niveau de l'instance de base de données, supprimez le groupe d'options de l'instance de base de données. Pour de plus amples informations, veuillez consulter Désactivation de la journalisation des audits DB2.

Exemples

L'exemple suivant désactive la journalisation des audits pour une base de données nomméeTESTDB.

```
db2 "call rdsadmin.disable_db_audit('TESTDB', ?)"
```

Procédures stockées pour les pools de mémoire tampon RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent les pools de mémoire tampon pour les bases de données Amazon RDS for Db2. Pour exécuter ces procédures, l'utilisateur principal doit d'abord se connecter à la `rdsadmin` base de données.

Ces procédures stockées sont utilisées dans diverses tâches. Cette liste n'est pas exhaustive.

- Tâches courantes pour les pools de mémoire tampon
- Génération de rapports de performance
- Copier les métadonnées d'une base de données avec db2look
- Création d'une base de données de référentiel pour IBM Db2 Data Management Console

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.create_bufferpool
- rdsadmin.alter_bufferpool
- rdsadmin.drop_bufferpool

rdsadmin.create_bufferpool

Crée un pool de mémoire tampon.

Syntaxe

```
db2 "call rdsadmin.create_bufferpool(
    'database_name',
    'buffer_pool_name',
    buffer_pool_size,
    'immediate',
    'automatic',
    page_size,
    number_block_pages,
    block_size)"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données sur laquelle exécuter la commande. Le type de données est`varchar`.

buffer_pool_name

Nom du pool de mémoire tampon à créer. Le type de données est`varchar`.

Les paramètres suivants sont facultatifs :

buffer_pool_size

Taille du pool de mémoire tampon en nombre de pages. Le type de données est`integer`. L'argument par défaut est `-1`.

immediate

Spécifie si la commande s'exécute immédiatement. Le type de données est`char`. L'argument par défaut est Y.

automatic

Spécifie s'il faut définir le pool de tampons sur automatique. Le type de données est`char`. L'argument par défaut est Y.

page_size

Taille de page du pool de mémoire tampon. Le type de données est`integer`. Valeurs valides: `4096`, `8192`, `16384`, `32768`. L'argument par défaut est `8192`.

number_block_pages

Le nombre de pages de blocs dans les pools de mémoire tampon. Le type de données est`integer`. L'argument par défaut est `0`.

block_size

Taille de bloc pour les pages de bloc. Le type de données est`integer`. Valeurs valides : `2` à `256`. L'argument par défaut est `32`.

Notes d’utilisation

Pour plus d'informations sur la vérification de l'état de création d'un pool de mémoire tampon, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : Création d'un pool de mémoire tampon avec des paramètres par défaut

L'exemple suivant crée un pool de mémoire tampon appelé BP8 pour une base de données appelée TESTDB avec des paramètres par défaut, de sorte que le pool de mémoire tampon utilise une taille de page de 8 Ko.

```
db2 "call rdsadmin.create_bufferpool(
    'TESTDB',
    'BP8')"
```

Exemple 2 : Création d'un pool de mémoire tampon à exécuter immédiatement avec allocation automatique

L'exemple suivant crée un pool de mémoire tampon appelé BP16 pour une base de données appelée TESTDB qui utilise une taille de page de 16 Ko avec un nombre initial de pages de 1 000 et qui est définie sur automatique. DB2 exécute la commande immédiatement. Si vous utilisez un nombre de pages initial de -1, Db2 utilisera l'allocation automatique des pages.

```
db2 "call rdsadmin.create_bufferpool(
    'TESTDB',
    'BP16',
    1000,
    'Y',
    'Y',
    16384)"
```

Exemple 3 : Création d'un pool de mémoire tampon à exécuter immédiatement à l'aide de pages de blocs

L'exemple suivant crée un pool de mémoire tampon appelé BP16 pour une base de données appeléeTESTDB. Ce pool de mémoire tampon a une taille de page de 16 Ko avec un nombre de pages initial de 10 000. Db2 exécute immédiatement la commande en utilisant 500 pages de blocs d'une taille de bloc de 512.

```
db2 "call rdsadmin.create_bufferpool(
    'TESTDB',
    'BP16',
    10000,
    'Y',
    'Y',
    16384,
    500,
    512)"
```

rdsadmin.alter_bufferpool

Modifie un pool tampon.

Syntaxe

```
db2 "call rdsadmin.alter_bufferpool(
    'database_name',
    'buffer_pool_name',
    buffer_pool_size,
    'immediate',
    'automatic',
    change_number_blocks,
    number_block_pages,
    block_size)"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données sur laquelle exécuter la commande. Le type de données est`varchar`.

buffer_pool_name

Nom du pool de mémoire tampon à modifier. Le type de données est`varchar`.

buffer_pool_size

Taille du pool de mémoire tampon en nombre de pages. Le type de données est`integer`.

Les paramètres suivants sont facultatifs :

immediate

Spécifie si la commande s'exécute immédiatement. Le type de données est`char`. L'argument par défaut est `Y`.

automatic

Spécifie s'il faut définir le pool de tampons sur automatique. Le type de données est`char`. L'argument par défaut est `N`.

change_number_blocks

Spécifie si le nombre de pages de blocs dans le pool de mémoire tampon est modifié. Le type de données est`char`. L'argument par défaut est `N`.

number_block_pages

Le nombre de pages de blocs dans les pools de mémoire tampon. Le type de données est`integer`. L'argument par défaut est `0`.

block_size

Taille de bloc pour les pages de bloc. Le type de données est`integer`. Valeurs valides : `2` à `256`. L'argument par défaut est `32`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de modification d'un pool de mémoire tampon, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant modifie un pool de mémoire tampon appelé `BP16` pour une base de données appelée `TESTDB` non automatique et modifie la taille à 10 000 pages. DB2 exécute cette commande immédiatement.

```
db2 "call rdsadmin.alter_bufferpool(
    'TESTDB',
    'BP16',
    10000,
    'Y',
```

```
    'N')"
```

rdsadmin.drop_bufferpool

Supprime un pool de mémoire tampon.

Syntaxe

```
db2 "call rdsadmin.drop_bufferpool(
    'database_name',
    'buffer_pool_name'"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données à laquelle appartient le pool de tampons. Le type de données est`varchar`.

buffer_pool_name

Nom du pool de mémoire tampon à supprimer. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de suppression d'un pool de mémoire tampon, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime un pool de mémoire tampon appelé `BP16` pour une base de données appelée`TESTDB`.

```
db2 "call rdsadmin.drop_bufferpool(
    'TESTDB',
    'BP16')"
```

Procédures stockées pour les bases de données pour RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent les bases de données pour Amazon RDS for Db2. Pour exécuter ces procédures, l'utilisateur principal doit d'abord se connecter à la `rdsadmin` base de données.

Ces procédures stockées sont utilisées dans diverses tâches. Cette liste n'est pas exhaustive.

- Tâches courantes pour les bases de données
- Création de bases de données avec classement EBCDIC
- Collecte d'informations sur les bases de données
- Modification des paramètres de configuration de base de données
- Migration de Linux vers Linux
- Migration de Linux vers Linux avec un temps d'arrêt quasi nul

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.create_database
- rdsadmin.deactivate_database
- rdsadmin.activate_database
- rdsadmin.reactivate_database
- rdsadmin.drop_database
- rdsadmin.update_db_param
- rdsadmin.set_configuration
- rdsadmin.show_configuration
- rdsadmin.restore_database
- rdsadmin.rollforward_database
- rdsadmin.rollforward_status
- rdsadmin.complete_rollforward
- rdsadmin.db2pd_command

- rdsadmin.force_application
- rdsadmin.set_archive_log_retention
- rdsadmin.show_archive_log_retention
- rdsadmin.list_archive_log_information

rdsadmin.create_database

Crée une base de données.

Syntaxe

```
db2 "call rdsadmin.create_database('database_name')"
```

Paramètres

> **Note**
>
> Cette procédure stockée ne valide pas la combinaison des paramètres requis. Lorsque vous appelezrdsadmin.get_task_status, la fonction définie par l'utilisateur peut renvoyer une erreur en raison d'une combinaison de `database_codesetdatabase_territory`, et `database_collation` cela n'est pas valide. Pour plus d'informations, consultez la section Choix de la page de code, du territoire et du classement pour votre base de données dans la IBM Db2 documentation.

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données à créer. Le type de données est`varchar`.

Les paramètres suivants sont facultatifs :

database_page_size

Taille de page par défaut de la base de données. Valeurs valides: `4096`, `8192`, `16384`, `32768`. Le type de données est`integer`. L'argument par défaut est `8192`.

> **Important**
>
> Amazon RDS prend en charge l'atomicité d'écriture pour les pages 4 KiB, 8 KiB et 16 KiB. En revanche, les pages de 32 Kio risquent d'être déchirées ou d'écrire des données partielles sur le bureau. Si vous utilisez des pages 32 KiB, nous vous recommandons d'activer la point-in-time restauration et les sauvegardes automatisées. Sinon, vous risquez de ne pas pouvoir récupérer des pages déchirées. Pour plus d'informations, consultez the section called “Présentation des sauvegardes” et the section called “Point-in-time rétablissement”.

database_code_set

Code défini pour la base de données. Le type de données est`varchar`. L'argument par défaut est `UTF-8`.

database_territory

Le code de pays à deux lettres de la base de données. Le type de données est`varchar`. L'argument par défaut est `US`.

database_collation

Séquence de classement qui détermine la manière dont les chaînes de caractères stockées dans la base de données sont triées et comparées. Le type de données est`varchar`.

Valeurs valides :

- `COMPATIBILITY`— Une séquence de classement IBM Db2 version 2.
- `EBCDIC_819_037`— Page de code latin ISO, classement ; CCSID 037 (EBCDIC, anglais américain).
- `EBCDIC_819_500`— Page de code latin ISO, classement ; CCSID 500 (EBCDIC International).
- `EBCDIC_850_037`— Page de code latin ASCII, collation ; CCSID 037 (EBCDIC, anglais américain).
- `EBCDIC_850_500`— Page de code latin ASCII, classement ; CCSID 500 (EBCDIC International).
- `EBCDIC_932_5026`— Page de code ASCII en japonais, collation ; CCSID 5026 (EBCDIC, anglais américain).

- `EBCDIC_932_5035`— Page de code ASCII en japonais, classement ; CCSID 5035 (EBCDIC International).
- `EBCDIC_1252_037`— Page de code latin de Windows, classement ; CCSID 037 (EBCDIC, anglais américain).
- `EBCDIC_1252_500`— Page de code latin de Windows, classement ; CCSID 500 (EBCDIC International).
- `IDENTITY`— Classement par défaut. Les chaînes sont comparées octet par octet.
- `IDENTITY_16BIT`— Le schéma de codage de compatibilité pour UTF-16 : séquence de classement 8 bits (CESU-8). Pour plus d'informations, consultez le rapport technique Unicode #26 sur le site Web du consortium Unicode.
- `NLSCHAR`— À utiliser uniquement avec la page de code thaï (CP874).
- `SYSTEM`— Si vous utilisez`SYSTEM`, la base de données utilise automatiquement la séquence de classement pour `database_codeset` et`database_territory`.

L'argument par défaut est `IDENTITY`.

En outre, RDS pour Db2 prend en charge les groupes de collations suivants : `language-aware-collation` et. `locale-sensitive-collation` Pour plus d'informations, consultez la section Choix d'un classement pour une base de données Unicode dans la IBM Db2 documentation.

`database_autoconfigure_str`

La syntaxe de la `AUTOCONFIGURE` commande, par exemple, `'AUTOCONFIGURE APPLY DB'`. Le type de données est`varchar`. La valeur par défaut est une chaîne vide ou nulle.

Pour plus d'informations, consultez AUTOCONFIGUREla section commande dans la IBM Db2 documentation.

Notes d'utilisation

Si vous envisagez de modifier le `db2_compatibility_vector` paramètre, modifiez-le avant de créer une base de données. Pour de plus amples informations, veuillez consulter Définition du paramètre db2_compatibility_vector.

Considérations spéciales :

- La `CREATE DATABASE` commande envoyée à l'instance Db2 utilise l'`RESTRICTIVE`option.
- RDS pour DB2 utilise `AUTOMATIC STORAGE` uniquement des tablespaces.

- RDS pour DB2 utilise les valeurs par défaut pour `NUMSEGS` et. `DFT_EXTENT_SZ`
- RDS pour DB2 utilise le chiffrement du stockage et ne prend pas en charge le chiffrement de base de données.

Pour plus d'informations sur ces considérations, consultez la section CREATE DATABASEcommande dans la IBM Db2 documentation.

Avant d'appeler`rdsadmin.create_database`, vous devez vous connecter à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *`master_username`* et *`master_password`* par votre RDS pour les informations d'instance de base de données DB2 :

```
db2 connect to rdsadmin user master_username using master_password
```

Pour plus d'informations sur la vérification de l'état de création d'une base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.create_database` l'appel, consultezthe section called “Erreurs de procédure stockée”.

Exemples

L'exemple suivant crée une base de données appelée `TESTJP` avec une combinaison correcte des *`database_collation`* paramètres *`database_code_setdatabase_territory`*,, et pour le Japon :

```
db2 "call rdsadmin.create_database('TESTJP', 4096, 'IBM-437', 'JP', 'SYSTEM')"
```

rdsadmin.deactivate_database

Désactive une base de données.

Syntaxe

```
db2 "call rdsadmin.deactivate_database(
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données à désactiver. Le type de données est`varchar`.

Notes d'utilisation

Vous pouvez désactiver les bases de données pour économiser les ressources de mémoire ou pour apporter d'autres modifications à la configuration des bases de données. Pour remettre en ligne les bases de données désactivées, appelez la procédure rdsadmin.activate_database stockée.

Pour plus d'informations sur la vérification de l'état de désactivation d'une base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.deactivate_database` l'appel, consultezthe section called "Erreurs de procédure stockée".

Exemples

L'exemple suivant désactive une base de données appeléeTESTDB.

```
db2 "call rdsadmin.deactivate_database(?, 'TESTDB')"
```

rdsadmin.activate_database

Active une base de données.

Pour plus d'informations sur les différences entre rdsadmin.reactivate_database et `rdsadmin.activate_database`, consultez Notes d'utilisation.

Syntaxe

```
db2 "call rdsadmin.activate_database(
    ?,
```

```
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données à activer. Le type de données est`varchar`.

Notes d'utilisation

Toutes les bases de données sont activées par défaut lors de leur création. Si vous désactivez une base de données sur une instance de base de données autonome pour économiser des ressources mémoire ou pour apporter d'autres modifications à la configuration de la base de données, appelez la procédure `rdsadmin.activate_database` stockée pour réactiver la base de données.

Pour plus d'informations sur la vérification de l'état d'activation d'une base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.activate_database` l'appel, consultezthe section called "Erreurs de procédure stockée".

Exemples

L'exemple suivant active une base de données appeléeTESTDB.

```
db2 "call rdsadmin.activate_database(?, 'TESTDB')"
```

rdsadmin.reactivate_database

Réactive une base de données.

Pour plus d'informations sur les différences entre rdsadmin.activate_database et`rdsadmin.reactivate_database`, voirNotes d'utilisation.

Syntaxe

```
db2 "call rdsadmin.reactivate_database(
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données à réactiver. Le type de données est`varchar`.

Notes d'utilisation

Lorsque vous appelez la procédure `rdsadmin.reactivate_database` stockée, celle-ci désactive d'abord la base de données en appelant la procédure rdsadmin.deactivate_database stockée, puis active la base de données en appelant la procédure rdsadmin.activate_database stockée.

Après avoir modifié les configurations de base de données, vous devrez peut-être réactiver une base de données sur une instance de base de données RDS pour DB2. Pour déterminer si vous devez réactiver une base de données, connectez-vous à la base de données et `db2 get db cfg show detail` exécutez-la.

Pour une base de données sur une instance de base de données autonome, vous pouvez utiliser la procédure de `rdsadmin.reactivate_database` stockage pour réactiver la base de données. Ou, si vous avez déjà appelé la procédure rdsadmin.deactivate_database stockée, vous pouvez l'rdsadmin.activate_databaseappeler à la place.

Pour plus d'informations sur la vérification de l'état de réactivation d'une base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.reactivate_database` l'appel, consultezthe section called “Erreurs de procédure stockée”.

Exemples

L'exemple suivant réactive une base de données appeléeTESTDB.

```
db2 "call rdsadmin.reactivate_database(?, 'TESTDB')"
```

rdsadmin.drop_database

Supprime une base de données.

Syntaxe

```
db2 "call rdsadmin.drop_database('database_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données à supprimer. Le type de données est`varchar`.

Notes d'utilisation

Vous pouvez supprimer une base de données en appelant `rdsadmin.drop_database` uniquement si les conditions suivantes sont remplies :

- Vous n'avez pas spécifié le nom de la base de données lorsque vous avez créé votre instance de base de données RDS pour DB2 à l'aide de la console Amazon RDS ou du. AWS CLI Pour de plus amples informations, veuillez consulter Création d'une instance de base de données.
- Vous avez créé la base de données en appelant la procédure the section called "rdsadmin.create_database" stockée.
- Vous avez restauré la base de données à partir d'une image hors ligne ou sauvegardée en appelant la procédure the section called "rdsadmin.restore_database" stockée.

Avant d'appeler`rdsadmin.drop_database`, vous devez vous connecter à la `rdsadmin` base de données. Dans l'exemple suivant, remplacez *master_username* et *master_password* par votre RDS pour les informations d'instance de base de données DB2 :

```
db2 connect to rdsadmin user master_username using master_password
```

Pour plus d'informations sur la vérification de l'état de suppression d'une base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.drop_database` l'appel, consultezthe section called “Erreurs de procédure stockée”.

Exemples

L'exemple suivant supprime une base de données appelée `TESTDB` :

```
db2 "call rdsadmin.drop_database('TESTDB')"
```

rdsadmin.update_db_param

Actualise les paramètres de base de données.

Syntaxe

```
db2 "call rdsadmin.update_db_param(
    'database_name',
    'parameter_to_modify',
    'changed_value',
    'restart_database')"
```

Paramètres

Les paramètres suivants sont obligatoires :

`database_name`

Nom de la base de données pour laquelle exécuter la tâche. Le type de données est`varchar`.

`parameter_to_modify`

Nom du paramètre à modifier. Le type de données est`varchar`. Pour de plus amples informations, veuillez consulter Paramètres Amazon RDS pour DB2.

`changed_value`

La valeur à laquelle modifier la valeur du paramètre. Le type de données est`varchar`.

Le paramètre suivant est facultatif :

restart_database

Spécifie si RDS redémarre la base de données si un redémarrage est nécessaire. Le type de données est`varchar`. Pour modifier `logprimary` et`logfilsiz`, définissez ce paramètre sur`'YES'`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de mise à jour des paramètres de base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.update_db_param` l'appel, consultezthe section called "Erreurs de procédure stockée".

Exemples

Exemple 1 : mise à jour d'un paramètre

L'exemple suivant met à jour le `archretrydelay` paramètre `100` pour une base de données appelée `TESTDB` :

```
db2 "call rdsadmin.update_db_param(
    'TESTDB',
    'archretrydelay',
    '100')"
```

Exemple 2 : report de la validation d'objets

L'exemple suivant reporte la validation des objets créés sur une base de données appelée `TESTDB` pour éviter la vérification des dépendances :

```
db2 "call rdsadmin.update_db_param(
    'TESTDB',
    'auto_reval',
    'deferred_force')"
```

rdsadmin.set_configuration

Configure les paramètres spécifiques de la base de données.

Syntaxe

```
db2 "call rdsadmin.set_configuration(
    'name',
    'value')"
```

Paramètres

Les paramètres suivants sont obligatoires :

name

Nom du paramètre de configuration. Le type de données est`varchar`.

value

La valeur du paramètre de configuration. Le type de données est`varchar`.

Notes d'utilisation

Le tableau suivant indique les paramètres de configuration que vous pouvez contrôler`rdsadmin.set_configuration`.

Name (Nom)	Description
`RESTORE_DATABASE_NUM_BUFFERS`	Nombre de tampons à créer lors d'une opération de restauration. Cette valeur doit être inférieure à la taille de mémoire totale de la classe d'instance de base de données. Si ce paramètre n'est pas configuré, Db2 détermine la valeur à utiliser lors de l'opération de restauration. Pour plus d'informations, consultez la documentation IBM Db2.
`RESTORE_DATABASE_PARALLELISM`	Nombre de manipulateurs de mémoire tampon à créer lors d'une opération de restauration. Cette valeur doit être inférieure au double du nombre de v CPUs pour l'instance de base de données. Si ce paramètre n'est pas configuré, Db2 détermine la valeur à utiliser lors de l'opération de restauration. Pour plus d'informations, consultez la documentation IBM Db2.

Exemples

Exemple 1 : Spécification du nombre de manipulateurs de mémoire tampon à créer

L'exemple suivant définit la `RESTORE_DATABASE_PARALLELISM` configuration sur`8`.

```
db2 "call rdsadmin.set_configuration(
    'RESTORE_DATABASE_PARALLELISM',
    '8')"
```

Exemple 2 : Spécification du nombre de tampons à créer

L'exemple suivant définit la `RESTORE_DATABASE_NUM_BUFFERS` configuration sur`150`.

```
db2 "call rdsadmin.set_configuration(
    'RESTORE_DATABASE_NUM_BUFFERS',
    '150')"
```

rdsadmin.show_configuration

Renvoie les paramètres actuels que vous pouvez définir à l'aide de la procédure stockée`rdsadmin.set_configuration`.

Syntaxe

```
db2 "call rdsadmin.show_configuration(
    'name')"
```

Paramètres

Le paramètre suivant est facultatif :

`name`

Nom du paramètre de configuration pour lequel les informations doivent être renvoyées. Le type de données est`varchar`.

Les noms de configuration suivants sont valides :

- RESTORE_DATABASE_NUM_BUFFERS — Nombre de tampons à créer lors d'une opération de restauration.
- RESTORE_DATABASE_PARALLELISM — Nombre de manipulateurs de mémoire tampon à créer lors d'une opération de restauration.

Notes d'utilisation

Si vous ne spécifiez pas le nom d'un paramètre de configuration, `rdsadmin.show_configuration` renvoie des informations pour tous les paramètres de configuration que vous pouvez définir à l'aide de la procédure stockée`rdsadmin.set_configuration`.

Exemples

L'exemple suivant renvoie des informations sur la `RESTORE_DATABASE_PARALLELISM` configuration actuelle.

```
db2 "call rdsadmin.show_configuration(
    'RESTORE_DATABASE_PARALLELISM')"
```

rdsadmin.restore_database

Restaure une base de données depuis un compartiment Amazon S3 vers votre instance de base de données RDS pour DB2.

Syntaxe

```
db2 "call rdsadmin.restore_database(
    ?,
    'database_name',
    's3_bucket_name',
    's3_prefix',
    restore_timestamp,
    'backup_type')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données cible à restaurer dans RDS pour DB2. Le type de données est`varchar`.

Par exemple, si le nom de la base de données source était `TESTDB` et que vous l'avez défini *database_name* sur`NEWDB`, Amazon RDS effectue la restauration `NEWDB` en tant que base de données source.

s3_bucket_name

Le nom du compartiment Amazon S3 dans lequel réside votre sauvegarde. Le type de données est`varchar`.

s3_prefix

Le préfixe à utiliser pour la mise en correspondance des fichiers lors du téléchargement. Le type de données est`varchar`.

Si ce paramètre est vide, tous les fichiers du compartiment Amazon S3 seront téléchargés. Voici un exemple de préfixe :

```
backupfolder/SAMPLE.0.rdsdb.DBPART000.20230615010101
```

restore_timestamp

Horodatage de l'image de sauvegarde de la base de données. Le type de données est`varchar`.

L'horodatage est inclus dans le nom du fichier de sauvegarde. Par exemple, `20230615010101` est l'horodatage du nom du fichier. `SAMPLE.0.rdsdb.DBPART000.20230615010101.001`

backup_type

Type de sauvegarde. Le type de données est`varchar`. Valeurs valides : `OFFLINE`, `ONLINE`.

À utiliser `ONLINE` pour des migrations quasiment sans interruption de service. Pour de plus amples informations, veuillez consulter Migration à partir de Linux to Linux avec un temps d'arrêt quasi nul pour Amazon RDS pour DB2.

Notes d’utilisation

Vous pouvez utiliser cette procédure stockée pour migrer une base de données DB2 vers une instance de base de données RDS pour DB2. Pour de plus amples informations, veuillez consulter

Utilisation de AWS services pour migrer des données de Db2 vers Amazon RDS pour Db2. Vous pouvez également utiliser cette procédure stockée pour créer plusieurs copies de la même base de données portant des noms de base de données différents et utilisant la même image de restauration.

Avant d'appeler la procédure stockée, prenez en compte les points suivants :

- Avant de restaurer une base de données, vous devez allouer un espace de stockage pour votre instance de base de données RDS for DB2 égal ou supérieur à la somme de la taille de votre sauvegarde et de la base de données DB2 d'origine sur le disque. Pour de plus amples informations, veuillez consulter Insufficient disk space.
- Lorsque vous restaurez la sauvegarde, Amazon RDS extrait le fichier de sauvegarde sur votre instance de base de données RDS pour DB2. Chaque fichier de sauvegarde doit être inférieur ou égal à 5 To. Si un fichier de sauvegarde dépasse 5 To, vous devez diviser celui-ci en plusieurs fichiers plus petits.
- Pour restaurer tous les fichiers à l'aide de la procédure `rdsadmin.restore_database` stockée, n'incluez pas le suffixe du numéro de fichier après l'horodatage dans les noms de fichiers. Par exemple, les fichiers suivants sont *`s3_prefix`* `backupfolder/SAMPLE.0.rdsdb.DBPART000.20230615010101` restaurés :

```
SAMPLE.0.rdsdb.DBPART000.20230615010101.001
SAMPLE.0.rdsdb.DBPART000.20230615010101.002
SAMPLE.0.rdsdb.DBPART000.20230615010101.003
SAMPLE.0.rdsdb.DBPART000.20230615010101.004
SAMPLE.0.rdsdb.DBPART000.20230615010101.005
```

- RDS pour DB2 ne prend pas en charge le stockage non automatique. Pour de plus amples informations, veuillez consulter Tablespaces not restored.
- RDS pour DB2 ne prend pas en charge les routines non clôturées. Pour de plus amples informations, veuillez consulter Non-fenced routines not allowed.
- Pour améliorer les performances des opérations de restauration de base de données, vous pouvez configurer le nombre de tampons et de manipulateurs de tampons à utiliser par RDS. Pour vérifier la configuration actuelle, utilisezthe section called “rdsadmin.show_configuration”. Pour modifier la configuration, utilisezthe section called “rdsadmin.set_configuration”.

Pour mettre la base de données en ligne et appliquer des journaux de transactions supplémentaires après la restauration de la base de données, consultezrdsadmin.rollforward_database.

Pour plus d'informations sur la vérification de l'état de restauration de votre base de données, consultezrdsadmin.get_task_status.

Pour les messages d'erreur renvoyés lors de `rdsadmin.restore_database` l'appel, consultezthe section called “Erreurs de procédure stockée”.

Exemples

L'exemple suivant restaure une sauvegarde hors ligne avec un seul fichier ou plusieurs fichiers dotés des caractéristiques *`s3_prefix`* `backupfolder/SAMPLE.0.rdsdb.DBPART000.20230615010101` suivantes :

```
db2 "call rdsadmin.restore_database(
    ?,
    'SAMPLE',
    'amzn-s3-demo-bucket',
    'backupfolder/SAMPLE.0.rdsdb.DBPART000.20230615010101',
    20230615010101,
    'OFFLINE')"
```

rdsadmin.rollforward_database

Met la base de données en ligne et applique des journaux de transactions supplémentaires après avoir restauré une base de données en appelantrdsadmin.restore_database.

Syntaxe

```
db2 "call rdsadmin.rollforward_database(
    ?,
    'database_name',
    's3_bucket_name',
    s3_prefix,
    'rollforward_to_option',
    'complete_rollforward')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données sur laquelle effectuer l'opération. Le type de données est`varchar`.

s3_bucket_name

Le nom du compartiment Amazon S3 dans lequel réside votre sauvegarde. Le type de données est`varchar`.

s3_prefix

Le préfixe à utiliser pour la mise en correspondance des fichiers lors du téléchargement. Le type de données est`varchar`.

Si ce paramètre est vide, tous les fichiers du compartiment S3 seront téléchargés. L'exemple suivant est un exemple de préfixe :

```
backupfolder/SAMPLE.0.rdsdb.DBPART000.20230615010101
```

Les paramètres d'entrée suivants sont facultatifs :

rollforward_to_option

Le point vers lequel vous souhaitez avancer. Le type de données est`varchar`. Valeurs valides :`END_OF_LOGS`, `END_OF_BACKUP` avec l'horodatage au format. `YYYY-MM-DD-HH.MM.SS` L'argument par défaut est `END_OF_LOGS`.

complete_rollforward

Spécifie s'il faut terminer le processus de reconduction. Le type de données est`varchar`. L'argument par défaut est `TRUE`.

SiTRUE, une fois terminée, la base de données est en ligne et accessible. Si c'est le casFALSE, la base de données reste dans un `ROLL-FORWARD PENDING` état.

Notes d'utilisation

Vous pouvez l'utiliser `rds.rollforward_database` pour une sauvegarde en ligne avec des journaux d'inclusion produits sur site dans de nombreux scénarios différents.

Scénario 1 : restauration de la base de données, report des journaux inclus et mise en ligne de la base de données

Une fois `rdsadmin.restore_database()` terminé, utilisez la syntaxe de l'exemple 1 pour mettre en ligne la base de données contenant les journaux de transactions.

Scénario 2 : mise en ligne de la base de données sans report des journaux inclus.

Une fois `rdsadmin.restore_database()` terminé, utilisez la syntaxe de l'exemple 2 pour mettre en ligne la base de données sans les journaux de transactions.

Scénario 3 : report des journaux inclus dans la sauvegarde et application de journaux de transactions supplémentaires au fur et à mesure qu'ils sont produits sur site

Une fois l'`rdsadmin.restore_database()`opération terminée, utilisez la syntaxe de l'exemple 3 ou de l'exemple 4 pour transférer les journaux sans mettre la base de données en ligne.

Si vous définissez `complete_rollforward` cette option`FALSE`, votre base de données est en `ROLL-FORWARD PENDING` état et hors ligne. Pour mettre la base de données en ligne, vous devez appelerrdsadmin.complete_rollforward.

Pour plus d'informations sur la vérification de l'état du report de la base de données, consultezrdsadmin.rollforward_status.

Exemples

Exemple 1 : mise en ligne d'une base de données contenant des journaux de transactions

L'exemple suivant passe à une sauvegarde en ligne de la base de données avec les journaux de transactions, puis met la base de données en ligne :

```
db2 "call rdsadmin.rollforward_database(
    ?,
    null,
    null,
    'END_OF_LOGS',
    'TRUE')"
```

Exemple 2 : mise en ligne d'une base de données sans journaux de transactions

L'exemple suivant passe à une sauvegarde en ligne de la base de données sans journaux de transactions, puis met la base de données en ligne :

```
db2 "call rdsadmin.rollforward_database(
    ?,
    'TESTDB',
    'amzn-s3-demo-bucket',
    'logsfolder/,
    'END_OF_BACKUP',
    'TRUE')"
```

Exemple 3 : ne pas mettre en ligne une base de données contenant des journaux de transactions

L'exemple suivant passe à une sauvegarde en ligne de la base de données avec les journaux de transactions, puis ne met pas la base de données en ligne :

```
db2 "call rdsadmin.rollforward_database(
    ?,
    'TESTDB',
    null,
    'onlinebackup/TESTDB',
    'END_OF_LOGS',
    'FALSE')"
```

Exemple 4 : ne pas mettre en ligne une base de données contenant des journaux de transactions supplémentaires

L'exemple suivant passe à une sauvegarde en ligne de la base de données avec des journaux de transactions supplémentaires, puis ne met pas la base de données en ligne :

```
db2 "call rdsadmin.rollforward_database(
    ?,
    'TESTDB',
    'amzn-s3-demo-bucket',
    'logsfolder/S0000155.LOG',
    'END_OF_LOGS',
    'FALSE')"
```

rdsadmin.rollforward_status

Renvoie le résultat de`ROLLFORWARD DATABASE database_name QUERY STATUS`.

Syntaxe

```
db2 "call rdsadmin.rollforward_status(
```

```
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données sur laquelle effectuer l'opération. Le type de données est`varchar`.

Notes d'utilisation

Après avoir appelérdsadmin.rollforward_database, vous pouvez appeler `rdsadmin.rollforward_status` pour vérifier l'état du rollforward dans la base de données.

Pour plus d'informations sur la vérification de l'état de cette procédure stockée, consultezrdsadmin.get_task_status.

rdsadmin.complete_rollforward

Met en ligne la base de données d'un `ROLL-FORWARD PENDING` État.

Syntaxe

```
db2 "call rdsadmin.complete_rollforward(
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données que vous souhaitez mettre en ligne. Le type de données est`varchar`.

Notes d'utilisation

Si vous avez appelé rdsadmin.rollforward_database avec `complete_rollforward` set to`FALSE`, cela signifie que votre base de données est en `ROLL-FORWARD PENDING` état et hors ligne. Pour terminer le processus de reconduction et mettre la base de données en ligne, appelez. `rdsadmin.complete_rollforward`

Pour plus d'informations sur la vérification de l'état d'achèvement du processus de report, voirrdsadmin.rollforward_status.

Exemples

L'exemple suivant met la TESTDB base de données en ligne :

```
db2 "call rdsadmin.complete_rollforward(
    ?,
    'TESTDB')"
```

rdsadmin.db2pd_command

Collecte des informations sur une base de données RDS pour DB2.

Syntaxe

```
db2 "call rdsadmin.db2pd_command('db2pd_cmd')"
```

Paramètres

Le paramètre d'entrée suivant est obligatoire :

db2pd_cmd

Nom de la db2pd commande que vous souhaitez exécuter. Le type de données est`varchar`.

Le paramètre doit commencer par un tiret. Pour obtenir la liste des paramètres, voir db2pd--Surveiller et résoudre les problèmes liés à la commande de base de données DB2 dans la documentation. IBM Db2

Les options suivantes ne sont pas prises en charge :

- `-addnode`
- `-alldatabases`
- `-alldbp`
- `-alldbs`
- `-allmembers`
- `-alm_in_memory`
- `-cfinfo`
- `-cfpool`
- `-command`
- `-dbpartitionnum`
- `-debug`
- `-dump`
- `-everything`
- `-file | -o`
- `-ha`
- `-interactive`
- `-member`
- `-pages`

> **Note**
>
> `-pages summary` est pris en charge.

- `-pdcollection`
- `-repeat`
- `-stack`

- `-totalmem`

  La `file` sous-option n'est pas prise en charge, `db2pd -db testdb -tcbstats file=tcbstat.out` par exemple.

  L'utilisation de l'`stacks`option n'est pas prise en charge, par exemple`db2pd -edus interval=5 top=10 stacks`.

Notes d'utilisation

Cette procédure stockée rassemble des informations qui peuvent aider à surveiller et à dépanner les bases de données RDS pour DB2.

La procédure stockée utilise l'IBM`db2pd`utilitaire pour exécuter différentes commandes. L'`db2pd`utilitaire nécessite une `SYSADM` autorisation, que l'utilisateur principal de RDS pour DB2 ne possède pas. Cependant, avec la procédure stockée Amazon RDS, l'utilisateur principal peut utiliser l'utilitaire pour exécuter diverses commandes. Pour plus d'informations sur cet utilitaire, consultez la section db2pd--Surveiller et dépanner la commande de base de données DB2 dans la documentation. IBM Db2

La sortie est limitée à un maximum de 2 Go.

Pour plus d'informations sur la vérification de l'état de la collecte d'informations sur la base de données, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : retour de la disponibilité d'une instance de base de données

L'exemple suivant renvoie le temps de disponibilité d'une instance de base de données RDS pour DB2 :

```
db2 "call rdsadmin.db2pd_command('-')"
```

Exemple 2 : retour du temps de disponibilité de la base de données

L'exemple suivant renvoie le temps de disponibilité d'une base de données appelée `TESTDB` :

```
db2 "call rdsadmin.db2pd_command('-db TESTDB -')"
```

Exemple 3 : retour de l'utilisation de la mémoire d'une instance de base de données

L'exemple suivant renvoie l'utilisation de la mémoire d'une instance de base de données RDS pour DB2 :

```
db2 "call rdsadmin.db2pd_command('-dbptnmem')"
```

Exemple 4 : renvoyer des ensembles de mémoire d'une instance de base de données et d'une base de données

L'exemple suivant renvoie les ensembles de mémoire d'une instance de base de données RDS pour DB2 et d'une base de données appelée : `TESTDB`

```
db2 "call rdsadmin.db2pd_command('-inst -db TESTDB -memsets')"
```

rdsadmin.force_application

Force les applications à quitter une base de données RDS pour DB2.

Syntaxe

```
db2 "call rdsadmin.force_application(
    ?,
    'applications')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

applications

Les applications que vous souhaitez forcer à quitter une base de données RDS pour DB2. Le type de données est`varchar`. Valeurs valides : `ALL` ou *application_handle*.

Séparez les noms de plusieurs applications par des virgules. Exemple : «*application_handle_1*, *application_handle_2* ».

Notes d'utilisation

Cette procédure stockée force toutes les applications à quitter une base de données afin que vous puissiez effectuer la maintenance.

La procédure stockée utilise la IBM `FORCE APPLICATION` commande. La `FORCE APPLICATION` commande nécessite`SYSADM`, ou `SYSCTRL` autorisation`SYSMAINT`, que l'utilisateur principal de RDS pour DB2 ne possède pas. Cependant, avec la procédure stockée Amazon RDS, l'utilisateur principal peut utiliser la commande. Pour plus d'informations, voir la commande FORCE APPLICATION dans la IBM Db2 documentation.

Pour plus d'informations sur la vérification de l'état du forçage des applications hors d'une base de données, consultezrdsadmin.get_task_status.

Exemples

Exemple 1 : Spécification de toutes les applications

L'exemple suivant force toutes les applications à quitter une base de données RDS pour DB2 :

```
db2 "call rdsadmin.force_application(
    ?,
    'ALL')"
```

Exemple 2 : Spécification de plusieurs applications

L'exemple suivant force les descripteurs 9991 d'application et la 1192 désactivation d'une base de données RDS pour DB2 : 8891

```
db2 "call rdsadmin.force_application(
    ?,
    '9991, 8891, 1192')"
```

rdsadmin.set_archive_log_retention

Configure la durée (en heures) de conservation des fichiers journaux d'archivage pour la base de données RDS pour DB2 spécifiée.

Syntaxe

```
db2 "call rdsadmin.set_archive_log_retention(
    ?,
    'database_name',
    'archive_log_retention_hours')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Les paramètres d'entrée suivants sont requis :

database_name

Nom de la base de données pour laquelle configurer la conservation des journaux d'archivage. Le type de données est`varchar`.

archive_log_retention_hours

Le nombre d'heures pendant lesquelles les fichiers journaux d'archivage sont conservés. Le type de données est`smallint`. La valeur par défaut est`0`, et le maximum est de 168 (7 jours).

Si la valeur est`0`, Amazon RDS ne conserve pas les fichiers journaux d'archivage.

Notes d'utilisation

Par défaut, RDS pour DB2 conserve les journaux pendant 5 minutes. Nous vous recommandons d'utiliser des outils de réplication tels que AWS DMS pour la capture des données de modification (CDC) ou IBM Q Replication de configurer la conservation des journaux dans ces outils pendant plus de 5 minutes.

Vous pouvez consulter le paramètre actuel de conservation des journaux d'archivage en appelantthe section called "rdsadmin.show_archive_log_retention".

Vous ne pouvez pas configurer le paramètre de conservation des journaux d'archivage sur la `rdsadmin` base de données.

Exemples

Exemple 1 : définition de la durée de rétention

L'exemple suivant définit la durée de conservation du journal d'archivage pour une base de données appelée TESTDB à 24 heures.

```
db2 "call rdsadmin.set_archive_log_retention(
    ?,
    'TESTDB',
    '24')"
```

Exemple 2 : désactivation de la durée de rétention

L'exemple suivant désactive la conservation des journaux d'archivage pour une base de données appeléeTESTDB.

```
db2 "call rdsadmin.set_archive_log_retention(
    ?,
    'TESTDB',
    '0')"
```

rdsadmin.show_archive_log_retention

Renvoie le paramètre actuel de conservation du journal d'archivage pour la base de données spécifiée.

Syntaxe

```
db2 "call rdsadmin.show_archive_log_retention(
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données pour laquelle le paramètre de conservation du journal d'archivage doit être affiché. Le type de données est`varchar`.

Exemples

L'exemple suivant montre le paramètre de conservation des journaux d'archivage pour une base de données appeléeTESTDB.

```
db2 "call rdsadmin.show_archive_log_retention(
    ?
    'TESTDB')"
```

rdsadmin.list_archive_log_information

Renvoie des informations sur les fichiers journaux d'archive, tels que la taille, la date et l'heure de création, ainsi que le nom des fichiers journaux individuels pour la base de données spécifiée. Il renvoie également la quantité totale de stockage utilisée par les fichiers journaux de la base de données.

Syntaxe

```
db2 "call rdsadmin.list_archive_log_information(
    ?,
    'database_name')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre accepte uniquement?.

Le paramètre d'entrée suivant est obligatoire :

database_name

Nom de la base de données pour laquelle répertorier les informations du journal d'archivage. Le type de données est`varchar`.

Exemples

L'exemple suivant renvoie les informations du journal d'archivage d'une base de données appeléeTESTDB.

```
db2 "call rdsadmin.list_archive_log_information(
    ?
    'TESTDB')"
```

Procédures stockées pour l'accès au stockage RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent l'accès au stockage RDS pour les bases de données DB2 qui utilisent Amazon S3 pour la migration des données. Pour de plus amples informations, veuillez consulter the section called “Migration avec Amazon S3”.

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.catalog_storage_access
- rdsadmin.uncatalog_storage_access

rdsadmin.catalog_storage_access

Catalogue un alias de stockage permettant d'accéder à un compartiment Amazon S3 contenant des fichiers de données DB2.

Syntaxe

```
db2 "call rdsadmin.catalog_storage_access(
    ?,
    'alias',
    's3_bucket_name',
    'grantee_type',
    'grantee'
    )"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Le type de données est. `varchar`

Les paramètres d'entrée suivants sont requis :

alias

Le nom d'alias permettant d'accéder au stockage à distance dans un compartiment Amazon S3. Le type de données est. `varchar`

s3_bucket_name

Le nom du compartiment Amazon S3 dans lequel se trouvent vos données. Le type de données est`varchar`.

grantee_type

Type de bénéficiaire devant recevoir l'autorisation. Le type de données est`varchar`. Valeurs valides : `USER`, `GROUP`.

grantee

L'utilisateur ou le groupe à qui l'autorisation doit être accordée. Le type de données est`varchar`.

Notes d'utilisation

Amazon RDS inclut l'alias catalogué dans le IAM rôle que vous avez ajouté à votre instance de base de données RDS for Db2. Si vous supprimez le IAM rôle de votre instance de base de données, Amazon RDS supprime l'alias. Pour de plus amples informations, veuillez consulter the section called "Migration avec Amazon S3".

Pour plus d'informations sur la vérification de l'état du catalogage de votre alias, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant enregistre un alias appelé`SAMPLE`. L'utilisateur `jorge_souza` est autorisé à accéder au compartiment Amazon S3 appelé`amzn-s3-demo-bucket`.

```
db2 "call rdsadmin.catalog_storage_access(
    ?,
    'SAMPLE',
    'amzn-s3-demo-bucket',
    'USER',
    'jorge_souza')"
```

rdsadmin.uncatalog_storage_access

Supprime un alias d'accès au stockage.

Syntaxe

```
db2 "call rdsadmin.uncatalog_storage_access(
    ?,
    'alias')"
```

Paramètres

Le paramètre de sortie suivant est requis :

?

Marqueur de paramètre qui génère un message d'erreur. Le type de données est. `varchar`

Le paramètre d'entrée suivant est obligatoire :

`alias`

Nom de l'alias de stockage à supprimer. Le type de données est. `varchar`

Notes d’utilisation

Pour plus d'informations sur la vérification de l'état de suppression de votre alias, consultezrdsadmin.get_task_status.

Exemples

L'exemple suivant supprime un alias appelé`SAMPLE`. Cet alias ne permet plus d'accéder au compartiment Amazon S3 auquel il était associé.

```
db2 "call rdsadmin.uncatalog_storage_access(
    ?,
    'SAMPLE')"
```

Procédures stockées pour les espaces de table pour RDS pour Db2

Les procédures stockées intégrées décrites dans cette rubrique gèrent les espaces de table pour les bases de données Amazon RDS for DB2. Pour exécuter ces procédures, l'utilisateur principal doit d'abord se connecter à la `rdsadmin` base de données.

Ces procédures stockées sont utilisées dans diverses tâches. Cette liste n'est pas exhaustive.

- Tâches courantes pour les tablespaces
- Génération de rapports de performance
- Copier les métadonnées d'une base de données avec db2look
- Création d'une base de données de référentiel pour IBM Db2 Data Management Console

Reportez-vous aux procédures stockées intégrées suivantes pour obtenir des informations sur leur syntaxe, leurs paramètres, leurs notes d'utilisation et des exemples.

Procédures stockées

- rdsadmin.create_tablespace
- rdsadmin.alter_tablespace
- rdsadmin.rename_tablespace
- rdsadmin.drop_tablespace

rdsadmin.create_tablespace

Crée un tablespace.

Syntaxe

```
db2 "call rdsadmin.create_tablespace(
    'database_name',
    'tablespace_name',
    'buffer_pool_name',
    tablespace_page_size,
    tablespace_initial_size,
    tablespace_increase_size,
    'tablespace_type',
    'tablespace_prefetch_size')"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données dans laquelle créer le tablespace. Le type de données est`varchar`.

tablespace_name

Nom du tablespace à créer. Le type de données est`varchar`.

Le nom du tablespace est soumis aux restrictions suivantes :

- Il ne peut pas être identique au nom d'un tablespace existant dans cette base de données.
- Il ne peut contenir que les caractères`_$#@a-zA-Z0-9`.
- Cela ne peut pas commencer par _ ou$.
- Ça ne peut pas commencer parSYS.

Les paramètres suivants sont facultatifs :

buffer_pool_name

Nom du pool de mémoire tampon à attribuer au tablespace. Le type de données est`varchar`. La valeur par défaut est une chaîne vide.

> **Important**
>
> Vous devez déjà disposer d'un pool de mémoire tampon de même taille de page à associer au tablespace.

tablespace_page_size

Taille de page du tablespace en octets. Le type de données est`integer`. Valeurs valides: `4096`, `8192`, `16384`, `32768`. La taille de page par défaut est celle utilisée lorsque vous avez créé la base de données en appelantrdsadmin.create_database.

> **Important**
>
> Amazon RDS prend en charge l'atomicité d'écriture pour les pages 4 KiB, 8 KiB et 16 KiB. En revanche, les pages de 32 Kio risquent d'être déchirées ou d'écrire des données

partielles sur le bureau. Si vous utilisez des pages 32 KiB, nous vous recommandons d'activer la point-in-time restauration et les sauvegardes automatisées. Sinon, vous risquez de ne pas pouvoir récupérer des pages déchirées. Pour plus d'informations, consultez the section called "Présentation des sauvegardes" et the section called "Point-in-time rétablissement".

tablespace_initial_size

Taille initiale du tablespace en kilo-octets (Ko). Le type de données est`integer`. Valeurs valides : 48 ou supérieures. La valeur par défaut est NULL.

Si vous ne définissez aucune valeur, Db2 définit une valeur appropriée pour vous.

> **Note**
>
> Ce paramètre ne s'applique pas aux espaces disque logiques temporaires car le système gère les espaces disque logiques temporaires.

tablespace_increase_size

Pourcentage d'augmentation du tablespace lorsqu'il est plein. Le type de données est`integer`. Valeurs valides : 1 —`100`. La valeur par défaut est NULL.

Si vous ne définissez aucune valeur, Db2 définit une valeur appropriée pour vous.

> **Note**
>
> Ce paramètre ne s'applique pas aux espaces disque logiques temporaires car le système gère les espaces disque logiques temporaires.

tablespace_type

Type du tablespace. Le type de données est`char`. Valeurs valides : `U` (pour les données utilisateur), `T` (pour les données temporaires de l'utilisateur) ou `S` (pour les données temporaires du système). L'argument par défaut est `U`.

tablespace_prefetch_size

Taille de page de prélecture du tablespace. Le type de données est`char`. Valeurs valides : `AUTOMATIC` (sans distinction majuscules/minuscules) ou entiers positifs non nuls inférieurs ou égaux à 32767.

Notes d'utilisation

RDS pour DB2 crée toujours une base de données volumineuse.

Pour plus d'informations sur la vérification de l'état de création d'un tablespace, consultez. rdsadmin.get_task_status

Exemples

Exemple 1 : création d'un espace disque logique et attribution d'un pool de mémoire tampon

L'exemple suivant crée un tablespace appelé SP8 et affecte un pool de mémoire tampon appelé BP8 pour une base de données appelée. TESTDB Le tablespace a une taille de page initiale de 4 096 octets, un tablespace initial de 1 000 Ko et une augmentation de taille de table définie à 50 %.

```
db2 "call rdsadmin.create_tablespace(
    'TESTDB',
    'SP8',
    'BP8',
    4096,
    1000,
    50)"
```

Exemple 2 : création d'un espace disque logique temporaire et attribution d'un pool de mémoire tampon

L'exemple suivant crée un tablespace temporaire appelé. SP8 Il attribue un pool de mémoire tampon appelé BP8 d'une taille de 8 KiB pour une base de données appeléeTESTDB.

```
db2 "call rdsadmin.create_tablespace(
    'TESTDB',
    'SP8',
    'BP8',
    8192,
    NULL,
```

```
    NULL,
    'T')"
```

Exemple 3 : création d'un espace disque logique et attribution d'une taille de page de prélecture

L'exemple suivant crée un tablespace appelé `SP8` pour une base de données appelée. `TESTDB` L'espace disque logique a une taille d'augmentation initiale de l'espace disque logique `50` et une taille de page de prélecture de. `800`

```
db2 "call rdsadmin.create_tablespace(
    'TESTDB',
    'SP8',
    NULL,
    NULL,
    NULL,
    50,
    NULL,
    '800')"
```

rdsadmin.alter_tablespace

Modifie un tablespace.

Syntaxe

```
db2 "call rdsadmin.alter_tablespace(
    'database_name',
    'tablespace_name',
    'buffer_pool_name',
    tablespace_increase_size,
    'max_size',
    'reduce_max',
    'reduce_stop',
    'reduce_value',
    'lower_high_water',
    'lower_high_water_stop',
    'switch_online',
    'tablespace_prefetch_size')"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données qui utilise le tablespace. Le type de données est`varchar`.

tablespace_name

Nom du tablespace à modifier. Le type de données est`varchar`.

Les paramètres suivants sont facultatifs :

buffer_pool_name

Nom du pool de mémoire tampon à attribuer au tablespace. Le type de données est`varchar`. La valeur par défaut est une chaîne vide.

> **Important**
>
> Vous devez déjà disposer d'un pool de mémoire tampon de même taille de page à associer au tablespace.

tablespace_increase_size

Pourcentage d'augmentation du tablespace lorsqu'il est plein. Le type de données est`integer`. Valeurs valides : `1` —`100`. L'argument par défaut est `0`.

max_size

Taille maximale du tablespace. Le type de données est`varchar`. Valeurs valides : *integer* `K` `M` | | `G` ou`NONE`. L'argument par défaut est `NONE`.

reduce_max

Spécifie s'il faut réduire le maximum de la limite maximale. Le type de données est`char`. L'argument par défaut est `N`.

reduce_stop

Spécifie s'il faut interrompre une `reduce_value` commande `reduce_max` ou une commande précédente. Le type de données est`char`. L'argument par défaut est `N`.

reduce_value

Le nombre ou le pourcentage de réduction du point culminant de l'espace disque logique de. Le type de données est`varchar`. Valeurs valides : *integer* `K` `M` | | `G` ou `1` —`100`. L'argument par défaut est `N`.

lower_high_water

Spécifie s'il faut exécuter la `ALTER TABLESPACE LOWER HIGH WATER MARK` commande. Le type de données est`char`. L'argument par défaut est `N`.

lower_high_water_stop

Spécifie s'il faut exécuter la `ALTER TABLESPACE LOWER HIGH WATER MARK STOP` commande. Le type de données est`char`. L'argument par défaut est `N`.

switch_online

Spécifie s'il faut exécuter la `ALTER TABLESPACE SWITCH ONLINE` commande. Le type de données est`char`. L'argument par défaut est `N`.

tablespace_prefetch_size

Taille de page de prélecture du tablespace. Le type de données est`char`. Valeurs valides : `AUTOMATIC` (sans distinction majuscules/minuscules) ou entiers positifs non nuls inférieurs ou égaux à 32767.

> **Note**
>
> Ce paramètre ne fonctionne qu'avec `buffer_pool_name`,`table_increase_size`,`max_size`, et`switch_online`. Cela ne fonctionne pas avec `reduce_max`,`reduce_stop`,`reduce_value`,`lower_high_water`, et`lower_high_water_stop`.

Notes d'utilisation

Avant d'appeler la procédure stockée, prenez en compte les points suivants :

- La procédure `rdsadmin.alter_tablespace` stockée ne fonctionnera pas sur un tablespace `tablespace_type` défini sur `T` pour les données temporaires de l'utilisateur.
- Les paramètres facultatifs `reduce_max`,`reduce_stop`,`reduce_value`,, `lower_high_water`,`lower_high_water_stop`, et s'`switch_online`excluent mutuellement.

Vous ne pouvez pas les combiner avec d'autres paramètres facultatifs`buffer_pool_name`, tels que ceux de la `rdsadmin.alter_tablespace` commande. Pour de plus amples informations, veuillez consulter Statement not valid.

Pour plus d'informations sur la vérification de l'état de modification d'un espace disque logique, consultez. rdsadmin.get_task_status

Pour les messages d'erreur renvoyés lors de l'appel de procédures stockées, consultezthe section called “Erreurs de procédure stockée”.

Exemples

Exemple 1 : Abaisser le niveau du haut du niveau des eaux

L'exemple suivant modifie un tablespace appelé SP8 et affecte un pool de mémoire tampon appelé BP8 pour une base de données appelée TESTDB pour abaisser le seuil maximum.

```
db2 "call rdsadmin.alter_tablespace(
    'TESTDB',
    'SP8',
    'BP8',
    NULL,
    NULL,
    'Y')"
```

Exemple 2 : Réduire le niveau des hautes eaux

L'exemple suivant exécute la `REDUCE MAX` commande sur un tablespace appelé `TBSP_TEST` dans la base de données. `TESTDB`

```
db2 "call rdsadmin.alter_tablespace(
    'TESTDB',
    'TBSP_TEST',
    NULL,
    NULL,
    NULL,
    'Y')"
```

Exemple 3 : Interruption de commandes pour réduire le niveau de saturation

L'exemple suivant exécute la `REDUCE STOP` commande sur un tablespace appelé `TBSP_TEST` dans la base de données. `TESTDB`

```
db2 "call rdsadmin.alter_tablespace(
    'TESTDB',
    'TBSP_TEST',
    NULL,
    NULL,
    NULL,
    NULL,
    'Y')"
```

Exemple 4 : Modification de la taille de page de prélecture existante

L'exemple suivant exécute la `ALTER TABLESPACE SWITCH ONLINE` commande sur un tablespace appelé `TSBP_TEST` et modifie la taille de page de prélecture existante en. `64`

```
db2 "call rdsadmin.alter_tablespace(
    'TESTDB',
    'TBSP_TEST',
    NULL,
    NULL,
    NULL,
    NULL,
    NULL,
    NULL,
    NULL,
    NULL,
    'Y',
    '64')"
```

rdsadmin.rename_tablespace

Renomme un tablespace.

Syntaxe

```
db2 "call rdsadmin.rename_tablespace(
    ?,
    'database_name',
    'source_tablespace_name',
```

```
    'target_tablespace_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

?

Marqueur de paramètre qui génère un message d'erreur. Ce paramètre n'accepte que ?

database_name

Nom de la base de données à laquelle appartient le tablespace. Le type de données est`varchar`.

source_tablespace_name

Nom du tablespace à renommer. Le type de données est`varchar`.

target_tablespace_name

Le nouveau nom du tablespace. Le type de données est`varchar`.

Le nouveau nom comporte les restrictions suivantes :

- Il ne peut pas être identique au nom d'un tablespace existant.
- Il ne peut contenir que les caractères`_$#@a-zA-Z0-9`.
- Cela ne peut pas commencer par _ ou$.
- Ça ne peut pas commencer par`SYS`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état du changement de nom d'un espace disque logique, consultez. rdsadmin.get_task_status

Vous ne pouvez pas renommer les tablespaces appartenant à la base de données. `rdsadmin`

Exemples

L'exemple suivant renomme un tablespace appelé `SP9` dans une base `SP8` de données appelée. `TESTDB`

```
db2 "call rdsadmin.rename_tablespace(
```

```
    ?,
    'TESTDB',
    'SP8',
    'SP9')"
```

rdsadmin.drop_tablespace

Supprime un tablespace.

Syntaxe

```
db2 "call rdsadmin.drop_tablespace(
    'database_name',
    'tablespace_name')"
```

Paramètres

Les paramètres suivants sont obligatoires :

database_name

Nom de la base de données à laquelle appartient le tablespace. Le type de données est`varchar`.

tablespace_name

Nom du tablespace à supprimer. Le type de données est`varchar`.

Notes d'utilisation

Pour plus d'informations sur la vérification de l'état de suppression d'un tablespace, consultez. rdsadmin.get_task_status

Exemples

L'exemple suivant supprime un tablespace appelé SP8 depuis une base de données appelée. TESTDB

```
db2 "call rdsadmin.drop_tablespace(
    'TESTDB',
    'SP8')"
```

Référence RDS des fonctions définies par l'utilisateur Amazon pour Db2

Les fonctions définies par l'utilisateur suivantes sont disponibles pour les instances Amazon RDS DB exécutant le moteur Db2.

Rubriques

- rdsadmin.get_task_status
- rdsadmin.list_databases

rdsadmin.get_task_status

Renvoie le statut d'une tâche.

Syntaxe

```
db2 "select task_id, task_type, database_name, lifecycle,
    varchar(bson_to_json(task_input_params), 500) as task_params,
    cast(task_output as varchar(500)) as task_output
    from table(rdsadmin.get_task_status(task_id,'database_name','task_type'))"
```

Paramètres

Les paramètres suivants sont facultatifs. Si vous ne fournissez aucun paramètre, la fonction définie par l'utilisateur renvoie le statut de toutes les tâches pour toutes les bases de données. Amazon RDS conserve l'historique des tâches pendant 35 jours.

task_id

ID de la tâche en cours d'exécution. Cet identifiant est renvoyé lorsque vous exécutez une tâche. Par défaut: `0`.

database_name

Nom de la base de données pour laquelle la tâche est exécutée.

task_type

Type de tâche à interroger. Valeurs valides :ADD_GROUPS,ADD_USER,ALTER_BUFFERPOOL,,ALTER_TABLESPACE,CHANGE_PASSWORD,COMP

Notes d’utilisation

Vous pouvez utiliser la fonction `rdsadmin.get_task_status` définie par l'utilisateur pour vérifier l'état des tâches suivantes pour Amazon RDS for Db2. Cette liste n'est pas exhaustive.

- Création, modification ou suppression d'un pool de mémoire tampon
- Création, modification ou suppression d'un tablespace
- Création ou suppression d'une base de données
- Restauration d'une sauvegarde de base de données depuis Amazon S3
- Prolongation progressive des journaux de base de données depuis Amazon S3

Exemples

L'exemple suivant affiche les colonnes renvoyées lors de `rdsadmin.get_task_status` l'appel.

```
db2 "describe select * from table(rdsadmin.get_task_status())"
```

L'exemple suivant répertorie le statut de toutes les tâches.

```
db2 "select task_id, task_type, database_name, lifecycle,
    varchar(bson_to_json(task_input_params), 500) as task_params,
    cast(task_output as varchar(500)) as task_output
    from table(rdsadmin.get_task_status(null,null,null))"
```

L'exemple suivant répertorie le statut d'une tâche spécifique.

```
db2 "select task_id, task_type, database_name,
    varchar(bson_to_json(task_input_params), 500) as task_params
    from table(rdsadmin.get_task_status(1,null,null))"
```

L'exemple suivant répertorie le statut d'une tâche et d'une base de données spécifiques.

```
db2 "select task_id, task_type, database_name,
    varchar(bson_to_json(task_input_params), 500) as task_params
    from table(rdsadmin.get_task_status(2,'SAMPLE',null))"
```

L'exemple suivant répertorie le statut de toutes les `ADD_GROUPS` tâches.

```
db2 "select task_id, task_type, database_name,
    varchar(bson_to_json(task_input_params), 500) as task_params
    from table(rdsadmin.get_task_status(null,null,'add_groups'))"
```

L'exemple suivant répertorie le statut de toutes les tâches pour une base de données spécifique.

```
db2 "select task_id, task_type, database_name,
    varchar(bson_to_json(task_input_params), 500) as task_params
    from table(rdsadmin.get_task_status(null,'testdb', null))"
```

L'exemple suivant affiche les JSON valeurs sous forme de colonnes.

```
db2 "select varchar(r.task_type,25) as task_type, varchar(r.lifecycle,10) as lifecycle,
 r.created_at, u.* from
    table(rdsadmin.get_task_status(null,null,'restore_db')) as r,
 json_table(r.task_input_params, 'strict $' columns(s3_prefix varchar(500)
    null on empty, s3_bucket_name varchar(500) null on empty) error on error ) as U"
```

Réponse

La fonction `rdsadmin.get_task_status` définie par l'utilisateur renvoie les colonnes suivantes :

`TASK_ID`

ID de la tâche

`TASK_TYPE`

Cela dépend des paramètres d'entrée.

- `ADD_GROUPS`— Ajoute des groupes.
- `ADD_USER`— Ajoute un utilisateur.
- `ALTER_BUFFERPOOL`— Modifie un pool tampon.
- `ALTER_TABLESPACE`— Modifie un tablespace.
- `CHANGE_PASSWORD` — Modifie le mot de passe d'un utilisateur.
- `COMPLETE_ROLLFORWARD`— Termine une `rdsadmin.rollforward_database` tâche et active une base de données.
- `CREATE_BUFFERPOOL`— Crée un pool de mémoire tampon.
- `CREATE_DATABASE`— Crée une base de données.

- CREATE_ROLE— Crée un rôle DB2 pour un utilisateur.
- CREATE_TABLESPACE— Crée un tablespace.
- DROP_BUFFERPOOL— Supprime un pool de mémoire tampon.
- DROP_DATABASE— Supprime une base de données.
- DROP_TABLESPACE— Supprime un tablespace.
- LIST_USERS— Liste tous les utilisateurs.
- REMOVE_GROUPS— Supprime des groupes.
- REMOVE_USER— Supprime un utilisateur.
- RESTORE_DB— Restaure une base de données complète.
- ROLLFORWARD_DB_LOG— Exécute une `rdsadmin.rollforward_database` tâche sur les journaux de base de données.
- ROLLFORWARD_STATUS — Renvoie le statut d'une `rdsadmin.rollforward_database` tâche.
- UPDATE_DB_PARAM— Met à jour les paramètres des données.

`DATABASE_NAME`

Nom de la base de données à laquelle la tâche est associée.

`COMPLETED_WORK_BYTES`

Nombre d'octets restaurés par la tâche.

`DURATION_MINS`

Le temps nécessaire pour terminer la tâche.

`LIFECYCLE`

État de la tâche. Statuts possibles :

- CREATED— Une fois qu'une tâche est soumise à AmazonRDS, Amazon RDS définit le statut surCREATED.
- IN_PROGRESS— Après le démarrage d'une tâche, Amazon RDS définit le statut surIN_PROGRESS. Le passage d'un statut à peut prendre jusqu'à 5 minutesIN_PROGRESS. CREATED
- SUCCESS— Une fois qu'une tâche est terminée, Amazon RDS définit le statut surSUCCESS.
- ERROR— Si une tâche de restauration échoue, Amazon RDS définit le statut surERROR. Pour plus d'informations sur cette erreur, consultezTASK_OUPUT.

`CREATED_BY`

Celui `authid` qui a créé la commande.

`CREATED_AT`

Date et heure de création de la tâche.

`LAST_UPDATED_AT`

Date et heure de dernière mise à jour de la tâche.

`TASK_INPUT_PARAMS`

Les paramètres varient en fonction du type de tâche. Tous les paramètres d'entrée sont représentés sous forme d'JSONobjet. Par exemple, les JSON clés de la `RESTORE_DB` tâche sont les suivantes :

- `DBNAME`
- `RESTORE_TIMESTAMP`
- `S3_BUCKET_NAME`
- `S3_PREFIX`

`TASK_OUTPUT`

Informations supplémentaires sur la tâche. Si une erreur se produit lors de la restauration native, cette colonne contient des informations sur l'erreur.

Exemples de réponses

L'exemple de réponse suivant montre qu'une base de données appelée `TESTJP` a été créée avec succès. Pour plus d'informations, consultez la procédure the section called “rdsadmin.create_database” stockée.

```
`1 SUCCESS CREATE_DATABASE RDSDB 2023-10-24-18.32.44.962689 2023-10-24-18.34.50.038523
 1 TESTJP { "CODESET" : "IBM-437", "TERRITORY" : "JP", "COLLATION" : "SYSTEM",
 "AUTOCONFIGURE_CMD" : "", "PAGESIZE" : 4096 }
2023-10-24-18.33.30.079048 Task execution has started.

2023-10-24-18.34.50.038523 Task execution has completed successfully`.
```

L'exemple de réponse suivant explique pourquoi la suppression d'une base de données a échoué. Pour plus d'informations, consultez la procédure the section called “rdsadmin.drop_database” stockée.

```
1 ERROR DROP_DATABASE RDSDB 2023-10-10-16.33.03.744122 2023-10-10-16.33.30.143797 -
 2023-10-10-16.33.30.098857 Task execution has started.
2023-10-10-16.33.30.143797 Caught exception during executing task id 1, Aborting task.
Reason Dropping database created via rds CreateDBInstance api is not allowed.
Only database created using rdsadmin.create_database can be dropped
```

L'exemple de réponse suivant montre la restauration réussie d'une base de données. Pour plus d'informations, consultez la procédure the section called “rdsadmin.restore_database” stockée.

```
1 RESTORE_DB  SAMPLE  SUCCESS

{ "S3_BUCKET_NAME" : "amzn-s3-demo-bucket", "S3_PREFIX" :
 "SAMPLE.0.rdsdb3.DBPART000.20230413183211.001", "RESTORE_TIMESTAMP" :
 "20230413183211", "BACKUP_TYPE" : "offline" }




2023-11-06-18.31.03.115795 Task execution has started.
2023-11-06-18.31.04.300231 Preparing to download
2023-11-06-18.31.08.368827 Download complete. Starting Restore
2023-11-06-18.33.13.891356 Task Completed Successfully
```

rdsadmin.list_databases

Renvoie une liste de toutes les bases de données exécutées sur une instance de base de données RDS for DB2.

Syntaxe

```
db2 "select * from table(rdsadmin.list_databases())"
```

Notes d’utilisation

Cette fonction définie par l'utilisateur ne précise pas si les bases de données sont activées ou désactivées.

Si vos bases de données ne figurent pas dans la liste, appelez la fonction the section called “rdsadmin.get_task_status” définie par l'utilisateur et recherchez les messages d'erreur.

Réponse

La fonction `rdsadmin.list_databases` définie par l'utilisateur renvoie les colonnes suivantes :

`DATABASE_NAME`

Nom d'une base de données.

`CREATE_TIME`

Date et heure de création de la base de données.

Exemples de réponses

L'exemple de réponse suivant montre une liste de bases de données et l'heure à laquelle elles ont été créées. `rdsadmin`est une base de données RDS gérée par Amazon et qui apparaît toujours dans la sortie.

```
DATABASE_NAME   CREATE_TIME
--------------- --------------------------
rdsadmin        2024-10-22-03.37.48.535671
TEST            2024-10-22-03.39.36.818679
TEST1           2024-10-22-03.57.15.218009
TEST2           2024-10-22-03.59.28.029556
```

Résolution des problèmes liés à Amazon RDS pour DB2

Le contenu suivant peut vous aider à résoudre les problèmes que vous rencontrez avec RDS pour DB2.

Pour plus d'informations sur les problèmes généraux de résolution des problèmes liés à Amazon RDS, consultezDépannage d'Amazon RDS.

Rubriques

- Erreur de connexion de base de données
- I/O Erreur de fichier
- Erreurs de procédure stockée

Erreur de connexion de base de données

Le message d'erreur suivant indique qu'une base de données n'a pas pu se connecter car le serveur ne dispose pas de suffisamment de mémoire.

```
SQL1643C The database manager failed to allocate shared memory because the
database manager instance memory limit has been reached.
```

Augmentez la mémoire de votre instance de base de données, puis réessayez de vous connecter à votre base de données. Pour plus d'informations sur l'utilisation de la mémoire et les recommandations relatives aux bases de données, consultezthe section called “Plusieurs bases de données DB2”. Pour plus d'informations sur la mise à jour de la mémoire d'une base de données RDS pour DB2, consultez. the section called “rdsadmin.update_db_param”

I/O Erreur de fichier

Vous pouvez rencontrer une I/O erreur de fichier pour différentes raisons, par exemple lorsque vous utilisez la `LOAD` commande ou que vous appelez la procédure `rdsadmin.restore_database` stockée.

Dans cet exemple, vous exécutez la `LOAD` commande suivante.

```
db2 "call sysproc.admin_cmd('load from "DB2REMOTE://s3test//public/datapump/t6.del" of
 del lobs from "DB2REMOTE://s3test/public/datapump/" modified by lobsinfile MESSAGES ON
 SERVER insert INTO RDSDB.t6 nonrecoverable ')"
```

La LOAD commande renvoie le message suivant :

```
 Result set 1
 --------------

 ROWS_READ            ROWS_SKIPPED         ROWS_LOADED          ROWS_REJECTED
ROWS_DELETED         ROWS_COMMITTED       ROWS_PARTITIONED     NUM_AGENTINFO_ENTRIES
MSG_RETRIEVAL




MSG_REMOVAL
 -------------------- -------------------- -------------------- --------------------
-------------------- -------------------- -------------------- --------------------
--------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------------------
                    -                    -                    -                    -
                   -                    -                    -                    -
SELECT SQLCODE, MSG FROM TABLE(SYSPROC.ADMIN_GET_MSGS('1594987316_285548770')) AS MSG




                                                                                CALL
SYSPROC.ADMIN_REMOVE_MSGS('1594987316_285548770')

 1 record(s) selected.

 Return Status = 0

SQL20397W  Routine "SYSPROC.ADMIN_CMD" execution has completed, but at least
one error, "SQL1652", was encountered during the execution. More information
is available.  SQLSTATE=01H52
```

Pour afficher le message d'erreur, vous devez exécuter la commande SQL comme suggéré dans la réponse précédente. `SELECT SQLCODE, MSG FROM TABLE(SYSPROC.ADMIN_GET_MSGS('1594987316_285548770')) AS MSG`renvoie le message suivant :

```
SQLCODE   MSG
---------
 ------------------------------------------------------------------------------------------
SQL2025N  An I/O error occurred. Error code “438”. Media on which this error occurred:
 “DB2REMOTE://s3test//public/datapump/t6.del”

SQL3500W The utility is beginning the LOAD phase at time “07/05/2024 21:21:48.082954”

SQL1652N File I/O error occurred
```

Les journaux de diagnostic Db2 contiennent un fichier journal similaire au suivant :

```
2024-07-05-21.20.09.440609+000 I1191321E864          LEVEL: Error
PID     : 2710                 TID : 139619509200640 PROC : db2sysc 0
INSTANCE: rdsdb                NODE : 000            DB   : NTP
APPHDL  : 0-12180              APPID: xxx.xx.x.xxx.xxxxx.xxxxxxxxxxxx
UOWID   : 5                    ACTID: 1
AUTHID  : ADMIN                HOSTNAME: ip-xx-xx-x-xx
EDUID   : 147                  EDUNAME: db2lmr 0
FUNCTION: DB2 UDB, oper system services, sqloS3Client_GetObjectInfo, probe:219
MESSAGE : ZRC=0x870F01B6=-2029059658=SQLO_FAILED
          "An unexpected error is encountered"
DATA #1 : String, 29 bytes
S3:HeadObject request failed.
DATA #2 : signed integer, 4 bytes
99
DATA #3 : String, 0 bytes
Object not dumped: Address: 0x00007EFC08A9AE38 Size: 0 Reason: Zero-length data
DATA #4 : String, 33 bytes
curlCode: 28, Timeout was reached
```

Cette I/O erreur de fichier peut être le résultat de différents scénarios. Par exemple, le VPC associé au groupe de sécurité utilisé pour créer votre instance de base de données RDS pour DB2 peut ne pas disposer d'un point de terminaison de passerelle Amazon S3. Ce point de terminaison est essentiel pour permettre à RDS pour Db2 d'accéder à Amazon S3. Si votre instance de base de données RDS pour DB2 se trouve dans des sous-réseaux privés, un point de terminaison de passerelle Amazon S3 est requis. Vous pouvez spécifier si votre instance de base de données utilise des sous-réseaux privés ou publics en configurant des groupes de sous-réseaux Amazon RDS. Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.

Rubriques

- Étape 1 : créer un point de terminaison de passerelle VPC pour Amazon S3
- Étape 2 : Vérifiez que le point de terminaison de votre passerelle VPC pour Amazon S3 existe

Étape 1 : créer un point de terminaison de passerelle VPC pour Amazon S3

Pour que votre instance de base de données RDS for Db2 interagisse avec Amazon S3, créez un VPC, puis un point de terminaison de passerelle Amazon S3 que les sous-réseaux privés pourront utiliser.

Pour créer un point de terminaison de passerelle VPC pour S3

1. Créez un VPC. Pour plus d'informations, consultez la section Créer un VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.
2. Créez un point de terminaison Amazon S3 pour les sous-réseaux privés à utiliser. Pour plus d'informations, consultez la section Points de terminaison de la passerelle dans le AWS PrivateLink Guide.

Étape 2 : Vérifiez que le point de terminaison de votre passerelle VPC pour Amazon S3 existe

Vérifiez que vous avez correctement créé un point de terminaison de passerelle Amazon S3 à l'aide du AWS Management Console ou du AWS CLI.

console

Pour confirmer un point de terminaison d'une passerelle Amazon S3

1. Connectez-vous à la console Amazon VPC AWS Management Console et ouvrez-la à l'adresse. https://console.aws.amazon.com/vpc
2. Dans le coin supérieur droit de la console, choisissez celui Région AWS de votre VPC.
3. Sélectionnez le VPC que vous avez créé.
4. Dans l'onglet Carte des ressources, sous Connexions réseau, vérifiez qu'un point de terminaison de passerelle Amazon S3 est répertorié.

AWS CLI

Pour confirmer un point de terminaison de passerelle Amazon S3, exécutez la describe-vpc-endpointscommande. Dans l'exemple suivant, remplacez *vpc_id* par l'ID VPC, par votre et *region* par le Région AWS nom *profile* de votre profil.

Pour LinuxmacOS, ou Unix :

```
aws ec2 describe-vpc-endpoints \
    --filters "Name=vpc-id,Values=$vpc_id" \
    "Name=service-name,\
    Values=com.amazonaws.${region}.s3" \
    --region $region --profile=$profile \
    --query "VpcEndpoints[*].VpcEndpointId" --output text
```

Dans Windows :

```
aws ec2 describe-vpc-endpoints ^
    --filters "Name=vpc-id,Values=$vpc_id" ^
    "Name=service-name,^
    Values=com.amazonaws.${region}.s3" ^
    --region $region --profile=$profile ^
    --query "VpcEndpoints[*].VpcEndpointId" --output text
```

Cette commande produit une sortie similaire à l'exemple suivant s'il existe un point de terminaison de passerelle Amazon S3.

```
[
    "vpce-0ea810434ff0b97e4"
]
```

Cette commande produit un résultat similaire à celui de l'exemple suivant s'il n'existe pas de point de terminaison de passerelle Amazon S3.

```
[]
```

Si aucun point de terminaison de passerelle Amazon S3 n'est répertorié, alorsÉtape 1 : créer un point de terminaison de passerelle VPC pour Amazon S3.

Erreurs de procédure stockée

Cette section décrit les différentes erreurs renvoyées lors de l'appel de procédures stockées et explique comment les résoudre.

Catégorie	Erreurs de procédure stockée
Bases de données	Erreurs rdsadmin.activate_database
Bases de données	Erreurs rdsadmin.create_database
Bases de données	Erreurs rdsadmin.deactivate_database
Bases de données	Erreurs rdsadmin.drop_database
Bases de données	Erreurs rdsadmin.reactivate_database
Bases de données	Erreurs rdsadmin.restore_database
Bases de données	Erreurs rdsadmin.update_db_param
Espaces de table	Erreurs rdsadmin.alter_tablespace

Erreurs rdsadmin.activate_database

Les erreurs suivantes peuvent se produire lorsque vous appelez la procédure the section called “rdsadmin.activate_database” stockée.

Erreur	Message d’erreur
Failed to allocate shared memory	`SQL1643C The database manager failed to allocate shared memory because the database manager instance memory limit has been reached.`
Unable to activate because of running processes	`The database can’t be activated because it's in the process of being created or restored.`

Impossible d'allouer de la mémoire partagée

Le message d'erreur suivant indique que la procédure stockée n'a pas réussi à activer une base de données car l'instance de base de données ne dispose pas de suffisamment de mémoire.

```
SQL1643C The database manager failed to allocate shared memory because the database
 manager instance memory limit has been reached.
```

Augmentez la mémoire de votre instance de base de données, puis appelez à nouveau la procédure `rdsadmin.activate_database` stockée. Pour plus d'informations sur l'utilisation de la mémoire et les recommandations relatives aux bases de données, consultezthe section called “Plusieurs bases de données DB2”.

Impossible d'activer en raison de processus en cours

Le message d'erreur suivant indique que la procédure stockée n'a pas pu activer de base de données car elle est en cours d'exécution. `rdsadmin.create_database` `rdsadmin.restore_database`

```
The database can’t be activated because it's in the process of being created or
 restored.
```

Patientez quelques minutes, puis appelez à nouveau la procédure `rdsadmin.activate_database` stockée.

Erreurs rdsadmin.alter_tablespace

Les erreurs suivantes peuvent se produire lorsque vous appelez la procédure the section called “rdsadmin.alter_tablespace” stockée.

Erreur	Message d’erreur

Statement not valid	`DB21034E The command was processed as an SQL statement because it was not a valid Command Line Processor command. During SQL processing it returned:`
	
	`SQL1763N Invalid ALTER TABLESPACE statement for table space "TBSP_TEST" due to reason "12"`

tablespace_prefetch_size value not valid	`Invalid tablespace_prefetch_size. Set value to AUTOMATIC or to a non-zero positive numerical value.`

Erreur	Message d'erreur
tablespace_prefetch_size numerical value not valid	`Invalid tablespace_prefetch_size. The number of pages can't be greater than 32767.`
Parameter can't be used with tablespace_prefetch_size	`You can't use tablespace_prefetch_size with {parameter }.`
Tablespace change failed	`The change to tablespace { tablespace_name}  failed because you can only alter LARGE or REGULAR tablespaces.`

Déclaration non valide

Le message d'erreur suivant indique que la procédure stockée a combiné des paramètres facultatifs mutuellement exclusifs avec d'autres paramètres facultatifs. Les paramètres facultatifs `reduce_max, reduce_stopreduce_value`,`lower_high_water`,`lower_high_water_stop`, et `switch_online` de la procédure `rdsadmin.alter_tablespace` stockée s'excluent mutuellement. Vous ne pouvez pas les combiner avec d'autres paramètres facultatifs`buffer_pool_name`, comme dans la procédure `rdsadmin.alter_tablespace` stockée. Si vous les combinez, Db2 renvoie ce message d'erreur lorsque vous appelez la fonction `rdsadmin.get_task_status` définie par l'utilisateur.

```
DB21034E The command was processed as an SQL statement because it was not a valid
 Command Line Processor command. During SQL processing it returned:
SQL1763N Invalid ALTER TABLESPACE statement for table space "TBSP_TEST" due to reason
 "12"
```

Appelez à nouveau la procédure `rdsadmin.alter_tablespace` stockée sans combiner des paramètres facultatifs mutuellement exclusifs avec d'autres paramètres facultatifs. Appelez ensuite la fonction `rdsadmin.get_task_status` définie par l'utilisateur. Pour plus d'informations, consultez rdsadmin.alter_tablespace et rdsadmin.get_task_status.

valeur tablespace_prefetch_size non valide

Le message d'erreur suivant indique que vous n'avez pas défini `tablespace_prefetch_size` de valeur numérique positive `AUTOMATIC` ou non positive. Par exemple, vous avez essayé de le définir sur`testinput`.

```
Invalid tablespace_prefetch_size. Set value to AUTOMATIC or to a non-zero positive
 numerical value.
```

Appelez à nouveau la procédure `rdsadmin.alter_tablespace` stockée et réglez-la `tablespace_prefetch_size` sur `AUTOMATIC` ou sur une valeur numérique non positive.

valeur numérique tablespace_prefetch_size non valide

Le message d'erreur suivant indique que vous avez défini une valeur numérique supérieure `tablespace_prefetch_size` à 32 767.

```
Invalid tablespace_prefetch_size. The number of pages can't be greater than 32767.
```

Appelez à nouveau la procédure `rdsadmin.alter_tablespace` stockée et `tablespace_prefetch_size` réglez-la sur une valeur numérique positive différente de zéro inférieure ou égale à 32767.

Le paramètre ne peut pas être utilisé avec tablespace_prefetch_size

Le message d'erreur suivant indique que vous avez essayé de l'utiliser `tablespace_prefetch_size` avec un paramètre incompatible.

```
You can't use tablespace_prefetch_size with {parameter}.
```

Appelez à nouveau la procédure `rdsadmin.alter_tablespace` stockée et utilisez-la uniquement `tablespace_prefetch_size` avec des paramètres compatibles. Pour plus d'informations sur les paramètres que vous pouvez utiliser`tablespace_prefetch_size`, consultez[rdsadmin.alter_tablespace](#).

Le changement de tablespace a échoué

Le message d'erreur suivant indique que vous avez essayé de modifier un tablespace.

```
The change to tablespace {tablespace_name} failed because you can only alter LARGE or
 REGULAR tablespaces.
```

Erreurs rdsadmin.create_database

L'erreur suivante peut se produire lorsque vous appelez la procédure the section called “rdsadmin.create_database” stockée.

Erreur	Message d’erreur
Failed to allocate shared memory	`SQL1643C The database manager failed to allocate shared memory because the database manager instance memory limit has been reached.`

Impossible d'allouer de la mémoire partagée

Le message d'erreur suivant indique que la procédure stockée n'a pas réussi à créer une base de données car l'instance de base de données ne dispose pas de suffisamment de mémoire.

```
SQL1643C The database manager failed to allocate shared memory because the database
 manager instance memory limit has been reached.
```

Augmentez la mémoire de votre instance de base de données, puis appelez à nouveau la procédure `rdsadmin.create_database` stockée. Pour plus d'informations sur l'utilisation de la mémoire et les recommandations relatives aux bases de données, consultezthe section called “Plusieurs bases de données DB2”.

Pour confirmer que la base de données a été créée, appelez la fonction the section called “rdsadmin.list_databases” définie par l'utilisateur et vérifiez que la nouvelle base de données est répertoriée.

Erreurs rdsadmin.deactivate_database

L'erreur suivante peut se produire lorsque vous appelez la procédure the section called “rdsadmin.deactivate_database” stockée.

Erreur	Message d’erreur
Unable to deactivate because of running processes	`The database can’t be deactivated because it's in the process of being created or restored.`

Impossible de désactiver en raison de processus en cours

Le message d'erreur suivant indique que la procédure stockée n'a pas pu désactiver une base de données car elle est en cours d'exécution. `rdsadmin.create_database` `rdsadmin.restore_database`

```
The database can’t be deactivated because it's in the process of being created or
 restored.
```

Patientez quelques minutes, puis appelez à nouveau la procédure `rdsadmin.deactivate_database` stockée.

Erreurs rdsadmin.drop_database

Les erreurs suivantes peuvent se produire lorsque vous appelez la procédure the section called “rdsadmin.drop_database” stockée.

Erreur	Message d’erreur
Database name doesn't exist	`SQL0438N Application raised error or warning with diagnostic text: "Cannot drop database. Database with provided name does not exist". SQLSTATE=99993`
Return status = 0	`Return Status = 0`
Dropping database not allowed	`1 ERROR DROP_DATABASE RDSDB 2023-10-10-16.33.0 3.744122 2023-10-10-16.33.30.143797 - 2023-10-1 0-16.33.30.098857 Task execution has started. 2023-10-10-16.33.30.143797 Caught exception during executing task id 1, Aborting task. Reason Dropping database created via rds CreateDBInstance api is not allowed. Only database created using rdsadmin. create_database can be dropped`

Le nom de la base de données n'existe pas

Le message d'erreur suivant indique que vous avez transmis un nom de base de données incorrect dans la procédure `rdsadmin.drop_database` stockée.

```
SQL0438N Application raised error or warning with diagnostic text: "Cannot
drop database. Database with provided name does not exist". SQLSTATE=99993
```

Appelez à nouveau la procédure `rdsadmin.drop_database` stockée avec un nom de base de données correct. Pour confirmer que la base de données a été supprimée, appelez la fonction the section called “rdsadmin.list_databases” définie par l'utilisateur et vérifiez que la base de données supprimée n'est pas répertoriée.

État du retour = 0

Le message d'erreur suivant indique que la procédure stockée n'a pas pu être terminée.

```
Return Status = 0
```

Après réception`Return Status = 0`, appelez la fonction définie par l'utilisateur rdsadmin.get_task_status.

La suppression de la base de données est interdite

Le message d'erreur suivant indique que vous avez créé la base de données à l'aide de la console Amazon RDS ou du AWS CLI. Vous ne pouvez utiliser la procédure `rdsadmin.drop_database` stockée que si vous avez créé la base de données en appelant la procédure the section called “rdsadmin.create_database” stockée.

```
1 ERROR DROP_DATABASE RDSDB 2023-10-10-16.33.03.744122 2023-10-10-16.33.30.143797 -
 2023-10-10-16.33.30.098857 Task execution has started.
2023-10-10-16.33.30.143797 Caught exception during executing task id 1, Aborting task.
Reason Dropping database created via rds CreateDBInstance api is not allowed.
Only database created using rdsadmin.create_database can be dropped
```

Pour supprimer une base de données que vous avez créée à l'aide de la console Amazon RDS ou du AWS CLI, utilisez un client pour vous connecter à la base de données, puis exécutez la commande appropriée.

Erreurs rdsadmin.reactivate_database

L'erreur suivante peut se produire lorsque vous appelez la procédure the section called “rdsadmin.reactivate_database” stockée.

Erreur	Message d'erreur
Failed to allocate shared memory	`SQL1643C The database manager failed to allocate shared memory because the database manager instance memory limit has been reached.`
Unable to reactivate because of running processes	`The database can't be reactivated because it's in the process of being created or restored.`

Impossible d'allouer de la mémoire partagée

Le message d'erreur suivant indique que la procédure stockée n'a pas réussi à activer une base de données car l'instance de base de données ne dispose pas de suffisamment de mémoire.

```
SQL1643C The database manager failed to allocate shared memory because the database
 manager instance memory limit has been reached.
```

Augmentez la mémoire de votre instance de base de données, puis appelez à nouveau la procédure `rdsadmin.activate_database` stockée. Pour plus d'informations sur l'utilisation de la mémoire et les recommandations relatives aux bases de données, consultezthe section called "Plusieurs bases de données DB2".

Impossible de réactiver en raison de processus en cours

Le message d'erreur suivant indique que la procédure stockée n'a pas pu réactiver une base de données car elle est en cours d'exécution. `rdsadmin.create_database` `rdsadmin.restore_database`

```
The database can't be reactivated because it's in the process of being created or
 restored.
```

Patientez quelques minutes, puis appelez à nouveau la procédure `rdsadmin.reactivate_database` stockée.

Erreurs rdsadmin.restore_database

Les erreurs suivantes peuvent se produire lorsque vous appelez la procédure the section called "rdsadmin.restore_database" stockée :

Erreur	Message d'erreur
Insufficient disk space	`Aborting task. Reason Restoring your database failed because of insufficient disk space. Increase the storage for your DB instance and rerun the rdsadmin.restore_database stored procedure.`
Internal error	`Caught exception during executing task id 104, Aborting task. Reason Internal Error`
Non-fenced routines not allowed	`Caught exception during executing task id 2, Aborting task. Reason Non fenced routines are not allowed. Please delete the routines and retry the restore.`
Tablespaces not restored	`Reason SQL0970N The system attempted to write to a read-only file. Reason SQL2563W The Restore process has completed successfully. However one or more table spaces from the backup were not restored.`

Espace disque insuffisant

Le message d'erreur suivant indique que l'espace disque de votre instance de base de données est insuffisant pour restaurer votre base de données :

```
Aborting task. Reason Restoring your database failed because of insufficient
 disk space. Increase the storage for your DB instance and rerun the
 rdsadmin.restore_database stored procedure.
```

L'espace libre sur votre instance de base de données doit être plus du double de la taille de votre image de sauvegarde. Si votre image de sauvegarde est compressée, l'espace libre sur votre instance de base de données doit être plus du triple de la taille de votre image de sauvegarde. Pour de plus amples informations, veuillez consulter the section called “Augmentation de la capacité de stockage d'une instance de base de données”.

Augmentez votre espace disque, puis appelez à nouveau la procédure `rdsadmin.restore_database` stockée. Pour confirmer que la base de données a été restaurée, appelez la fonction the section called “rdsadmin.list_databases” définie par l'utilisateur et vérifiez que la base de données restaurée est répertoriée.

Erreur interne

Le message d'erreur suivant indique que la procédure stockée a rencontré une erreur interne :

```
Caught exception during executing task id 104, Aborting task. Reason Internal Error
```

Contactez AWS Support.

Les routines non clôturées ne sont pas autorisées

Le message d'erreur suivant indique que votre base de données contient des routines non clôturées :

```
Caught exception during executing task id 2, Aborting task. Reason Non fenced routines
 are not allowed. Please delete the routines and retry the restore.
```

RDS pour DB2 ne prend pas en charge les routines non clôturées. Supprimez les routines non clôturées de la base de données source, puis appelez `rdsadmin.restore_database` à nouveau. Pour confirmer que la base de données a été restaurée, appelez la fonction the section called “rdsadmin.list_databases” définie par l'utilisateur et vérifiez que la base de données restaurée est répertoriée. Pour de plus amples informations, veuillez consulter the section called “Routines non clôturées”.

Tablespaces non restaurés

Le message d'erreur suivant indique que RDS for Db2 a correctement restauré votre base de données, mais n'a pas pu restaurer un ou plusieurs tablespaces :

```
Reason SQL0970N The system attempted to write to a read-only file.
Reason SQL2563W The Restore process has completed successfully. However one or more
 table spaces from the backup were not restored.
```

RDS pour DB2 ne prend pas en charge le stockage non automatique. Convertissez le stockage non automatique en stockage automatique, puis appelez `rdsadmin.restore_database` à nouveau. Pour plus d'informations, consultez la section Conversion d'une base de données de stockage non automatique pour utiliser le stockage automatique dans la IBM Db2 documentation.

Les bases de données avec stockage non automatique de SMS nécessitent une restauration manuelle. Si votre base de données contient un stockage non automatique des SMS, contactez le AWS Support.

Pour plus d'informations sur le stockage non automatique et les migrations ponctuelles, consultezthe section called “Tablespaces de stockage non automatiques pendant la migration”.

Erreurs rdsadmin.update_db_param

L'erreur suivante peut se produire lorsque vous appelez la procédure the section called “rdsadmin.update_db_param” stockée.

Erreur	Message d’erreur
Parameter not supported or modifiable	`SQL0438N Application raised error or warning with diagnostic text: "Parameter is either not supported or not modifiable to customers". SQLSTATE=99993`

Paramètre non pris en charge ou non modifiable

Le message d'erreur suivant indique que vous avez essayé de modifier un paramètre de configuration de base de données qui n'est pas pris en charge ou qui n'est pas modifiable.

```
SQL0438N Application raised error or warning with diagnostic text: "Parameter
is either not supported or not modifiable to customers". SQLSTATE=99993
```

Vous pouvez voir quels paramètres sont modifiables en consultant vos groupes de paramètres. Pour de plus amples informations, veuillez consulter the section called “Afficher les valeurs des paramètres d'un groupe de paramètres de base de données”.

Amazon RDS for MariaDB

Amazon RDS prend en charge plusieurs versions de MariaDB pour les instances de base de données. Pour obtenir des informations complètes sur les versions prises en charge, consultezVersions de MariaDB sur Amazon RDS.

Pour créer une instance de base de données MariaDB, utilisez les interfaces ou les outils de gestion Amazon RDS. Vous pouvez ensuite utiliser les outils Amazon RDS pour effectuer des actions de gestion pour l'instance de base de données. Ces actions incluent :

- Reconfiguration ou redimensionnement de l'instance de base de données
- Autorisation des connexions à l'instance de base de données
- Création et restauration à partir de sauvegardes ou d'instantanés
- Création de secondaires Multi-AZ
- Création de réplicas en lecture
- Surveillance des performances de votre instance de base de données

Pour stocker les données de votre instance de base de données et y accéder, utilisez les applications et les utilitaires MariaDB standard.

MariaDB est disponible dans toutes les Régions AWS. Pour plus d'informations sur Régions AWS, voirRégions, zones de disponibilité et zones locales.

Vous pouvez utiliser Amazon RDS for MariaDB afin de développer des applications conformes à la loi HIPAA. Vous pouvez stocker les informations relatives à la santé, y compris les données de santé protégées (PHI, Protected Health Information), selon les termes d'un accord de partenariat (BAA, Business Associate Agreement) avec AWS. Pour de plus amples informations, veuillez consulter HIPAA compliance (français non garanti). AWS Les services concernés ont été entièrement évalués par un auditeur tiers et ont donné lieu à une certification, à une attestation de conformité ou à une autorisation d'exploitation (ATO). Pour de plus amples informations, veuillez consulter les services AWS concernés par le programme de conformité.

Avant de créer une instance de base de données, effectuez les étapes de la section Configuration de votre environnement Amazon RDS. Lorsque vous créez une instance de base de données, l'utilisateur principal RDS obtient des privilèges d'administrateur de base de données (avec certaines

restrictions). Utilisez ce compte pour des tâches administratives telles que la création de comptes de base de données supplémentaires.

Vous pouvez créer ce qui suit :

- Instances DB
- Instantanés de base de données
- Point-in-time restaure
- Sauvegardes automatiques
- Sauvegardes manuelles

Vous pouvez utiliser des instances de base de données exécutant MariaDB dans un cloud privé virtuel (VPC) basé sur Amazon VPC. Vous pouvez également ajouter des fonctionnalités à votre instance de base de données MariaDB en activant diverses options. Amazon RDS prend en charge les déploiements multi-AZ pour MariaDB comme solution de basculement haute disponibilité.

> **Important**
>
> Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui nécessitent des privilèges avancés. Vous pouvez accéder à votre base de données en utilisant des client SQL standard tels que le client mysql. Toutefois, vous ne pouvez pas accéder directement à l'hôte en utilisant Telnet ou Secure Shell (SSH).

Rubriques

- Support des fonctionnalités MariaDB sur Amazon RDS
- Versions de MariaDB sur Amazon RDS
- Connexion à votre instance de base de données MariaDB
- Sécurisation des connexions d'instance de base de données MariaDB
- Amélioration des performances des requêtes pour RDS for MariaDB avec Amazon RDS Optimized Reads
- Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB

- Mises à niveau du moteur de base de données MariaDB
- Mise à niveau d'une version du moteur de snapshots de base de données MariaDB
- Importation de données dans une instance de base de données Amazon RDS for MariaDB
- Utilisation de la réplication MariaDB sur Amazon RDS
- Options pour le moteur de base de données MariaDB
- Paramètres pour MariaDB
- Migration des données d'un instantané My SQL DB vers une instance de base de données MariaDB
- Référence MariaDB sur SQL Amazon RDS
- Fuseau horaire local pour les instances de base de données MariaDB
- Limites et problèmes connus pour RDS for MariaDB

Support des fonctionnalités MariaDB sur Amazon RDS

RDScar MariaDB prend en charge la plupart des fonctionnalités et capacités de MariaDB. Certaines fonctions peuvent avoir une prise en charge limitée ou des privilèges restreints.

Vous pouvez filtrer les nouvelles RDS fonctionnalités d'Amazon dans la base de données What's New with ? page. Pour les produits, choisissez Amazon RDS. Ensuite, effectuez une recherche à l'aide de mots clés tels que **`MariaDB 2023`**.

> **Note**
>
> Les listes suivantes ne sont pas exhaustives.

Pour plus d'informations sur la prise en charge des fonctionnalités MariaDB sur RDS Amazon, consultez les rubriques suivantes.

Rubriques

- Moteurs de stockage pris en charge pour MariaDB sur Amazon RDS
- Réchauffement du cache pour MariaDB sur Amazon RDS
- Fonctionnalités de MariaDB non prises en charge par Amazon RDS

Support des fonctionnalités de MariaDB sur Amazon RDS pour les versions majeures de MariaDB

Dans les sections suivantes, vous trouverez des informations sur la prise en charge des fonctionnalités de MariaDB sur Amazon RDS pour les versions majeures de MariaDB :

Rubriques

- Support de MariaDB 11.4 sur Amazon RDS
- Support de MariaDB 10.11 sur Amazon RDS
- Support de MariaDB 10.6 sur Amazon RDS
- Support de MariaDB 10.5 sur Amazon RDS
- Support de MariaDB 10.4 sur Amazon RDS

Pour plus d'informations sur les versions mineures prises en charge d'Amazon RDS pour MariaDB, consultez. Versions de MariaDB sur Amazon RDS

Support de MariaDB 11.4 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MariaDB version 11.4 ou supérieure.

- Bibliothèque cryptographique — pour RDS MariaDB, Open a été remplacé par AWS Libcrypto (AWS-LC), qui est certifié SSL 140-3. FIPS
- Plugin Simple Password Check — Vous pouvez utiliser le plugin MariaDB Simple Password Check pour vérifier si un mot de passe contient au moins un certain nombre de caractères d'un type spécifique. Pour de plus amples informations, veuillez consulter the section called “Plug-ins de validation de mots”.
- Plugin Cracklib Password Check — Vous pouvez utiliser le plugin MariaDB Cracklib Password Check pour vérifier la solidité des nouveaux mots de passe. Pour de plus amples informations, veuillez consulter the section called “Plug-ins de validation de mots”.
- Améliorations InnoDB : ces améliorations incluent les éléments suivants :
  - La mémoire tampon des modifications a été supprimée. Pour plus d'informations, consultez InnoDB Change Buffering.
  - La défragmentation InnoDB a été supprimée. Pour plus d'informations, consultez la section Défragmentation d'InnoDB.

- Nouveau privilège — L'utilisateur administrateur dispose désormais également de `SHOW CREATE ROUTINE` ce privilège. Ce privilège permet au bénéficiaire de consulter l'instruction de `SHOW CREATE` définition d'une routine appartenant à un autre utilisateur. Pour plus d'informations, consultez Privilèges de base de données.
- Amélioration de la réplication : les instances de base de données MariaDB version 11.4 prennent en charge l'indexation des journaux binaires. Vous pouvez créer un GTID index pour chaque fichier binlog. Ces index améliorent les performances de réplication en réduisant le temps nécessaire pour localiser unGTID. Pour plus d'informations, consultez Binlog Indexing.
- Paramètres obsolètes ou supprimés — Les paramètres suivants ont été déconseillés ou supprimés pour les instances de base de données MariaDB version 11.4 :
  - `engine_condition_pushdown`est supprimé de optimizer_switch
  - innodb_change_buffer_max_size
  - innodb_defragment
  - `TLSv1.0`et `TLSv1.1` sont supprimés de tls_version
- Nouvelles valeurs par défaut pour un paramètre — La valeur par défaut du paramètre innodb_undo_tablespaces est passée de à. `0` `3`
- Nouvelles valeurs valides pour les paramètres — Les paramètres suivants ont de nouvelles valeurs valides pour les instances de base de données MariaDB version 11.4 :
  - Les valeurs valides pour le paramètre binlog_row_image incluent désormais. `FULL_NODUP`
  - Les valeurs valides pour le MODE paramètre OLD_ sont désormais inclusesNO_NULL_COLLATION_IDS.
- Nouveaux paramètres — Les paramètres suivants sont nouveaux pour les instances de base de données MariaDB version 11.4 :
  - Le paramètre transaction_isolation remplace le paramètre tx_isolation.
  - Le paramètre transaction_read_only remplace le paramètre tx_read_only.
  - Le paramètre block_encryption_mode définit le mode de chiffrement par blocs par défaut pour les fonctions AES_ ENCRYPT () et AES _ (). DECRYPT
  - Le character_set_collations définit les remplacements pour les classements par défaut des jeux de caractères.
  - Les fichiers binlog_gtid_index, binlog_gtid_index_page_size et binlog_gtid_index_span_mindéfinissent les propriétés de l'index binlog. GTID Pour plus d'informations, consultez Binlog Indexing.

Pour une liste de toutes les fonctionnalités de MariaDB 11.4 et de leur documentation, consultez les sections Modifications et améliorations apportées à MariaDB 11.4 et Notes de version - série MariaDB 11.4 sur le site Web de MariaDB.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctionnalités de MariaDB non prises en charge par Amazon RDS.

Support de MariaDB 10.11 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MariaDB version 10.11 ou supérieure.

- Plug-in Password Reuse Check : vous pouvez utiliser le plug-in MariaDB Password Reuse Check pour empêcher les utilisateurs de réutiliser les mots de passe et pour définir la période de conservation des mots de passe. Pour plus d'informations, consultez Plug-in Password Reuse Check (langue française non garantie).
- GRANTPUBLICAutorisation TO — Vous pouvez accorder des privilèges à tous les utilisateurs ayant accès à votre serveur. Pour plus d'informations, consultez GRANTTO PUBLIC.
- Séparation SUPER et READ ONLY ADMIN privilèges — Vous pouvez supprimer les READ ONLY ADMIN privilèges de tous les utilisateurs, même ceux qui en bénéficiaient SUPER auparavant.
- Sécurité : vous pouvez maintenant définir l'option `--ssl` par défaut pour votre client MariaDB. MariaDB ne se SSL désactive plus silencieusement si la configuration est incorrecte.
- SQLcommandes et fonctions — Vous pouvez désormais utiliser la `SHOW ANALYZE FORMAT=JSON` commande et les fonctions `ROW_NUMBERSFORMAT`, et`RANDOM_BYTES`. `SFORMAT`autorise le formatage de chaînes et est activé par défaut. Vous pouvez convertir une partition en table et une table en partition en une seule commande. Il existe également plusieurs améliorations concernant les fonctions `JSON_*()`. Les fonctions `DES_ENCRYPT` et `DES_DECRYPT` ont été déconseillées pour les versions 10.10 et supérieures. Pour de plus amples informations, veuillez consulter SFORMAT.
- Améliorations InnoDB : ces améliorations incluent les éléments suivants :
  - Améliorations des performances dans le journal redo afin de réduire l'amplification d'écriture et améliorer la simultanéité.
  - Possibilité de modifier l'espace de table d'annulation sans réinitialiser le répertoire de données. Cette amélioration réduit le surcoût du plan de contrôle. Elle requiert un redémarrage, mais pas la réinitialisation après la modification de l'espace de table d'annulation.
  - Prise en charge de `CHECK TABLE … EXTENDED` et des index décroissants en interne.
  - Améliorations apportées à l'insertion en vrac.

- Modifications du journal binaire : ces modifications incluent les éléments suivants :
  - Journalisation `ALTER` en deux phases pour réduire la latence de réplication. Le paramètre `binlog_alter_two_phase` est désactivé par défaut, mais peut être activé par le biais de groupes de paramètres.
  - Journalisation `explicit_defaults_for_timestamp`.
  - Plus de journalisation `INCIDENT_EVENT` si la transaction peut être annulée en toute sécurité.
- Améliorations de la réplication — Les instances de base de données MariaDB version 10.11 GTID utilisent la réplication par défaut si le maître la prend en charge. De plus, `Seconds_Behind_Master` est plus précis.
- Clients : vous pouvez utiliser de nouvelles options de ligne de commande pour `mysqlbinlog` et `mariadb-dump`. Vous pouvez utiliser `mariadb-dump` pour vider et restaurer les données d'historique.
- Gestion des versions du système : vous pouvez modifier l'historique. MariaDB crée automatiquement de nouvelles partitions.
- Atomique DDL — `CREATE OR REPLACE` est désormais atomique. Soit l'instruction réussit, soit elle est complètement inversée.
- Écriture du journal redo : le journal redo écrit de manière asynchrone.
- Fonctions stockées : les fonctions stockées prennent désormais en charge les mêmes paramètres `IN`, `OUT` et `INOUT` que dans les procédures stockées.
- Paramètres déconseillés ou supprimés : les paramètres suivants sont devenus obsolètes ou ont été supprimés pour les instances de base de données MariaDB version 10.11 :
  - innodb_change_buffering
  - innodb_disallow_writes
  - innodb_log_write_ahead_size
  - innodb_prefix_index_cluster_optimization
  - keep_files_on_create
  - old
- Paramètres dynamiques : les paramètres suivants sont désormais dynamiques pour les instances de base de données MariaDB version 10.11 :
  - innodb_log_file_size
  - innodb_write_io_threads
  - innodb_read_io_threads

- Nouvelles valeurs par défaut pour les paramètres : les paramètres suivants ont de nouvelles valeurs par défaut pour les instances de base de données MariaDB version 10.11 :
  - La valeur par défaut du paramètre explicit_defaults_for_timestamp est passée de `OFF` à `ON`.
  - La valeur par défaut du paramètre optimizer_prune_level est passée de 1 à 2.
- Nouvelles valeurs valides pour les paramètres : les paramètres suivants ont de nouvelles valeurs valides pour les instances de base de données MariaDB version 10.11 :
  - Les valeurs valides pour le paramètre old ont été fusionnées à celles du paramètre old_mode.
  - Les valeurs valides pour le paramètre histogram_type incluent désormais `JSON_HB`.
  - La plage des valeurs valides pour le paramètre innodb_log_buffer_size est maintenant de `262144` à `4294967295` (de 256 Ko à 4 096 Mo).
  - La plage des valeurs valides pour le paramètre innodb_log_file_size est maintenant de `4194304` à `512GB` (de 4 Mo à 512 Go).
  - Les valeurs valides pour le paramètre optimizer_prune_level incluent désormais 2.
- Nouveaux paramètres : les paramètres suivants sont nouveaux pour les instances de base de données MariaDB version 10.11 :
  - Le paramètre binlog_alter_two_phase peut améliorer les performances de réplication.
  - Le paramètre log_slow_min_examined_row_limit peut améliorer les performances.
  - Le paramètre log_slow_query et le paramètre log_slow_query_file sont des alias pour `slow_query_log` et `slow_query_log_file`, respectivement.
  - optimizer_extra_pruning_depth
  - system_versioning_insert_history

Pour une liste de toutes les fonctionnalités de MariaDB 10.11 et de leur documentation, voir Modifications et améliorations apportées à MariaDB 10.11 et Notes de version - série MariaDB 10.11 sur le site Web de MariaDB.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctionnalités de MariaDB non prises en charge par Amazon RDS.

Support de MariaDB 10.6 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MariaDB version 10.6 ou supérieure :

- MyRocks moteur de stockage : vous pouvez utiliser le moteur MyRocks de stockage RDS pour MariaDB afin d'optimiser la consommation de stockage de vos applications Web hautes performances et gourmandes en écriture. Pour plus d'informations, consultez Moteurs de stockage pris en charge pour MariaDB sur Amazon RDS et MyRocks.
- AWS Identity and Access Management (IAM) Authentification de base de données — Vous pouvez utiliser l'authentification de IAM base de données pour une meilleure sécurité et une gestion centralisée des connexions à vos instances de base de données MariaDB. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.
- Options de mise à niveau — Vous pouvez désormais passer à la RDS version 10.6 de MariaDB à partir de n'importe quelle version majeure antérieure (10.3, 10.4, 10.5). Vous pouvez également restaurer un instantané d'une instance de base de données My SQL 5.6 ou 5.7 existante sur une instance MariaDB 10.6. Pour de plus amples informations, veuillez consulter Mises à niveau du moteur de base de données MariaDB.
- Réplication retardée: vous pouvez désormais définir une période configurable pour laquelle un réplica en lecture est en retard par rapport à la base de données source. Dans une configuration de réplication MariaDB standard, le délai de réplication entre la source et le réplica est minime. Avec la réplication différée, vous pouvez définir un délai intentionnel comme stratégie de reprise après sinistre. Pour de plus amples informations, veuillez consulter Configuration de la réplication différée avec MariaDB.
- SQLCompatibilité avec Oracle PL/ — En utilisant RDS la version 10.6 de MariaDB, vous pouvez migrer plus facilement vos anciennes applications Oracle vers Amazon. RDS Pour plus d'informations, voir SQL_ MODE = ORACLE.
- Atomic DDL — Vos instructions Dynamic Data Language (DDL) peuvent être relativement sûres en cas de crash avec la version 10.6 de RDS MariaDB. `CREATE TABLE`,`ALTER TABLE`, `RENAME TABLEDROP TABLE`, `DROP DATABASE` et les DDL instructions associées sont désormais atomiques. Soit l'instruction réussit, soit elle est complètement inversée. Pour plus d'informations, consultez Atomic DDL.
- Autres améliorations — Ces améliorations incluent une `JSON_TABLE` fonction permettant de transformer les JSON données au SQL format relationnel et un chargement plus rapide des données de tables vides avec Innodb. Ils incluent également de nouveaux `sys_schema` à des fins d'analyse et de dépannage, d'amélioration de l'optimiseur pour ignorer les index inutilisés et d'amélioration des performances. Pour plus d'informations, voir JSON_ TABLE.
- Nouvelles valeurs par défaut pour les paramètres : les paramètres suivants disposent de nouvelles valeurs par défaut pour les instances de base de données MariaDB version 10.6 :

- La valeur par défaut des paramètres suivants est passée de `utf8` à `utf8mb3` :
  - character_set_client
  - character_set_connection
  - character_set_results
  - character_set_system

  Bien que les valeurs par défaut aient changé pour ces paramètres, il n'y a pas de changement fonctionnel. Pour plus d'informations, consultez Supported Character Sets and Collations (Jeux de caractères et classements pris en charge) dans la documentation MariaDB.
- La valeur par défaut du paramètre collation_connection est passée de `utf8_general_ci` à `utf8mb3_general_ci`. Bien que les valeurs par défaut aient changé pour ces paramètres, il n'y a pas de changement fonctionnel.
- La valeur par défaut du paramètre old_mode est passé de non défini à `UTF8_IS_UTF8MB3`. Bien que les valeurs par défaut aient changé pour ces paramètres, il n'y a pas de changement fonctionnel.

Pour accéder à la liste complète des fonctions MariaDB 10.6 ainsi qu'à la documentation associée, veuillez consulter Modifications et améliorations apportées à MariaDB 10.6 et Notes de mise à jour – Série MariaDB 10.6 sur le site Web de MariaDB.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctionnalités de MariaDB non prises en charge par Amazon RDS.

Support de MariaDB 10.5 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MariaDB version 10.5 ou ultérieure :

- Améliorations d'InnoDB – MariaDB version 10.5 inclut les améliorations d'InnoDB. Pour plus d'informations, consultez InnoDB: Performance Improvements etc. (InnoDB : Améliorations liées aux performances, etc.) dans la documentation MariaDB.
- Mises à jour du schéma de performances – MariaDB version 10.5 inclut les mises à jour du schéma de performances. Pour plus d'informations, consultez la section Mises à jour du schéma de performance pour correspondre à mon instrumentation et à mes tables SQL 5.7 dans la documentation MariaDB.

- Un seul fichier dans le journal redo d'InnoDB – Dans les versions de MariaDB antérieures à la version 10.5, la valeur du paramètre `innodb_log_files_in_group` était définie sur 2. Dans MariaDB version 10.5, la valeur de ce paramètre est définie sur 1.

  Si vous procédez à une mise à niveau vers MariaDB version 10.5 et que vous ne modifiez pas les paramètres, la valeur du paramètre `innodb_log_file_size` reste inchangée. Mais elle s'applique à un seul fichier journal au lieu de deux. En conséquence, votre instance de base de données MariaDB version 10.5 mise à niveau utilise la moitié de la taille du journal redo qu'elle utilisait avant la mise à niveau. Ce changement peut avoir un impact notable sur les performances. Pour résoudre ce problème, vous pouvez doubler la valeur du paramètre `innodb_log_file_size`. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).
- SHOWSLAVESTATUScommande non prise en charge — Dans les versions de MariaDB antérieures à la version 10.5, la commande nécessitait `SHOW SLAVE STATUS` le privilège. `REPLICATION SLAVE` Dans MariaDB version 10.5, la commande `SHOW REPLICA STATUS` équivalente requiert le privilège `REPLICATION REPLICA ADMIN`. Ce nouveau privilège n'est pas accordé à l'utilisateur RDS principal.

  Au lieu d'utiliser la commande `SHOW REPLICA STATUS`, exécutez la nouvelle procédure stockée `mysql.rds_replica_status` pour renvoyer des informations similaires. Pour de plus amples informations, veuillez consulter mysql.rds_replica_status.
- SHOWRELAYLOGEVENTScommande non prise en charge — Dans les versions de MariaDB antérieures à la version 10.5, la commande nécessitait `SHOW RELAYLOG EVENTS` le privilège. `REPLICATION SLAVE` Dans MariaDB version 10.5, cette commande requiert le privilège `REPLICATION REPLICA ADMIN`. Ce nouveau privilège n'est pas accordé à l'utilisateur RDS principal.
- Nouvelles valeurs par défaut pour les paramètres – Les paramètres suivants disposent de nouvelles valeurs par défaut pour les instances de base de données MariaDB version 10.5 :
  - La valeur par défaut du paramètre max_connections a été remplacée par `LEAST({DBInstanceClassMemory/25165760},12000)`. Pour plus d'informations sur la fonction de paramètre `LEAST`, consultez Fonctions de paramètre de bases de données.
  - La valeur par défaut du paramètre innodb_adaptive_hash_index a été remplacée par `OFF` (`0`).
  - La valeur par défaut du paramètre innodb_checksum_algorithm a été remplacée par `full_crc32`.
  - La valeur par défaut du paramètre innodb_log_file_size a été remplacée par 2 Go.

Pour accéder à la liste complète des fonctions MariaDB 10.5 ainsi qu'à la documentation associée, consultez Modifications et améliorations apportées à MariaDB 10.5 et Notes de mise à jour - Série MariaDB 10.5 sur le site Web de MariaDB.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctionnalités de MariaDB non prises en charge par Amazon RDS.

Support de MariaDB 10.4 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MariaDB version 10.4 ou ultérieure :

- Améliorations de la sécurité des comptes utilisateur – Améliorations de l'expiration des mots de passe et du verrouillage des comptes
- Améliorations de l'optimiseur – Fonction Optimizer Trace
- Améliorations apportées à InnoDB — DROPCOLUMNSupport instantané et `VARCHAR` extension instantanée pour et `ROW_FORMAT=DYNAMIC` `ROW_FORMAT=COMPACT`
- Nouveaux paramètres – Notamment : tcp_nodedelay, tls_version et gtid_cleanup_batch_size

Pour obtenir la liste de toutes les fonctions MariaDB 10.4 et leur documentation, veuillez consulter Changes and Improvements in MariaDB 10.4 (Modifications et améliorations dans MariaDB 10.4) et Release Notes - MariaDB 10.4 Series (Notes de mise à jour - MariaDB 10.4 Series) sur le site web de MariaDB.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctionnalités de MariaDB non prises en charge par Amazon RDS.

Moteurs de stockage pris en charge pour MariaDB sur Amazon RDS

RDScar MariaDB prend en charge les moteurs de stockage suivants.

Rubriques

- Le moteur de stockage InnoDB
- Le moteur MyRocks de stockage

Les autres moteurs de stockage ne sont actuellement pas pris en charge par RDS MariaDB.

Le moteur de stockage InnoDB

Bien que MariaDB prenne en charge plusieurs moteurs de stockage avec diverses capacités, toutes ne sont pas optimisées pour la récupération sur incident et la durabilité des données. InnoDB est le moteur de stockage recommandé pour les instances de base de données MariaDB sur Amazon. RDS RDSLes fonctionnalités Amazon telles que la point-in-time restauration et la restauration de snapshots nécessitent un moteur de stockage récupérable et ne sont prises en charge que pour le moteur de stockage recommandé pour la version MariaDB.

Pour plus d'informations, veuillez consulter InnoDB.

Le moteur MyRocks de stockage

Le moteur MyRocks de stockage est disponible RDS pour MariaDB version 10.6 et supérieure. Avant d'utiliser le moteur MyRocks de stockage dans une base de données de production, nous vous recommandons d'effectuer des analyses comparatives et des tests approfondis afin de vérifier les avantages potentiels par rapport à InnoDB pour votre cas d'utilisation.

Le groupe de paramètres par défaut pour MariaDB version 10.6 inclut des paramètres. MyRocks Pour plus d'informations, consultez Paramètres pour MariaDB et Groupes de paramètres pour (Amazon RDS).

Pour créer une table utilisant le moteur de MyRocks stockage, spécifiez-le `ENGINE=RocksDB` dans l'`CREATE TABLE`instruction. L'exemple suivant crée une table qui utilise le moteur MyRocks de stockage.

```
CREATE TABLE test (a INT NOT NULL, b CHAR(10)) ENGINE=RocksDB;
```

Nous vous recommandons vivement de ne pas exécuter de transactions couvrant à la fois InnoDB et MyRocks des tables. MariaDB ne ACID garantit pas (atomicité, cohérence, isolation, durabilité) les transactions entre les moteurs de stockage. Bien qu'il soit possible d'avoir à la fois InnoDB et MyRocks des tables dans une instance de base de données, nous ne recommandons pas cette approche, sauf lors d'une migration d'un moteur de stockage à l'autre. Lorsque InnoDB et des MyRocks tables existent dans une instance de base de données, chaque moteur de stockage possède son propre pool de mémoire tampon, ce qui peut entraîner une dégradation des performances.

MyRocks ne supporte pas `SERIALIZABLE` l'isolation ou les verrous interstitiels. Par conséquent, vous ne pouvez généralement pas l'utiliser MyRocks avec la réplication basée sur des instructions. Pour plus d'informations, reportez-vous à la section MyRocks et Réplication.

Actuellement, vous ne pouvez modifier que les MyRocks paramètres suivants :

- `rocksdb_block_cache_size`
- `rocksdb_bulk_load`
- `rocksdb_bulk_load_size`
- `rocksdb_deadlock_detect`
- `rocksdb_deadlock_detect_depth`
- `rocksdb_max_latest_deadlocks`

Le moteur MyRocks de stockage et le moteur de stockage InnoDB peuvent se disputer la mémoire en fonction des paramètres `rocksdb_block_cache_size` et`innodb_buffer_pool_size`. Dans certains cas, il se peut que vous ayez l'intention d'utiliser le moteur MyRocks de stockage uniquement sur une instance de base de données particulière. Dans l'affirmative, nous vous recommandons de définir le paramètre `innodb_buffer_pool_size minimal` à une valeur minimale et de définir le paramètre `rocksdb_block_cache_size` à une valeur aussi haute que possible.

Vous pouvez accéder aux fichiers MyRocks journaux à l'aide `DownloadDBLogFilePortion`des opérations `DescribeDBLogFiles`et.

Pour plus d'informations MyRocks, consultez MyRocksle site Web de MariaDB.

Réchauffement du cache pour MariaDB sur Amazon RDS

La préparation du cache InnoDB peut fournir des gains de performances pour votre instance de base de données MariaDB en enregistrant l'état actuel du pool de mémoires tampons lorsque l'instance de base de données est arrêtée, puis en rechargeant le pool de mémoires tampons à partir des informations enregistrées au démarrage de l'instance de base de données. Cette approche contourne la nécessité de « préparer » le pool de tampons à partir d'une utilisation normale de la base de données et précharge à la place le pool de tampons avec les pages des requêtes courantes connues. Pour plus d'informations sur la préparation du cache, consultez Vidage et restauration du pool de tampons dans la documentation MariaDB.

La préparation du cache est activée par défaut sur les instances de base de données MariaDB versions 10.3 et ultérieures. Pour l'activer, définissez les paramètres `innodb_buffer_pool_dump_at_shutdown` et `innodb_buffer_pool_load_at_startup` avec la valeur 1 dans le groupe de paramètres de votre instance de base de données. La

modification de ces valeurs dans un groupe de paramètres affecte toutes les instances de base de données MariaDB qui utilisent ce groupe de paramètres. Pour activer la préparation du cache pour des instances de base de données MariaDB spécifiques, vous aurez peut-être à créer un groupe de paramètres pour ces instances de base de données. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

La préparation du cache fournit principalement une amélioration des performances pour les instances de bases de données qui utilisent le stockage standard. Si vous utilisez le PIOPS stockage, vous ne constatez généralement pas d'avantage significatif en termes de performances.

Important

Si votre instance de base de données MariaDB ne se ferme pas normalement, comme lors d'un basculement, l'état du pool de tampons n'est pas enregistré sur le disque. Dans ce cas, MariaDB charge n'importe quel fichier du pool de tampons disponible au redémarrage de l'instance de base de données. Il n'en résulte aucun dommage, mais le pool de tampons restauré peut ne pas refléter l'état le plus récent du pool de tampons avant le redémarrage. Pour vous assurer d'avoir un état récent du pool de mémoires tampons disponible afin de préparer le cache au démarrage, il est recommandé que vous vidiez régulièrement le pool de mémoires tampons « à la demande ». Vous pouvez vider ou charger le pool de tampons à la demande.

Vous pouvez créer un événement pour vider le pool de tampons automatiquement et à intervalles réguliers. Par exemple, l'instruction suivante crée un événement nommé `periodic_buffer_pool_dump` qui vide le pool de mémoires tampons toutes les heures.

```
CREATE EVENT periodic_buffer_pool_dump
   ON SCHEDULE EVERY 1 HOUR
   DO CALL mysql.rds_innodb_buffer_pool_dump_now();
```

Pour plus d'informations, consultez Événements dans la documentation MariaDB.

Vidage et chargement du pool de tampons à la demande

Vous pouvez enregistrer et charger le cache à la demande à l'aide des procédures stockées suivantes :

- Pour vider l'état actuel du pool de mémoires tampons sur le disque, appelez la procédure stockée mysql.rds_innodb_buffer_pool_dump_now.

- Pour charger l'état enregistré du pool de mémoires tampons à partir du disque, appelez la procédure stockée mysql.rds_innodb_buffer_pool_load_now.
- Pour annuler une opération de chargement en cours, appelez la procédure stockée mysql.rds_innodb_buffer_pool_load_abort.

Fonctionnalités de MariaDB non prises en charge par Amazon RDS

Les fonctionnalités MariaDB suivantes ne sont pas prises en charge sur Amazon : RDS

- Moteur de stockage S3
- Plugin d'authentification — GSSAPI
- Plug-in d'authentification – Socket Unix
- AWS Plugin de chiffrement pour la gestion des clés
- Réplication différée pour les versions MariaDB inférieures à 10.6
- Chiffrement au repos MariaDB natif pour InnoDB et Aria

  Vous pouvez activer le chiffrement au repos pour une instance de base de données MariaDB en suivant les instructions de Chiffrement des ressources Amazon RDS.
- HandlerSocket
- JSONtype de table pour les versions de MariaDB inférieures à 10.6
- MariaDB ColumnStore
- MariaDB Galera Cluster
- Réplication multi-source
- MyRocks moteur de stockage pour les versions de MariaDB inférieures à 10.6
- Plugin de validation de mot de passe`simple_password_check`, et `cracklib_password_check` pour les versions de MariaDB inférieures à 11.4
- Moteur de stockage Spider
- Moteur de stockage Sphinx
- Moteur de stockage TokuDB
- Attributs d'objet spécifiques au moteur de stockage, tels que décrits dans la section Nouveaux attributs définis par le moteur dans la documentation Table/Field/Index MariaDB
- Chiffrement de table et d'espace de tables
- Plug-in Hashicorp Key Management

- Exécution de deux mises à niveau en parallèle

Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données et restreint l'accès à certaines procédures et tables du système qui nécessitent des privilèges avancés. Amazon RDS prend en charge l'accès aux bases de données sur une instance de base de données à l'aide de n'importe quelle application SQL client standard. Amazon RDS n'autorise pas l'accès direct de l'hôte à une instance de base de données via Telnet, Secure Shell (SSH) ou Windows Remote Desktop Connection.

Versions de MariaDB sur Amazon RDS

Dans MariaDB, les numéros de version sont organisés en versions X.Y.Z. Dans la terminologie Amazon RDS, X.Y indique la version majeure et Z le numéro de la version mineure. Pour les implémentations Amazon RDS, un changement de version sera considéré comme majeur si le numéro de la version majeure change, par exemple, en passant de la version 10.5 à la version 10.6. Un changement de version est considéré comme mineur si seul le numéro de version mineure change, par exemple en passant de la version 10.6.14 à la version 10.6.16.

Rubriques

- Versions de MariaDB mineures prises en charge sur Amazon RDS
- Versions de MariaDB majeures prises en charge sur Amazon RDS
- Utilisation de l'environnement de prévisualisation de base de données
- MariaDB version 11.8 dans l'environnement de prévisualisation de la base de données
- MariaDB version 11.7 dans l'environnement de prévisualisation de la base de données
- MariaDB version 11.4 dans l'environnement de prévisualisation de la base de données
- Versions rendues obsolètes pour Amazon RDS for MariaDB

Versions de MariaDB mineures prises en charge sur Amazon RDS

Amazon RDS prend actuellement en charge les versions mineures suivantes de MariaDB.

> **Note**
>
> Les dates avec seulement un mois et une année sont approximatives et sont mises à jour avec une date exacte quand elles sont connues.

Le tableau suivant présente les versions mineures de MariaDB 11.4 actuellement prises en charge par Amazon RDS.

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
11,4.8	6 août 2025	13 août 2025	septembre 2026

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
11,4.7	22 mai 2025	4 juin 2025	septembre 2026
11.4.5	4 février 2025	24 février 2025	mars 2026
11,4.4	1er novembre 2024	20 décembre 2024	mars 2026
11.4.3	8 août 2024	15 octobre 2024	mars 2026

Le tableau suivant présente les versions mineures de MariaDB 10.11 actuellement prises en charge par Amazon RDS.

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
10,1,114	6 août 2025	13 août 2025	septembre 2026
10,1,113	22 mai 2025	4 juin 2025	septembre 2026
10,11,11	4 février 2025	24 février 2025	mars 2026
10,1,110	1er novembre 2024	20 décembre 2024	mars 2026
10,11.9	8 août 2024	4 septembre 2024	mars 2026
10,11.8	16 mai 2024	14 juin 2024	octobre 2025

Le tableau suivant présente les versions mineures de MariaDB 10.6 actuellement prises en charge par Amazon RDS.

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
10,6,23	6 août 2025	13 août 2025	août 2026
10,6,22	6 mai 2025	20 mai 2025	Juillet 2026
10,6,21	4 février 2025	24 février 2025	mars 2026
10,6,20	1er novembre 2024	20 décembre 2024	mars 2026
10,6,19	8 août 2024	4 septembre 2024	mars 2026
10,6,18	16 mai 2024	14 juin 2024	octobre 2025

Le tableau suivant présente les versions mineures de MariaDB 10.5 actuellement prises en charge par Amazon RDS.

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
10,5,29	6 mai 2025	20 mai 2025	février 2026
10,5,28	4 février 2025	24 février 2025	février 2026
10,5,27	1er novembre 2024	20 décembre 2024	février 2026
10,5,26	8 août 2024	4 septembre 2024	octobre 2025
10,5,25	16 mai 2024	14 juin 2024	octobre 2025

Le tableau suivant présente les versions mineures de MariaDB 10.4 actuellement prises en charge par Amazon RDS.

Version du moteur MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
10,4,34	16 mai 2024	14 juin 2024	28 février 2025
10,4,33	7 février 2024	26 février 2024	28 février 2025
10,4,32	13 novembre 2023	12 décembre 2023	28 février 2025
10,4,31	14 août 2023	7 septembre 2023	28 février 2025
10,4,30	7 juin 2023	22 juin 2023	28 février 2025
10,4,29	10 mai 2023	15 juin 2023	28 février 2025

Vous pouvez spécifier n'importe quelle version MariaDB actuellement prise en charge lorsque vous créez une instance de base de données. Vous pouvez spécifier la version majeure (par exemple, MariaDB 10.5) et toute version mineure prise en charge pour la version majeure spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version prise en charge, généralement la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée. Pour voir la liste des versions prises en charge, ainsi que les valeurs par défaut pour les instances de base de données nouvellement créées, utilisez la `describe-db-engine-versions` AWS CLI commande.

Par exemple, pour répertorier les versions de moteur prises en charge pour RDS for MariaDB, exécutez la commande CLI suivante :

```
aws rds describe-db-engine-versions --engine mariadb --query "*[].
{Engine:Engine,EngineVersion:EngineVersion}" --output text
```

La version par défaut de MariaDB peut varier selon la Région AWS. Pour créer une instance de base de données avec une version mineure spécifique, spécifiez la version mineure lors de la création de l'instance de base de données. Vous pouvez déterminer la version mineure par défaut pour un Région AWS en exécutant la AWS CLI commande suivante :

```
aws rds describe-db-engine-versions --default-only --engine mariadb
 --engine-version major_engine_version --region region --query "*[].
{Engine:Engine,EngineVersion:EngineVersion}" --output text
```

*major_engine_version*Remplacez-le par la version principale du moteur, puis remplacez *region* par le Région AWS. Par exemple, la AWS CLI commande suivante renvoie la version mineure du moteur MariaDB par défaut pour la version majeure 10.5 et pour l'ouest des États-Unis (Oregon) Région AWS (us-west-2) :

```
aws rds describe-db-engine-versions --default-only --engine mariadb --engine-version
 10.5 --region us-west-2 --query "*[].{Engine:Engine,EngineVersion:EngineVersion}" --
output text
```

Versions mineures de MariaDB sur Amazon RDS

Versions mineures

- Version 11.4.8 de MariaDB
- Version 11.4.7 de MariaDB
- Version 11.4.5 de MariaDB
- Version 11.4.4 de MariaDB
- Version 10.11.14 de MariaDB
- Version 10.11.13 de MariaDB
- Version 10.11.11 de MariaDB
- Version 10.11.10 de MariaDB
- Version 10.6.23 de MariaDB
- Version 10.6.22 de MariaDB
- Version 10.6.21 de MariaDB
- Version 10.6.20 de MariaDB
- Version 10.5.29 de MariaDB
- Version 10.5.28 de MariaDB
- Version 10.5.27 de MariaDB

Version 11.4.8 de MariaDB

La version 11.4.8 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Ajout du plugin `rds_security_audit` de base de données, un plugin interne Amazon RDS MariaDB qui collecte les métriques liées à la sécurité. Aucune action du client n'est requise.

Version 11.4.7 de MariaDB

La version 11.4.7 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

Version 11.4.5 de MariaDB

La version 11.4.5 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025a`.

Version 11.4.4 de MariaDB

La version 11.4.4 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Annulation de deux modifications apportées à la communauté MariaDB qui point-in-time entraînaient l'échec de la restauration (PITR). Pour plus d'informations, consultez le numéro MDEV-35528 de MariaDB Server Jira.

Version 10.11.14 de MariaDB

La version 10.11.14 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Version 10.11.13 de MariaDB

La version 10.11.13 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

Version 10.11.11 de MariaDB

La version 10.11.11 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025a`.

Version 10.11.10 de MariaDB

La version 10.11.10 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Annulation de deux modifications apportées à la communauté MariaDB qui point-in-time entraînaient l'échec de la restauration (PITR). Pour plus d'informations, consultez le numéro MDEV-35528 de MariaDB Server Jira.

Version 10.6.23 de MariaDB

La version 10.6.23 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Version 10.6.22 de MariaDB

La version 10.6.22 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

Version 10.6.21 de MariaDB

La version 10.6.21 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025a`.

Version 10.6.20 de MariaDB

La version 10.6.20 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Annulation de deux modifications apportées à la communauté MariaDB qui point-in-time entraînaient l'échec de la restauration (PITR). Pour plus d'informations, consultez le numéro MDEV-35528 de MariaDB Server Jira.

Version 10.5.29 de MariaDB

La version 10.5.29 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

Version 10.5.28 de MariaDB

La version 10.5.28 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-`citzdata2025a`.

Version 10.5.27 de MariaDB

La version 10.5.27 de MariaDB est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MariaDB et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Annulation de deux modifications apportées à la communauté MariaDB qui point-in-time entraînaient l'échec de la restauration (PITR). Pour plus d'informations, consultez le numéro MDEV-35528 de MariaDB Server Jira.

Versions de MariaDB majeures prises en charge sur Amazon RDS

Les versions majeures de RDS for MariaDB restent disponibles au moins jusqu'à la fin de vie de la version correspondante de la communauté. Vous pouvez utiliser les dates suivantes pour planifier vos cycles de test et de mise à niveau. Si Amazon étend le support d'une version de RDS for MariaDB au-delà de la date initialement prévue, ce tableau sera mis à jour pour refléter la nouvelle date.

> **Note**
>
> Les dates avec seulement un mois et une année sont approximatives et sont mises à jour avec une date exacte quand elles sont connues.
> Vous pouvez également consulter les informations relatives aux dates de support des principales versions du moteur en exécutant la AWS CLI commande describe-db-major-engine-versions ou en utilisant l'opération d'API DBMajorEngineVersionsDescribe RDS.

Version majeure de MariaDB	Date de parution communautaire	Date de parution de RDS	Date de fin de vie de la communauté	Date de fin de la prise en charge standard de RDS
MariaDB 11.4	8 août 2024	15 octobre 2024	mai 2029	mai 2029
MariaDB 10.11	16 février 2023	21 août 2023	16 février 2028	Février 2028
MariaDB 10.6	6 juillet 2021	3 février 2022	6 juillet 2026	Juillet 2026
MariaDB 10.5	24 juin 2020	21 janvier 2021	24 juin 2025	février 2026
MariaDB 10.4	18 juin 2019	6 avril 2020	18 juin 2024	février 2025

Utilisation de l'environnement de prévisualisation de base de données

Les instances de base de données RDS pour MariaDB dans l'environnement de prévisualisation de base de données sont fonctionnellement similaires aux autres instances de base de données RDS pour MariaDB. Cependant, vous ne pouvez pas utiliser l'environnement de prévisualisation de base de données pour les charges de travail de production.

Les environnements de prévisualisation présentent les limitations suivantes :

- Amazon RDS supprime toutes les instances de base de données 60 jours après leur création, en même temps que leurs sauvegardes et leurs instantanés.
- Vous ne pouvez utiliser que les stockages SSD à usage général et les stockages SSD IOPS provisionnés.
- Vous ne pouvez pas obtenir d'aide Support avec les instances de base de données. Vous pouvez plutôt publier vos questions sur la communauté de AWS questions-réponses gérée,AWS Re:post.
- Vous ne pouvez pas copier un instantané d'instance de base de données dans un environnement de production.

Les options suivantes sont prises en charge par la prévisualisation.

- Vous pouvez créer des instances de base de données à l'aide des classes d'instance de base de données db.m6i, db.r6i, db.m6g, db.m5, db.t3, db.r6g et db.r5. Pour plus d'informations sur les classes d'instances RDS, consultez Classes d'instances de base de données .

- Vous pouvez utiliser des déploiements mono-AZ et multi-AZ.
- Vous pouvez utiliser les fonctions standard de vidage et de chargement de MariaDB pour exporter des bases de données depuis ou importer des bases de données vers l'environnement Database Preview.

Fonctions non prises en charge dans l'environnement de prévisualisation de base de données

Les fonctions suivantes ne sont pas disponibles dans l'environnement de prévisualisation de base de données :

- Copie d'instantanés entre Régions
- Réplicas en lecture entre Régions
- RDS Proxy (Proxy RDS)

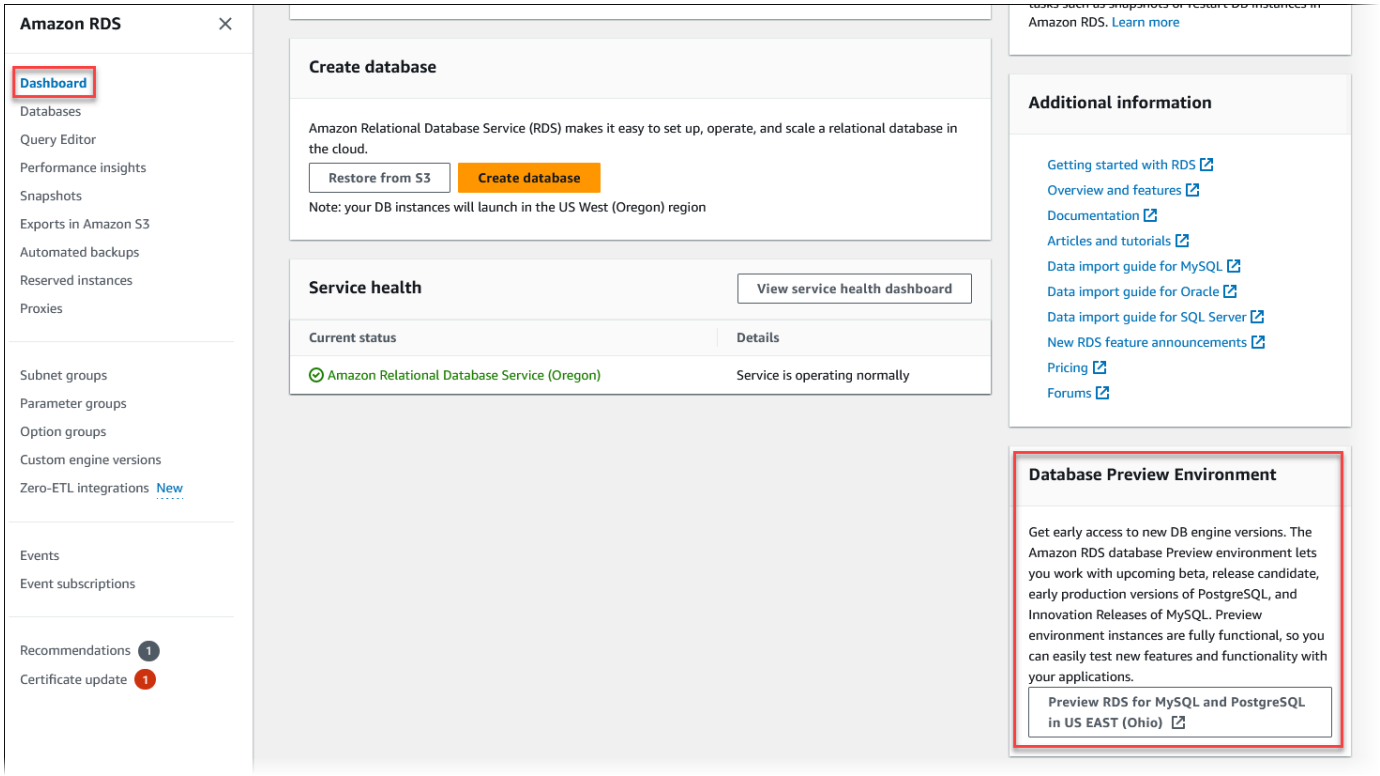
Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données

Vous pouvez créer une instance de base de données dans l'environnement Database Preview à l'aide de l'API AWS Management Console AWS CLI, ou RDS.

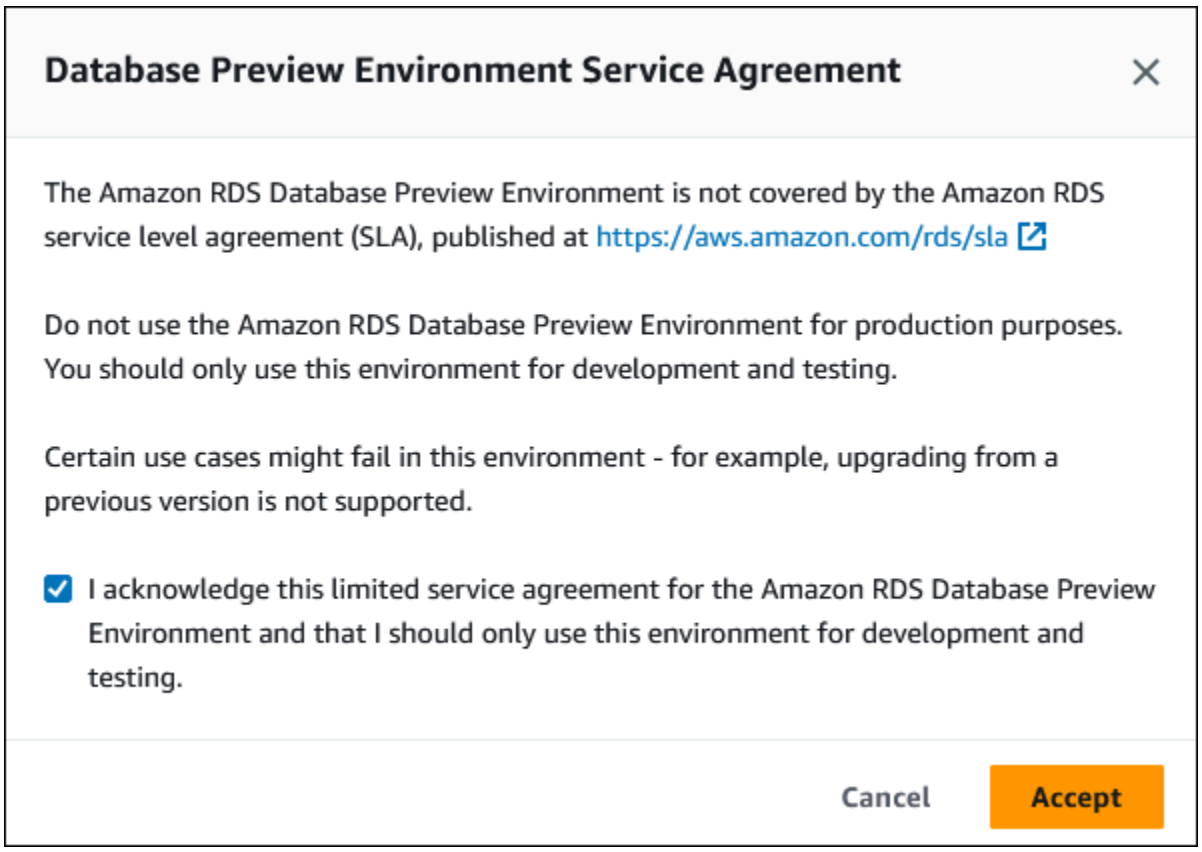
console

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Dashboard (Tableau de bord) dans le panneau de navigation.
3. Sur la page Tableau de bord, recherchez la section Environnement de prévisualisation de base de données, comme illustré dans l'image suivante.



Vous pouvez accéder directement à l'environnement de prévisualisation de base de données. Avant de poursuivre, vous devez reconnaître et accepter les limites.



4. Pour créer l'instance de base de données RDS pour MariaDB, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure console dans Création d'une instance de base de données.

AWS CLI

Pour créer une instance de base de données dans l'environnement Database Preview à l'aide du AWS CLI, utilisez le point de terminaison suivant.

```
rds-preview.us-east-2.amazonaws.com
```

Pour créer l'instance de base de données RDS pour MariaDB, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure AWS CLI dans Création d'une instance de base de données.

API RDS

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données à l'aide de l'API RDS, utilisez le point de terminaison suivant.

```
rds-preview.us-east-2.amazonaws.com
```

Pour créer l'instance de base de données RDS pour MariaDB, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure API RDS dans Création d'une instance de base de données.

MariaDB version 11.8 dans l'environnement de prévisualisation de la base de données

La version 11.8 de MariaDB est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 11.8 de MariaDB contient plusieurs améliorations décrites dans Modifications et améliorations apportées à MariaDB 11.8. Cette version inclut également la prise en charge du type de données vectorielles, de l'indexation et de la recherche. Pour plus d'informations, consultez la section Présentation des vecteurs dans la documentation de MariaDB.

Vous pouvez utiliser l'environnement Database Preview pour tester vos charges de travail par rapport à cette version avant qu'elle ne soit entièrement disponible Régions AWS pour les charges de travail de production. Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement de prévisualisation de base de données”. Pour accéder à l'environnement en préversion à partir de la console, sélectionnez rds-preview/.

MariaDB version 11.7 dans l'environnement de prévisualisation de la base de données

La version 11.7 de MariaDB est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 11.7 de MariaDB contient plusieurs améliorations décrites dans Modifications et améliorations apportées à MariaDB 11.7. Cette version inclut également la prise en charge du type de données vectorielles, de l'indexation et de la recherche. Pour plus d'informations, consultez la section Présentation des vecteurs dans la documentation de MariaDB.

Vous pouvez utiliser l'environnement Database Preview pour tester vos charges de travail par rapport à cette version avant qu'elle ne soit entièrement disponible Régions AWS pour les charges de travail de production. Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement de prévisualisation de base de données”. Pour accéder à l'environnement en préversion à partir de la console, sélectionnez rds-preview/.

MariaDB version 11.4 dans l'environnement de prévisualisation de la base de données

La version 11.4 de MariaDB est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 11.4 de MariaDB contient plusieurs améliorations décrites dans Modifications et améliorations apportées à MariaDB 11.4. Vous pouvez utiliser l'environnement Database Preview pour tester vos charges de travail par rapport à cette version avant qu'elle ne soit entièrement disponible Régions AWS pour les charges de travail de production.

Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement de prévisualisation de base de données”. Pour accéder à l'environnement en préversion à partir de la console, sélectionnez rds-preview/.

Versions rendues obsolètes pour Amazon RDS for MariaDB

Les versions 10.0, 10.1, 10.2 et 10.3 d'Amazon RDS for MariaDB sont obsolètes.

Pour plus d'informations sur la politique d'obsolescence d'Amazon RDS pour MariaDB, consultez Amazon RDS. FAQs

Connexion à votre instance de base de données MariaDB

Une fois qu'Amazon RDS a provisionné votre instance de base de données, vous pouvez utiliser n'importe quelle application cliente MariaDB standard pour vous connecter à l'instance. Dans la chaîne de connexion, vous spécifiez l'adresse DNS (système de noms de domaine) du point de terminaison de l'instance de base de données en tant que paramètre hôte. Vous spécifiez également le numéro de port du point de terminaison de l'instance de base de données en tant que paramètre de port.

Vous pouvez vous connecter à une instance de base de données Amazon RDS pour MariaDB à l'aide d'outils tels que le client de ligne de commande MySQL. Pour plus d'informations sur l'utilisation du client de ligne de commande MySQL, consultez Client de ligne de commande mysql dans la documentation MariaDB. Heidi est une application basée sur l'interface utilisateur graphique que vous pouvez utiliser pour la connexion. Pour en savoir plus, consultez la page Télécharger HeidiSQL. Pour plus d'informations sur l'installation de MySQL (y compris le client de ligne de commande MySQL), consultez Installation et mise à niveau de MySQL.

La plupart des distributions Linux incluent le client MariaDB au lieu du client MySQL Oracle. Pour installer le client de ligne de commande MySQL sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install mariadb105
```

Pour installer le client de ligne de commande MySQL sur Amazon Linux 2, exécutez la commande suivante :

```
sudo yum install mariadb
```

Pour installer le client de ligne de commande MySQL sur la plupart des distributions Linux basées sur DEB, exécutez la commande suivante.

```
apt-get install mariadb-client
```

Pour vérifier la version de votre client de ligne de commande MySQL, exécutez la commande suivante.

```
mysql --version
```

Pour lire la documentation MySQL pour votre version de client actuelle, exécutez la commande suivante.

```
man mysql
```

Pour se connecter à une instance de base de données depuis l'extérieur d'un VPC (Virtual Private Cloud) basé sur Amazon VPC, l'instance de base de données doit être accessible publiquement. En outre, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité de l'instance de base de données, et d'autres exigences doivent être satisfaites. Pour plus d'informations, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Vous pouvez utiliser le chiffrement SSL sur les connexions à une instance de base de données MariaDB. Pour plus d'informations, veuillez consulter SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS.

Pour rechercher et vous connecter à une instance de base de données RDS pour MariaDB, consultez les rubriques suivantes.

Rubriques

- Recherche des informations de connexion pour une instance de base de données MariaDB
- Connexion depuis le client de SQL ligne de commande My (non crypté) pour MariaDB RDS
- Connexion RDS à MariaDB avec le AWS JDBC pilote et le pilote Python ; AWS
- Dépannage des connexions à votre instance de base de données MariaDB

Recherche des informations de connexion pour une instance de base de données MariaDB

Les informations de connexion d'une instance de base de données incluent son point de terminaison, son port et un utilisateur de base de données valide, tel que l'utilisateur principal. Par exemple, supposons qu'une valeur de point de terminaison soit `mydb.123456789012.us-east-1.rds.amazonaws.com`. Dans ce cas, la valeur du port est `3306`, et l'utilisateur de base de données est `admin`. Compte tenu de ces informations, vous spécifiez les valeurs suivantes dans une chaîne de connexion :

- Pour l'hôte ou le nom ou le DNS nom d'hôte, spécifiez`mydb.123456789012.us-east-1.rds.amazonaws.com`.

- Pour un port, spécifiez `3306`.
- Pour l'utilisateur, spécifiez `admin`.

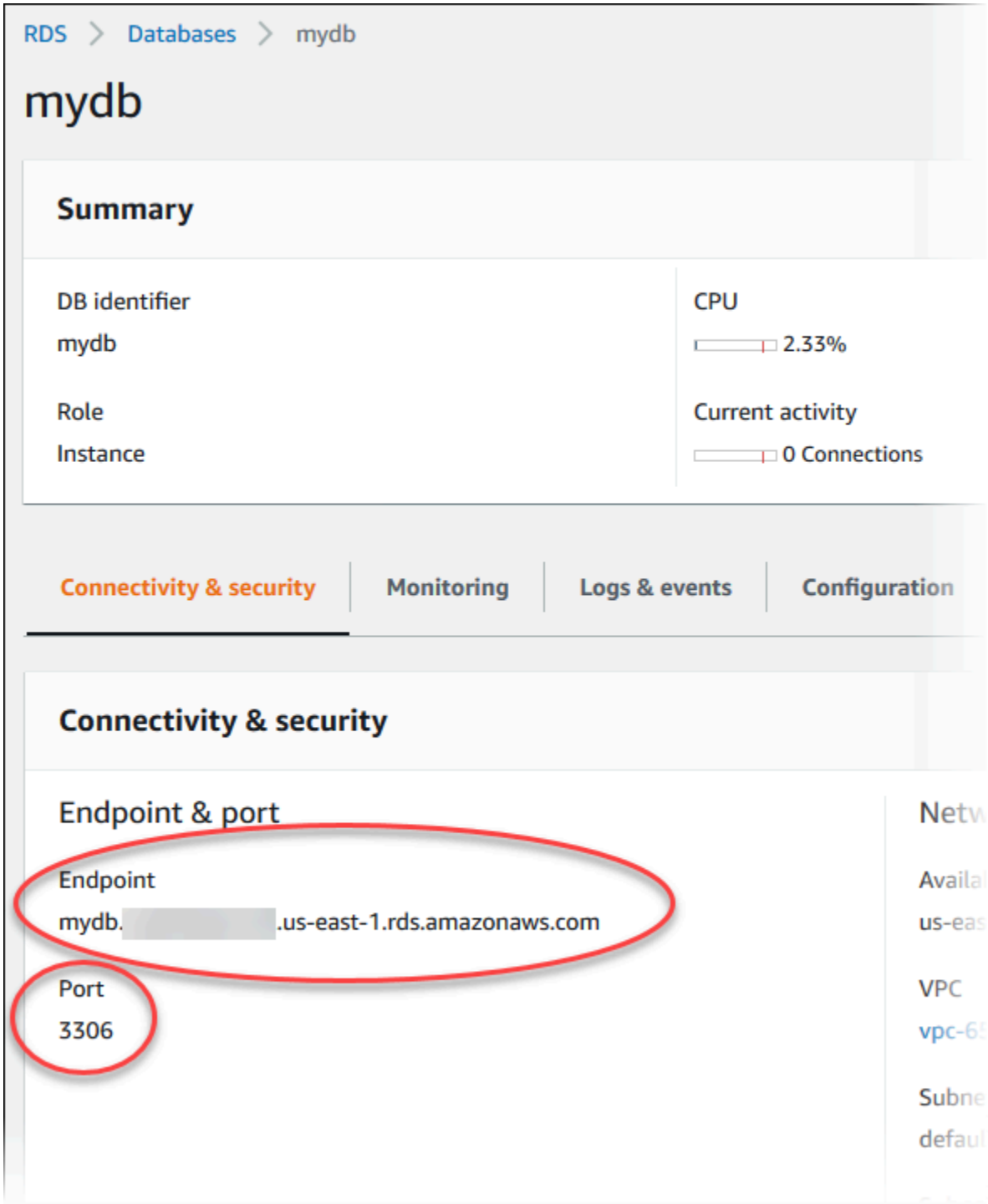
Pour vous connecter à une instance de base de données, utilisez n'importe quel client pour le moteur de base de données MariaDB. Par exemple, vous pouvez utiliser le client de SQL ligne de commande My ou My SQL Workbench.

Pour trouver les informations de connexion d'une instance de base de données, vous pouvez utiliser la AWS Management Consoledescribe-db-instancescommande, the AWS Command Line Interface (AWS CLI) ou l'escribeDBInstancesopération Amazon RDS API D pour répertorier ses détails.

Console

Pour trouver les informations de connexion d'une instance de base de données dans AWS Management Console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases (Bases de données) pour afficher la liste de vos instances de base de données.
3. Choisissez le nom de l'instance de base de données MariaDB pour afficher ses détails.
4. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



5. Si vous devez rechercher le nom d'utilisateur principal, choisissez l'onglet Configuration et affichez la valeur Master username (Identifiant principal.

AWS CLI

Pour trouver les informations de connexion d'une instance de base de données MariaDB à l'aide de, exécutez AWS CLI la commande. describe-db-instances Dans l'appel, recherchez l'ID d'instance de base de données, le point de terminaison, le port et l'identifiant principal.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-instances \
  --filters "Name=engine,Values=mariadb" \
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Dans Windows:

```
aws rds describe-db-instances ^
  --filters "Name=engine,Values=mariadb" ^
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Votre sortie doit ressembler à ce qui suit.

```
[
    [
        "mydb1",
        "mydb1.123456789012.us-east-1.rds.amazonaws.com",
        3306,
        "admin"
    ],
    [
        "mydb2",
        "mydb2.123456789012.us-east-1.rds.amazonaws.com",
        3306,
        "admin"
    ]
]
```

RDS API

Pour trouver les informations de connexion d'une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'escribeDBInstancesopération D. Dans la sortie, recherchez les valeurs de l'adresse du point de terminaison, du port du point de terminaison et du nom d'utilisateur principal.

Connexion depuis le client de SQL ligne de commande My (non crypté) pour MariaDB RDS

> **Important**
>
> N'utilisez une SQL connexion My non cryptée que lorsque le client et le serveur sont connectés VPC et que le réseau est fiable. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Connexion à votre instance de base de données MariaDB sur RDS Amazon TLS avec/ depuis le client SSL de ligne de commande SQL My (crypté).

Pour vous connecter à une instance de base de données à l'aide du client de SQL ligne de commande My, entrez la commande suivante dans une invite de commande sur un ordinateur client. Vous êtes alors connecté à une base de données sur une instance de base de données MariaDB. Remplacez le DNS nom (point de terminaison) de votre instance de base de données par *<endpoint>* et le nom d'utilisateur principal que vous avez utilisé pour *<mymasteruser>*. Entrez le mot de passe principal que vous avez utilisé lorsque vous y êtes invité.

```
mysql -h <endpoint> -P 3306 -u <mymasteruser> -p
```

Après avoir entré le mot de passe pour l'utilisateur, le résultat suivant doit normalement s'afficher.

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.6.10-MariaDB-log Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Connexion RDS à MariaDB avec le AWS JDBC pilote et le pilote Python ; AWS

Connectez-vous à quatre instances RDS de base de données MariaDB avec AWS JDBC le pilote et le AWS pilote Python. Pour plus d'informations, consultez les rubriques suivantes.

Rubriques

- Connexion RDS à MariaDB avec le pilote Amazon Web Services ()AWS JDBC
- Connexion RDS à MariaDB avec le pilote Python Amazon Web Services ()AWS

Connexion RDS à MariaDB avec le pilote Amazon Web Services ()AWS JDBC

Le JDBC pilote Amazon Web Services (AWS) est conçu comme un JDBC wrapper avancé. Ce wrapper complète et étend les fonctionnalités d'un JDBC pilote existant. Le pilote est compatible directement avec le pilote communautaire My SQL Connector/J et le pilote communautaire MariaDB Connector/J.

Pour installer le AWS JDBC pilote, ajoutez le fichier .jar du AWS JDBC pilote (situé dans l'applicationCLASSPATH) et conservez les références au pilote communautaire correspondant. Mettez à jour le URL préfixe de connexion correspondant comme suit :

- `jdbc:mysql://` sur `jdbc:aws-wrapper:mysql://`
- `jdbc:mariadb://` sur `jdbc:aws-wrapper:mariadb://`

Pour plus d'informations sur le AWS JDBC pilote et des instructions complètes pour son utilisation, consultez le GitHub référentiel de JDBC pilotes Amazon Web Services (AWS).

Connexion RDS à MariaDB avec le pilote Python Amazon Web Services ()AWS

Le pilote Python Amazon Web Services (AWS) est conçu comme un wrapper Python avancé. Ce wrapper complète et étend les fonctionnalités du pilote open source Psycopg. Le pilote AWS Python prend en charge les versions 3.8 et supérieures de Python. Vous pouvez installer le `aws-advanced-python-wrapper` package à l'aide de la `pip` commande, en même temps que les packages `psycopg` open source.

Pour plus d'informations sur le pilote AWS Python et des instructions complètes pour son utilisation, consultez le GitHub référentiel de pilotes Python Amazon Web Services (AWS).

Dépannage des connexions à votre instance de base de données MariaDB

Les deux causes les plus courantes d'échec de connexion à une nouvelle instance de base de données sont :

- L'instance de base de données a été créée à l'aide d'un groupe de sécurité qui n'autorise pas les connexions depuis l'appareil ou l'EC2instance Amazon sur laquelle l'application ou l'utilitaire MariaDB est exécuté. L'instance de base de données doit disposer d'un groupe de VPC sécurité qui autorise les connexions. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.

  Vous pouvez ajouter ou modifier une règle entrante dans le groupe de sécurité. Pour Source, choisissez Mon IP. Cela autorise à accéder à l'instance de base de données à partir de l'adresse IP détectée dans votre navigateur.
- L'instance de base de données a été créée à l'aide du port par défaut 3306, et votre entreprise dispose de règles de pare-feu bloquant les connexions à ce port depuis les appareils de votre réseau d'entreprise. Pour corriger le problème, recréez l'instance avec un port différent.

Pour de plus amples informations sur les problèmes de connexion, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

Sécurisation des connexions d'instance de base de données MariaDB

Vous pouvez gérer la sécurité de vos instances de base de données MariaDB.

Rubriques

- Sécurité de MariaDB sur Amazon RDS
- Utilisation des plugins de validation de mot de passe RDS pour MariaDB
- Chiffrement des connexions client SSL avec/ TLS vers les instances de base de données MariaDB sur Amazon RDS
- Mise à jour des applications pour se connecter aux instances MariaDB à l'aide de nouveaux certificats/SSLTLS

Sécurité de MariaDB sur Amazon RDS

La sécurité des instances de base de données MariaDB est gérée à trois niveaux :

- AWS Identity and Access Management contrôle qui peut effectuer des actions de RDS gestion Amazon sur les instances de base de données. Lorsque vous vous connectez à AWS l'aide IAM d'informations d'identification, votre IAM compte doit disposer de IAM politiques accordant les autorisations requises pour effectuer les opérations RDS de gestion Amazon. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.
- Lorsque vous créez une instance de base de données, vous utilisez un groupe VPC de sécurité pour contrôler quels appareils et EC2 instances Amazon peuvent ouvrir des connexions au point de terminaison et au port de l'instance de base de données. Ces connexions peuvent être établies à l'aide de Secure Socket Layer (SSL) et de Transport Layer Security (TLS). En outre, les règles de pare-feu de votre entreprise peuvent contrôler si les appareils en cours d'exécution dans votre entreprise peuvent ouvrir des connexions à l'instance de base de données.
- Une fois qu'une connexion a été ouverte sur une instance de base de données MariaDB, l'authentification de la connexion et les autorisations sont appliquées de la même manière que dans une instance autonome de MariaDB. Les commandes telles que `CREATE USER`, `RENAME USER`, `GRANT`, `REVOKE` et `SET PASSWORD` fonctionnent de la même façon que dans les bases de données autonomes, comme le fait la modification directe des tables du schéma de base de données.

Lorsque vous créez une RDS instance de base de données Amazon, l'utilisateur principal dispose des privilèges par défaut suivants :

- `alter`
- `alter routine`
- `create`
- `create routine`
- `create temporary tables`
- `create user`
- `create view`
- `delete`
- `drop`
- `event`
- `execute`
- `grant option`
- `index`
- `insert`
- `lock tables`
- `process`
- `references`
- `reload`

  Ce privilège est limité sur les instances de base de données MariaDB. Il n'autorise pas l'accès aux `FLUSH TABLES WITH READ LOCK` opérations du `FLUSH LOGS` système d'exploitation.
- `replication client`
- `replication slave`
- `select`
- `show create routine`

  Ce privilège s'applique uniquement aux instances de base de données MariaDB exécutant la version 11.4 et supérieure.

- `show databases`
- `show view`
- `trigger`
- `update`

Pour plus d'informations sur ces privilèges, consultez Gestion des comptes d'utilisateur dans la documentation MariaDB.

> **Note**
>
> Bien que vous puissiez supprimer l'utilisateur principal sur une instance de base de données, il est déconseillé d'agir ainsi. Pour recréer l'utilisateur principal, utilisez le `ModifyDBInstance` API ou le `modify-db-instance` AWS CLI et spécifiez un nouveau mot de passe utilisateur principal avec le paramètre approprié. Si l'utilisateur maître n'existe pas dans l'instance, il est créé avec le mot de passe spécifié.

Pour fournir des services de gestion à chaque instance de base de données, l'utilisateur `rdsadmin` est créé lors de la création de l'instance de base de données. Les tentatives de supprimer, renommer et modifier le mot de passe du compte `rdsadmin`, ou d'en modifier les privilèges, génèrent une erreur.

Pour autoriser la gestion de l'instance de base de données, les commandes standard `kill` et `kill_query` ont fait l'objet de restrictions. Les RDS commandes Amazon `mysql.rds_killmysql.rds_kill_query`, et `mysql.rds_kill_query_id` sont fournies pour être utilisées dans MariaDB et également dans SQL My afin que vous puissiez mettre fin aux sessions utilisateur ou aux requêtes sur les instances de base de données.

Utilisation des plugins de validation de mot de passe RDS pour MariaDB

À partir RDS de la version 11.4 de MariaDB, vous pouvez utiliser les plugins de validation de mot de passe suivants pour améliorer la sécurité de vos connexions à la base de données :

- simple_password_check — vérifie si un mot de passe contient au moins un certain nombre de caractères d'un type spécifique.
- cracklib_password_check — vérifie si un mot de passe apparaît dans un fichier de dictionnaire de la bibliothèque. CrackLib

Pour activer ces plug-ins, définissez la valeur du paramètre `simple_password_check` ou sur `FORCE_PLUS_PERMANENT` dans le groupe `cracklib_password_check` de paramètres de base de données associé à l'instance de base de données. Lorsque cette valeur est définie, le plugin ne peut pas être désinstallé à l'aide de l'`UNINSTALL PLUGIN`instruction au moment de l'exécution.

Pour désactiver ces plug-ins, définissez la valeur du paramètre `simple_password_check` ou sur `OFF` dans le groupe `cracklib_password_check` de paramètres de base de données associé à l'instance de base de données. Lorsque cette valeur est définie, les règles de validation du plugin ne s'appliquent plus aux nouveaux mots de passe.

Pour plus d'informations sur la définition des valeurs des paramètres dans les groupes de paramètres, consultezthe section called “Modification de paramètres dans un groupe de paramètres de bases de données”.

Chiffrement des connexions client SSL avec/ TLS vers les instances de base de données MariaDB sur Amazon RDS

Secure Sockets Layer (SSL) est un protocole standard pour sécuriser les connexions réseau entre le client et le serveur. Après SSL la version 3.0, le nom a été changé en Transport Layer Security (TLS). Amazon RDS prend en charge le SSL TLS chiffrement/chiffrement pour les instances de base de données MariaDB. À l'aide deSSL/TLS, vous pouvez chiffrer une connexion entre votre client d'application et votre instance de base de données MariaDB. SSL/TLSle support est disponible dans tous les domaines Régions AWS.

Avec AmazonRDS, vous pouvez sécuriser les données en transit en chiffrant les connexions client aux instances de base de données MariaDB SSL avecTLS/, en SSL exigeantTLS/pour toutes les connexions à une instance de base de données MariaDB et en vous connectant depuis le client de ligne de commande My avec/(crypté). SQL SSL TLS Les sections suivantes fournissent des conseils sur la configuration et l'utilisation deSSL/TLSencryption pour les instances de base de données MariaDB sur Amazon. RDS

Rubriques

- SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS
- ExigerSSL/TLSpour des comptes utilisateur spécifiques à une instance de base de données MariaDB sur Amazon RDS
- ExigerSSL/TLSpour toutes les connexions à une instance de base de données MariaDB sur Amazon RDS

- Connexion à votre instance de base de données MariaDB sur RDS Amazon TLS avec/ depuis le client SSL de ligne de commande SQL My (crypté)

SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS

Amazon RDS crée un SSL/TLS certificate and installs the certificate on the DB instance when Amazon RDS provisions the instance. These certificates are signed by a certificate authority. The SSL/TLS certificate includes the DB instance endpoint as the Common Name (CN) for the SSL/TLS certificat pour se prémunir contre les attaques par usurpation d'identité.

Un TLS certificatSSL/créé par Amazon RDS est l'entité racine fiable et devrait fonctionner dans la plupart des cas, mais il peut échouer si votre application n'accepte pas les chaînes de certificats. Si votre application n'accepte pas les chaînes de certificats, essayez d'utiliser un certificat intermédiaire pour vous connecter à votre Région AWS. Par exemple, vous devez utiliser un certificat intermédiaire pour vous connecter aux AWS GovCloud (US) régions avecSSL/TLS.

Pour plus d'informations sur le téléchargement de certificats, veuillez consulter . Pour plus d'informations sur l'utilisation deSSL/TLSavec MySQL, consultezMise à jour des applications pour se connecter aux instances MariaDB à l'aide de nouveaux certificats/SSLTLS.

Amazon RDS pour MariaDB prend en charge les versions 1.3, 1.2, 1.1 et 1.0 de Transport Layer Security TLS (). TLSle support dépend de la version mineure de MariaDB. Le tableau suivant montre la TLS prise en charge des versions mineures de MariaDB.

Version de TLS	MariaDB 11.4	MariaDB 10.11	MariaDB 10.6	MariaDB 10.5	MariaDB 10.4
TLS1.3	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures
TLS1.2	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures	Toutes les versions mineures
TLS1.1	Non pris en charge	10.11.6 et versions antérieures	10.6.16 et versions antérieures	10.5.23 et versions antérieures	10.4.32 et versions antérieures

Version de TLS	MariaDB 11.4	MariaDB 10.11	MariaDB 10.6	MariaDB 10.5	MariaDB 10.4
TLS 1.0	Non pris en charge	10.11.6 et versions antérieures	10.6.16 et versions antérieures	10.5.23 et versions antérieures	10.4.32 et versions antérieures

ExigerSSL/TLSpour des comptes utilisateur spécifiques à une instance de base de données MariaDB sur Amazon RDS

Vous pouvez exiger le TLS chiffrementSSL/pour les connexions de compte utilisateur spécifiées à vos instances de base de données MariaDB sur Amazon. RDS La protection des informations sensibles contre tout accès non autorisé ou toute interception est essentielle pour appliquer les politiques de sécurité lorsque la confidentialité des données est une préoccupation.

Pour exiger des TLS connexionsSSL/pour des comptes utilisateurs spécifiques, utilisez l'une des instructions suivantes, en fonction de votre SQL version My version, pour exiger des TLS connexionsSSL/sur le compte utilisateur`encrypted_user`.

Pour cela, utilisez l'instruction suivante.

```
ALTER USER 'encrypted_user'@'%' REQUIRE SSL;
```

Pour plus d'informations sur les TLS connexionsSSL/avec MariaDB, consultez la section Sécurisation des connexions pour le client et le serveur dans la documentation de MariaDB.

ExigerSSL/TLSpour toutes les connexions à une instance de base de données MariaDB sur Amazon RDS

Utilisez le `require_secure_transport` paramètre pour exiger que toutes les connexions utilisateur à votre instance de base de données MariaDB utilisent/. SSL TLS Par défaut, le paramètre `require_secure_transport` est défini sur `OFF`. Vous pouvez définir le `require_secure_transport` paramètre sur pour `ON` requérirSSL/TLSpour les connexions à votre instance de base de données.

> **Note**
>
> Le paramètre `require_secure_transport` est uniquement prise en charge pour MariaDB versions 10.5 et ultérieures.

Vous pouvez définir la valeur du paramètre `require_secure_transport` en mettant à jour le groupe de paramètres de base de données pour votre instance de base de données. Vous n'avez pas besoin de redémarrer votre instance de base de données pour que la modification prenne effet.

Lorsque le paramètre `require_secure_transport` est défini sur `ON` pour une instance de base de données, un client de base de données peut s'y connecter s'il peut établir une connexion chiffrée. Sinon, un message d'erreur similaire au suivant est renvoyé au client :

```
ERROR 1045 (28000): Access denied for user 'USER'@'localhost' (using password: YES |
 NO)
```

Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Pour plus d'informations sur le paramètre `require_secure_transport`, consultez la documentation MariaDB.

Connexion à votre instance de base de données MariaDB sur RDS Amazon TLS avec/ depuis le client SSL de ligne de commande SQL My (crypté)

Les paramètres du programme `mysql` client sont légèrement différents si vous utilisez la version My SQL 5.7, la version My SQL 8.0 ou la version MariaDB.

Pour savoir quelle version vous avez, exécutez la commande `mysql` avec l'option `--version`. Dans l'exemple suivant, la sortie indique que le programme client provient de MariaDB.

```
$ mysql --version
mysql  Ver 15.1 Distrib 10.5.15-MariaDB, for osx10.15 (x86_64) using readline 5.1
```

La plupart des distributions Linux, telles qu'Amazon Linux, CentOS et DebianSUSE, ont remplacé My par SQL MariaDB, et la version qu'elles contiennent provient de MariaDB. `mysql`

Pour vous connecter à votre instance de base de données à l'aide deSSL/TLS, procédez comme suit :

Pour vous connecter à une instance de base de données SSL avec/ à TLS l'aide du client de ligne de SQL commande My

1. Téléchargez un certificat racine qui fonctionne pour tous Régions AWS.

   Pour plus d'informations sur le téléchargement de certificats, veuillez consulter .

2. Utilisez un client de SQL ligne de commande My pour vous connecter à une instance de base de données avec le chiffrementSSL/TLS. Pour le `-h` paramètre, remplacez le DNS nom (point de terminaison) de votre instance de base de données. Remplacez le `--ssl-ca` paramètre par le nom du fichierSSL/TLScertificate. Pour le paramètre `-P`, remplacez le port pour votre instance de base de données. Pour le paramètre `-u`, remplacez le nom d'utilisateur d'un utilisateur de base de données valide, par exemple l'utilisateur principal. Entrez le mot de passe de l'utilisateur principal quand vous y êtes invité.

   L'exemple suivant montre comment lancer le client à l'aide du paramètre `--ssl-ca` en utilisant le client MariaDB.

   ```
   mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl -P 3306 -u myadmin -p
   ```

   Pour exiger que la TLS connexionSSL/vérifie le point de terminaison de l'instance de base de données par rapport au point de terminaison indiqué dans le TLS certificatSSL/, entrez la commande suivante :

   ```
   mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl-verify-server-cert -P 3306 -u myadmin -p
   ```

   L'exemple suivant montre comment lancer le client à l'aide du `--ssl-ca` paramètre à l'aide du client My SQL 5.7 ou version ultérieure :

   ```
   mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl-mode=REQUIRED -P 3306 -u myadmin -p
   ```

3. Entrez le mot de passe de l'utilisateur principal quand vous y êtes invité.

Vous devez visualiser des résultats similaires à ce qui suit.

```
Welcome to the MariaDB monitor.  Commands end with ; or \g.
```

```
Your MariaDB connection id is 31
Server version: 10.6.10-MariaDB-log Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Mise à jour des applications pour se connecter aux instances MariaDB à l'aide de nouveaux certificats/SSLTLS

Depuis le 13 janvier 2023, Amazon RDS a publié de nouveaux certificats d'autorité de certification (CA) pour la connexion à vos RDS instances de base de données à l'aide de Secure Socket Layer ou de Transport Layer Security (SSL/TLS). Vous trouverez ci-après des informations sur la mise à jour de vos applications afin d'utiliser les nouveaux certificats.

Cette rubrique peut vous aider à déterminer si vos applications nécessitent une vérification du certificat pour se connecter à vos instances de bases de données.

> **Note**
>
> Certaines applications sont configurées pour se connecter à MariaDB uniquement si la vérification du certificat sur le serveur s'effectue avec succès. Pour ces applications, vous devez mettre à jour les magasins d'approbations des applications clientes afin d'inclure les nouveaux certificats de l'autorité de certification.
> Vous pouvez spécifier les SSL modes suivants : `disabled`, `preferred`, et `required`.
> Lorsque vous utilisez le `preferred` SSL mode et que le certificat CA n'existe pas ou n'est pas à jour, la connexion revient à ne pas être utilisée SSL et se connecte toujours correctement.
> Nous recommandons d'éviter le mode `preferred`. En mode `preferred`, si la connexion rencontre un certificat non valide, elle cesse d'utiliser le chiffrement et continue sans chiffrement.

Une fois que vous avez mis à jour les certificats de l'autorité de certification dans les magasins d'approbations des applications clientes, vous pouvez soumettre les certificats de vos instances de bases de données à une rotation. Nous vous recommandons vivement de tester ces procédures

dans un environnement de développement ou intermédiaire avant de les implémenter dans vos environnements de production.

Pour de plus amples informations sur la rotation de certificats, veuillez consulter Rotation de votre certificat SSL/TLS. Pour en savoir plus sur le téléchargement de certificats, consultez . Pour plus d'informations sur l'utilisation deSSL/TLSavec les instances de base de données MariaDB, consultez. SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS

Rubriques

- Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter
- Mise à jour du magasin d'approbations de votre application
- Exemple de code Java pour établir des SSL connexions

Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter

Vous pouvez vérifier si les JDBC clients et Mes SQL clients ont besoin d'une vérification de certificat pour se connecter.

JDBC

L'exemple suivant avec My SQL Connector/J 8.0 montre un moyen de vérifier les propriétés de JDBC connexion d'une application afin de déterminer si les connexions réussies nécessitent un certificat valide. Pour plus d'informations sur toutes les options de JDBC connexion pour MySQL, consultez la section Propriétés de configuration dans la SQL documentation My.

Lorsque vous utilisez My SQL Connector/J 8.0, une SSL connexion doit être vérifiée par rapport au certificat CA du serveur si les propriétés de votre connexion sont `sslMode` définies sur `VERIFY_CA` ou`VERIFY_IDENTITY`, comme dans l'exemple suivant.

```
Properties properties = new Properties();
properties.setProperty("sslMode", "VERIFY_IDENTITY");
properties.put("user", DB_USER);
properties.put("password", DB_PASSWORD);
```

> **Note**
>
> Si vous utilisez le My SQL Java Connector v5.1.38 ou version ultérieure, ou le My SQL Java Connector v8.0.9 ou version ultérieure pour vous connecter à vos bases de données, même si vous n'avez pas explicitement configuré vos applications pour utiliserSSL/TLSlors de la connexion à vos bases de données, ces pilotes clients utilisent par défaut/. SSL TLS En outre, lorsqu'ils utilisentSSL/TLS, ils vérifient partiellement le certificat et ne parviennent pas à se connecter si le certificat du serveur de base de données est expiré.
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Mon SQL

Les exemples suivants avec My SQL Client montrent deux manières de vérifier Ma SQL connexion d'un script afin de déterminer si les connexions réussies nécessitent un certificat valide. Pour plus d'informations sur toutes les options de connexion avec My SQL Client, consultez la section Configuration côté client pour les connexions chiffrées dans la documentation MySQL.

Lorsque vous utilisez le client My SQL 5.7 ou My SQL 8.0, une SSL connexion doit être vérifiée par rapport au certificat CA du serveur, s'il s'agit de l'`--ssl-mode`option que vous spécifiez `VERIFY_CA` ou`VERIFY_IDENTITY`, comme dans l'exemple suivant.

```
mysql -h mysql-database.rds.amazonaws.com -uadmin -ppassword --ssl-ca=/tmp/ssl-cert.pem
 --ssl-mode=VERIFY_CA
```

Lorsque vous utilisez le client My SQL 5.6, une SSL connexion doit être vérifiée par rapport au certificat CA du serveur si vous spécifiez l'`--ssl-verify-server-cert`option, comme dans l'exemple suivant.

```
mysql -h mysql-database.rds.amazonaws.com -uadmin -ppassword --ssl-ca=/tmp/ssl-cert.pem
 --ssl-verify-server-cert
```

Mise à jour du magasin d'approbations de votre application

Pour plus d'informations sur la mise à jour du trust store pour Mes SQL applications, consultez la section Utilisation deTLS/SSLavec MariaDB Connector/J dans la documentation de MariaDB.

Pour plus d'informations sur le téléchargement du certificat racine, consultez .

Pour obtenir des exemples de scripts qui importent des certificats, consultez Exemple de script pour importer les certificats dans votre magasin d'approbations.

> **Note**
>
> Lors de la mise à jour du magasin d'approbations, vous pouvez conserver les certificats plus anciens en complément de l'ajout des nouveaux certificats.

Si vous utilisez le pilote MariaDB JDBC Connector/J dans une application, définissez les propriétés suivantes dans l'application.

```
System.setProperty("javax.net.ssl.trustStore", certs);
System.setProperty("javax.net.ssl.trustStorePassword", "password");
```

Lorsque vous démarrez l'application, définissez les propriétés suivantes.

```
java -Djavax.net.ssl.trustStore=/path_to_truststore/MyTruststore.jks -
Djavax.net.ssl.trustStorePassword=my_truststore_password com.companyName.MyApplication
```

> **Note**
>
> Spécifiez des mots de passe autres que ceux indiqués ici, en tant que bonne pratique de sécurité.

Exemple de code Java pour établir des SSL connexions

L'exemple de code suivant montre comment configurer la SSL connexion à l'aide deJDBC.

```
private static final String DB_USER = "admin";

        private static final String DB_USER = "user name";
        private static final String DB_PASSWORD = "password";
        // This key store has only the prod root ca.
        private static final String KEY_STORE_FILE_PATH = "file-path-to-keystore";
        private static final String KEY_STORE_PASS = "keystore-password";

    public static void main(String[] args) throws Exception {
        Class.forName("org.mariadb.jdbc.Driver");

        System.setProperty("javax.net.ssl.trustStore", KEY_STORE_FILE_PATH);
        System.setProperty("javax.net.ssl.trustStorePassword", KEY_STORE_PASS);

        Properties properties = new Properties();
        properties.put("user", DB_USER);
        properties.put("password", DB_PASSWORD);


        Connection connection = DriverManager.getConnection("jdbc:mysql://ssl-mariadb-
public.cni62e2e7kwh.us-east-1.rds.amazonaws.com:3306?useSSL=true",properties);
        Statement stmt=connection.createStatement();

        ResultSet rs=stmt.executeQuery("SELECT 1 from dual");

        return;
    }
```

> **Important**
>
> Une fois que vous avez déterminé que vos connexions à la base de données utilisentSSL/TLSet que vous avez mis à jour le magasin de confiance des applications, vous pouvez mettre à jour votre base de données pour utiliser les certificats rds-ca-rsa 2048-g1. Pour obtenir des instructions, veuillez consulter l'étape 3 dans Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données.

Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Amélioration des performances des requêtes pour RDS for MariaDB avec Amazon RDS Optimized Reads

Vous pouvez accélérer le traitement des requêtes pour RDS for MariaDB avec Amazon RDS Optimized Reads. Une instance de base de données RDS for MariaDB qui utilise RDS Optimized Reads peut traiter les requêtes jusqu'à 2 fois plus rapidement qu'une instance de base de données qui ne l'utilise pas.

Rubriques

- Présentation de RDS Optimized Reads
- Cas d'utilisation pour RDS Optimized Reads
- Bonnes pratiques relatives à RDS Optimized Reads
- Utilisation de RDS Optimized Reads
- Surveillance des instances de base de données qui utilisent RDS Optimized Reads
- Limites pour RDS Optimized Reads

Présentation de RDS Optimized Reads

Lorsque vous utilisez une instance de base de données RDS for MariaDB sur laquelle RDS Optimized Reads est activé, votre instance de base de données présente des performances de requête plus rapides grâce à l'utilisation d'un stockage d'instances. Un stockage d'instance fournit un stockage temporaire de niveau bloc pour votre instance de base de données. Le stockage repose sur des disques SSD (Solid State Drive) NVMe (Non-Volatile Memory Express) qui sont physiquement attachés au serveur hôte. Ce stockage est optimisé pour une faible latence, de hautes performances d'E/S aléatoires et un haut débit de lecture séquentielle.

RDS Optimized Reads est activé par défaut lorsqu'une instance de base de données utilise une classe d'instances de base de données avec un stockage d'instances, tel que db.m5d ou db.m6gd. Avec RDS Optimized Reads, certains objets temporaires sont stockés dans le stockage d'instances. Ces objets temporaires incluent des fichiers temporaires internes, des tables temporaires internes sur disque, des fichiers de mappage de mémoire et des fichiers de cache de journal binaire (binlog). Pour plus d'informations sur le stockage d'instances, consultez Stockage d'instances Amazon EC2 dans le Guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux.

Les charges de travail qui génèrent des objets temporaires dans MariaDB pour le traitement des requêtes peuvent tirer parti du stockage d'instances pour accélérer le traitement des requêtes.

Ce type de charge de travail inclut les requêtes impliquant des tris, des agrégations de hachage, des jointures à charge élevée, des expressions de table communes (CTE) et des requêtes sur des colonnes non indexées. Ces volumes de stockage d'instances fournissent des IOPS et des performances supérieures, quelles que soient les configurations de stockage utilisées pour le stockage Amazon EBS persistant. Étant donné que RDS Optimized Reads décharge les opérations sur les objets temporaires dans le stockage d'instances, les opérations d'entrée/sortie par seconde (IOPS) ou le débit du stockage persistant (Amazon EBS) peuvent désormais être utilisés pour les opérations sur les objets persistants. Ces opérations incluent des lectures et des écritures régulières de fichiers de données, ainsi que des opérations de moteur en arrière-plan, telles que le vidage et la fusion de mémoires tampon par insertion.

> **Note**
>
> Les instantanés RDS manuels et automatisés ne contiennent que des fichiers de moteur pour les objets persistants. Les objets temporaires créés dans le stockage d'instances ne sont pas inclus dans les instantanés RDS.

Cas d'utilisation pour RDS Optimized Reads

Si vous avez des charges de travail qui dépendent fortement d'objets temporaires, tels que des tables ou des fichiers internes, pour l'exécution de leurs requêtes, vous pouvez tirer parti de l'activation de RDS Optimized Reads. Les cas d'utilisation suivants sont propices à RDS Optimized Reads :

- Applications exécutant des requêtes analytiques avec des expressions de table communes (CTE), des tables dérivées et des opérations de regroupement complexes
- Réplicas en lecture qui traitent un trafic de lecture important avec des requêtes non optimisées
- Applications exécutant des requêtes de création de rapport à la demande ou dynamiques impliquant des opérations complexes, telles que des requêtes avec des clauses `GROUP BY` et `ORDER BY`
- Charges de travail utilisant des tables temporaires internes pour le traitement des requêtes

  Vous pouvez surveiller la variable de statut du moteur `created_tmp_disk_tables` pour déterminer le nombre de tables temporaires sur disque créées sur votre instance de base de données.
- Applications qui créent de grandes tables temporaires, directement ou dans le cadre de procédures, pour stocker des résultats intermédiaires

- Requêtes de base de données qui regroupent ou trient des colonnes non indexées

Bonnes pratiques relatives à RDS Optimized Reads

Utilisez les bonnes pratiques suivantes pour RDS Optimized Reads :

- Ajoutez une logique de nouvelle tentative pour les requêtes en lecture seule au cas où elles échoueraient en raison d'un stockage d'instances complet pendant l'exécution.
- Surveillez l'espace de stockage disponible sur le magasin d'instances à l'aide de la CloudWatch métrique`FreeLocalStorage`. Si le stockage d'instances atteint sa limite en raison de la charge de travail sur l'instance de base de données, modifiez l'instance de base de données pour utiliser une classe d'instances de base de données plus grande.
- Lorsque votre instance de base de données dispose de suffisamment de mémoire mais atteint toujours la limite de stockage sur le stockage d'instances, augmentez la valeur `binlog_cache_size` pour conserver en mémoire les entrées binlog spécifiques à la session. Cette configuration empêche l'écriture des entrées binlog dans les fichiers de cache binlog temporaires sur le disque.

  Le paramètre `binlog_cache_size` est spécifique à la session. Vous pouvez modifier cette valeur pour chaque nouvelle session. Le réglage de ce paramètre peut augmenter l'utilisation de la mémoire sur l'instance de base de données pendant les pics de charge de travail. Par conséquent, envisagez d'augmenter la valeur du paramètre en fonction du modèle de charge de travail de votre application et de la mémoire disponible sur l'instance de base de données.
- Utilisez la valeur par défaut `MIXED` pour `binlog_format`. En fonction de la taille des transactions, le réglage de `binlog_format` sur `ROW` peut entraîner la création de fichiers de cache binlog volumineux sur le stockage d'instances.
- Évitez d'effectuer des modifications en bloc dans une transaction unique. Ces types de transactions peuvent générer de gros fichiers de cache binlog sur le stockage d'instances et peuvent provoquer des problèmes lorsque le stockage d'instances est plein. Envisagez de diviser les écritures en plusieurs petites transactions afin de réduire au maximum l'utilisation de l'espace de stockage pour les fichiers de cache binlog.

Utilisation de RDS Optimized Reads

Lorsque vous provisionnez une instance de base de données RDS for MariaDB avec l'une des classes d'instances de base de données suivantes dans le cadre d'un déploiement d'instance de

base de données mono-AZ ou multi-AZ, l'instance de base de données utilise automatiquement les lectures optimisées pour RDS.

Pour activer RDS Optimized Reads, effectuez l'une des actions suivantes :

- Créez une instance de base de données RDS for MariaDB en utilisant l'une de ces classes d'instances de base de données. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données RDS for MariaDB afin d'utiliser l'une de ces classes d'instances de base de données. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Les lectures optimisées RDS sont disponibles partout Régions AWS où une ou plusieurs classes d'instances de base de données avec stockage SSD NVMe local sont prises en charge. Pour plus d'informations sur les classes d'instances de base de données, consultez the section called “Classes d'instances de base de données”.

La disponibilité des classes d'instances de base de données diffère pour Régions AWS. Pour déterminer si une classe d'instance de base de données est prise en charge dans une classe spécifique Région AWS, consultezthe section called “Déterminer le support des classes d'instance de base de données dans Régions AWS”.

Si vous ne souhaitez pas utiliser RDS Optimized Reads, modifiez votre instance de base de données afin qu'elle n'utilise pas une classe d'instances de base de données prenant en charge cette fonctionnalité.

Surveillance des instances de base de données qui utilisent RDS Optimized Reads

Vous pouvez surveiller les instances de base de données qui utilisent des lectures optimisées RDS avec les CloudWatch métriques suivantes :

- `FreeLocalStorage`
- `ReadIOPSLocalStorage`
- `ReadLatencyLocalStorage`
- `ReadThroughputLocalStorage`
- `WriteIOPSLocalStorage`

- `WriteLatencyLocalStorage`
- `WriteThroughputLocalStorage`

Ces métriques fournissent des données sur le stockage disponible dans le stockage d'instances, les IOPS et le débit. Pour plus d'informations sur ces métriques, consultez Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS.

Limites pour RDS Optimized Reads

Les limites suivantes s'appliquent à RDS Optimized Reads :

- RDS Optimized Reads est pris en charge pour les versions RDS for MariaDB suivantes :
  - 10.11.4 et versions 10.11 ultérieures
  - Versions 10.6.7 et 10.6 ultérieures
  - Versions 10.5.16 et 10.5 ultérieures
  - Versions 10.4.25 et 10.4 ultérieures

  Pour obtenir des informations sur les versions de RDS for MariaDB, consultez Versions de MariaDB sur Amazon RDS.
- Vous ne pouvez pas remplacer l'emplacement des objets temporaires par un stockage persistant (Amazon EBS) dans les classes d'instances de base de données qui prennent en charge RDS Optimized Reads.
- Lorsque la journalisation binaire est activée sur une instance de base de données, la taille maximale des transactions est limitée par la taille du stockage d'instances. Pour MariaDB, toute session qui nécessite plus de stockage que la valeur de `binlog_cache_size` écrit les modifications des transactions dans les fichiers de cache de journaux binaires temporaires, qui sont créés dans le stockage d'instances.
- Les transactions peuvent échouer lorsque le stockage d'instances est plein.

Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB

Vous pouvez améliorer les performances des transactions d'écriture avec l'option Écritures optimisées pour RDS for MariaDB. Lorsque votre base de données RDS for MariaDB utilise Écritures optimisées pour RDS, elle peut atteindre un débit de transactions d'écriture jusqu'à deux fois supérieur.

Rubriques

- Présentation de l'option Écritures optimisées pour RDS
- Utilisation de l'option Écritures optimisées pour RDS
- Activation de l'option Écritures optimisées pour RDS sur une base de données existante
- Limites pour l'option Écritures optimisées pour RDS

Présentation de l'option Écritures optimisées pour RDS

Lorsque vous activez l'option Écritures optimisées pour RDS, vos bases de données RDS for MariaDB n'écrivent qu'une seule fois lors du vidage des données dans un stockage durable sans avoir besoin du tampon à double écriture. Les bases de données continuent de protéger les propriétés ACID pour les transactions de base de données fiables, ainsi que des performances améliorées.

Les bases de données relationnelles, comme MariaDB, fournissent les propriétés ACID d'atomicité, de cohérence, d'isolation et de durabilité pour des transactions de base de données fiables. Pour fournir ces propriétés, MariaDB utilise une zone de stockage de données appelée tampon à double écriture qui empêche les erreurs d'écriture de page partielles. Ces erreurs se produisent en cas de panne matérielle alors que la base de données met à jour une page, par exemple en cas de panne de courant. Une base de données MariaDB peut détecter les écritures de page partielles et récupérer une copie de la page dans le tampon à double écriture. Cette technique offre une protection, mais elle entraîne également des opérations d'écriture supplémentaires. Pour plus d'informations sur le tampon à double écriture MariaDB, consultez Doublewrite Buffer (Tampon à double écriture) dans la documentation MariaDB.

Quand Écritures optimisées pour RDS est activé, les bases de données RDS for MariaDB n'écrivent qu'une seule fois lors du vidage des données dans un stockage durable sans utiliser le tampon à double écriture. Écritures optimisées pour RDS est utile si vous exécutez des charges de travail

lourdes en écriture sur vos bases de données RDS for MariaDB. Parmi les bases de données soumises à de lourdes charges de travail en écriture, citons celles qui prennent en charge les paiements numériques, les transactions financières et les applications de jeu.

Ces bases de données s'exécutent sur des classes d'instances de base de données qui utilisent le système AWS Nitro. En raison de la configuration matérielle dans ces systèmes, la base de données peut écrire des pages de 16 Kio directement dans des fichiers de données de manière fiable et durable, en une seule étape. Le système AWS Nitro permet des écritures optimisées RDS.

Vous pouvez définir le nouveau paramètre de base de données `rds.optimized_writes` pour contrôler la fonction Écritures optimisées pour RDS pour les bases de données RDS for MariaDB. Accédez à ce paramètre dans les groupes de paramètres de base de données de RDS for MariaDB pour les versions suivantes :

- Versions 11.4.3 et supérieures 11.4
- 10.11.4 et versions 10.11 ultérieures
- 10.6.10 et versions 10.6 ultérieures

Définissez ce paramètre sur l'une des valeurs suivantes :

- `AUTO` : activer RDS Optimized Writes si la base de données le prend en charge. Désactiver RDS Optimized Writes si la base de données ne le prend pas en charge. Il s'agit de la valeur par défaut.
- `OFF` : désactiver l'option Écritures optimisées pour RDS même si la base de données le prend en charge.

Si vous migrez une base de données RDS for MariaDB configurée pour utiliser Écritures optimisées pour RDS dans une classe d'instances de base de données qui ne prend pas en charge cette fonctionnalité, RDS désactive automatiquement Écritures optimisées pour RDS pour la base de données.

Lorsque Écritures optimisées pour RDS est désactivé, la base de données utilise le tampon à double écriture MariaDB.

Pour déterminer si une base de données RDS for MariaDB utilise Écritures optimisées pour RDS, consultez la valeur actuelle du paramètre `innodb_doublewrite` pour la base de données. Si la base de données utilise des écritures optimisées RDS, ce paramètre est défini sur `FALSE` (`0`).

Utilisation de l'option Écritures optimisées pour RDS

Vous pouvez activer les écritures optimisées RDS lorsque vous créez une base de données RDS pour MariaDB à l'aide de la console RDS, de l'API RDS ou de l'API RDS. AWS CLI L'option Écritures optimisées pour RDS est automatiquement activé lorsque les deux conditions suivantes s'appliquent dans le cadre de la création de la base de données :

- Vous spécifiez une version du moteur de base de données et une classe d'instances de base de données qui prennent en charge l'option Écritures optimisées pour RDS.
  - La fonctionnalité Écritures optimisées pour RDS est prise en charge pour les versions RDS for MariaDB suivantes :
    - Versions 11.4.3 et supérieures 11.4
    - 10.11.4 et versions 10.11 ultérieures
    - 10.6.10 et versions 10.6 ultérieures

    Pour obtenir des informations sur les versions de RDS for MariaDB, consultez Versions de MariaDB sur Amazon RDS.
  - Écritures optimisées pour RDS est pris en charge pour les bases de données RDS for MariaDB qui utilisent les classes d'instances de base de données suivantes :
    - db.m7i
    - db.m7g
    - db.m6g
    - db.m6gd
    - db.m6i
    - db.m5
    - db.m5d
    - db.r7i
    - db.r7g
    - db.r6g
    - db.r6gd
    - db.r6i
    - db.r5
    - db.r5b

- db.r5d
- db.x2idn
- db.x2iedn

Pour plus d'informations sur les classes d'instances de base de données, consultez the section called “Classes d'instances de base de données”.

La disponibilité des classes d'instances de base de données diffère pour Régions AWS. Pour déterminer si une classe d'instance de base de données est prise en charge dans une classe spécifique Région AWS, consultezthe section called “Déterminer le support des classes d'instance de base de données dans Régions AWS”.

- Dans le groupe de paramètres associé à la base de données, le paramètre `rds.optimized_writes` est défini sur `AUTO`. Dans les groupes de paramètres par défaut, ce paramètre est toujours défini sur `AUTO`.

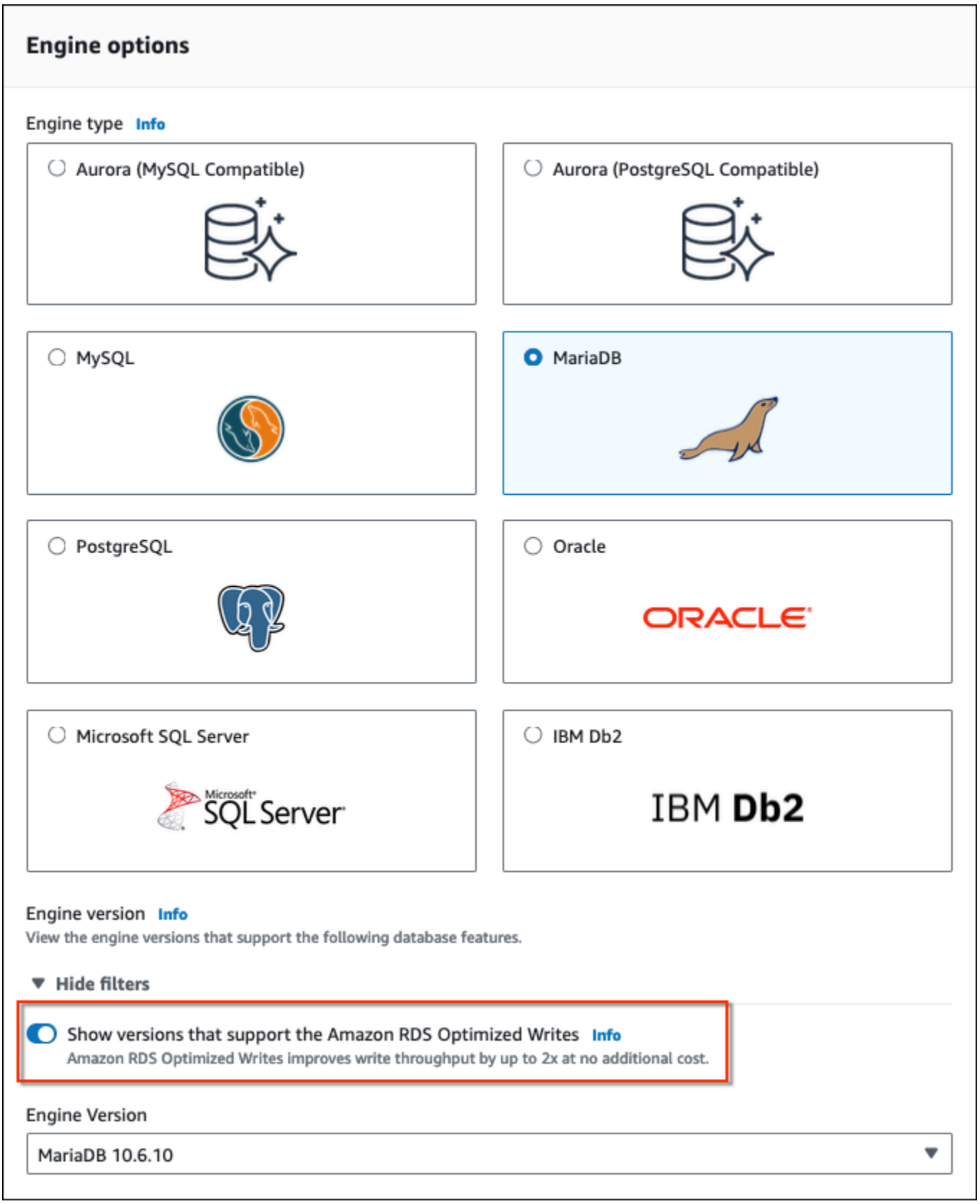
Si vous voulez utiliser une version du moteur de base de données et une classe d'instances de base de données qui prennent en charge Écritures optimisées pour RDS, mais que vous ne voulez pas utiliser cette fonction, spécifiez alors un groupe de paramètres personnalisé quand vous créez la base de données. Dans le groupe de paramètres, définissez le paramètre `rds.optimized_writes` sur `OFF`. Si vous souhaitez que la base de données utilise l'option Écritures optimisées pour RDS ultérieurement, vous pouvez définir ce paramètre sur `AUTO` pour l'activer. Pour obtenir des informations sur la création des groupes de paramètres personnalisés et sur la définition des paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.

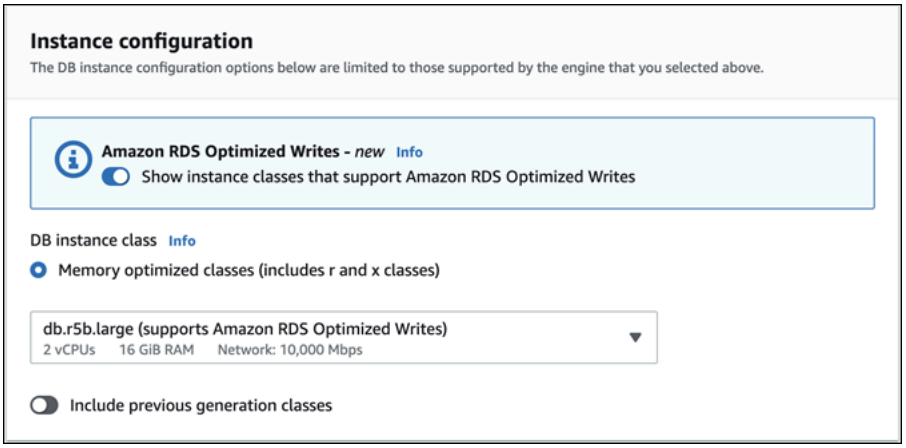
Console

Lorsque vous utilisez la console RDS pour créer une base de données RDS for MariaDB, vous pouvez filtrer les versions du moteur de base de données et les classes d'instances de base de données qui prennent en charge Écritures optimisées pour RDS. Après avoir activé les filtres, vous pouvez choisir parmi les versions du moteur de base de données et les classes d'instances de base de données disponibles.

Pour choisir une version du moteur de base de données prenant en charge RDS Optimized Writes, filtrez les versions du moteur de base de données RDS for MariaDB qui le prennent en charge dans Engine version (Version du moteur), puis choisissez une version.


Engine options
Engine type Info
Aurora (MySQL Compatible)
Aurora (PostgreSQL Compatible)
MySQL
MariaDB
PostgreSQL
Oracle
ORACLE
Microsoft SQL Server
Microsoft SQL Server
IBM Db2
IBM Db2
Engine version Info
View the engine versions that support the following database features.
Hide filters
Show versions that support the Amazon RDS Optimized Writes Info
Amazon RDS Optimized Writes improves write throughput by up to 2x at no additional cost.
Engine Version
MariaDB 10.6.10

Dans la section Instance configuration (Configuration de l'instance), filtrez les classes d'instances de base de données qui prennent en charge l'option Écritures optimisées pour RDS, puis choisissez une classe d'instances de base de données.



Après avoir effectué ces sélections, vous pouvez choisir d'autres paramètres qui répondent à vos besoins et terminer la création de la base de données RDS for MariaDB à l'aide de la console.

AWS CLI

Pour créer une instance de base de données à l'aide de AWS CLI, exécutez la create-db-instancecommande. Veillez à ce que les valeurs `--engine-version` et `--db-instance-class` prennent en charge RDS Optimized Writes. De plus, veillez à ce que le paramètre `rds.optimized_writes` du groupe de paramètres associé à l'instance de base de données soit défini sur AUT0. Cet exemple associe le groupe de paramètres par défaut à l'instance de base de données.

Example Création d'une instance de base de données qui utilise RDS Optimized Writes

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --engine mariadb \
    --engine-version 10.6.10 \
    --db-instance-class db.r5b.large \
    --manage-master-user-password \
    --master-username admin \
    --allocated-storage 200
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --engine mariadb ^
    --engine-version 10.6.10 ^
    --db-instance-class db.r5b.large ^
    --manage-master-user-password ^
    --master-username admin ^
    --allocated-storage 200
```

API RDS

Vous pouvez créer une instance de base de données à l'aide de l'DBInstanceopération Create. Quand vous utilisez cette opération, veillez à ce que les valeurs `EngineVersion` et `DBInstanceClass` prennent en charge RDS Optimized Writes. De plus, veillez à ce que le paramètre `rds.optimized_writes` du groupe de paramètres associé à l'instance de base de données soit défini sur `AUTO`.

Activation de l'option Écritures optimisées pour RDS sur une base de données existante

Pour modifier une base de données RDS for MariaDB existante afin d'activer l'option Écritures optimisées pour RDS, la base de données doit avoir été créée avec une version du moteur de base de données et une classe d'instance de base de données prises en charge. En outre, la base de données doit avoir été créée après la publication de l'option Écritures optimisées pour RDS le 7 mars 2023, car la configuration requise du système de fichiers sous-jacent est incompatible avec celle des bases de données créées avant sa publication. Si ces conditions sont remplies, vous pouvez activer l'option Écritures optimisées pour RDS en définissant le paramètre `rds.optimized_writes` sur `AUTO`.

Si votre base de données n'a pas été créée avec une version de moteur, une classe d'instance ou une configuration de système de fichiers prise en charge, vous pouvez utiliser le Blue/Green Deployments to migrate to a supported configuration. While creating the blue/green déploiement RDS. Procédez comme suit :

- Sélectionnez Activer l'option Écritures optimisées pour RDS sur une base de données verte, puis spécifiez une version du moteur et une classe d'instance de base de données qui prennent en charge l'option Écritures optimisées pour RDS. Pour obtenir la liste des versions de moteur et des classes d'instance prises en charge, consultez the section called “Utilisation avec une nouvelle base de données”.

- Sous Stockage, choisissez Mettre à niveau la configuration du système de fichiers de stockage. Cette option met à niveau la base de données vers une configuration de système de fichiers sous-jacent compatible.

Lorsque vous créez le déploiement bleu/vert, si le paramètre `rds.optimized_writes` est défini sur AUTO, l'option Écritures optimisées pour RDS sera automatiquement activé dans l'environnement vert. Vous pouvez ensuite basculer le déploiement bleu/vert, qui favorise l'environnement vert comme nouvel environnement de production.

Pour de plus amples informations, veuillez consulter the section called “Création d'un déploiement bleu/vert”.

Limites pour l'option Écritures optimisées pour RDS

Lorsque vous restaurez une base de données RDS for MariaDB à partir d'un instantané, vous pouvez activer l'option Écritures optimisées pour RDS pour cette base de données seulement si toutes les conditions suivantes s'appliquent :

- L'instantané a été créé à partir d'une base de données qui prend en charge RDS Optimized Writes.
- L'instantané a été créé à partir d'une base de données qui a été créée après le lancement d'Écritures optimisées pour RDS.
- L'instantané est restauré en une base de données qui prend en charge RDS Optimized Writes.
- La base de données restaurée est associée à un groupe de paramètres où le paramètre `rds.optimized_writes` est défini sur AUTO.

Mises à niveau du moteur de base de données MariaDB

Lorsqu'Amazon RDS prend en charge une nouvelle version d'un moteur de base de données, vous pouvez mettre à niveau vos instances de base de données vers la nouvelle version. Il existe deux types de mises à niveau pour les instances de base de données MariaDB : les mises à niveau de version majeure et les mises à niveau de version mineure.

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. En conséquence, vous devez effectuer manuellement les mises à niveau de version majeure de vos instances de base de données. Vous pouvez lancer une mise à niveau de version majeure en modifiant votre instance de base de données. Cependant, avant d'effectuer une mise à niveau de version majeure, nous vous recommandons de suivre les instructions décrites dans Mises à niveau de version majeures RDS pour MariaDB.

En revanche, une mise à niveau de version mineure contient uniquement des modifications rétrocompatibles avec les applications existantes. Vous pouvez lancer manuellement une mise à niveau de version mineure en modifiant votre instance de base de données. Vous pouvez également activer l'option Mise à niveau automatique des versions mineures lorsque vous créez ou modifiez une instance de base de données. Cela signifie que votre instance de base de données est automatiquement mise à niveau après qu'Amazon a RDS testé et approuvé la nouvelle version. Pour de plus amples informations sur la mise à niveau, veuillez consulter Mise à niveau de la version du moteur d'une instance de base de données.

Si votre instance de base de données MariaDB utilise des réplicas en lecture, vous devez mettre à niveau tous les réplicas en lecture avant de mettre à niveau l'instance source. Si votre instance de base de données se trouve dans un déploiement Multi-AZ, les deux réplicas, enregistreur et de secours, sont mis à niveau. Votre instance de base de données peut ne pas être disponible tant que la mise à niveau n'est pas terminée.

Pour plus d'informations sur les versions MariaDB prises en charge et la gestion des versions, consultez Versions de MariaDB sur Amazon RDS.

Les mises à niveau du moteur de base de données nécessitent un temps d'arrêt. La durée du temps d'arrêt varie en fonction de la taille de votre instance de base de données.

> **Tip**
>
> Vous pouvez minimiser le temps d'arrêt nécessaire à la mise à niveau de l'instance de base de données en utilisant un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Rubriques

- Considérations relatives aux mises à niveau de MariaDB
- Trouver des cibles de mise à niveau valides
- Numéros de version de MariaDB
- RDSnuméros de version RDS pour MariaDB
- Mises à niveau de version majeures RDS pour MariaDB
- Mise à niveau d'une instance de base de données MariaDB
- Mises à niveau automatiques des versions mineures RDS pour MariaDB
- Utilisation d'une réplique en lecture pour réduire les temps d'arrêt lors de la mise à niveau d'une base RDS de données pour MariaDB

Considérations relatives aux mises à niveau de MariaDB

Amazon RDS prend au moins deux instantanés de base de données au cours du processus de mise à niveau. Amazon RDS prend jusqu'à deux instantanés de l'instance de base de données avant d'apporter des modifications à la mise à niveau. Si la mise à niveau ne fonctionne pas pour vos bases de données, vous pouvez restaurer l'un de ces instantanés pour créer une instance de base de données exécutant l'ancienne version. Amazon RDS prend un autre instantané de l'instance de base de données lorsque la mise à niveau est terminée. Amazon RDS prend ces instantanés, qu'il AWS Backup gère ou non les sauvegardes de l'instance de base de données.

> **Note**
>
> Amazon RDS ne prend des instantanés de base de données que si vous avez défini la période de rétention des sauvegardes pour votre instance de base de données sur un nombre supérieur à 0. Pour modifier la période de rétention des sauvegardes, consultez Modification d'une RDS instance de base de données Amazon.

Une fois la mise à niveau terminée, vous ne pouvez pas rétablir la version précédente du moteur de base de données. Si vous souhaitez revenir à la version précédente, restaurez le premier instantané de base de données pris pour créer une nouvelle instance de base de données.

C'est vous qui décidez quand mettre à niveau votre instance de base de données vers une nouvelle version prise en charge par AmazonRDS. Ce niveau de contrôle vous aide à maintenir la compatibilité avec des versions de base de données spécifiques et à tester les nouvelles versions avec votre application avant un déploiement en production. Lorsque vous êtes prêt, vous pouvez effectuer des mises à niveau de version aux moments qui conviennent le mieux à votre planning.

Si votre instance DB utilise la réplication en lecture, vous devez mettre à niveau tous les réplicas en lecture avant de mettre à niveau l'instance source.

Si votre instance de base de données se trouve dans un déploiement multi-AZ, les deux instances de base de données principale et de secours sont mises à niveau. Les instances de base de données principales et de secours sont mises à niveau en même temps, et vous êtes confronté à une panne jusqu'à ce que la mise à niveau soit terminée. La durée de la panne varie selon votre moteur de base de données, la version du moteur et la taille de votre instance de base de données.

Trouver des cibles de mise à niveau valides

Lorsque vous utilisez le AWS Management Console pour mettre à niveau une instance de base de données, il affiche les cibles de mise à niveau valides pour l'instance de base de données. Vous pouvez également exécuter la AWS CLI commande suivante pour identifier les cibles de mise à niveau valides pour une instance de base de données :

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine mariadb \
  --engine-version version_number \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine mariadb ^
  --engine-version version_number ^
```

```
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Par exemple, pour identifier les cibles de mise à niveau valides pour une instance de base de données MariaDB version 10.5.17, exécutez la commande suivante : AWS CLI

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine mariadb \
  --engine-version 10.5.17 \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine mariadb ^
  --engine-version 10.5.17 ^
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Numéros de version de MariaDB

La séquence de numérotation des versions RDS pour le moteur de base de données MariaDB est soit sous la forme major.minor.patch. YYYYMMDDou major.minor.patch, par exemple, 10.11.5.R2.20231201 ou 10.4.30. Le format utilisé dépend de la version du moteur MariaDB.

majeur

Le numéro de version principal est à la fois le nombre entier et la première partie fractionnaire du numéro de version, par exemple 10.11. Une mise à niveau majeure augmente la partie majeure du numéro de version. Par exemple, une mise à niveau de 10.5 .20 vers 10.6.12 est une mise à niveau de version majeure, où 10.5 et 10.6 sont les numéros de version principaux.

mineur

Le numéro de version secondaire est la troisième partie du numéro de version, par exemple le 5 dans la version 10.11.5.

patch

Le correctif est la quatrième partie du numéro de version, par exemple le R2 dans 10.11.5.R2. Une version de RDS correctif inclut des corrections de bogues importantes ajoutées à une version mineure après sa sortie.

YYYYMMDD

La date est la cinquième partie du numéro de version, par exemple, le 20231201 dans 10.11.5.R2.20231201. Une version RDS datée est un correctif de sécurité qui inclut des correctifs de sécurité importants ajoutés à une version mineure après sa publication. Il n'inclut aucun correctif susceptible de modifier le comportement d'un moteur.

Le tableau suivant explique le schéma de dénomination RDS pour la version 10.11 de MariaDB.

10.11 version mineure	Schéma de dénomination
≥5	Les nouvelles instances de base de données utilisent major.minor.patch. YYMMDD, par exemple, 10.11.5.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple 10.11.5.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.
< 5	Les instances de base de données existantes utilisent major.minor.patch, par exemple 10.11.4.R2.

Le tableau suivant explique le schéma de dénomination de RDS MariaDB version 10.6.

10.6 version mineure	Schéma de dénomination
≥ 14	Les nouvelles instances de base de données utilisent major.minor.patch. YYMMDD, par exemple, 10.6.14.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple 10.6.14.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.

10.6 version mineure	Schéma de dénomination
< 14	Les instances de base de données existantes utilisent major.minor.patch, par exemple 10.6.13.R2.

Le tableau suivant explique le schéma de dénomination RDS pour la version 10.5 de MariaDB.

10.5 version mineure	Schéma de dénomination
≥ 21	Les nouvelles instances de base de données utilisent major.minor.patch. YYMMDD, par exemple, 10.5.21.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple, 10.5.21.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.
< 21	Les instances de base de données existantes utilisent major.minor.patch, par exemple 10.5.20.R2.

Le tableau suivant explique le schéma de dénomination de RDS MariaDB version 10.4.

10.4 version mineure	Schéma de dénomination
≥ 30	Les nouvelles instances de base de données utilisent major.minor.patch. YYMMDD, par exemple, 10.4.30.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple, 10.4.30.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.
< 30	Les instances de base de données existantes utilisent major.minor.patch, par exemple 10.4.29.R2.

RDSnuméros de version RDS pour MariaDB

RDSles numéros de version utilisent le schéma de dénomination *major*.*minor*.*patch* ou le schéma de *major*.*minor*.*patch*.*YYYYMMDD* dénomination. Une version de RDS correctif inclut des corrections de bogues importantes ajoutées à une version mineure après sa sortie. Une version RDS datée (*YYMMDD*) est un correctif de sécurité. Un correctif de sécurité n'inclut aucun correctif susceptible de modifier le comportement du moteur.

Pour identifier le numéro de RDS version Amazon de votre base de données, vous devez d'abord créer l'`rds_tools`extension à l'aide de la commande suivante :

```
CREATE EXTENSION rds_tools;
```

Vous pouvez trouver le numéro de RDS version de votre RDS base de données MariaDB à l'aide de la requête suivante : SQL

```
mysql> select mysql.rds_version();
```

Par exemple, l'interrogation d'une base de données RDS pour MariaDB 10.6.14 renvoie le résultat suivant :

```
+---------------------+
| mysql.rds_version() |
+---------------------+
| 10.6.14.R2.20231201  |
+---------------------+
1 row in set (0.01 sec)
```

Mises à niveau de version majeures RDS pour MariaDB

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. Par conséquent, Amazon RDS n'applique pas automatiquement les mises à niveau des versions majeures. Vous devez modifier manuellement votre instance de base de données. Nous vous recommandons de tester soigneusement toute mise à niveau avant de l'appliquer à vos instances de production.

Amazon RDS prend en charge les mises à niveau sur place suivantes pour les versions majeures du moteur de base de données MariaDB :

- Toute version de MariaDB vers MariaDB 11.4
- De toute version de MariaDB vers MariaDB 10.11
- Toute version MariaDB vers MariaDB 10.6
- MariaDB 10.4 vers MariaDB 10.5

Si vous utilisez un groupe de paramètres personnalisé et que vous effectuez une mise à niveau de version majeure, vous devez spécifier un groupe de paramètres par défaut pour la nouvelle version du moteur de base de données ou créer votre propre groupe de paramètres personnalisé pour la nouvelle version du moteur de base de données. L'association du nouveau groupe de paramètres avec l'instance de base de données exige un redémarrage de la base de données initié par le client après la mise à niveau. Le statut du groupe de paramètres de l'instance indique `pending-reboot` si l'instance doit être redémarrée pour que les modifications du groupe de paramètres soient appliquées. L'état du groupe de paramètres d'une instance peut être consulté dans le AWS Management Console ou en exécutant un appel « describe » tel que`describe-db-instances`.

Mise à niveau d'une instance de base de données MariaDB

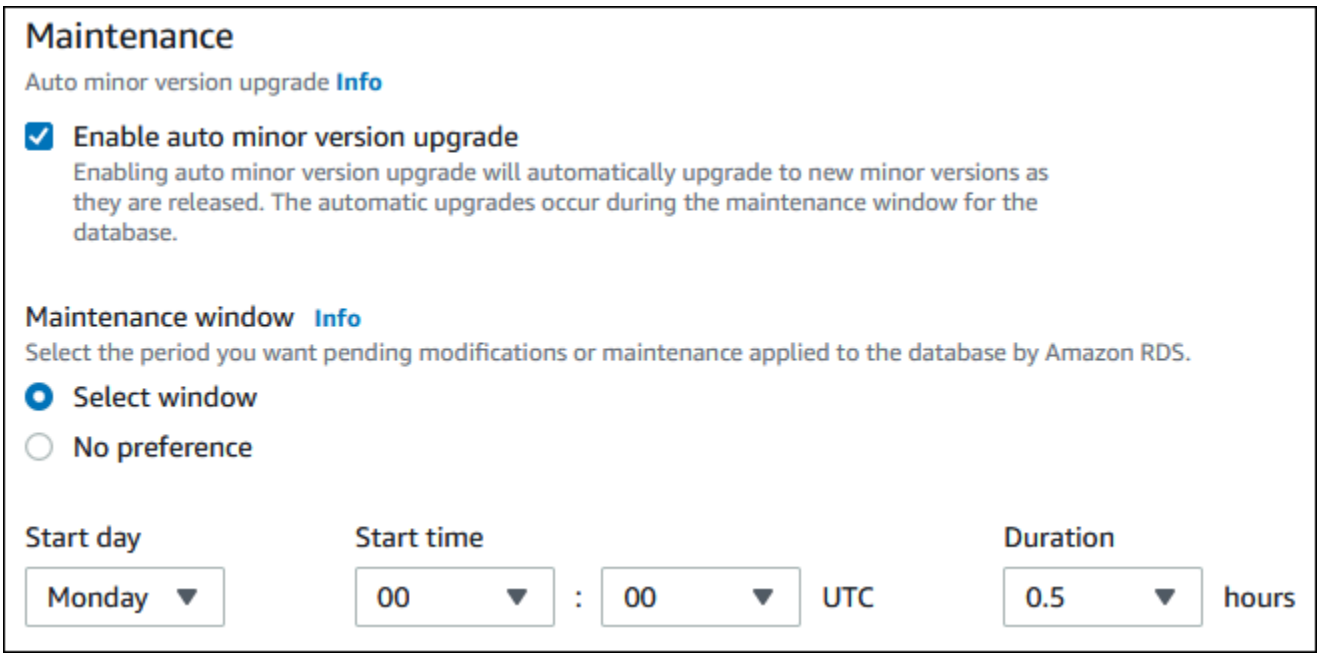
Pour plus d'informations sur la mise à niveau manuelle ou automatique d'une instance de base de données MariaDB, consultez la section Mise à niveau de la version du moteur d'une instance de base de données.

Mises à niveau automatiques des versions mineures RDS pour MariaDB

Si vous spécifiez les paramètres suivants lors de la création ou de la modification d'une instance de base de données, celle-ci peut être mise à niveau automatiquement.

- Le paramètre Mise à niveau automatique des versions mineures est activé.
- Le paramètre Période de conservation des sauvegardes est supérieur à 0.

Dans le AWS Management Console, ces paramètres se trouvent sous Configuration supplémentaire. L'image suivante illustre le réglage Auto minor version upgrade (Mise à niveau automatique de versions mineures).



Pour plus d'informations sur ces paramètres, consultez la page Paramètres des instances de base de données.

Dans certains cas, RDS pour les versions majeures de MariaDB, dans d' Régions AWS autres, une version mineure est désignée comme version de mise RDS à niveau automatique. Une fois qu'une version mineure a été testée et approuvée par AmazonRDS, la mise à niveau de la version mineure s'effectue automatiquement pendant votre période de maintenance. RDSne définit pas automatiquement les nouvelles versions mineures publiées comme version de mise à niveau automatique. Avant qu'RDS ne désigne une version de mise à niveau automatique plus récente, plusieurs critères sont pris en compte, à savoir :

- Problèmes de sécurité connus
- Bogues dans MariaDB Community
- Stabilité globale du parc depuis la publication de la version mineure

> **Note**
>
> Support pour l'utilisation des TLS versions 1.0 et 1.1 a été supprimé à partir de versions mineures spécifiques de MariaDB. Pour plus d'informations sur les versions mineures de MariaDB prises en charge, consultez. the section called "SSL/TLSsupport pour MariaDB"

Vous pouvez exécuter la AWS CLI commande suivante pour déterminer la version cible de mise à niveau mineure automatique actuelle pour une version mineure de MariaDB spécifiée dans une version spécifique. Région AWS

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine mariadb \
--engine-version minor_version \
--region region \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine mariadb ^
--engine-version minor_version ^
--region region ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output text
```

Par exemple, la AWS CLI commande suivante détermine la cible de mise à niveau mineure automatique pour la version mineure 10.5.16 de MariaDB dans l'est des États-Unis (Ohio) (us-east-2). Région AWS

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine mariadb \
--engine-version 10.5.16 \
--region us-east-2 \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output table
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
```

```
--engine mariadb ^
--engine-version 10.5.16 ^
--region us-east-2 ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output table
```

Votre sortie est similaire à ce qui suit.

```
---------------------------------
|    DescribeDBEngineVersions    |
+--------------+-----------------+
|  AutoUpgrade |  EngineVersion  |
+--------------+-----------------+
|  True        |  10.5.17        |
|  False       |  10.5.18        |
|  False       |  10.5.19        |
|  False       |  10.6.5         |
|  False       |  10.6.7         |
|  False       |  10.6.8         |
|  False       |  10.6.10        |
|  False       |  10.6.11        |
|  False       |  10.6.12        |
+--------------+-----------------+
```

Dans cet exemple, la valeur de `AutoUpgrade` est `True` pour MariaDB version 10.5.17. Ainsi, la cible de mise à niveau mineure automatique est MariaDB version 10.5.17, comme mis en évidence dans la sortie.

Une instance de base de données MariaDB est automatiquement mise à niveau pendant votre fenêtre de maintenance si les critères suivants sont réunis :

- Le paramètre Mise à niveau automatique des versions mineures est activé.
- Le paramètre Période de conservation des sauvegardes est supérieur à 0.
- L'instance de base de données exécute une version mineure du moteur de base de données qui est inférieure à la version mineure de la mise à niveau automatique actuelle.

Pour plus d'informations, consultez Mise à niveau automatique de la version mineure du moteur.

Utilisation d'une réplique en lecture pour réduire les temps d'arrêt lors de la mise à niveau d'une base RDS de données pour MariaDB

Dans la plupart des cas, un déploiement bleu/vert est la meilleure option pour réduire les temps d'arrêt lors de la mise à niveau d'une instance de base de données MariaDB. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Si vous ne pouvez pas utiliser un déploiement bleu/vert et que votre instance de base de données MariaDB est en cours d'utilisation avec une application de production, vous pouvez utiliser la procédure suivante pour mettre à niveau la version de la base de données pour votre instance de base de données. Cette procédure peut réduire les temps d'arrêt de votre application.

En utilisant un réplica en lecture, vous pouvez effectuer la plupart des étapes de maintenance à l'avance et ainsi réduire les modifications nécessaires lors d'une panne réelle. Cette technique vous permet de tester et de préparer la nouvelle instance de base de données sans apporter de modifications à votre instance de base de données existante.

La procédure suivante illustre un exemple de mise à niveau de MariaDB version 10.5 vers MariaDB version 10.6. Vous pouvez utiliser les mêmes étapes générales pour des mises à niveau vers d'autres versions majeures.

Pour mettre à niveau une base de données MariaDB alors qu'une instance de base de données est en cours d'utilisation

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Créez un réplica en lecture de votre instance de base de données MariaDB 10.5. Ce processus crée une copie pouvant être mise à niveau de votre base de données. D'autres réplicas en lecture de l'instance de base de données peuvent également exister.

    a. Sur la console, choisissez Bases de données, puis sélectionnez l'instance de base de données que vous souhaitez mettre à niveau.
    b. Sous Actions, choisissez Créer des réplicas en lecture.
    c. Spécifiez une valeur pour DB instance identifier (Identifiant de l'instance DB) pour votre réplica en lecture et assurez-vous que la DB instance class (Classe d'instance DB) et les autres paramètres correspondent à votre instance de base de données MariaDB 10.5.
    d. Choisissez Créer un réplica en lecture.

3. (Facultatif) Lorsque le réplica en lecture a été créé et que le champ État indique Disponible, convertissez le réplica en lecture en déploiement multi-AZ et activez les sauvegardes.

   Par défaut, un réplica en lecture est créé en tant que déploiement mono-AZ et les sauvegardes sont désactivées. Dans la mesure où le réplica en lecture finira par devenir l'instance de base de données de production, nous vous recommandons de configurer un déploiement multi-AZ et d'activer les sauvegardes dès maintenant.

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de créer.

   b. Sélectionnez Modify.

   c. Dans le champ Déploiement multi-AZ, choisissez Créer une instance de secours.

   d. Dans le champ Backup Retention Period (Période de rétention des sauvegardes), choisissez une valeur positive différente de zéro (par exemple, 3 jours), puis sélectionnez Continue (Continuer).

   e. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.

   f. Choisissez Modifier l'instance DB.

4. Lorsque le champ Status (Statut) du réplica en lecture indique Available (Disponible), procédez à sa mise à niveau vers MySQL 10.6.

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de créer.

   b. Sélectionnez Modify.

   c. Pour DB engine version (Version du moteur de base de données), choisissez la version MariaDB 10.6 vers laquelle effectuer la mise à niveau, puis Continue (Continuer).

   d. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.

   e. Choisissez Modifier l'instance de base de données pour démarrer la mise à niveau.

5. Lorsque la mise à niveau est terminée et que le statut indique Disponible, vérifiez que la réplique de lecture mise à niveau correspond up-to-date à l'instance de base de données MariaDB 10.5 source. Pour vérifier, connectez-vous au réplica en lecture et exécutez la commande `SHOW REPLICA STATUS`. Si le `Seconds_Behind_Master` champ l'est`0`, la réplication l'est up-to-date.

> **Note**
>
> Les versions précédentes de MariaDB utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MariaDB antérieure à la version 10.6, utilisez alors `SHOW SLAVE STATUS`.

6. (Facultatif) Créez un réplica en lecture de votre réplica en lecture.

   Si vous souhaitez que l'instance de base de données dispose d'un réplica en lecture une fois celle-ci promue en tant qu'instance de base de données autonome, vous pouvez créer le réplica en lecture dès maintenant.

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de mettre à niveau.

   b. Sous Actions, choisissez Créer des réplicas en lecture.

   c. Spécifiez une valeur pour DB instance identifier (Identifiant de l'instance DB) pour votre réplica en lecture et assurez-vous que la DB instance class (Classe d'instance DB) et les autres paramètres correspondent à votre instance de base de données MariaDB 10.5.

   d. Choisissez Créer un réplica en lecture.

7. (Facultatif) Configurez un groupe de paramètres de base de données personnalisé pour le réplica en lecture.

   Si vous souhaitez que l'instance de base de données utilise un groupe de paramètres personnalisé une fois celle-ci promue en tant qu'instance de base de données autonome, vous pouvez créer le groupe de paramètres de base de données dès maintenant et l'associer au réplica en lecture.

   a. Créez un groupe de paramètres de base de données personnalisé pour MariaDB 10.6. Pour obtenir des instructions, consultez Création d'un groupe de paramètres de base de données dans RDSAmazon.

   b. Modifiez les paramètres que vous souhaitez modifier dans le groupe de paramètres de base de données fraîchement créé. Pour obtenir des instructions, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

   c. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture.

   d. Sélectionnez Modify.

e. Pour DB parameter group (Groupe de paramètres DB), choisissez le groupe de paramètres de base de données MariaDB 10.6 que vous venez de créer, puis sélectionnez Continue (Continuer).

f. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.

g. Choisissez Modifier l'instance de base de données pour démarrer la mise à niveau.

8. Faites de votre réplica en lecture MariaDB 10.6 une instance de base de données autonome.

> **Important**
>
> Une fois promu en tant qu'instance de base de données autonome, votre réplica en lecture MariaDB 10.6 cesse d'être un réplica de votre instance de base de données MariaDB 10.5. Nous vous conseillons d'effectuer la promotion de votre réplica en lecture MariaDB 10.6 au cours d'une fenêtre de maintenance lorsque votre instance de base de données MariaDB 10.5 source est en mode lecture seule et que toutes les opérations d'écriture sont suspendues. Une fois la promotion terminée, vous pouvez diriger vos opérations d'écriture vers l'instance de base de données MariaDB 10.6 mise à niveau pour garantir qu'aucune opération d'écriture ne se perde.
> En outre, nous vous recommandons, avant de promouvoir votre réplique de lecture MariaDB 10.6, d'effectuer toutes les opérations de langage de définition de données DDL () nécessaires sur votre réplique de lecture MariaDB 10.6. Par exemple, la création d'index. Cette approche permet d'éviter tout effet négatif sur les performances du réplica en lecture MariaDB 10.6 après sa promotion. Pour promouvoir un réplica en lecture.

a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de mettre à niveau.

b. Pour Actions, choisissez Promote (Promouvoir).

c. Choisissez Oui pour activer les sauvegardes automatiques pour l'instance du réplica en lecture. Pour plus d'informations, consultez Présentation des sauvegardes.

d. Choisissez Continuer.

e. Choisissez Promouvoir le réplica en lecture.

9. Vous disposez à présent d'une version mise à niveau de votre base de données MariaDB. À ce stade, vous pouvez diriger vos applications vers la nouvelle instance de base de données MariaDB 10.6.

Mise à niveau d'une version du moteur de snapshots de base de données MariaDB

Avec Amazon RDS, vous pouvez créer un instantané de base de données de volume de stockage de votre instance de base de données MariaDB. Lorsque vous créez un instantané de base de données, celui-ci est basé sur la version du moteur utilisée par votre instance de base de données. Vous pouvez mettre à niveau la version du moteur pour vos instantanés de base de données.

Pour RDS pour MariaDB, vous pouvez effectuer une mise à niveau vers toutes les versions de moteur disponibles. Vous pouvez mettre à niveau des instantanés de base de données chiffrés ou non chiffrés.

Pour afficher les versions de moteur disponibles pour votre instantané de base de données RDS pour MariaDB, utilisez l'exemple suivant. AWS CLI

```
aws rds describe-db-engine-versions --engine mariadb --include-all --engine-
version example-engine-version --query "DBEngineVersions[*].ValidUpgradeTarget[*].
{EngineVersion:EngineVersion}" --output text
```

Si vous ne voyez aucun résultat pour votre instantané, la version de votre moteur est peut-être obsolète. Si la version de votre moteur est obsolète, nous vous recommandons de passer à la dernière cible de mise à niveau de la version majeure ou à l'une des autres cibles de mise à niveau disponibles pour cette version. Pour de plus amples informations, veuillez consulter Options de mise à niveau pour les instantanés de base de données avec des versions de moteur non prises en charge pour RDS pour MariaDB.

Après avoir restauré un instantané de base de données mis à niveau vers une nouvelle version de moteur, veillez à vérifier que la mise à jour est réussie. Pour de plus amples informations sur une mise à niveau des versions majeures, veuillez consulter the section called “Mises à niveau du moteur de base de données MariaDB”. Pour savoir comment restaurer un instantané de base de données, consultez the section called “Restauration vers une instance de base de données”.

> **Note**
>
> Vous ne pouvez pas mettre à niveau les instantanés de base de données automatisés créés au cours du processus de sauvegarde automatique.

Vous pouvez mettre à niveau un instantané de base de données à l'aide de l'API AWS Management Console AWS CLI, ou RDS.

Console

Pour mettre à niveau une version du moteur de snapshots de base de données à l'aide de AWS Management Console, procédez comme suit.

Pour mettre à niveau un instantané de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané que vous souhaitez mettre à niveau.
4. Pour Actions, choisissez Upgrade Snapshot (Mettre à niveau l'instantané). La page Upgrade Snapshot (Mettre à niveau l'instantané) s'affiche.
5. Choisissez la New engine version (Version du nouveau moteur) vers laquelle mettre à niveau.
6. Choisissez Save changes (Enregistrer les changements) pour mettre à niveau l'instantané.

   Pendant le processus de mise à niveau, toutes les actions d'instantané sont désactivées pour l'instantané de base de données. En outre, le statut du snapshot de base de données passe de Disponible à Mise à niveau, puis passe à Actif une fois terminé. Si le snapshot de base de données ne peut pas être mis à niveau en raison de problèmes de corruption du snapshot, le statut passe à Indisponible. Vous ne pouvez pas récupérer l'instantané lorsqu'il a ce statut.

   > **Note**
   >
   > Si la mise à niveau de l'instantané de base de données échoue, l'instantané revient à l'état d'origine avec la version originale.

AWS CLI

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, exécutez la AWS CLI modify-db-snapshotcommande.

Options

- `--db-snapshot-identifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être unique pour un Amazon Resource Name (ARN). Pour plus d'informations, consultez Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `--engine-version` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot \

    --db-snapshot-identifier my_db_snapshot \
    --engine-version new_version
```

Dans Windows:

```
aws rds modify-db-snapshot ^
    --db-snapshot-identifier my_db_snapshot ^
    --engine-version new_version
```

Amazon RDS API

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, appelez l'DBSnapshotopération RDS API Modify.

Paramètres

- `DBSnapshotIdentifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être unique pour un Amazon Resource Name (ARN). Pour plus d'informations, consultez Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `EngineVersion` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Options de mise à niveau pour les instantanés de base de données avec des versions de moteur non prises en charge pour RDS pour MariaDB

Le tableau suivant indique les versions de moteur vers lesquelles vous pouvez effectuer une mise à niveau à partir d'une version de moteur non prise en charge pour les instantanés de base de données RDS pour MariaDB.

> **Note**
>
> Il se peut que vous deviez mettre à niveau votre instantané de base de données plusieurs fois pour passer à la version du moteur que vous avez choisie.

Version du moteur de capture d'écran DB	Versions du moteur disponibles pour la mise à niveau
10,0,17, 10,0,24, 10,0,28, 10,0,31, 10,0,32, 10,0,34, 10,0,35	10,0,38, 10,1,48, 10,6,19, 10,6,20, 10,11,9, 10,1,110, 11,4.3, 11,4,4
10.1.14, 10.1.16, 10.1.19, 10.1.23, 10.1.26, 10.1.31, 10.1.34	10,1,48, 10,2,44, 10,6,19, 10,6,20, 10,11,9, 10,1,110, 11,4.3, 11,4,4
10.2.11, 10.2.12, 10.2.15, 10.2.21, 10.2.32, 10.2.37, 10.2.39, 10.2.40, 10.2.41, 10.2.43	10.2.44, 10.3.39, 10.6.19, 10.6.20, 10.6.21, 10.11.9, 10.11.10, 10.11.11, 11.4.3, 11.4.4
10.3.8, 10.3.13, 10.3.20, 10.3.23, 10.3.28, 10.3.31, 10.3.32	10.3.39, 10.4.34, 10.6.19, 10.6.20 10.6.21, 10.11.9, 10.11.10, 10.11.11, 11.4.3, 11.4.4
10.4.8	10,4,34, 10,5,16, 10,5,18, 10,5,20, 10,5,21, 10,5,22, 10,5,23, 10,5,24, 10,5,25, 10,5,26, 10,5,27, 10,5,28, 10,6,19, 10,6,20, 10,6,21, 10.11.4, 10.6.17, 10.11.5, 10.11.6, 10.11.7, 10.11.8, 10.11.9, 10.11.10, 10.11.11 11

Importation de données dans une instance de base de données Amazon RDS for MariaDB

Vous pouvez utiliser plusieurs techniques différentes pour importer des données dans une instance de base de données RDS for MariaDB. La meilleure approche dépend de plusieurs facteurs :

- Source des données
- Quantité de données
- Importation ponctuelle ou continue
- Durée du temps d'arrêt

Si vous migrez également une application avec les données, il est important de prendre en compte le temps d'arrêt.

Le tableau suivant répertorie les techniques d'importation de données dans une instance de base de données RDS pour MariaDB :

> **Note**
>
> Amazon RDS prend uniquement en charge l'importation depuis Amazon S3 vers RDS pour les instances de base de données MySQL. L'importation de sauvegardes créées avec `mariadb-backup` dans RDS pour MariaDB n'est actuellement pas prise en charge.

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Base de données MariaDB existante sur site ou sur	N'importe quel compte	En continu	Minimale	Configurez la réplication avec une base de données MariaDB existante comme source de réplication. Vous pouvez configurer la réplication dans une instance de base de données	Configuration d'une réplication de position

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Amazon EC2				MariaDB à l'aide des identificateurs de transaction globaux de MariaDB (GTIDs) lorsque l'instance externe est MariaDB version 10.0.24 ou supérieure, ou en utilisant les coordonnées du journal binaire pour les instances de MariaDB sur les versions antérieures à 10.0.24. MariaDB est GTIDs implémenté différemment de GTIDs MySQL, qui n'est pas pris en charge par Amazon RDS.	de fichier journal binaire avec une instance source externe Importation de données vers une instance de base de données Amazon RDS for MariaDB avec réduction des temps d'arrêt

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Toute base de données existante	N'importe quel compte	Une seule fois ou en continu	Minimale	AWS Database Migration Service À utiliser pour migrer la base de données avec un temps d'arrêt minimal et, pour de nombreux moteurs de base de données, poursuivre la réplication continue.	Présentation d'AWS Database Migration Service et Utilisation d'une base de données compatible MySQL comme cible pour AWS DMS dans le Guide de l'utilisateur AWS Database Migration Service

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Instance de base de données MariaDB existante	N'importe quel compte	Une seule fois ou en continu	Minimale	Créez un réplica en lecture pour la réplication continue. Promouvez le réplica en lecture pour la création unique d'une nouvelle instance de base de données.	Utilisation des réplicas en lecture d'instance de base de données
Base de données MariaDB existante	Petite	Une seule fois	Momentanée	Copiez les données directement sur votre instance de base de données MariaDB à l'aide d'un utilitaire de ligne de commande.	Importation de données depuis une base de données MariaDB externe vers une instance de base de données Amazon RDS for MariaDB

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Données non stockées dans une base de données existante	Medium	Une seule fois	Momentanée	Créez des fichiers plats et importez-les à l'aide des instructions `LOAD DATA LOCAL INFILE` MariaDB.	Importation de données depuis n'importe quelle source vers une instance de base de données Amazon RDS for MariaDB

> **Note**
>
> La base de données système `mysql` contient les informations d'authentification et d'autorisation requises pour se connecter à l'instance de base de données et accéder aux données. Supprimer, modifier, renommer ou tronquer des tables, des données ou d'autres contenus de la `mysql` base de données dans votre instance de base de données peut entraîner des erreurs et rendre l'instance de base de données et vos données inaccessibles. Dans ce cas, vous pouvez restaurer l'instance de base de données à partir d'un instantané à l'aide du AWS CLI restore-db-instance-from-db-snapshotcommande. Vous pouvez récupérer l'instance de base de données à l'aide de restore-db-instance-to-point-in-timecommande.

Considérations relatives à l'importation de données pour MariaDB

Le contenu suivant contient des informations techniques relatives au chargement de données dans MariaDB. Ce contenu est destiné aux utilisateurs qui connaissent l'architecture du serveur MariaDB.

Journalisation binaire

L'activation de la journalisation binaire réduit les performances de chargement des données et nécessite jusqu'à quatre fois plus d'espace disque que la journalisation désactivée. La taille des transactions utilisées pour charger les données influe directement sur les performances du système et les besoins en espace disque. Les transactions plus importantes nécessitent davantage de ressources.

Taille de la transaction

La taille des transactions influence les aspects suivants du chargement de données MariaDB :

- Consommation de ressources
- Utilisation de l'espace disque
- Processus de CV
- Il est temps de récupérer
- Format d'entrée (fichiers plats ou SQL)

Cette section décrit comment la taille de la transaction influe sur la journalisation binaire. En outre, elle plaide en faveur de la désactivation de cette même journalisation lors de chargements de données volumineux. Vous pouvez activer et désactiver la journalisation binaire en définissant la période de conservation automatique des sauvegardes Amazon RDS. Les valeurs différentes de zéro activent la journalisation binaire, tandis que la valeur zéro la désactive. Pour de plus amples informations, veuillez consulter Période de rétention des sauvegardes.

Cette section décrit également l'impact des transactions importantes sur InnoDB et explique pourquoi il est important de réduire la taille des transactions.

Petites transactions

Pour les petites transactions, la journalisation binaire multiplie par deux le nombre d'écritures sur disque requises pour charger les données. Elle peut ainsi affecter gravement les performances des autres sessions de base de données et accroître le temps requis pour charger les données. La dégradation subie dépend en partie des facteurs suivants :

- Taux de téléchargement
- Autres activités de base de données ayant lieu pendant le chargement
- Capacité de votre instance de base de données Amazon RDS

Les journaux binaires consomment également un espace disque à peu près égal à la quantité de données chargées jusqu'à ce que les journaux soient sauvegardés et supprimés. Amazon RDS minimise ce problème en sauvegardant et en supprimant fréquemment les journaux binaires.

Transactions volumineuses

Pour les transactions importantes, la journalisation binaire triple les IOPS et l'utilisation du disque pour les raisons suivantes :

- Le cache du journal binaire stocke temporairement les données de transaction sur le disque.
- Ce cache augmente avec la taille de la transaction, ce qui consomme de l'espace disque.
- Lorsque la transaction (validation ou annulation) est terminée, le système copie le cache dans le journal binaire.

Ce processus crée trois copies des données :

- Les données d'origine
- Le cache sur le disque
- La dernière entrée du journal binaire

Chaque opération d'écriture entraîne des E/S supplémentaires, ce qui a un impact supplémentaire sur les performances.

De ce fait, la journalisation binaire nécessite trois fois plus d'espace disque que la journalisation désactivée. Par exemple, le chargement de 10 GiB de données en une seule transaction crée trois copies :

- 10 GiB pour les données du tableau
- 10 GiB pour le cache du journal binaire
- 10 GiB pour le fichier journal binaire

L'espace disque temporaire total requis est de 30 GiB.

Considérations importantes relatives à l'espace disque :

- Le fichier de cache est conservé jusqu'à la fin de la session ou jusqu'à ce qu'une nouvelle transaction crée un autre cache.
- Le journal binaire est conservé jusqu'à ce qu'il soit sauvegardé et peut contenir 20 GiB (cache et journal) pendant une période prolongée.

Si vous avez l'`LOAD DATA LOCAL INFILE`habitude de charger les données, Data Recovery crée une quatrième copie au cas où la base de données doive être restaurée à partir d'une sauvegarde effectuée avant le chargement. Pendant la restauration, MariaDB extrait les données du journal binaire dans un fichier plat. MariaDB s'exécute ensuite. `LOAD DATA LOCAL INFILE` Sur la base de l'exemple précédent, cette restauration nécessite un espace disque temporaire total de 40 Go, soit 10 Go chacun pour la table, le cache, le journal et le fichier local. Sans au moins 40 GiB d'espace disque disponible, la restauration échoue.

Optimisation des charges de données importantes

Pour les chargements de données importants, désactivez la journalisation binaire afin de réduire les frais généraux et les besoins en espace disque. Vous pouvez désactiver la journalisation binaire en définissant la période de conservation des sauvegardes sur 0. Une fois le chargement terminé, restaurez la période de conservation des sauvegardes à la valeur appropriée différente de zéro. Pour plus d'informations, reportez-vous Modification d'une RDS instance de base de données Amazon à la section « Durée de conservation des sauvegardes » dans le tableau des paramètres.

> **Note**
>
> Si l'instance de base de données est une instance de base de données source pour les répliques de lecture, vous ne pouvez pas définir la période de rétention des sauvegardes sur 0.

Avant de charger les données, nous vous recommandons de créer un instantané de base de données. Pour de plus amples informations, veuillez consulter Gestion des sauvegardes manuelles.

InnoDB

Les informations suivantes sur les options d'annulation de journalisation et de restauration permettent de limiter les transactions InnoDB afin d'optimiser les performances de la base de données.

Comprendre la journalisation des annulations par InnoDB

L'annulation est un mécanisme de journalisation qui permet l'annulation des transactions et prend en charge le contrôle de simultanéité multiversion (MVCC).

Pour les versions 10.11 et antérieures de MariaDB, les journaux d'annulation sont stockés dans le tablespace du système InnoDB (généralement ibdata1) et sont conservés jusqu'à ce que le thread de purge les supprime. Par conséquent, des transactions de chargement de données importantes peuvent entraîner une augmentation importante de l'espace disque logique du système et consommer de l'espace disque que vous ne pouvez récupérer que si vous recréez la base de données.

Pour toutes les versions de MariaDB, le thread de purge doit attendre que la transaction active la plus ancienne soit validée ou annulée pour supprimer les journaux d'annulation. Si la base de données traite d'autres transactions pendant le chargement, leurs journaux d'annulation s'accumulent également et ne peuvent pas être supprimés, même si les transactions sont validées et qu'aucune autre transaction n'a besoin des journaux d'annulation pour MVCC. Dans ce cas, toutes les transactions, y compris les transactions en lecture seule, ralentissent. Ce ralentissement se produit parce que toutes les transactions accèdent à toutes les lignes modifiées par toutes les transactions, et pas seulement par les transactions de chargement. En effet, les transactions doivent parcourir les journaux d'annulation que les transactions de chargement de longue durée ont empêchées d'être purgées lors du nettoyage du journal d'annulation. Cela affecte les performances de toute opération accédant aux lignes modifiées.

Options de récupération des transactions InnoDB

Bien qu'InnoDB optimise les opérations de validation, les annulations de transactions importantes sont lentes. Pour accélérer la restauration, effectuez une point-in-time restauration ou restaurez un instantané de base de données. Pour plus d’informations, consultez Point-in-time rétablissement et Restauration vers une instance de base de données.

Formats d'importation de données

MariaDB prend en charge deux formats d'importation de données : les fichiers plats et le SQL. Passez en revue les informations relatives à chaque format afin de déterminer l'option la mieux adaptée à vos besoins.

Fichiers plats

Pour les petites transactions, chargez des fichiers plats avec`LOAD DATA LOCAL INFILE`. Ce format d'importation de données peut offrir les avantages suivants par rapport à l'utilisation de SQL :

- Moins de trafic réseau
- Réduction des coûts de transmission de données
- Diminution des frais de traitement des bases
- Traitement plus rapide

`LOAD DATA LOCAL INFILE`charge l'intégralité du fichier plat en une seule transaction. Réduisez la taille de chaque fichier pour bénéficier des avantages suivants :

- Fonctionnalité de reprise : vous pouvez suivre les fichiers qui ont été chargés. Si un problème survient pendant le chargement, vous pouvez reprendre là où vous vous étiez arrêté. Il se peut que vous deviez retransmettre certaines données vers Amazon RDS, mais dans le cas de petits fichiers, le volume retransmis est minime.
- Chargement parallèle des données : si vous disposez de suffisamment d'IOPS et de bande passante réseau pour le chargement d'un seul fichier, le chargement en parallèle peut vous faire gagner du temps.
- Contrôle du taux de charge : si le chargement de vos données a un impact négatif sur les autres processus, vous pouvez contrôler le taux de charge en augmentant l'intervalle entre les fichiers.

Les transactions importantes réduisent les avantages liés `LOAD DATA LOCAL INFILE` à l'importation de données. Lorsque vous ne pouvez pas diviser une grande quantité de données en fichiers plus petits, pensez à utiliser le langage SQL.

SQL

Le SQL présente un avantage majeur par rapport aux fichiers plats : vous pouvez facilement réduire la taille des transactions. Cependant, le chargement du SQL peut prendre beaucoup plus de temps que celui des fichiers plats. De plus, après un échec, il peut être difficile de déterminer où reprendre : vous ne pouvez pas redémarrer les fichiers mariadb-dump. Si un échec survient lors du chargement du fichier mariadb-dump, vous devez modifier ou remplacer le fichier avant que le chargement ne puisse reprendre. Ou bien, après avoir corrigé la cause de l'échec, vous pouvez restaurer le fichier à l'endroit où il se trouvait avant le chargement et renvoyer le fichier. Pour de plus amples informations, veuillez consulter Point-in-time rétablissement.

Utilisation des instantanés de base de données Amazon RDS pour les points de contrôle de base de données

Si vous chargez des données sur de longues durées, par exemple des heures ou des jours, sans journalisation binaire, utilisez des instantanés de base de données pour fournir des points de contrôle périodiques en matière de sécurité des données. Chaque instantané de base de données crée une copie cohérente de votre instance de base de données qui sert de point de restauration lors de défaillances du système ou d'événements de corruption de données. Les instantanés de base de données étant rapides, les points de contrôle fréquents ont un impact minimal sur les performances de chargement. Vous pouvez supprimer des instantanés de base de données précédents sans affecter la durabilité ou les capacités de restauration de la base de données. Pour plus d'informations sur les instantanés de base de données, consultezGestion des sauvegardes manuelles.

Réduction des temps de chargement des bases de données

Voici quelques conseils supplémentaires pour réduire les temps de chargement :

- Créez tous les index secondaires avant de charger les données dans les bases de données MariaDB. Contrairement aux autres systèmes de base de données, MariaDB reconstruit l'intégralité de la table lors de l'ajout ou de la modification d'index secondaires. Ce processus crée une nouvelle table avec les modifications d'index, copie toutes les données et supprime la table d'origine.
- Chargez les données dans l'ordre des clés primaires. Pour les tables InnoDB, cela peut réduire les temps de chargement de 75 % à 80 % et la taille des fichiers de données de 50 %.
- Désactivez les contraintes liées aux clés étrangères en définissant `foreign_key_checks` sur`0`. Cela est souvent nécessaire pour les fichiers plats chargés avec`LOAD DATA LOCAL INFILE`. Quel que soit le chargement, la désactivation des vérifications par clé étrangère accélère le chargement des données. Une fois le chargement terminé, réactivez les contraintes en configurant `1` et `foreign_key_checks` en vérifiant les données.
- Chargez les données en parallèle, sauf si vous approchez d'une limite de ressources. Pour permettre le chargement simultané sur plusieurs segments de table, utilisez des tables partitionnées le cas échéant.
- Pour réduire la charge d'exécution du code SQL, combinez plusieurs `INSERT` instructions en `INSERT` opérations uniques à valeurs multiples. `mariadb-dump`implémente automatiquement cette optimisation.

- Réduisez les opérations d'E/S du journal InnoDB en `innodb_flush_log_at_trx_commit` réglant sur. `0` Une fois le chargement terminé, restaurez `innodb_flush_log_at_trx_commit` vers`1`.

  > **Warning**
  >
  > Le réglage `innodb_flush_log_at_trx_commit` sur `0` oblige InnoDB à vider ses journaux toutes les secondes plutôt qu'à chaque validation. Ce paramètre améliore les performances mais peut entraîner la perte de transactions en cas de défaillance du système.

- Si vous chargez des données dans une instance de base de données qui ne possède pas de répliques de lecture, définissez `sync_binlog` cette valeur sur. `0` Une fois le chargement terminé, restaurez `sync_binlog parameter` vers`1`.
- Chargez les données dans une instance de base de données mono-AZ avant de convertir l'instance de base de données en un déploiement multi-AZ. Si l'instance de base de données utilise déjà un déploiement multi-AZ, nous ne recommandons pas de passer à un déploiement mono-AZ pour le chargement des données. Cela n'apporte que des améliorations marginales.

Importation de données depuis une base de données MariaDB externe vers une instance de base de données Amazon RDS for MariaDB

Vous pouvez importer des données d'une base de données MariaDB ou MySQL existante vers une instance de base de données RDS pour MariaDB ou RDS pour MySQL. Pour ce faire, copiez la base de données avec mysqldump et redirigez-la directement vers l'instance de base de données RDS pour MariaDB ou RDS pour MySQL. L'utilitaire de ligne de commande `mysqldump` est généralement utilisé pour effectuer des sauvegardes et des transferts de données d'un serveur MariaDB ou MySQL vers un autre. Il est inclus dans les logiciels clients MySQL et MariaDB.

Dans MariaDB 10.5, le client s'appelle mariadb-dump. À partir de MariaDB 11.0.1, vous devez utiliser à la place de. `mariadb-dump` `mysqldump`

> **Note**
>
> Si vous importez ou exportez de grandes quantités de données avec une instance de base de données MySQL, le transfert de données vers et depuis Amazon RDS est plus fiable et plus rapide à l'aide de fichiers de `xtrabackup` sauvegarde et d'Amazon S3. Pour de plus

amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.
Amazon RDS ne prend pas en charge `xtrabackup` MariaDB ou. `mariabackup` Amazon RDS prend uniquement en charge l'importation depuis Amazon S3 pour MySQL.

Une `mysqldump` commande classique pour déplacer des données d'une base de données externe vers une instance de base de données Amazon RDS ressemble à l'exemple suivant. Remplacez les valeurs par vos propres informations. Pour les versions 11.0.1 et supérieures de MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

```
mysqldump -u local_user \
    --databases database_name \
    --single-transaction \
    --compress \
    --order-by-primary  \
    --routines=0 \
    --triggers=0 \
    --events=0 \
    -plocal_password | mysql -u RDS_user \
        --port=port_number \
        --host=host_name \
        -pRDS_password
```

Important

Veillez à ne pas laisser d'espace entre l'option `-p` et le mot de passe saisi.
Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans cet exemple.

Assurez-vous que vous êtes conscient des recommandations et des considérations suivantes :

- Excluez les schémas suivants du fichier de vidage :
  - `sys`
  - `performance_schema`
  - `information_schema`

  L'`mariadb-dump`utilitaire `mysqldump` and exclut ces schémas par défaut.

- Si vous devez migrer des utilisateurs et des privilèges, pensez à utiliser un outil qui génère le langage de contrôle des données (DCL) pour les recréer, tel que l'pt-show-grantsutilitaire.
- Pour effectuer l'importation, assurez-vous que l'utilisateur qui l'effectue a accès à l'instance de base de données. Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité.

Les paramètres utilisés sont les suivants :

- `-u local_user` – Utilisez ce paramètre pour spécifier un nom d'utilisateur. Lors de la première utilisation de ce paramètre, spécifiez le nom d'un compte utilisateur sur la base de données MariaDB ou MySQL locale que vous identifiez `--databases` avec le paramètre.
- `--databases database_name` : utilisez ce paramètre pour spécifier le nom de la base de données sur l'instance MariaDB ou MySQL locale que vous souhaitez importer dans Amazon RDS.
- `--single-transaction` – Utilisez ce paramètre pour vérifier que toutes les données chargées depuis la base de données locale sont en cohérence avec un point dans le temps unique. S'il existe d'autres processus qui modifient les données pendant que `mysqldump` les lit, l'utilisation de ce paramètre permet de maintenir l'intégrité des données.
- `--compress` – Utilisez ce paramètre pour réduire la consommation de bande passante réseau par compression des données à partir de la base de données locale avant de les envoyer vers Amazon RDS.
- `--order-by-primary` – Utilisez ce paramètre pour réduire le temps de chargement en triant les données de chaque tableau sur par clé primaire.
- `--routines`— À utiliser si des routines telles que des procédures stockées ou des fonctions existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les routines pendant le processus d'importation. Ensuite, recréez manuellement les routines dans la base de données Amazon RDS.
- `--triggers`— À utiliser si des déclencheurs existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les déclencheurs pendant le processus d'importation. Ensuite, recréez manuellement les déclencheurs dans la base de données Amazon RDS.
- `--events`— À utiliser si des événements existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les événements pendant le processus d'importation. Recréez ensuite manuellement les événements dans la base de données Amazon RDS.
- `-plocal_password` – Utilisez ce paramètre pour spécifier un mot de passe. Lors de la première utilisation de ce paramètre, spécifiez le mot de passe du compte utilisateur que vous identifiez avec le premier `-u` paramètre.

- `-u` *`RDS_user`* – Utilisez ce paramètre pour spécifier un nom d'utilisateur. Lors de la deuxième utilisation de ce paramètre, spécifiez le nom d'un compte utilisateur sur la base de données par défaut pour l'instance de base de données MariaDB ou MySQL que vous identifiez `--host` avec le paramètre.
- `--port` *`port_number`* : utilisez ce paramètre pour spécifier le port pour votre instance de base de données MariaDB ou MySQL. Par défaut, il s'agit de 3306, sauf si vous avez modifié la valeur lors de la création de l'instance de base de données.
- `--host` *`host_name`* : utilisez ce paramètre pour spécifier le nom du système de nom de domaine (DNS) du point de terminaison de l'instance de base de données Amazon RDS, par exemple, `myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez trouver la valeur du point de terminaison dans les détails de l'instance de base de données dans la console Amazon RDS.
- `-p`*`RDS_password`* – Utilisez ce paramètre pour spécifier un mot de passe. Lors de la seconde utilisation de ce paramètre, vous spécifiez le mot de passe du compte utilisateur identifié par le second paramètre `-u`.

Assurez-vous de créer manuellement les procédures stockées, déclencheurs, fonctions ou événements dans votre base de données Amazon RDS. Si la base de données que vous copiez contient l'un de ces objets, excluez-le lorsque vous exécutez `mysqldump` ou`mariadb-dump`. Pour ce faire, incluez les paramètres suivants dans votre `mariadb-dump` commande `mysqldump` or :

- `--routines=0`
- `--triggers=0`
- `--events=0`

Exemple

L'exemple suivant copie l'`world`exemple de base de données sur l'hôte local vers une instance de base de données RDS pour MySQL. Remplacez les valeurs par vos propres informations. Pour copier l'exemple de base de données sur une instance de base de données RDS pour MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

Dans Linux, macOS, ou Unix:

```
sudo mysqldump -u local_user \
    --databases world \
```

```
    --single-transaction \
    --compress \
    --order-by-primary  \
    --routines=0 \
    --triggers=0 \
    --events=0 \
    -plocal_password | mysql -u rds_user \
        --port=3306 \
        --host=my_instance.123456789012.us-east-1.rds.amazonaws.com \
        -pRDS_password
```

Dans Windows:

Exécutez la commande suivante dans une invite de commande qui a été ouverte en cliquant avec le bouton droit sur Invite de commandes dans le menu des programmes Windows et en choisissant Exécuter en tant qu'administrateur. Remplacez les valeurs par vos propres informations. Pour copier l'exemple de base de données sur une instance de base de données RDS pour MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

```
mysqldump -u local_user ^
    --databases world ^
    --single-transaction ^
    --compress ^
    --order-by-primary  ^
    --routines=0 ^
    --triggers=0 ^
    --events=0 ^
    -plocal_password | mysql -u RDS_user ^
        --port=3306 ^
        --host=my_instance.123456789012.us-east-1.rds.amazonaws.com ^
        -pRDS_password
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

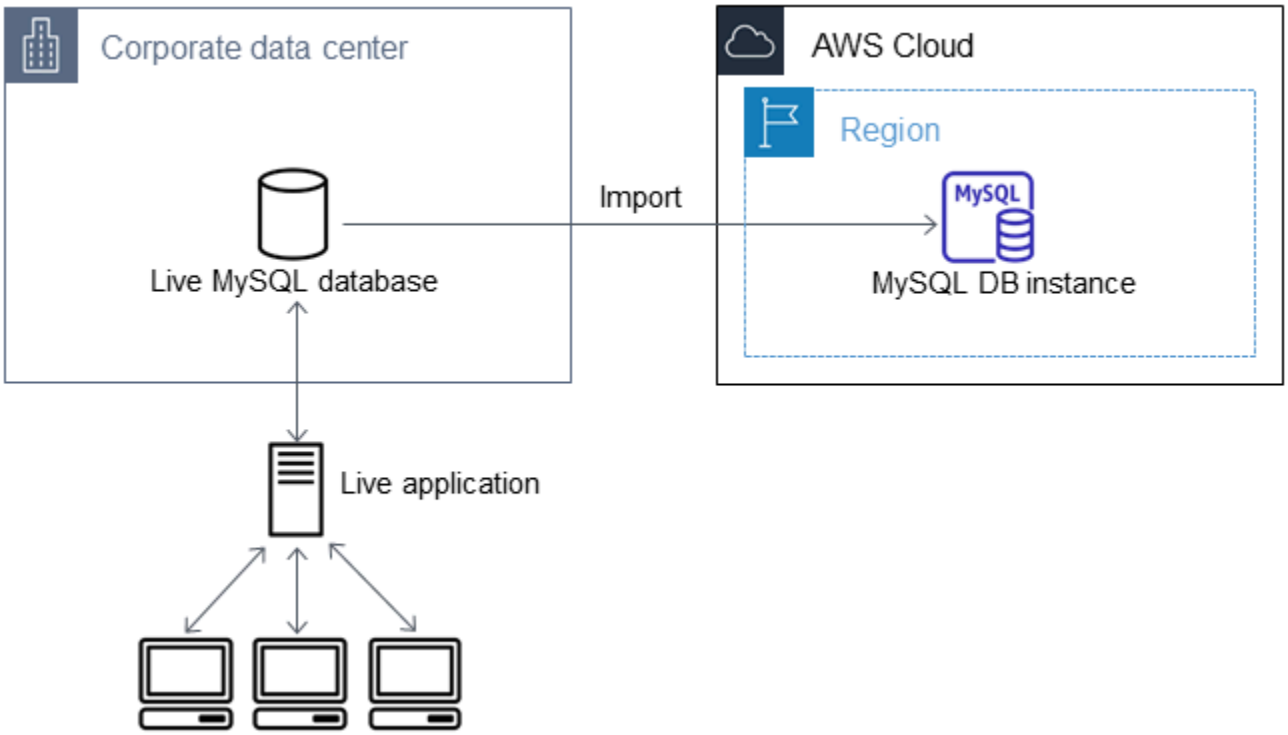
Importation de données vers une instance de base de données Amazon RDS for MariaDB avec réduction des temps d'arrêt

Dans certains cas, vous devrez peut-être importer des données d'une base de données MariaDB ou MySQL externe qui prend en charge une application en direct vers une instance de base de données RDS pour MariaDB, une instance de base de données RDS pour MySQL ou un cluster de base de données multi-AZ RDS pour MySQL. Utilisez la procédure suivante pour réduire l'impact sur la disponibilité des applications. Cette procédure peut s'avérer également utile si vous travaillez avec une base de données très volumineuse. À l'aide de cette procédure, vous pouvez réduire le coût de l'importation en réduisant la quantité de données transmises sur le réseau AWS.

Dans cette procédure, vous transférez une copie des données de votre base de données vers une EC2 instance Amazon et vous importez les données dans une nouvelle base de données Amazon RDS. Vous utilisez ensuite la réplication pour intégrer la base de données Amazon RDS up-to-date à votre instance externe active, avant de rediriger votre application vers la base de données Amazon RDS. Si l'instance externe est MariaDB 10.0.24 ou version ultérieure et que l'instance cible est RDS pour MariaDB, configurez la réplication MariaDB en fonction des identifiants de transaction globaux (). GTIDs Sinon, configurez la réplication en fonction des coordonnées des journaux binaires. Nous recommandons la réplication GTID si votre base de données externe la prend en charge, car la réplication GTID est une méthode plus fiable. Pour plus d'informations, consultez Identificateurs de transaction mondiaux dans la documentation MariaDB.

> **Note**
>
> Si vous souhaitez importer des données dans une instance de base de données RDS pour MySQL et que votre scénario le permet, nous vous recommandons de déplacer les données vers et depuis Amazon RDS à l'aide de fichiers de sauvegarde et d'Amazon S3. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.

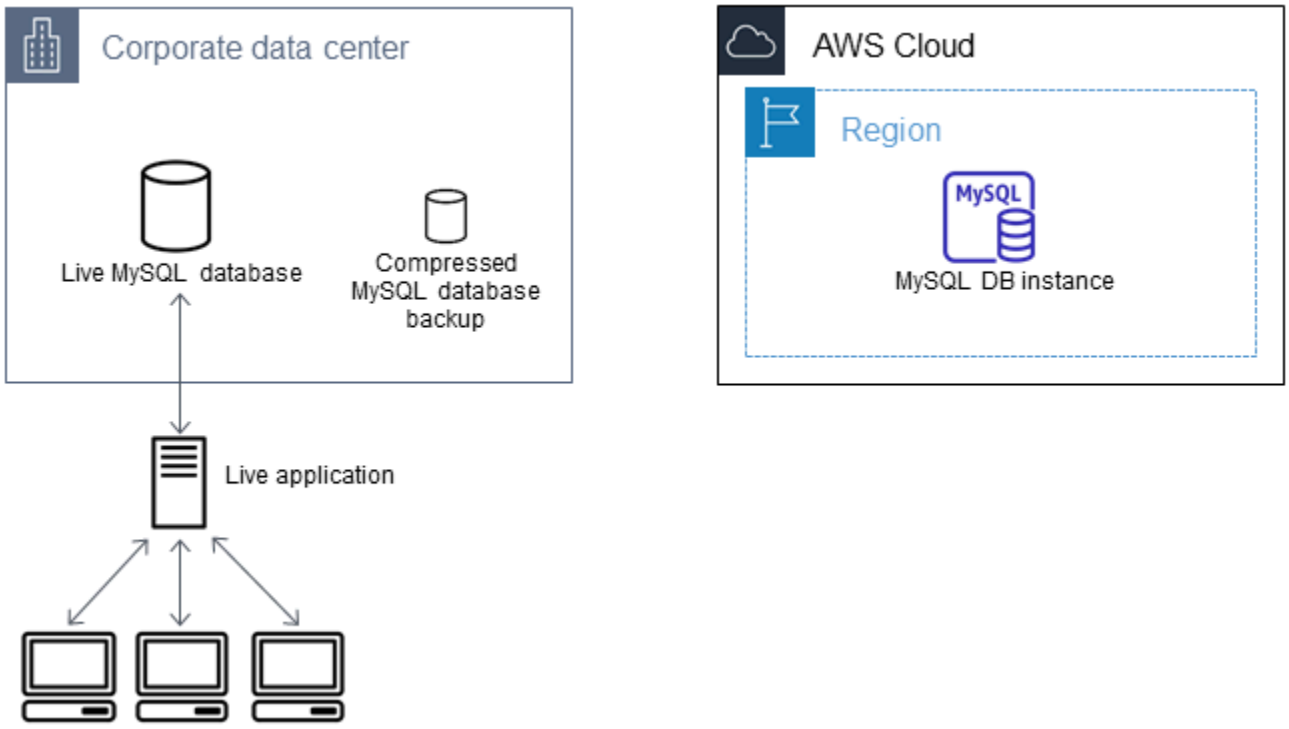
Le schéma suivant montre l'importation d'une base de données MySQL externe dans une base de données MySQL sur Amazon RDS.



Tâche 1 : créer une copie de votre base de données existante

La première étape du processus de migration d'une grande quantité de données vers une base de données RDS for MariaDB ou RDS for MySQL avec un temps d'arrêt minimal consiste à créer une copie des données sources.

Le schéma suivant montre la création d'une sauvegarde de la base de données MySQL.



Vous pouvez recourir à l'utilitaire `mysqldump` pour créer une sauvegarde de base de données au format SQL ou texte délimité. Dans MariaDB 10.5, le client s'appelle mariadb-dump. À partir de MariaDB 11.0.1, vous devez utiliser à la place de. `mariadb-dump` `mysqldump` Nous vous recommandons d'effectuer un test avec chaque format dans un environnement autre que celui de production afin de déterminer la méthode qui minimise le temps d'exécution de mysqldump.

Nous vous recommandons également d'évaluer les `mysqldump` performances par rapport aux avantages offerts en utilisant le format de texte délimité pour le chargement. Une sauvegarde à l'aide du format texte délimité crée un fichier texte séparé par des tabulations pour chaque table vidée. Pour réduire le temps nécessaire à l'importation de votre base de données, vous pouvez charger ces fichiers en parallèle en utilisant la commande `LOAD DATA LOCAL INFILE`. Pour plus d'informations, consultez Charger les données pour MariaDB ou Charger les données pour MySQL.

Avant de commencer l'opération de sauvegarde, assurez-vous de définir les options de réplication sur la base de données MariaDB ou MySQL que vous copiez vers Amazon RDS. Les options de réplication incluent l'activation de la journalisation binaire et la configuration d'un ID de serveur unique. La définition de ces options oblige votre serveur à démarrer la journalisation des transactions de base de données et le prépare à être une instance de réplication source ultérieurement dans le processus.

Assurez-vous que vous êtes conscient des recommandations et des considérations suivantes :

- Utilisez l'option `--single-transaction` avec `mysqldump`, car elle vide un état cohérent de la base de données. Pour vous assurer de la validité d'un fichier de vidage, n'exécutez pas les instructions DDL (Data Definition Language) pendant l'exécution de `mysqldump`. Vous pouvez planifier une fenêtre de maintenance pour ces opérations.
- Excluez les schémas suivants du fichier de vidage :
  - `sys`
  - `performance_schema`
  - `information_schema`

  L'`mariadb-dump`utilitaire `mysqldump` and exclut ces schémas par défaut.
- Si vous devez migrer des utilisateurs et des privilèges, pensez à utiliser un outil qui génère le langage de contrôle des données (DCL) pour les recréer, tel que l'pt-show-grantsutilitaire.

Pour définir les options de réplication

1. Modifiez le fichier `my.cnf`. Ce fichier se trouve généralement sous`/etc`.

   ```
   sudo vi /etc/my.cnf
   ```

   Ajoutez les options `log_bin` et `server_id` à la section `[mysqld]`. L'option `log_bin` fournit un identifiant de nom de fichier pour les fichiers journaux binaires. L'option `server_id` fournit un identifiant unique pour le serveur dans les relations source/réplica.

   L'exemple suivant montre la `[mysqld]` section mise à jour d'un `my.cnf` fichier :

   ```
   [mysqld]
   log-bin=mysql-bin
   server-id=1
   ```

   Pour plus d'informations, consultez la section Définition de la configuration de la source de réplication dans la documentation MySQL.
2. Pour la réplication avec un cluster de base de données multi-AZ, définissez le `GTID_MODE` paramètre `ENFORCE_GTID_CONSISTENCY` et sur`ON`.

   ```
   mysql> SET @@GLOBAL.ENFORCE_GTID_CONSISTENCY = ON;
   ```

```
mysql> SET @@GLOBAL.GTID_MODE = ON;
```

Ces paramètres ne sont pas requis pour la réplication avec une instance de base de données.

3. Redémarrez le service `mysql`.

```
sudo service mysqld restart
```

Pour créer une copie de sauvegarde de votre base de données existante

1. Créez une sauvegarde de vos données à l'aide de l'`mysqldump`utilitaire ou de l'`mariadb-dump`utilitaire, en spécifiant le format SQL ou le format texte délimité.

   Pour MySQL 8.0.25 et versions antérieures, spécifiez de `--master-data=2` créer un fichier de sauvegarde qui peut être utilisé pour démarrer la réplication entre les serveurs. Pour MySQL 8.0.26 et versions supérieures, spécifiez de `--source-data=2` créer un fichier de sauvegarde qui peut être utilisé pour démarrer la réplication entre les serveurs. Pour plus d'informations, consultez mysqldump — A Database Backup Program dans la documentation MySQL.

   Pour améliorer les performances et garantir l'intégrité des données, utilisez les `--single-transaction` options `--order-by-primary` et pour`mysqldump`.

   Pour éviter d'inclure la base de données système MySQL dans la sauvegarde, n'utilisez pas l'`--all-databases`option with`mysqldump`. Pour plus d'informations, consultez la section Création d'un instantané de données à l'aide de mysqldump dans la documentation MySQL.

   Utilisez`chmod`, si nécessaire, pour vous assurer que le répertoire dans lequel le fichier de sauvegarde est créé est inscriptible.

   > **Important**
   >
   > Sur Windows, exécutez la fenêtre de commande en tant qu'administrateur.

   - Pour produire une sortie SQL, utilisez la commande suivante :

     Dans Linux, macOS, ou Unix:

```
sudo mysqldump \
    --databases database_name \
    --master-data=2  \
    --single-transaction \
    --order-by-primary \
    -r backup.sql \
    -u local_user \
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

Dans Windows:

```
mysqldump ^
    --databases database_name ^
    --master-data=2  ^
    --single-transaction ^
    --order-by-primary ^
    -r backup.sql ^
    -u local_user ^
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

- Pour produire une sortie en texte délimité, utilisez la commande suivante :

  Dans Linux, macOS, ou Unix:

```
sudo mysqldump \
    --tab=target_directory \
    --fields-terminated-by ',' \
    --fields-enclosed-by '"' \
```

```
    --lines-terminated-by 0x0d0a \
    database_name \
    --master-data=2 \
    --single-transaction \
    --order-by-primary \
    -ppassword
```

Dans Windows:

```
mysqldump ^
    --tab=target_directory ^
    --fields-terminated-by "," ^
    --fields-enclosed-by """ ^
    --lines-terminated-by 0x0d0a ^
    database_name ^
    --master-data=2 ^
    --single-transaction ^
    --order-by-primary ^
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.
> Assurez-vous de créer manuellement les procédures stockées, déclencheurs, fonctions ou événements dans votre base de données Amazon RDS. Si vous avez l'un de ces objets dans la base de données que vous copiez, excluez-les lorsque lors de l'exécution de `mysqldump`. Pour ce faire, incluez les arguments suivants dans votre `mysqldump` commande :
>
> - `--routines=0`
> - `--triggers=0`
> - `--events=0`

Pour MySQL 8.0.22 et versions antérieures et pour les versions MariaDB, lorsque vous exécutez `mysqldump` et spécifiez le format de texte délimité, un commentaire est renvoyé. `CHANGE MASTER TO` Ce commentaire contient le nom du fichier journal maître et son emplacement. Pour MySQL 8.0.23 et versions supérieures, lorsque vous `mysqldump` utilisez le format de texte

délimité, un commentaire est renvoyé. `CHANGE REPLICATION SOURCE TO` Ce commentaire contient le nom et la position du fichier journal source. Si l'instance externe est autre chose que MariaDB 10.0.24 et versions supérieures, ou MySQL 8.0.23 et versions supérieures, notez les valeurs pour et. `MASTER_LOG_FILE` `MASTER_LOG_POS` Vous avez besoin de ces valeurs lors de la configuration de la réplication.

La sortie suivante est renvoyée pour les versions de MariaDB et pour MySQL 8.0.22 et versions antérieures :

```
-- Position to start replication or point-in-time recovery from
--
-- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin-changelog.000031',
 MASTER_LOG_POS=107;
```

La sortie suivante est renvoyée pour MySQL 8.0.23 et versions supérieures :

```
-- Position to start replication or point-in-time recovery from
--
-- CHANGE SOURCE TO SOURCE_LOG_FILE='mysql-bin-changelog.000031',
 SOURCE_LOG_POS=107;
```

Pour MySQL 8.0.22 et versions antérieures, si vous utilisez le format SQL, vous pouvez obtenir le nom et la position du fichier journal principal dans le `CHANGE MASTER TO` commentaire du fichier de sauvegarde. Pour MySQL 8.0.23 et versions supérieures, si vous utilisez le format SQL, vous pouvez obtenir le nom et la position du fichier journal source dans le `CHANGE REPLICATION SOURCE TO` commentaire du fichier de sauvegarde. Si l'instance externe est MariaDB version 10.0.24 ou ultérieure, vous pouvez obtenir l'identifiant de transaction global à l'étape suivante.

2. Si l'instance externe que vous utilisez est MariaDB version 10.0.24 ou supérieure, utilisez la réplication basée sur GTID. Exécutez `SHOW MASTER STATUS` sur l'instance externe de MariaDB pour obtenir le nom et la position du fichier journal binaire, puis convertissez-les en GTID en les exécutant `BINLOG_GTID_POS` sur l'instance externe de MariaDB.

   ```
   SELECT BINLOG_GTID_POS('binary_log_file_name', binary_log_file_position);
   ```

   Notez le GTID renvoyé. Vous avez besoin du GTID pour configurer la réplication.

3. Compressez les données copiées afin de réduire la quantité de ressources réseau nécessaires pour copier vos données sur la base de données Amazon RDS. Notez la taille du fichier de sauvegarde. Vous avez besoin de ces informations pour déterminer la taille d'une EC2 instance Amazon à créer. Lorsque vous avez terminé, compressez le fichier de sauvegarde à l'aide de GZIP ou de votre utilitaire de compression favori.
   - Pour compresser la sortie SQL, utilisez la commande suivante :

     ```
     gzip backup.sql
     ```

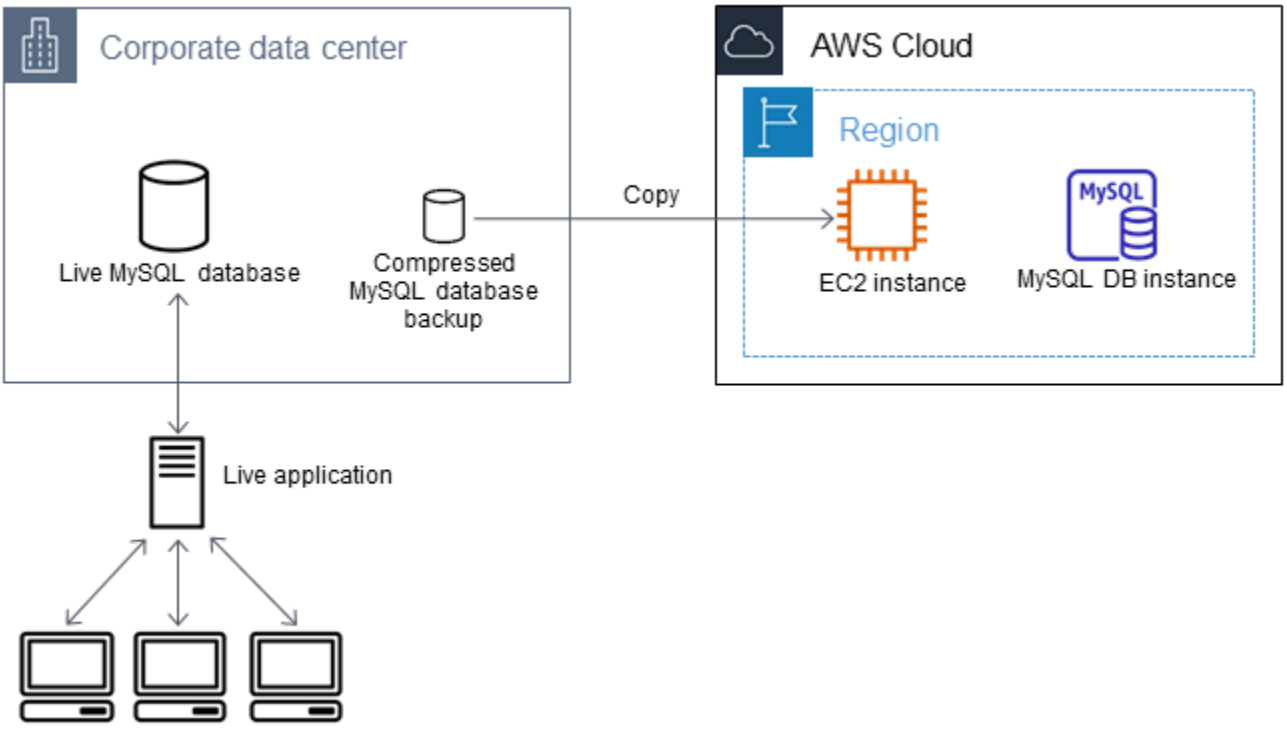
   - Pour compresser une sortie en texte délimité, utilisez la commande suivante :

     ```
     tar -zcvf backup.tar.gz target_directory
     ```

Tâche 2 : créer une EC2 instance Amazon et copier la base de données compressée

La copie de votre fichier de sauvegarde de base de données compressé sur une EC2 instance Amazon nécessite moins de ressources réseau que la copie directe de données non compressées entre des instances de base de données. Une fois que vos données sont dans Amazon EC2, vous pouvez les copier directement dans votre base de données MariaDB ou MySQL. Pour que vous puissiez économiser sur le coût des ressources réseau, votre EC2 instance Amazon doit être Région AWS identique à votre instance de base de données Amazon RDS. Le fait que l' EC2 instance Amazon soit Région AWS identique à votre base de données Amazon RDS réduit également la latence du réseau lors de l'importation.

Le schéma suivant montre comment copier la sauvegarde de la base de données sur une EC2 instance Amazon.



Pour créer une EC2 instance Amazon et copier vos données

1. Dans l' Région AWS endroit où vous prévoyez de créer la base de données Amazon RDS, créez un cloud privé virtuel (VPC), un groupe de sécurité VPC et un sous-réseau VPC. Assurez-vous que les règles entrantes de votre groupe de sécurité VPC autorisent les adresses IP requises pour que votre application se connecte à AWS. Vous pouvez spécifier une plage d'adresses IP, par exemple, ou `203.0.113.0/24` un autre groupe de sécurité VPC. Vous pouvez utiliser la console Amazon VPC pour créer et gérer des sous-réseaux et VPCs des groupes de sécurité. Pour plus d'informations, consultez Getting started with Amazon VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.
2. Ouvrez la EC2 console Amazon et choisissez celle qui doit contenir Région AWS à la fois votre EC2 instance Amazon et votre base de données Amazon RDS. Lancez une EC2 instance Amazon à l'aide du VPC, du sous-réseau et du groupe de sécurité que vous avez créés à l'étape 1. Vérifiez que vous sélectionnez un type d'instance avec un stockage suffisant pour le fichier de sauvegarde de votre base de données une fois qu'il est décompressé. Pour en savoir plus sur EC2 les instances Amazon, consultez Getting started with Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.
3. Pour vous connecter à votre base de données Amazon RDS depuis votre EC2 instance Amazon, modifiez votre groupe de sécurité VPC. Ajoutez une règle entrante spécifiant l'adresse IP privée de votre EC2 instance. Vous trouverez l'adresse IP privée dans l'onglet Détails du volet Instance de

la fenêtre de EC2 console. Pour modifier le groupe de sécurité VPC et ajouter une règle entrante, choisissez Security Groups dans le volet de navigation de la EC2 console, choisissez votre groupe de sécurité, puis ajoutez une règle entrante pour MySQL ou Aurora spécifiant l'adresse IP privée de votre instance. EC2 Pour savoir comment ajouter une règle entrante à un groupe de sécurité VPC, consultez la section Règles du groupe de sécurité dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

4. Copiez le fichier de sauvegarde de votre base de données compressée depuis votre système local vers votre EC2 instance Amazon. `chmod`Utilisez-le, si nécessaire, pour vous assurer que vous disposez d'une autorisation d'écriture pour le répertoire cible de l' EC2 instance Amazon. Vous pouvez utiliser `scp` ou un client SSH pour copier le fichier. La commande suivante est un exemple de `scp` commande :

```
scp -r -i key pair.pem backup.sql.gz ec2-user@EC2 DNS:/target_directory/backup.sql.gz
```

> **Important**
>
> Lorsque vous copiez des données sensibles, veillez à utiliser un protocole de transfert réseau sécurisé.

5. Connectez-vous à votre EC2 instance Amazon et installez les dernières mises à jour et les outils client MySQL à l'aide des commandes suivantes :

```
sudo yum update -y
sudo yum install mysql -y
```

Pour plus d'informations, consultez Connect to your instance for Linux dans le guide de l'utilisateur Amazon Elastic Compute Cloud.

> **Important**
>
> Cet exemple installe le client MySQL sur une Amazon Machine Image (AMI) pour une distribution Amazon Linux. Cet exemple n'installe pas le client MySQL sur une autre distribution, telle qu'Ubuntu ou Red Hat Enterprise Linux. Pour plus d'informations sur l'installation de MySQL, consultez la section Installation de MySQL dans la documentation MySQL.

6. Lorsque vous êtes connecté à votre EC2 instance Amazon, décompressez le fichier de sauvegarde de votre base de données. Les commandes suivantes sont des exemples.
   - Pour décompresser une sortie SQL, utilisez la commande suivante :

     ```
     gzip backup.sql.gz -d
     ```

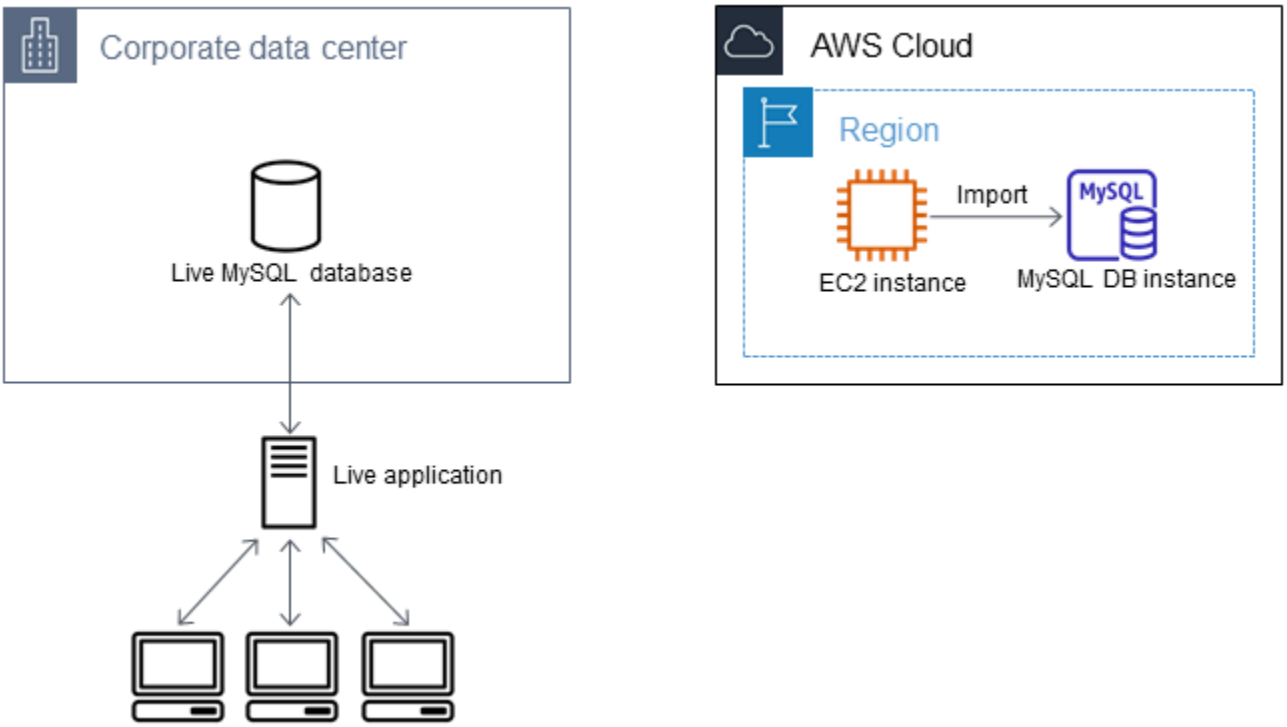
   - Pour décompresser une sortie texte délimité, utilisez la commande suivante :

     ```
     tar xzvf backup.tar.gz
     ```

Tâche 3 : créer une base de données MySQL ou MariaDB et importer des données depuis votre instance Amazon EC2

En créant une instance de base de données RDS pour MariaDB, une instance de base de données RDS pour MySQL ou un cluster de base de données multi-AZ RDS pour MySQL au même Région AWS titre que votre instance Amazon, vous pouvez importer le fichier de sauvegarde de base de données depuis EC2 Amazon plus rapidement que sur Internet. EC2

Le schéma suivant montre l'importation de la sauvegarde depuis une EC2 instance Amazon dans une base de données MySQL.



Pour créer une base de données MariaDB ou MySQL et importer vos données

1. Déterminez quelle classe d'instance de base de données et quelle quantité d'espace de stockage sont nécessaires pour prendre en charge la charge de travail attendue pour cette base de données Amazon RDS. Dans le cadre de ce processus, décidez de l'espace suffisant et de la capacité de traitement qui conviennent à vos procédures de chargement des données. Décidez également de ce qui est nécessaire pour gérer la charge de travail de production. Vous pouvez estimer ces éléments en fonction de la taille et des ressources de la base de données source MariaDB ou MySQL. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .
2. Créez une instance de base de données ou un cluster de base de données multi-AZ dans Région AWS celui qui contient votre EC2 instance Amazon.

   Pour créer un cluster de bases de données multi-AZ RDS pour MySQL, suivez les instructions de. Création d'un cluster de base de données multi-AZ pour Amazon RDS

   Pour créer une instance de base de données RDS pour MariaDB ou RDS pour MySQL, suivez les instructions Création d'une instance de base de données Amazon RDS et utilisez les directives suivantes :

   - Spécifiez une version du moteur de base de données compatible avec votre instance de base de données source.
     - Si votre instance source est MySQL 5.5.x, l'instance de base de données Amazon RDS doit être MySQL.
     - Si votre instance source est MySQL 5.6.x ou 5.7.x, l'instance de base de données Amazon RDS doit être MySQL ou MariaDB.
     - Si votre instance source est MySQL 8.0.x, l'instance de base de données Amazon RDS doit être MySQL 8.0.x.
     - Si votre instance source est MySQL 8.4.x, l'instance de base de données Amazon RDS doit être MySQL 8.4.x.
     - Si votre instance source est MariaDB 5.5 ou version ultérieure, l'instance de base de données Amazon RDS doit être MariaDB.
   - Spécifiez le même cloud privé virtuel (VPC) et le même groupe de sécurité VPC que pour votre instance Amazon. EC2 Cette approche garantit que votre EC2 instance Amazon et votre instance Amazon RDS sont visibles l'une pour l'autre sur le réseau. Assurez-vous que votre instance de base de données est accessible au public. Pour configurer la réplication avec

votre base de données source comme décrit dans la section suivante, votre instance de base de données doit être accessible au public.

- Ne configurez pas plusieurs zones de disponibilité, la rétention des sauvegardes ou les réplicas en lecture tant que vous n'avez pas importé la sauvegarde de la base de données. Lorsque l'importation est terminée, vous pouvez configurer l'option multi-AZ et la rétention des sauvegardes pour l'instance de production.

3. Vérifiez les options de configuration par défaut de la base de données Amazon RDS. Si le groupe de paramètres par défaut pour la base de données ne dispose pas des options de configuration que vous voulez, trouvez un autre groupe qui les possède ou créez un groupe de paramètres. Pour plus d'informations sur la création d'un groupe de paramètres, consultezGroupes de paramètres pour (Amazon RDS).
4. Connectez-vous à la nouvelle base de données Amazon RDS en tant qu'utilisateur principal. Créez les utilisateurs nécessaires pour prendre en charge les administrateurs, les applications et les services qui doivent accéder à l'instance de base de données. Le nom d'hôte de la base de données Amazon RDS est la valeur du point de terminaison de cette instance de base de données sans le numéro de port, par exemple. `mysampledb.123456789012.us-west-2.rds.amazonaws.com` Vous pouvez trouver la valeur du point de terminaison dans les détails de la base de données dans la console Amazon RDS.
5. Connectez-vous à votre EC2 instance Amazon. Pour plus d'informations, consultez Connect to your instance for Linux dans le guide de l'utilisateur Amazon Elastic Compute Cloud.
6. Connectez-vous à votre base de données Amazon RDS en tant qu'hôte distant depuis votre EC2 instance Amazon à l'aide de la `mysql` commande. La commande suivante est un exemple :

   ```
   mysql -h host_name -P 3306 -u db_master_user -p
   ```

   *host_name*Il s'agit du point de terminaison de la base de données Amazon RDS.
7. À l'`mysql`invite, exécutez la `source` commande et transmettez-lui le nom du fichier de vidage de votre base de données. Cette commande charge les données dans l'instance de base de données Amazon RDS.

   - Pour le format SQL, utilisez la commande suivante :

     ```
     mysql> source backup.sql;
     ```

- Pour le format texte délimité, créez d'abord la base de données, s'il ne s'agit pas de la base de données par défaut que vous avez créée lors de la configuration de la base de données Amazon RDS.

```
mysql> create database database_name;
mysql> use database_name;
```

Créez ensuite les tables.

```
mysql> source table1.sql
mysql> source table2.sql
etc...
```

Enfin, importez les données.

```
mysql> LOAD DATA LOCAL INFILE 'table1.txt' INTO TABLE table1 FIELDS TERMINATED BY
 ',' ENCLOSED BY '"' LINES TERMINATED BY '0x0d0a';
mysql> LOAD DATA LOCAL INFILE 'table2.txt' INTO TABLE table2 FIELDS TERMINATED BY
 ',' ENCLOSED BY '"' LINES TERMINATED BY '0x0d0a';
etc...
```

Pour améliorer les performances, vous pouvez exécuter ces opérations en parallèle à partir de plusieurs connexions de telle sorte que l'ensemble de vos tables soit créé et chargé simultanément.

> **Note**
>
> Si vous avez utilisé des options de formatage des données `mysqldump` lorsque vous avez initialement vidé le tableau, veillez à utiliser les mêmes options `LOAD DATA LOCAL INFILE` pour garantir une interprétation correcte du contenu du fichier de données.

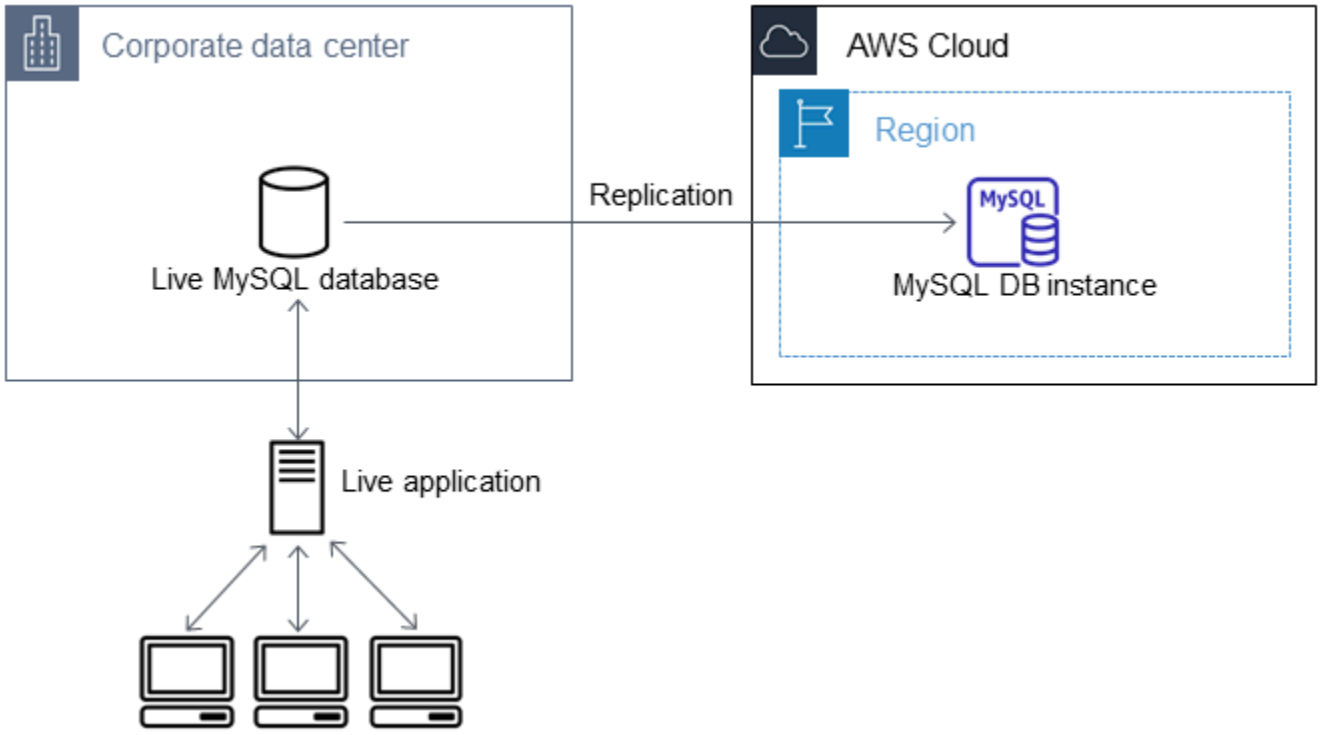
8. Exécutez une `SELECT` requête simple sur une ou deux des tables de la base de données importée pour vérifier que l'importation a réussi.

Si vous n'avez plus besoin de l' EC2 instance Amazon utilisée dans cette procédure, mettez-la hors service EC2 afin de réduire votre consommation de AWS ressources. Pour mettre fin à une EC2

instance, consultez la section Résiliation d'une instance dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Tâche 4 : répliquer les données de votre base de données externe vers votre nouvelle base de données Amazon RDS

Votre base de données source a probablement été mise à jour pendant la copie et le transfert des données vers la base de données MariaDB ou MySQL. Ainsi, vous pouvez utiliser la réplication pour intégrer la base de données up-to-date copiée à la base de données source.



Les autorisations requises pour démarrer la réplication sur une base de données Amazon RDS sont limitées et ne sont pas accessibles à votre utilisateur principal Amazon RDS. Pour cette raison, utilisez la procédure stockée Amazon RDS appropriée pour la version principale de votre moteur :

-
-
- mysql.rds_set_external_master_gtidpour configurer la réplication et mysql.rds_start_replication démarrer la réplication

Pour démarrer la réplication

Dans la tâche 1, lorsque vous définissez les options de réplication, vous avez activé la journalisation binaire et défini un ID de serveur unique pour votre base de données source. Maintenant, vous pouvez configurer votre base de données Amazon RDS comme réplica avec votre base de données active comme instance de réplication source.

1. Dans la console Amazon RDS, ajoutez l'adresse IP du serveur qui héberge la base de données source au groupe de sécurité VPC pour la base de données Amazon RDS. Pour plus d'informations sur la configuration d'un groupe de sécurité VPC, consultez la section Configurer les règles du groupe de sécurité dans le guide de l'utilisateur Amazon Virtual Private Cloud.

   Vous devrez peut-être également configurer votre réseau local pour autoriser les connexions depuis l'adresse IP de votre base de données Amazon RDS afin qu'elle puisse communiquer avec votre instance source. Pour trouver l'adresse IP de la base de données Amazon RDS, utilisez la `host` commande suivante :

   ```
   host host_name
   ```

   *`host_name`*Il s'agit du nom DNS du point de terminaison de la base de données Amazon RDS, par exemple`myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez trouver la valeur du point de terminaison dans les détails de l'instance de base de données dans la console Amazon RDS.
2. A l'aide du client de votre choix, connectez-vous à l'instance source et créez un utilisateur à utiliser pour la réplication. Ce compte est utilisé exclusivement pour la réplication et doit être limité à votre domaine pour améliorer la sécurité. La commande suivante est un exemple :

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
   ```

   > **Note**
   >
   > Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.
3. Pour l'instance source, attribuez les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` à votre utilisateur de réplication. Par exemple, pour accorder les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données à l'utilisateur « `repl_user` » de votre domaine, émettez la commande suivante :

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
```

4. Si vous avez utilisé le format SQL pour créer votre fichier de sauvegarde et que l'instance externe n'est pas MariaDB 10.0.24 ou version ultérieure, examinez le contenu de ce fichier en exécutant la commande suivante :

   ```
   cat backup.sql
   ```

   Le fichier inclut un commentaire `CHANGE MASTER TO` qui contient le nom du fichier journal maître et son emplacement. Ce commentaire est inclus dans le fichier de sauvegarde quand vous utilisez l'option `--master-data` avec `mysqldump`. Notez les valeurs pour `MASTER_LOG_FILE` et `MASTER_LOG_POS`.

   ```
   --
   -- Position to start replication or point-in-time recovery from
   --

   -- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin-changelog.000031', MASTER_LOG_POS=107;
   ```

   Si vous avez utilisé un format de texte délimité pour créer votre fichier de sauvegarde et que l'instance externe n'est pas MariaDB 10.0.24 ou une version ultérieure, vous devriez déjà disposer des coordonnées de journal binaires indiquées à l'étape 1 de la procédure Pour créer une copie de sauvegarde de votre base de données existante dans le cadre de la tâche 1.

   Si l'instance externe est MariaDB 10.0.24 ou version ultérieure, vous devriez déjà disposer du GTID à partir duquel démarrer la réplication, comme indiqué à l'étape 2 de la procédure Pour créer une copie de sauvegarde de votre base de données existante sous Tâche 1.

5. Transformez la base de données Amazon RDS en réplica. Si l'instance externe n'est pas MariaDB 10.0.24 ou version ultérieure, connectez-vous à la base de données Amazon RDS en tant qu'utilisateur principal et identifiez la base de données source en tant qu'instance de réplication source en utilisant la procédure stockée Amazon RDS appropriée :

   -
   -

   Si vous disposez d'un fichier de sauvegarde au format SQL, utilisez le nom du fichier journal principal et la position du journal principal que vous avez déterminés à l'étape 4. Si vous avez

utilisé le format texte délimité, utilisez le nom et la position que vous avez déterminés lors de la création des fichiers de sauvegarde. Les commandes suivantes sont des exemples :

MySQL 8.4 et versions supérieures

```
CALL mysql.rds_set_external_source ('myserver.mydomain.com', 3306,
    'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

MariaDB et MySQL 8.0 et versions antérieures

```
CALL mysql.rds_set_external_master ('myserver.mydomain.com', 3306,
    'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

Si l'instance externe est de version MariaDB 10.0.24 ou ultérieure, connectez-vous à la base de données Amazon RDS en tant qu'utilisateur principal et identifiez la base de données source comme instance de réplication source à l'aide de la commande mysql.rds_set_external_master_gtid. Utilisez le GTID que vous avez déterminé à l'étape 2 de la procédure Pour créer une copie de sauvegarde de votre base de données existante dans le cadre de la tâche 1. La commande suivante est un exemple :

```
CALL mysql.rds_set_external_master_gtid ('source_server_ip_address', 3306,
 'ReplicationUser', 'password', 'GTID', 1);
```

source_server_ip_address est l'adresse IP de l'instance de réplication source. Une adresse DNS EC2 privée n'est actuellement pas prise en charge.

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

6. Sur la base de données Amazon RDS, pour démarrer la réplication, exécutez la mysql.rds_start_replication commande suivante :

```
CALL mysql.rds_start_replication;
```

7. Sur la base de données Amazon RDS, pour déterminer à quel moment la réplique est à jour avec l'instance de réplication source, exécutez la commande SHOW REPLICA STATUS. Les résultats de la commande `SHOW REPLICA STATUS` incluent le champ `Seconds_Behind_Master`. Lorsque le `Seconds_Behind_Master` champ renvoie 0, la réplique est à jour avec l'instance de réplication source.

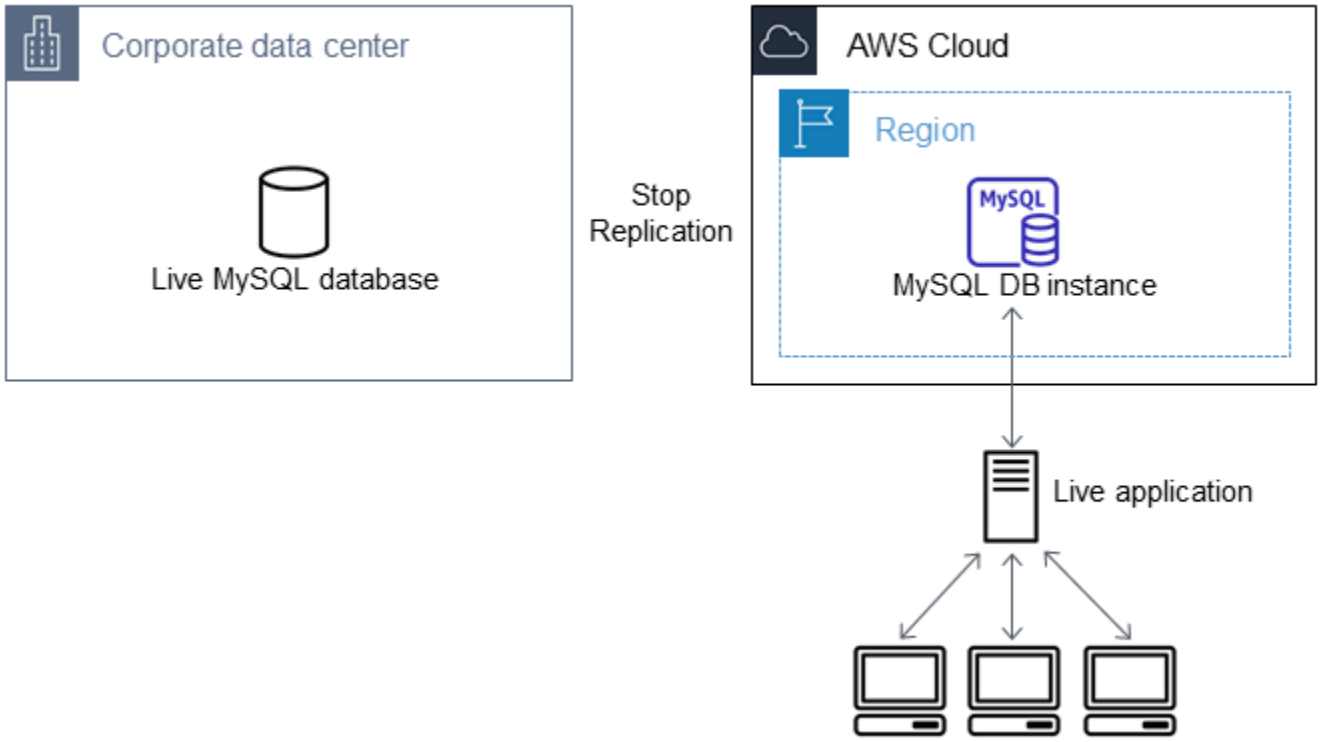
> **Note**
>
> Les versions précédentes de MySQL utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MySQL antérieure à la version 8.0.23, utilisez alors `SHOW SLAVE STATUS`.

   Pour une instance de base de données MariaDB 10.5, 10.6 ou 10.11, exécutez la procédure stockée au lieu de mysql.rds_replica_status la commande MySQL.

8. Une fois la base de données Amazon RDS à jour, activez les sauvegardes automatiques afin de pouvoir restaurer cette base de données si nécessaire. Vous pouvez activer ou modifier les sauvegardes automatisées de votre base de données Amazon RDS à l'aide de la console Amazon RDS. Pour de plus amples informations, veuillez consulter Présentation des sauvegardes.

Tâche 5 : rediriger votre application en ligne vers votre instance Amazon RDS

Une fois que la base de données MariaDB ou MySQL est à jour avec l'instance de réplication source, vous pouvez désormais mettre à jour votre application live pour utiliser l'instance Amazon RDS.



Pour rediriger votre application active vers votre base de données MariaDB ou MySQL et arrêter la réplication

1. Pour ajouter le groupe de sécurité VPC pour la base de données Amazon RDS, ajoutez l'adresse IP du serveur qui héberge l'application. Pour plus d'informations sur la modification d'un groupe de sécurité VPC, consultez Configurer les règles du groupe de sécurité dans le guide de l'utilisateur Amazon Virtual Private Cloud.
2. Vérifiez que le `Seconds_Behind_Master` champ des résultats de la commande SHOW REPLICA STATUS est égal à 0, ce qui indique que la réplique est à jour avec l'instance de réplication source.

```
SHOW REPLICA STATUS;
```

> **Note**
>
> Les versions précédentes de MySQL utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MySQL antérieure à la version 8.0.23, utilisez alors `SHOW SLAVE STATUS`.

Pour une instance de base de données MariaDB 10.5, 10.6 ou 10.11, exécutez la procédure mysql.rds_replica_status à la place de la commande MySQL.

3. Fermez toutes les connexions à la source une fois leurs transactions terminées.
4. Mettez à jour votre application pour utiliser la base de données Amazon RDS. Cette mise à jour implique généralement de modifier les paramètres de connexion pour identifier le nom d'hôte et le port de la base de données Amazon RDS, le compte utilisateur et le mot de passe avec lesquels se connecter, et la base de données à utiliser.
5. Connectez-vous à l'instance de base de données.

   Pour un cluster de bases de données multi-AZ, connectez-vous à l'instance de base de données d'écriture.
6. Arrêtez la réplication pour l'instance Amazon RDS à l'aide de la mysql.rds_stop_replication commande suivante :

   ```
   CALL mysql.rds_stop_replication;
   ```
7. Exécutez la procédure stockée Amazon RDS appropriée sur votre base de données Amazon RDS pour réinitialiser la configuration de réplication afin que cette instance ne soit plus identifiée comme une réplique.

   •

   •

   MySQL 8.4 et versions supérieures

   ```
   CALL mysql.rds_reset_external_source;
   ```

   MariaDB et MySQL 8.0 et versions antérieures

   ```
   CALL mysql.rds_reset_external_master;
   ```
8. Activez des fonctions Amazon RDS supplémentaires, telles que la prise en charge Multi-AZ et les réplicas en lecture. Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS et Utilisation des réplicas en lecture d'instance de base de données.

Importation de données depuis n'importe quelle source vers une instance de base de données Amazon RDS for MariaDB

Les étapes suivantes vous expliquent comment importer des données vers une instance de base de données Amazon RDS :

Étape 1 : Créer les fichiers plats contenant les données à charger

Utilisez un format courant, tel que CSV (valeurs séparées par des virgules), pour stocker les données à charger. Chaque table doit avoir son propre fichier. Vous ne pouvez pas combiner les données de plusieurs tables dans le même fichier. Attribuez à chaque fichier le même nom que celui de la table à laquelle il correspond. L'extension du fichier est laissée à votre libre choix. Par exemple, si le nom de la table est`sales`, le nom du fichier peut être `sales.csv` ou`sales.txt`.

Si possible, classez les données en fonction de la clé primaire de la table en cours de chargement. Cela améliore de façon spectaculaire les temps de chargement et réduit le stockage disque requis.

Cette procédure est d'autant plus rapide et efficace que les fichiers ont une petite taille. Si la taille non compressée d'un fichier est supérieure à 1 Gio, scindez-le en plusieurs fichiers et chargez chacun d'eux séparément.

Sur les systèmes Unix (Linux inclus), utilisez la commande `split`. Par exemple, la commande suivante fractionne le fichier `sales.csv` en plusieurs fichiers de moins d'1 Gio, le fractionnement n'intervenant qu'aux sauts de ligne (-C 1 024m). Les noms des nouveaux fichiers incluent des suffixes numériques croissants. La commande suivante génère des fichiers portant des noms tels que `sales.part_00` et`sales.part_01`.

```
split -C 1024m -d sales.csv sales.part_
```

Des utilitaires semblables sont disponibles sur les autres systèmes d'exploitation.

Vous pouvez stocker les fichiers plats n'importe où. Toutefois, lorsque vous chargez les données à l'étape 5, vous devez appeler le `mysql` shell depuis le même emplacement où se trouvent les fichiers, ou utiliser le chemin absolu des fichiers lors de l'exécution`LOAD DATA LOCAL INFILE`.

Étape 2 : empêcher les applications d'accéder à l'instance de base de données cible

Avant de démarrer un chargement important, empêchez toute activité de l'application d'accéder à l'instance de base de données cible sur laquelle vous prévoyez de charger. Nous le recommandons

particulièrement quand d'autres sessions sont susceptibles de modifier les tables chargées ou les tables qu'elles référencent. Cela réduit le risque de violation des contraintes intervenant pendant le chargement et améliore les performances de chargement. Dans le même temps, cela permet également de restaurer l'instance de base de données au point juste antérieur au chargement sans perdre les modifications effectuées par les processus non impliqués dans le chargement.

Il est vrai que cela peut ne pas être possible ou pratique. Si vous ne pouvez pas empêcher les applications d'accéder à l'instance de base de données avant le chargement, prenez les mesures nécessaires pour garantir la disponibilité et l'intégrité de vos données. Les étapes spécifiques requises varient grandement en fonction de cas d'utilisation spécifiques et des exigences du site.

Étape 3 : Créer un instantané de base de données

Si vous envisagez de charger des données dans une nouvelle instance de base de données qui ne contient aucune donnée, vous pouvez ignorer cette étape. Dans le cas contraire, nous vous recommandons de créer des instantanés de base de données de l'instance de base de données Amazon RDS cible avant et après le chargement des données. Les instantanés de base de données Amazon RDS sont des sauvegardes complètes de votre instance de base de données que vous pouvez utiliser pour restaurer votre instance de base de données à un état connu. Lorsque vous lancez un instantané de bases de données, les opérations I/O sur votre instance de base de données sont momentanément suspendues pendant la sauvegarde de la base de données.

La création d'un instantané de base de données juste avant le chargement vous permet, si besoin est, de restaurer la base de données à son état avant le chargement. Un instantané de base de données pris immédiatement après le chargement vous évite de devoir charger les données à nouveau en cas d'incident. Vous pouvez également utiliser des instantanés de base de données après le chargement pour importer des données dans de nouvelles instances de base de données.

L'exemple suivant exécute la AWS CLI create-db-snapshotcommande pour créer un instantané de base de données de l'`AcmeRDS`instance et attribuer l'identifiant à l'instantané de base de données`"preload"`.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-snapshot \
    --db-instance-identifier AcmeRDS \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds create-db-snapshot ^
    --db-instance-identifier AcmeRDS ^
    --db-snapshot-identifier preload
```

Vous pouvez aussi utiliser la restauration de la fonctionnalité d'instantané de bases de données pour créer des instances de bases de données de test dans le but de réaliser des essais ou pour annuler les modifications effectuées pendant le chargement.

Gardez à l'esprit que la restauration d'une base de données à partir d'un instantané de bases de données crée une nouvelle instance de base de données qui, comme toutes les instances de base de données, possède un point de terminaison et un identifiant unique. Si vous devez restaurer l'instance de base de données sans modifier le point de terminaison, vous devez d'abord supprimer l'instance de base de données de telle sorte que le point de terminaison puisse être réutilisé.

Par exemple, pour créer une instance de base de données pour les essais ou autres tests, vous attribuez à l'instance de base de données son propre identifiant. Dans cet exemple, l'identifiant est `AcmeRDS-2`. L'exemple se connecte à l'instance de base de données à l'aide du point de terminaison associé à `AcmeRDS-2`. Pour plus d'informations, consultez restore-db-instance-from-db-snapshot.

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier AcmeRDS-2 \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier AcmeRDS-2 ^
    --db-snapshot-identifier preload
```

Pour réutiliser le point de terminaison existant, il faut d'abord supprimer l'instance de base de données, puis donner le même identifiant à la base de données restaurée. Pour de plus amples informations, veuillez consulter delete-db-instance.

L'exemple suivant prend également un instantané de base de données final de l'instance de base de données avant de la supprimer. Cette action est facultative, mais recommandée.

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance \
    --db-instance-identifier AcmeRDS \
    --final-db-snapshot-identifier AcmeRDS-Final

aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier AcmeRDS \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds delete-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --final-db-snapshot-identifier AcmeRDS-Final

aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier AcmeRDS ^
    --db-snapshot-identifier preload
```

Étape 4 (facultatif) : désactiver les sauvegardes automatisées Amazon RDS

> **Warning**
>
> Ne désactivez pas les sauvegardes automatiques si vous devez effectuer une point-in-time restauration.

La désactivation des sauvegardes automatisées permet d'optimiser les performances et n'est pas nécessaire pour les chargements de données. La désactivation des sauvegardes automatiques efface toutes les sauvegardes existantes. Par conséquent, une fois les sauvegardes automatisées désactivées, la point-in-time restauration n'est pas possible. Les instantanés de bases de données manuels ne sont pas affectés par la désactivation des sauvegardes automatiques. Tous les instantanés manuels de base de données existants demeurent disponibles pour la restauration.

La désactivation des sauvegardes automatiques réduit le temps de chargement de près de 25 %, ainsi que la quantité d'espace de stockage requise pendant le chargement. Si vous envisagez de charger des données dans une nouvelle instance de base de données qui ne contient aucune donnée, la désactivation des sauvegardes constitue un moyen simple d'accélérer le chargement et d'éviter d'utiliser le stockage supplémentaire nécessaire pour les sauvegardes. Cependant, dans certains cas, vous pouvez envisager de charger dans une instance de base de données qui contient

déjà des données. Si tel est le cas, évaluez les avantages de la désactivation des sauvegardes par rapport à l'impact de la perte de performance point-in-time-recovery.

Les instances de bases de données ont les sauvegardes automatiques activées par défaut (avec une période de rétention égale à une journée). Pour désactiver les sauvegardes automatiques, définissez la période de rétention des sauvegardes à 0. Après le chargement, vous pouvez réactiver les sauvegardes en définissant la période de rétention des sauvegardes avec une valeur différente de zéro. Pour activer ou désactiver les sauvegardes, Amazon RDS arrête l'instance de base de données, puis la redémarre pour activer ou désactiver la connexion à MariaDB ou MySQL.

Exécutez la AWS CLI `modify-db-instance` commande pour définir la rétention des sauvegardes sur zéro et appliquez la modification immédiatement. Comme la définition de la période de rétention à la valeur zéro nécessite un redémarrage de l'instance de base de données, attendez que le redémarrage soit terminé avant de poursuivre. Pour de plus amples informations, veuillez consulter modify-db-instance.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier AcmeRDS \
    --apply-immediately \
    --backup-retention-period 0
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --apply-immediately ^
    --backup-retention-period 0
```

Vous pouvez vérifier l'état de votre instance de base de données à l'aide de la AWS CLI describe-db-instancescommande. L'exemple suivant affiche l'état de l'instance de base de données de l'`AcmeRDS`instance de base de données :

```
aws rds describe-db-instances --db-instance-identifier AcmeRDS --query "*[].
{DBInstanceStatus:DBInstanceStatus}"
```

Lorsque le statut de l'instance de base de données est définie`available`, vous êtes prêt à passer à l'étape suivante.

Étape 5 : Charger les données

Pour lire les lignes de vos fichiers plats dans les tables de base de données, utilisez l'`LOAD DATA LOCAL INFILE`instruction MySQL.

> **Note**
>
> Vous devez appeler le `mysql` shell depuis le même emplacement où se trouvent vos fichiers plats, ou utiliser le chemin absolu des fichiers lors de l'exécution`LOAD DATA LOCAL INFILE`.

L'exemple suivant montre comment charger les données d'un fichier nommé `sales.txt` dans une table nommée `Sales` dans la base de données :

```
mysql> LOAD DATA LOCAL INFILE 'sales.txt' INTO TABLE Sales FIELDS TERMINATED BY ' '
 ENCLOSED BY '' ESCAPED BY '\\';
Query OK, 1 row affected (0.01 sec)
Records: 1  Deleted: 0  Skipped: 0  Warnings: 0
```

Pour plus d'informations sur `LOAD DATA` cette instruction, consultez l'instruction LOAD DATA dans la documentation MySQL.

Étape 6 : réactiver les sauvegardes automatisées Amazon RDS

Si vous avez désactivé les sauvegardes automatisées Amazon RDS à l'étape 4, activez les sauvegardes automatisées une fois le chargement terminé en redéfinissant la période de conservation des sauvegardes à sa valeur de préchargement. Comme indiqué à l'étape 4, Amazon RDS redémarre l'instance de base de données. Préparez-vous donc à une brève interruption.

L'exemple suivant exécute la AWS CLI modify-db-instancecommande pour activer les sauvegardes automatiques pour l'`AcmeRDS`instance de base de données et définir la période de rétention sur un jour :

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier AcmeRDS \
    --backup-retention-period 1 \
```

```
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --backup-retention-period 1 ^
    --apply-immediately
```

Utilisation de la réplication MariaDB sur Amazon RDS

Vous utilisez généralement des répliques de lecture pour configurer la réplication entre les RDS instances de base de données Amazon. Pour obtenir des informations générales sur les réplicas en lecture, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données. Pour obtenir des informations spécifiques sur l'utilisation des répliques de lecture sur Amazon RDS pour MariaDB, consultez. Utilisation de réplicas en lecture MariaDB

Vous pouvez également configurer la réplication en fonction des coordonnées des journaux binaires pour une instance de base de données MariaDB. Pour les instances MariaDB, vous pouvez également configurer la réplication en fonction de global IDs transaction GTIDs (), ce qui améliore la sécurité en cas de crash. Pour de plus amples informations, veuillez consulter Configuration de la réplication GTID basée sur une instance source externe.

Les autres options de réplication disponibles avec RDS pour MariaDB sont les suivantes :

- Vous pouvez configurer la réplication entre une instance de base de données RDS pour MariaDB et une instance My SQL ou MariaDB externe à Amazon. RDS Pour plus d'informations sur la réplication de configuration avec une source externe, consultez Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.
- Vous pouvez configurer la réplication pour importer des bases de données depuis une instance My SQL ou MariaDB externe à RDS Amazon, ou pour exporter des bases de données vers de telles instances. Pour plus d'informations, consultez Importation de données vers une instance de base de données Amazon RDS for MariaDB avec réduction des temps d'arrêt et Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Pour chacune de ces options de réplication, vous pouvez utiliser la réplication basée sur les lignes, basée sur les instructions ou mixte. La réplication basée sur les lignes ne réplique que les lignes modifiées résultant d'une SQL instruction. La réplication basée sur des instructions reproduit l'intégralité de l'instruction. SQL La réplication mixte utilise la réplication basée sur des instructions lorsque cela est possible, mais passe à la réplication basée sur des lignes lorsque SQL des instructions présentant un danger pour la réplication basée sur des instructions sont exécutées. La réplication mixte est recommandée dans la plupart des cas. Le format de journalisation binaire de l'instance de base de données détermine si la réplication est basée sur les lignes, basée sur les instructions ou mixte. Pour plus d'informations sur la définition du format de journalisation binaire, consultez la section Configuration de la journalisation binaire MariaDB.

Pour plus d'informations sur la compatibilité de réplication entre les versions de MariaDB, consultez la section Compatibilité de réplication dans la documentation de MariaDB.

Rubriques

- Utilisation de réplicas en lecture MariaDB
- Configuration de la réplication GTID basée sur une instance source externe
- Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Utilisation de réplicas en lecture MariaDB

Vous trouverez ci-dessous des informations spécifiques sur l'utilisation des répliques de lecture sur Amazon RDS pour MariaDB. Pour obtenir des informations générales sur les réplicas en lecture et des instructions pour les utiliser, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

- Configuration des filtres de réplication avec MariaDB
- Configuration de la réplication différée avec MariaDB
- Mise à jour des réplicas en lecture avec MariaDB
- Utiliser des déploiements de réplicas en lecture Multi-AZ avec MariaDB
- Utilisation de répliques de lecture en cascade avec for MariaDB RDS
- Surveillance des réplicas en lecture MariaDB
- Démarrage et arrêt de la réplication avec des réplicas en lecture MariaDB
- Résolution d'un problème de réplica en lecture MariaDB

Configurer des réplicas en lecture avec MariaDB

Avant qu'une instance de base de données MariaDB puisse servir de source de réplication, assurez-vous d'activer les sauvegardes automatiques sur l'instance de base de données source en définissant la période de rétention des sauvegardes avec une valeur différente de 0. Cette exigence s'applique également à un réplica en lecture qui serait l'instance de base de données source d'un autre réplica en lecture.

Vous pouvez créer jusqu'à 15 réplicas en lecture à partir d'une seule instance de base de données au sein de la même région. Pour que la réplication fonctionne de façon efficace, chaque réplica en

lecture doit avoir la même quantité de ressources de calcul et de stockage que l'instance de base de données source. Si vous mettez à l'échelle l'instance de base de données source, faites-le également pour les réplicas en lecture.

RDScar MariaDB prend en charge les répliques de lecture en cascade. Pour apprendre à configurer des réplicas en lecture en cascade, consultez Utilisation de répliques de lecture en cascade avec for MariaDB RDS.

Vous pouvez exécuter simultanément plusieurs actions de création et suppression de réplicas en lecture qui référencent la même instance de base de données source. Lorsque vous effectuez ces actions, restez dans la limite de 15 réplicas en lecture pour chaque instance source.

Configuration des filtres de réplication avec MariaDB

Vous pouvez utiliser des filtres de réplication pour spécifier quelles bases de données et tables sont répliquées avec un réplica en lecture. Les filtres de réplication peuvent inclure des bases de données et des tables dans la réplication ou les exclure de la réplication.

Voici quelques cas d'utilisation pour les filtres de réplication :

- Pour réduire la taille d'un réplica en lecture. Avec le filtrage de réplication, vous pouvez exclure les bases de données et les tables qui ne sont pas nécessaires sur le réplica en lecture.
- Pour exclure des bases de données et des tables des réplicas en lecture, pour des raisons de sécurité.
- Pour répliquer différentes bases de données et tables pour des cas d'utilisation spécifiques au niveau de différents réplicas en lecture. Par exemple, vous pouvez utiliser des réplicas en lecture spécifiques pour l'analyse ou le partage.
- Pour une instance de base de données qui a lu des répliques dans différents Régions AWS, pour répliquer différentes bases de données ou tables dans différentes Régions AWS.

> **Note**
>
> Vous pouvez également utiliser des filtres de réplication pour spécifier quelles bases de données et tables sont répliquées avec une instance de base de données MariaDB principale configurée en tant que réplica dans une topologie de réplication entrante. Pour en savoir plus sur cette configuration, consultez Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

Rubriques

- Définition des paramètres de filtrage de réplication RDS pour MariaDB
- Limitations du filtrage de réplication RDS pour MariaDB
- Exemples de filtrage de réplication RDS pour MariaDB
- Affichage des filtres de réplication pour un réplica en lecture

Définition des paramètres de filtrage de réplication RDS pour MariaDB

Pour configurer des filtres de réplication, définissez les paramètres de filtrage de réplication suivants sur le réplica en lecture :

- `replicate-do-db` – Répliquer les modifications apportées aux bases de données spécifiées. Lorsque vous définissez ce paramètre pour un réplica en lecture, seules les bases de données spécifiées dans le paramètre sont répliquées.
- `replicate-ignore-db` – Ne pas répliquer les modifications apportées aux bases de données spécifiées. Lorsque le paramètre `replicate-do-db` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.
- `replicate-do-table` – Répliquer les modifications apportées aux tables spécifiées. Lorsque vous définissez ce paramètre pour un réplica en lecture, seules les tables spécifiées dans le paramètre sont répliquées. En outre, lorsque le paramètre `replicate-do-db` ou `replicate-ignore-db` est défini, la base de données qui inclut les tables spécifiées doit être incluse dans la réplication avec le réplica en lecture.
- `replicate-ignore-table` – Ne pas répliquer les modifications apportées aux tables spécifiées. Lorsque le paramètre `replicate-do-table` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.
- `replicate-wild-do-table` – Répliquer les tables en fonction des modèles de nom de base de données et nom de table spécifiés. Les caractères génériques `%` et `_` sont pris en charge. Lorsque le paramètre `replicate-do-db` ou `replicate-ignore-db` est défini, assurez-vous d'inclure la base de données qui comprend les tables spécifiées dans la réplication avec le réplica en lecture.
- `replicate-wild-ignore-table` – Ne pas répliquer les tables en fonction des modèles de nom de base de données et de nom de table spécifiés. Les caractères génériques `%` et `_` sont pris en charge. Lorsque le paramètre `replicate-do-table` ou `replicate-wild-do-table` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.

Les paramètres sont évalués dans l'ordre dans lequel ils sont répertoriés. Pour plus d'informations sur le fonctionnement de ces paramètres, consultez la documentation de MariaDB.

Par défaut, chacun de ces paramètres a une valeur vide. Sur chaque réplica en lecture, vous pouvez utiliser ces paramètres pour définir, modifier et supprimer des filtres de réplication. Lorsque vous définissez l'un de ces paramètres, séparez chaque filtre des autres par une virgule.

Vous pouvez utiliser les caractères génériques `%` et _ dans les paramètres `replicate-wild-do-table` et `replicate-wild-ignore-table`. Le caractère générique `%` correspond à un nombre quelconque de caractères, et le caractère générique _ ne correspond qu'à un seul caractère.

Le format de journalisation binaire de l'instance de base de données source est important pour la réplication, car il détermine l'enregistrement des modifications de données. Le réglage du paramètre `binlog_format` détermine si la réplication est basée sur les lignes ou les instructions. Pour de plus amples informations, veuillez consulter Configuration de la journalisation binaire MariaDB.

> **Note**
>
> Toutes les instructions du langage de définition de données (DDL) sont répliquées sous forme d'instructions, quel que soit le `binlog_format` paramètre de l'instance de base de données source.

Limitations du filtrage de réplication RDS pour MariaDB

Les limites suivantes s'appliquent au filtrage de réplication RDS pour MariaDB :

- Chaque paramètre de filtrage de réplication a une limite de 2 000 caractères.
- Les virgules ne sont pas prises en charge dans les filtres de réplication.
- Les options `binlog_do_db` et `binlog_ignore_db` de MariaDB pour le filtrage des journaux binaires ne sont pas prises en charge.
- Le filtrage de réplication ne prend pas en charge les transactions XA.

  Pour plus d'informations, consultez la section Restrictions relatives aux transactions XA dans la section Ma SQL documentation.
- Le filtrage de réplication n'est pas pris en charge RDS pour MariaDB version 10.2.

Exemples de filtrage de réplication RDS pour MariaDB

Pour configurer le filtrage de réplication pour un réplica en lecture, modifiez les paramètres de filtrage de réplication dans le groupe de paramètres associé au réplica en lecture.

> **Note**
>
> Vous ne pouvez pas modifier un groupe de paramètres par défaut. Si le réplica en lecture utilise un groupe de paramètres par défaut, créez un nouveau groupe de paramètres et associez-le au réplica en lecture. Pour plus d'informations sur les groupes de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).

Vous pouvez définir les paramètres d'un groupe de paramètres à l'aide du AWS Management Console, AWS CLI, ou RDSAPI. Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora). Lorsque vous définissez des paramètres dans un groupe de paramètres, toutes les instances de base de données associées au groupe de paramètres utilisent les réglages des paramètres. Si vous définissez les paramètres de filtrage de réplication dans un groupe de paramètres, assurez-vous que le groupe de paramètres est associé uniquement aux réplicas en lecture. Laissez les paramètres de filtrage de réplication vides pour les instances de base de données source.

Les exemples suivants définissent les paramètres à l'aide du AWS CLI. Ces exemples sont définis `ApplyMethod` de `immediate` telle sorte que les modifications de paramètres se produisent immédiatement après la fin de la CLI commande. Si vous souhaitez qu'une modification en attente soit appliquée après le redémarrage du réplica en lecture, définissez `ApplyMethod` sur `pending-reboot`.

Les exemples suivants définissent des filtres de réplication :

- Including databases in replication
- Including tables in replication
- Including tables in replication with wildcard characters
- Escaping wildcard characters in names
- Excluding databases from replication
- Excluding tables from replication

- Excluding tables from replication using wildcard characters

Example Inclusion de bases de données dans la réplication

L'exemple suivant inclut les bases de données `mydb1` et `mydb2` dans la réplication. Lorsque vous définissez `replicate-do-db` pour un réplica en lecture, seules les bases de données spécifiées dans le paramètre sont répliquées.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-do-db", "ParameterValue": "mydb1,mydb2",
 "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-do-db", "ParameterValue": "mydb1,mydb2",
 "ApplyMethod":"immediate"}]"
```

Example Inclusion de tables dans la réplication

L'exemple suivant inclut les tables `table1` et `table2` dans la base de données `mydb1` dans la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-do-table", "ParameterValue":
 "mydb1.table1,mydb1.table2", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-do-table", "ParameterValue":
 "mydb1.table1,mydb1.table2", "ApplyMethod":"immediate"}]"
```

Example Inclusion de tables dans la réplication à l'aide de caractères génériques

L'exemple suivant inclut des tables dont les noms commencent par `orders` et `returns` dans la base de données `mydb` dans la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-wild-do-table", "ParameterValue":
 "mydb.orders%,mydb.returns%", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-wild-do-table", "ParameterValue":
 "mydb.orders%,mydb.returns%", "ApplyMethod":"immediate"}]"
```

Example Échappement de caractères génériques dans les noms

L'exemple suivant montre comment utiliser le caractère d'échappement \ pour échapper à un caractère générique faisant partie d'un nom.

Supposons que vous avez plusieurs noms de tables dans la base de données `mydb1` qui commencent par `my_table`, et que vous souhaitez inclure ces tables dans la réplication. Les noms de table incluent un trait de soulignement, qui est également un caractère générique, de sorte que l'exemple échappe le trait de soulignement dans les noms de table.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-wild-do-table", "ParameterValue": "my
\_table%", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
```

```
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-wild-do-table", "ParameterValue": "my
\_table%", "ApplyMethod":"immediate"}]"
```

Example Exclusion de bases de données de la réplication

L'exemple suivant exclut les bases de données `mydb1` et `mydb2` de la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-ignore-db", "ParameterValue":
 "mydb1,mydb2", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-ignore-db", "ParameterValue":
 "mydb1,mydb2", "ApplyMethod":"immediate"}]"
```

Example Exclusion de tables de la réplication

L'exemple suivant exclut les tables `table1` et `table2` dans la base de données `mydb1` de la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-ignore-table", "ParameterValue":
 "mydb1.table1,mydb1.table2", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-ignore-table", "ParameterValue":
 "mydb1.table1,mydb1.table2", "ApplyMethod":"immediate"}]"
```

Example Exclusion de tables de la réplication à l'aide des caractères génériques

L'exemple suivant exclut de la réplication les tables dont les noms commencent par `orders` et `returns` dans la base de données `mydb`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "[{"ParameterName": "replicate-wild-ignore-table", "ParameterValue":
 "mydb.orders%,mydb.returns%", "ApplyMethod":"immediate"}]"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "[{"ParameterName": "replicate-wild-ignore-table", "ParameterValue":
 "mydb.orders%,mydb.returns%", "ApplyMethod":"immediate"}]"
```

Affichage des filtres de réplication pour un réplica en lecture

Vous pouvez afficher les filtres de réplication pour un réplica en lecture de la manière suivante :

- Vérifiez les réglages des paramètres de filtrage de réplication dans le groupe de paramètres associé au réplica en lecture.

  Pour obtenir des instructions, consultez Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon.
- Dans un client MariaDB, connectez-vous au réplica en lecture et exécutez l'instruction `SHOW REPLICA STATUS`.

  Dans la sortie, les champs suivants affichent les filtres de réplication pour le réplica en lecture :
  - `Replicate_Do_DB`
  - `Replicate_Ignore_DB`
  - `Replicate_Do_Table`
  - `Replicate_Ignore_Table`
  - `Replicate_Wild_Do_Table`
  - `Replicate_Wild_Ignore_Table`

Pour plus d'informations sur ces champs, consultez la section Vérification de l'état de réplication dans la section Ma SQL documentation.

> **Note**
>
> Les versions précédentes de MariaDB utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MariaDB antérieure à la version 10.5, utilisez alors `SHOW SLAVE STATUS`.

Configuration de la réplication différée avec MariaDB

Vous pouvez utiliser la réplication retardée comme stratégie pour la reprise après sinistre. Avec la réplication retardée, vous spécifiez la durée minimale, en secondes, pour retarder la réplication de la source vers la réplique de lecture. En cas de sinistre, par exemple la suppression accidentelle d'une table, vous appliquez la procédure suivante pour reprendre rapidement après le sinistre :

- Arrêtez la réplication vers le réplica en lecture avant que lui soit envoyée la modification qui a provoqué le sinistre.

  Pour arrêter la réplication, utilisez la procédure stockée mysql.rds_stop_replication.
- Effectuez la promotion du réplica en lecture pour qu'il devienne la nouvelle instance de base de données source, en suivant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

> **Note**
>
> - La réplication différée est prise en charge pour MariaDB 10.6 et versions ultérieures.
> - Utilisez des procédures stockées pour configurer la réplication retardée. Vous ne pouvez pas configurer la réplication différée avec AWS Management Console, le AWS CLI, ou Amazon RDSAPI.
> - Vous pouvez utiliser la réplication basée sur les identificateurs de transaction globaux (GTIDs) dans une configuration de réplication différée.

Rubriques

- Configuration de la réplication retardée pendant la création du réplica en lecture
- Modification de la réplication retardée pour un réplica en lecture existant
- Promotion d'un réplica en lecture

Configuration de la réplication retardée pendant la création du réplica en lecture

Pour configurer la réplication retardée pour tout réplica en lecture à venir créé à partir d'une instance de base de données, exécutez la procédure stockée mysql.rds_set_configuration avec le paramètre `target delay`.

Pour configurer la réplication retardée pendant la création du réplica en lecture

1. En utilisant un client MariaDB, connectez-vous à l'instance de base de données MariaDB qui sera la source des répliques en lecture en tant qu'utilisateur principal.
2. Exécutez la procédure stockée mysql.rds_set_configuration avec le paramètre `target delay`.

   Par exemple, exécutez la procédure stockée suivante pour indiquer que la réplication est retardée d'au moins une heure (3 600 secondes) pour tout réplica en lecture créé à partir de l'instance de base de données actuelle.

   ```
   call mysql.rds_set_configuration('target delay', 3600);
   ```

   > **Note**
   >
   > Après avoir exécuté cette procédure stockée, toute réplique de lecture que vous créez à l'aide du AWS CLI ou Amazon RDS API est configuré avec un délai de réplication du nombre de secondes spécifié.

Modification de la réplication retardée pour un réplica en lecture existant

Pour modifier la réplication retardée pour un réplica en lecture existant, exécutez la procédure stockée mysql.rds_set_source_delay.

Pour modifier la réplication retardée pour un réplica en lecture existant

1. En utilisant un client MariaDB, connectez-vous au réplica en lecture en tant qu'utilisateur principal.

2. Utilisez la procédure stockée mysql.rds_stop_replication pour arrêter la réplication.
3. Exécutez la procédure stockée mysql.rds_set_source_delay.

   Par exemple, exécutez la procédure stockée suivante pour indiquer que la réplication vers le réplica en lecture est retardée d'au moins une heure (3 600 secondes).

   ```
   call mysql.rds_set_source_delay(3600);
   ```
4. Utilisez la procédure stockée mysql.rds_start_replication pour lancer la réplication.

Promotion d'un réplica en lecture

Après l'arrêt de la réplication, dans un scénario de reprise après sinistre, vous pouvez promouvoir un réplica en lecture comme nouvelle instance de base de données source. Pour de plus amples informations sur la promotion d'un réplica en lecture, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.

Mise à jour des réplicas en lecture avec MariaDB

Les réplicas en lecture sont conçus pour prendre en charge les requêtes de lecture, mais vous pouvez avoir besoin de mises à jour ponctuelles. A titre d'exemple, vous pouvez avoir besoin d'ajouter un index, pour accélérer les types spécifiques de requêtes accédant au réplica. Vous pouvez autoriser les mises à jour en affectant au paramètre `read_only` la valeur 0 dans le groupe de paramètres de base de données pour le réplica en lecture.

Utiliser des déploiements de réplicas en lecture Multi-AZ avec MariaDB

Vous pouvez créer un réplica en lecture à partir de déploiements d'instance de base de données mono-AZ ou multi-AZ. Vous utilisez des déploiements multi-AZ pour améliorer la durabilité et la disponibilité des données critiques, mais vous ne pouvez pas utiliser une instance secondaire multi-AZ pour servir les requêtes en lecture seule. À la place, vous pouvez créer des réplicas en lecture à partir d'instances de base de données multi-AZ à trafic élevé pour décharger les requêtes en lecture seule. Si l'instance source d'un déploiement multi-AZ bascule vers l'instance secondaire, tous les réplicas en lecture associés se mettent automatiquement à utiliser l'instance secondaire (désormais principale) comme source de réplication. Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Vous pouvez créer un réplica en lecture en tant qu'instance de base de données Multi-AZ. Amazon RDS met en veille votre réplique dans une autre zone de disponibilité afin de prendre en charge le

basculement de la réplique. La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.

Utilisation de répliques de lecture en cascade avec for MariaDB RDS

RDScar MariaDB prend en charge les répliques de lecture en cascade. Avec les répliques de lecture en cascade, vous pouvez redimensionner les lectures sans surcharger votre source pour l'instance de base de données RDS MariaDB.

Avec les répliques de lecture en cascade, votre instance de base de données RDS MariaDB envoie des données à la première réplique de lecture de la chaîne. Ce réplica en lecture envoie ensuite les données au deuxième réplica de la chaîne, etc. Le résultat final est que toutes les répliques de lecture de la chaîne présentent les mêmes modifications que celles de l'instance de base RDS de données MariaDB, mais sans la surcharge uniquement sur l'instance de base de données source.

Vous pouvez créer une série de trois répliques de lecture au maximum dans une chaîne à partir d'une source RDS pour une instance de base de données MariaDB. Supposons, par exemple, que vous disposiez d'une instance de base de données RDS pour MariaDB,. `mariadb-main` Vous pouvez effectuer les actions suivantes :

- À partir de `mariadb-main`, créez le premier réplica en lecture de la chaîne, `read-replica-1`.
- Ensuite, à partir de `read-replica-1`, créez le réplica en lecture suivant dans la chaîne, `read-replica-2`.
- Enfin, à partir de `read-replica-2`, créez le troisième réplica en lecture de la chaîne, `read-replica-3`.

Vous ne pouvez pas créer un autre réplica en lecture au-delà de ce troisième réplica en lecture en cascade dans la série pour `mariadb-main`. Une série complète d'instances, depuis une instance de base de données source RDS pour MariaDB jusqu'à la fin d'une série de répliques de lecture en cascade, peut comprendre au maximum quatre instances de base de données.

Pour que les répliques de lecture en cascade fonctionnent, les sauvegardes automatiques doivent être activées pour chaque RDS source d'instance de base de données MariaDB. Pour activer les sauvegardes automatiques sur un réplica en lecture, commencez par créer le réplica en lecture, puis modifiez-le pour activer les sauvegardes automatiques. Pour de plus amples informations, veuillez consulter Création d'un réplica en lecture.

Comme pour tout réplica en lecture, vous pouvez promouvoir un réplica en lecture faisant partie d'une cascade. La promotion d'un réplica en lecture depuis une chaîne de réplicas en lecture retire ce réplica de la chaîne. Par exemple, supposons que vous souhaitez déplacer une partie de la charge de travail de votre instance de base de données `mariadb-main` vers une nouvelle instance destinée uniquement au service comptable. En prenant pour hypothèse la chaîne de trois réplicas en lecture de l'exemple, vous décidez de promouvoir `read-replica-2`. La chaîne est affectée comme suit :

- La promotion de `read-replica-2` le retire de la chaîne de réplication.
  - Il s'agit désormais d'une instance de base de données en lecture/écriture complète.
  - La réplication continue sur `read-replica-3`, tout comme avant la promotion.
- Votre `mariadb-main` continue la réplication sur `read-replica-1`.

Pour plus d'informations sur la promotion des réplicas en lecture, consultez Promotion d'un réplica en lecture en instance de bases de données autonome.

Surveillance des réplicas en lecture MariaDB

Pour les répliques de lecture de MariaDB, vous pouvez surveiller le délai de réplication sur Amazon en CloudWatch consultant la métrique Amazon. RDS `ReplicaLag` La métrique `ReplicaLag` contient la valeur du champ `Seconds_Behind_Master` de la commande `SHOW REPLICA STATUS`.

> **Note**
>
> Les versions précédentes de MariaDB utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MariaDB antérieure à la version 10.5, utilisez alors `SHOW SLAVE STATUS`.

Les causes courantes du retard de réplication pour MariaDB sont les suivantes :

- Une indisponibilité du réseau.
- L'écriture dans des tables avec des index sur un réplica en lecture. Si le paramètre `read_only` n'a pas pour valeur 0 sur le réplica en lecture, il peut interrompre la réplication.
- En utilisant un moteur de stockage non transactionnel tel que My. ISAM La réplication est uniquement prise en charge pour le moteur de stockage InnoDB sur MariaDB.

Lorsque la métrique `ReplicaLag` atteint 0, le réplica a rattrapé l'instance de bases de données source. Si la métrique `ReplicaLag` retourne -1, la réplication n'est actuellement pas active. `ReplicaLag`= -1 est équivalent à `Seconds_Behind_Master` = `NULL`.

Démarrage et arrêt de la réplication avec des réplicas en lecture MariaDB

Vous pouvez arrêter et redémarrer le processus de réplication sur une RDS instance de base de données Amazon en appelant les procédures stockées du système mysql.rds_stop_replication etmysql.rds_start_replication. Vous pouvez le faire lors de la réplication entre deux RDS instances Amazon pour des opérations de longue durée, telles que la création d'index volumineux. Vous devez également arrêter et démarrer la réplication lors de l'importation ou de l'exportation de bases de données. Pour plus d'informations, consultez Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt et Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Si la réplication est arrêtée pendant plus de 30 jours consécutifs, soit manuellement, soit en raison d'une erreur de réplication, Amazon RDS met fin à la réplication entre l'instance de base de données source et toutes les répliques en lecture. Cela permet d'éviter l'augmentation des besoins en stockage sur l'instance de bases de données source et d'importants délais de basculement. L'instance de base de données du réplica en lecture est toujours disponible. En revanche, la réplication ne peut pas être reprise, car les journaux binaires requis par le réplica en lecture sont supprimés de l'instance de base de données source une fois la réplication terminée. Vous pouvez créer un nouveau réplica en lecture pour l'instance de base de données source afin de rétablir la réplication.

Résolution d'un problème de réplica en lecture MariaDB

Les technologies de réplication pour MariaDB sont asynchrones. Par conséquent, des augmentations `BinLogDiskUsage` sur l'instance de base de données source et `ReplicaLag` sur le réplica en lecture sont prévisibles. Par exemple, un volume élevé d'opérations d'écriture sur l'instance de bases de données source peut se produire en parallèle. Tandis que les opérations d'écritures sur le réplica en lecture sont sérialisées à l'aide d'un seul thread d'E/S, ce qui peut conduire à un retard entre l'instance source et le réplica. Pour plus d'informations sur les réplicas en lecture seule dans la documentation MariaDB, consultez Présentation de la réplication.

Vous pouvez effectuer plusieurs opérations pour réduire le retard entre les mises à jour d'une instance de base de données source et les mises à jour suivantes appliquées au réplica en lecture, telles que les opérations suivantes :

- Dimensionnement d'un réplica en lecture pour qu'il ait une taille de stockage et une classe d'instance de base de données comparables à celles de l'instance de base de données source.
- Garantie que les réglages des paramètres dans les groupes de paramètres de base de données utilisés par l'instance de base de données source et le réplica en lecture sont compatibles. Pour obtenir plus d'informations et un exemple, reportez-vous à la présentation du paramètre `max_allowed_packet`, plus loin dans cette section.

Amazon RDS surveille l'état de réplication de vos répliques de lecture et met à jour le `Replication State` champ de l'instance de réplication de lecture en `Error` cas d'arrêt de la réplication pour une quelconque raison. Par exemple, si les DML requêtes exécutées sur votre réplique de lecture entrent en conflit avec les mises à jour effectuées sur l'instance de base de données source.

Vous pouvez passer en revue les détails de l'erreur associée et déclenchée par le moteur MariaDB, en consultant le champ `Replication Error`. Des événements indiquant l'état du réplica en lecture sont également générés, y compris RDS-EVENT-0045, RDS-EVENT-0046 et RDS-EVENT-0047. Pour plus d'informations sur les événements et l'abonnement aux événements, consultez Utilisation des notifications d'RDSévénements Amazon. Si un message d'erreur MariaDB est retourné, consultez l'erreur dans la documentation sur les messages d'erreur MariaDB.

Un problème courant susceptible de causer des erreurs de réplication se pose lorsque la valeur du paramètre `max_allowed_packet` d'un réplica en lecture est inférieure à celle du paramètre `max_allowed_packet` de l'instance de base de données source. Le `max_allowed_packet` paramètre est un paramètre personnalisé que vous pouvez définir dans un groupe de paramètres de base de données utilisé pour spécifier la taille maximale du DML code pouvant être exécuté sur la base de données. Dans certains cas, la valeur du paramètre `max_allowed_packet` dans le groupe de paramètres de base de données associé à une instance de base de données source est inférieure à la valeur du paramètre `max_allowed_packet` dans le groupe de paramètres de base de données associé au réplica en lecture de la source. Le processus de réplication peut alors générer une erreur (Packet bigger than 'max_allowed_packet' bytes) et arrêter la réplication. Vous pouvez corriger cette erreur en indiquant à la source et au réplica en lecture d'utiliser des groupes de paramètres de base de données avec les mêmes valeurs du paramètre `max_allowed_packet`.

Voici d'autres situations courantes susceptibles de causer des erreurs de réplication :

- Écriture sur les tables d'un réplica en lecture. Si vous créez des index sur un réplica en lecture, le paramètre `read_only` doit être défini sur 0 pour créer les index. Si vous écrivez dans des tables sur le réplica en lecture, cela peut interrompre la réplication.

- L'utilisation d'un moteur de stockage non transactionnel tel que les répliques ISAM My. read nécessite un moteur de stockage transactionnel. La réplication est uniquement prise en charge pour le moteur de stockage InnoDB sur MariaDB.
- Utilisation de requêtes non déterministes non sécurisées telles que `SYSDATE()`. Pour de plus amples informations, consultez Determination of safe and unsafe statements in binary logging.

Si vous décidez que vous pouvez ignorer une erreur en toute sécurité, vous pouvez suivre la procédure décrite dans Ignorer l'erreur de réplication actuelle pour RDS for My SQL. Dans le cas contraire, vous pouvez supprimer le réplica en lecture et créer une instance à l'aide du même identifiant d'instance de base de données de sorte que le point de terminaison reste le même que celui de votre ancien réplica en lecture. Si une erreur de réplication est corrigée, le champ `Replication State` prend la valeur replicating (réplication en cours).

Pour les instances de base de données MariaDB, dans certains cas, les réplicas en lecture ne peuvent pas être basculés vers l'instance secondaire si des événements de journaux binaires (binlog) ne sont pas vidés au cours de la panne. Dans ces situations, supprimez et recréez manuellement les réplicas en lecture. Vous pouvez réduire la probabilité que cela se produise en définissant les valeurs de paramètre suivantes : `sync_binlog=1` et `innodb_flush_log_at_trx_commit=1`. Ces paramètres peuvent réduire les performances. Testez donc leur impact avant d'implémenter les modifications dans un environnement de production.

Configuration de la réplication GTID basée sur une instance source externe

Vous pouvez configurer la réplication en fonction des identifiants de transaction globaux (GTIDs) depuis une instance externe de MariaDB de version 10.0.24 ou supérieure vers une instance de base de données pour MariaDB. RDS Suivez ces directives lorsque vous configurez une instance source externe et une réplique sur Amazon RDS :

- Surveillez les événements de basculement de l'instance de base de données RDS pour MariaDB qui constitue votre réplica. En cas de basculement, l'instance de base de données qui est votre réplica peut alors être recréée sur un nouvel hôte avec une autre adresse réseau. Pour plus d'informations sur la surveillance des événements de basculement, consultez Utilisation des notifications d'RDSévénements Amazon.
- Tenez à jour les journaux binaires sur votre instance source, jusqu'à ce que vous ayez vérifié qu'ils ont été appliqués au réplica. Cela garantit que vous pouvez restaurer votre instance source en cas de défaillance.

- Activez les sauvegardes automatisées sur votre instance de base de données MariaDB sur Amazon. RDS L'activation des sauvegardes automatiques garantit que vous pouvez restaurer votre réplica sur un instant donné si vous devez resynchroniser votre instance source et votre réplica. Pour plus d'informations sur les sauvegardes et les Point-In-Time restaurations, consultezSauvegarde, restauration et exportation de données.

> **Note**
>
> Les autorisations requises pour démarrer la réplication sur une instance de base de données MariaDB sont limitées et ne sont pas disponibles pour votre utilisateur principal AmazonRDS. Pour cette raison, vous devez utiliser les mysql.rds_start_replication commandes Amazon RDS mysql.rds_set_external_master_gtid et pour configurer la réplication entre votre base de données active et votre base de données RDS pour MariaDB.

Pour démarrer la réplication entre une instance source externe et une instance de base de données MariaDB sur RDS Amazon, suivez la procédure suivante.

Pour démarrer la réplication

1. Passez l'instance de base de données source MariaDB en lecture seule :

```
mysql> FLUSH TABLES WITH READ LOCK;
mysql> SET GLOBAL read_only = ON;
```

2. Obtenez le courant GTID de l'instance externe de MariaDB. Vous pouvez faire cela en utilisant `mysql` ou l'éditeur de requête de votre choix pour exécuter `SELECT @@gtid_current_pos;`.

   Le GTID est formaté comme`<domain-id>-<server-id>-<sequence-id>`. Un classique GTID ressemble à quelque chose comme ça**`0-1234510749-1728`**. Pour plus d'informations sur ces éléments GTIDs et leurs composants, consultez la section Global transaction ID dans la documentation de MariaDB.

3. Copiez la base de données de l'instance externe MariaDB vers l'instance de base de données MariaDB à l'aide de `mysqldump`. Pour les bases de données très volumineuses, il se peut que vous vouliez utiliser la procédure décrite dans Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt.

   Dans Linux, macOS, ou Unix:

```
mysqldump \
    --databases database_name \
    --single-transaction \
    --compress \
    --order-by-primary \
    -u local_user \
    -plocal_password | mysql \
        --host=hostname \
        --port=3306 \
        -u RDS_user_name \
        -pRDS_password
```

Dans Windows:

```
mysqldump ^
    --databases database_name ^
    --single-transaction ^
    --compress ^
    --order-by-primary \
    -u local_user \
    -plocal_password | mysql ^
        --host=hostname ^
        --port=3306 ^
        -u RDS_user_name ^
        -pRDS_password
```

> **Note**
>
> Veillez bien à ce qu'il n'y ait pas d'espace entre l'option `-p` et le mot de passe saisi. Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Utilisez les options `--host`, `--user (-u)`, `--port` et `-p` de la commande `mysql` pour spécifier le nom d'hôte, le nom d'utilisateur, le port et le mot de passe pour vous connecter à votre instance de base de données MariaDB. Le nom d'hôte est le DNS nom du point de terminaison de l'instance de base de données MariaDB, par exemple. `myinstance.123456789012.us-east-1.rds.amazonaws.com` Vous pouvez trouver la

valeur du point de terminaison dans les détails de l'instance dans l'Amazon RDS Management Console.

4. Transformez l'instance source MariaDB en instance accessible de nouveau en écriture.

```
mysql> SET GLOBAL read_only = OFF;
mysql> UNLOCK TABLES;
```

5. Dans la console RDS de gestion Amazon, ajoutez l'adresse IP du serveur qui héberge la base de données externe MariaDB au groupe de sécurité de VPC l'instance de base de données MariaDB. Pour plus d'informations sur la modification d'un groupe VPC de sécurité, consultez la section Groupes de sécurité correspondants VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

   L'adresse IP peut changer lorsque les conditions suivantes sont réunies :

   - Vous utilisez une adresse IP publique pour la communication entre l'instance source externe et l'instance de base de données.
   - L'instance source externe a été arrêtée et redémarrée.

   Si ces conditions sont réunies, vérifiez l'adresse IP avant de l'ajouter.

   Vous devrez peut-être aussi configurer votre réseau local de sorte à autoriser les connexions à partir de l'adresse IP de votre instance de base de données MariaDB, pour qu'elle puisse communiquer avec votre instance MariaDB externe. Pour obtenir l'adresse IP de l'instance de base de données MariaDB, utilisez la commande `host`.

```
host db_instance_endpoint
```

   Le nom d'hôte est le DNS nom du point de terminaison de l'instance de base de données MariaDB.

6. A l'aide du client de votre choix, connectez-vous à l'instance MariaDB externe et créez un utilisateur MariaDB à utiliser pour la réplication. Ce compte est utilisé exclusivement pour la réplication et doit être limité à votre domaine pour améliorer la sécurité. Voici un exemple.

```
CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

7. Pour l'instance MariaDB externe, attribuez les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` à votre utilisateur de réplication. Par exemple, pour accorder les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données à l'utilisateur « `repl_user` » de votre domaine, émettez la commande suivante.

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
```

8. Transformez l'instance de base de données MariaDB en réplica. Connectez-vous à l'instance de base de données MariaDB en tant qu'utilisateur principal et, à l'aide de la commande mysql.rds_set_external_master_gtid, identifiez la base de données MariaDB externe en tant qu'instance source de réplication. Utilisez celui GTID que vous avez déterminé à l'étape 2. Voici un exemple.

```
CALL mysql.rds_set_external_master_gtid ('mymasterserver.mydomain.com', 3306,
 'repl_user', 'password', 'GTID', 1);
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

9. Sur l'instance de base de données MariaDB, utilisez la commande mysql.rds_start_replication pour démarrer la réplication.

```
CALL mysql.rds_start_replication;
```

Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Vous pouvez configurer la réplication entre une instance de base de données RDS for My SQL ou MariaDB et une instance My SQL ou MariaDB externe à Amazon à RDS l'aide de la réplication de fichiers journaux binaires.

Rubriques

- Avant de commencer
- Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Avant de commencer

Vous pouvez configurer la réplication en utilisant la position du fichier journal binaire des transactions répliquées.

Les autorisations requises pour démarrer la réplication sur une RDS instance de base de données Amazon sont limitées et ne sont pas accessibles à votre utilisateur RDS principal Amazon. Pour cette raison, assurez-vous d'utiliser les mysql.rds_start_replication commandes Amazon RDS ou et pour configurer la réplication entre votre base de données active et votre base de RDS données Amazon.

Pour définir le format de journalisation binaire pour une base de données My SQL ou MariaDB, mettez à jour le paramètre. `binlog_format` Si votre instance de base de données utilise le groupe de paramètres d'instance de base de données par défaut, créez un nouveau groupe de paramètres de base de données pour modifier le `binlog_format` paramètre. Dans MariaDB et SQL My 8.0 et versions antérieures`binlog_format`, la valeur par défaut est. `MIXED` Cependant, vous pouvez également définir `binlog_format` sur `ROW` ou `STATEMENT` si vous avez besoin d'un format de journaux binaires (binlog) spécifique. Redémarrez votre instance de base de données pour que les modifications prennent effet. Dans My SQL 8.4 et versions supérieures, la `binlog_format` valeur par défaut est. `ROW`

Pour plus d'informations sur la configuration du paramètre `binlog_format`, consultez la section Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ. Pour plus d'informations sur les implications des différents types de SQL réplication My, consultez la section Avantages et inconvénients de la réplication basée sur des instructions et basée sur des lignes dans la documentation My. SQL

Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Suivez ces directives lorsque vous configurez une instance source externe et une réplique sur Amazon RDS :

- Surveillez les événements de basculement pour l'RDSinstance de base de données Amazon qui est votre réplique. En cas de basculement, l'instance de base de données qui est votre réplica peut alors être recréée sur un nouvel hôte avec une autre adresse réseau. Pour plus d'informations sur la surveillance des événements de basculement, consultez Utilisation des notifications d'RDSévénements Amazon.
- Tenez à jour les journaux binaires sur votre instance source jusqu'à ce que vous ayez vérifié qu'ils ont été appliqués au réplica. Cette maintenance garantit que vous pouvez restaurer votre instance source en cas de défaillance.
- Activez les sauvegardes automatisées sur votre RDS instance de base de données Amazon. L'activation des sauvegardes automatiques garantit que vous pouvez restaurer votre réplica sur un instant donné si vous devez resynchroniser votre instance source et votre réplica. Pour plus d'informations sur les sauvegardes et les point-in-time restaurations, consultezSauvegarde, restauration et exportation de données.

Pour configurer une réplication de position de fichier journal binaire avec une instance source externe

1. Rendez l'instance source My SQL ou MariaDB en lecture seule.

   ```
   mysql> FLUSH TABLES WITH READ LOCK;
   mysql> SET GLOBAL read_only = ON;
   ```

2. Exécutez la `SHOW MASTER STATUS` commande sur l'instance source My SQL ou MariaDB pour déterminer l'emplacement du journal binaire.

   Vous obtenez une sortie similaire à ce qui suit.

   ```
   File                        Position
   ------------------------------------
    mysql-bin-changelog.000031      107
   ------------------------------------
   ```

3. Copiez la base de données de l'instance externe vers l'instance de base de RDS données Amazon à l'aide de`mysqldump`. Pour les bases de données très volumineuses, il se peut

que vous vouliez utiliser la procédure décrite dans Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt.

Dans Linux, macOS, ou Unix:

```
mysqldump --databases database_name \
    --single-transaction \
    --compress \
    --order-by-primary \
    -u local_user \
    -plocal_password | mysql \
        --host=hostname \
        --port=3306 \
        -u RDS_user_name \
        -pRDS_password
```

Dans Windows:

```
mysqldump --databases database_name ^
    --single-transaction ^
    --compress ^
    --order-by-primary ^
    -u local_user ^
    -plocal_password | mysql ^
        --host=hostname ^
        --port=3306 ^
        -u RDS_user_name ^
        -pRDS_password
```

> **Note**
>
> Veillez bien à ce qu'il n'y ait pas d'espace entre l'option `-p` et le mot de passe saisi.

Pour spécifier le nom d'hôte, le nom d'utilisateur, le port et le mot de passe pour vous connecter à votre RDS instance de base de données Amazon `--host--user (-u)`, utilisez les `-p` options, `--port` et dans la `mysql` commande. Le nom d'hôte est le nom du service de nom de domaine (DNS) du point de terminaison de l'RDSinstance de base de données Amazon, par exemple`myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez

trouver la valeur du point de terminaison dans la AWS Management Console au niveau des détails de l'instance.

4. Rendez l'instance source My SQL ou MariaDB inscriptible à nouveau.

```
mysql> SET GLOBAL read_only = OFF;
mysql> UNLOCK TABLES;
```

Pour plus d'informations sur la création de sauvegardes destinées à la réplication, consultez la section Ma SQL documentation.

5. Dans le AWS Management Console, ajoutez l'adresse IP du serveur qui héberge la base de données externe au groupe de sécurité du cloud privé virtuel (VPC) pour l'RDSinstance de base de données Amazon. Pour plus d'informations sur la modification d'un groupe VPC de sécurité, consultez la section Groupes de sécurité correspondants VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

   L'adresse IP peut changer lorsque les conditions suivantes sont réunies :

   - Vous utilisez une adresse IP publique pour la communication entre l'instance source externe et l'instance de base de données.
   - L'instance source externe a été arrêtée et redémarrée.

   Si ces conditions sont réunies, vérifiez l'adresse IP avant de l'ajouter.

   Vous devrez peut-être également configurer votre réseau local pour autoriser les connexions à partir de l'adresse IP de votre RDS instance de base de données Amazon. Vous procédez ainsi pour que votre réseau local puisse communiquer avec votre instance externe My SQL ou MariaDB. Pour trouver l'adresse IP de l'RDSinstance de base de données Amazon, utilisez la `host` commande.

```
host db_instance_endpoint
```

   Le nom d'hôte est le DNS nom du point de terminaison de l'RDSinstance de base de données Amazon.

6. En utilisant le client de votre choix, connectez-vous à l'instance externe et créez un utilisateur à utiliser pour la réplication. Utilisez ce compte exclusivement pour la réplication et limitez-le à votre domaine pour améliorer la sécurité. Voici un exemple.

```
CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

7. Pour l'instance externe, attribuez les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` à votre utilisateur de réplication. Par exemple, pour accorder les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données à l'utilisateur « `repl_user` » de votre domaine, émettez la commande suivante.

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
```

8. Faites de l'RDSinstance de base de données Amazon la réplique. Pour ce faire, connectez-vous d'abord à l'RDSinstance de base de données Amazon en tant qu'utilisateur principal. Identifiez ensuite la base de données externe My SQL ou MariaDB comme instance source à l'aide de la commande or. Utilisez le nom et la position du fichier journal maître que vous avez déterminés à l'étape 2. Les commandes suivantes sont des exemples.

   Mon SQL 8,4

```
CALL mysql.rds_set_external_source ('mysourceserver.mydomain.com', 3306,
 'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

   MariaDB et My 8.0 et 5.7 SQL

```
CALL mysql.rds_set_external_master ('mymasterserver.mydomain.com', 3306,
 'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

> **Note**
>
> Sur RDS MySQL, vous pouvez choisir d'utiliser la réplication différée en exécutant la procédure mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL ou la procédure mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL stockée à la place. On RDS for MySQL, l'une des raisons d'utiliser la réplication différée est d'activer

la reprise après sinistre avec la procédure stockée. Actuellement, RDS MariaDB prend en charge la réplication différée mais ne prend pas en charge la procédure. `mysql.rds_start_replication_until`

9. Sur l'RDSinstance de base de données Amazon, émettez la mysql.rds_start_replication commande pour démarrer la réplication.

```
CALL mysql.rds_start_replication;
```

Options pour le moteur de base de données MariaDB

La section suivante décrit les options ou fonctions supplémentaires, disponibles pour les instances Amazon RDS exécutant le moteur de base de données MariaDB. Pour activer ces options, vous les ajoutez à un groupe d'options personnalisé, puis associer ce dernier à votre instance de base de données. Pour plus d'informations sur l'utilisation de groupes d'options, consultez Utilisation de groupes d'options.

Amazon RDS prend en charge les options suivantes pour MariaDB :

ID d'option	Versions du moteur
`MARIADB_AUDIT_PLUGIN`	MariaDB versions 10.3 et ultérieures

Prise en charge du plugin d'audit MariaDB

Amazon RDS prend en charge l'utilisation du plugin d'audit MariaDB sur les instances de base de données MariaDB. Le plugin d'audit MariaDB enregistre l'activité de la base de données, telle que les utilisateurs qui se connectent à la base de données, les requêtes exécutées sur la base de données et plus encore. L'enregistrement de l'activité de la base de données est stocké dans un fichier journal.

Paramètres de l'option du plugin d'audit

Amazon RDS prend en charge les paramètres suivants pour l'option de plugin d'audit MariaDB.

> **Note**
>
> Si vous ne configurez aucun paramètre d'option dans la console RDS, RDS utilise le paramètre par défaut.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
`SERVER_AUDIT_FILE_PATH`	`/rdsdbdata/log/audit/`	`/rdsdbdata/log/audit/`	Emplacement du fichier journal. Le fichier journal contient l'enregistrement de l'activité spécifiée dans `SERVER_AUDIT_EVENTS` .

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
			Pour plus d'informations, veuillez consulter Liste et affichage des fichiers journaux de base de données et Fichiers journaux de base de données MariaDB.
SERVER_AUDIT_FILE_ROTATE_SIZE	1–1000000000	1000000	Taille en octets qui, lorsqu'elle est atteinte, entraîne la rotation du fichier. Pour plus d'informations, consultez Rotation et conservation des journaux pour MariaDB.
SERVER_AUDIT_FILE_ROTATIONS	0–100	9	Nombre de rotations de journaux à enregistrer quand `server_audit_output_type=file` . S'il est défini sur 0, le fichier journal ne pivote jamais. Pour plus d'informations, veuillez consulter Rotation et conservation des journaux pour MariaDB et Téléchargement d'un fichier journal de base de données.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_EVENTS	CONNECT, QUERY, TABLE, QUERY_DDL , QUERY_DML , QUERY_DML_NO_SELECT , QUERY_DCL	CONNECT, QUERY	Types d'activités à enregistrer dans le journal. L'installation du plugin d'audit MariaDB est elle-même enregistrée. • CONNECT : Permet d'enregistrer les connexions à la base de données, réussies ou non, et les déconnexions de la base de données. • QUERY : Permet d'enregistrer le texte de toutes les requêtes exécutées sur la base de données. • TABLE : Permet d'enregistrer les tables affectées par des requêtes lorsque des requêtes sont exécutées sur la base de données. • QUERY_DDL : semblable à l'événement QUERY, mais renvoie uniquement les requêtes en langage de définition de données (DDL) (CREATE, ALTER, etc.). • QUERY_DML : semblable à l'événement QUERY, mais renvoie uniquement les requêtes en langage de manipulation de données (DML) (INSERT, UPDATE, SELECT, etc.). • QUERY_DML_NO_SELECT : Similaire à l'événement QUERY_DML , mais ne journalise pas les requêtes SELECT. • QUERY_DCL : semblable à l'événement QUERY, mais renvoie uniquement les requêtes en langage de contrôle de données (DCL) (GRANT, REVOKE, etc.).

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_INCL_USERS	Plusieurs valeurs séparées par des virgules	Aucune	Incluez uniquement l'activité des utilisateurs spécifiés. Par défaut, l'activité est enregistrée pour tous les utilisateurs. SERVER_AUDIT_INCL_USERS et SERVER_AUDIT_EXCL_USERS sont mutuellement exclusifs. Si vous ajoutez des valeurs à SERVER_AUDIT_INCL_USERS, assurez-vous qu'aucune valeur n'est ajoutée à SERVER_AUDIT_EXCL_USERS.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_EXCL_USERS	Plusieurs valeurs séparées par des virgules	Aucune	Excluez l'activité des utilisateurs spécifiés. Par défaut, l'activité est enregistrée pour tous les utilisateurs. SERVER_AUDIT_INCL_USERS et SERVER_AUDIT_EXCL_USERS sont mutuellement exclusifs. Si vous ajoutez des valeurs à SERVER_AUDIT_EXCL_USERS , assurez-vous qu'aucune valeur n'est ajoutée à SERVER_AUDIT_INCL_USERS . L'utilisateur rdsadmin interroge la base de données par seconde pour vérifier l'intégrité de la base de données. En fonction de vos autres paramètres, cette activité peut éventuellement provoquer un accroissement considérable et rapide de la taille de votre fichier journal. Si vous n'avez pas besoin d'enregistrer cette activité, ajoutez l'utilisateur rdsadmin à la liste SERVER_AUDIT_EXCL_USERS . **Note** CONNECTL'activité est toujours enregistrée pour tous les utilisateurs, même si l'utilisateur est spécifié pour ce paramètre d'option.
SERVER_AUDIT_LOGGING	ON	ON	La journalisation est active. La seule valeur valide est ON. Amazon RDS ne prend pas en charge la désactivation de la journalisation. Si vous souhaitez désactiver la journalisation, supprimez le plugin d'audit MariaDB. Pour plus d'informations, consultez Suppression du plugin d'audit MariaDB.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_QUERY_LOG_LIMIT	0–2147483647	1 024	Limite de longueur de la chaîne de requête dans un enregistrement.

Ajout du plugin d'audit MariaDB

Le processus général pour ajouter le plugin d'audit MariaDB à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Une fois que vous ajoutez le plugin d'audit MariaDB, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, l'audit commence immédiatement.

Pour ajouter le plugin d'audit MariaDB

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options de base de données personnalisé. Choisissez mariadb pour Engine (Moteur), puis la version 10.3 ou ultérieure pour Major engine version (Version majeure du moteur). Pour plus d'informations, consultez Création d'un groupe d'options.
2. Ajoutez l'option MARIADB_AUDIT_PLUGIN pour le groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options. Pour plus d'informations sur chaque paramètre, consultez Paramètres de l'option du plugin d'audit.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance de base de données et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Affichage et téléchargement du journal du plugin d'audit MariaDB

Une fois que vous activez le plugin d'audit MariaDB, vous accéder aux résultats dans les fichiers journaux de la même manière que tous les autres fichiers journaux texte. Les fichiers journaux d'audit se trouvent dans `/rdsdbdata/log/audit/`. Pour plus d'informations sur l'affichage du fichier journal dans la console, consultez Liste et affichage des fichiers journaux de base de données. Pour plus d'informations sur le téléchargement du fichier journal, consultez Téléchargement d'un fichier journal de base de données.

Modification des paramètres de plugin d'audit MariaDB

Une fois que vous activez le plugin d'audit MariaDB, vous pouvez modifier ses paramètres. Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option. Pour plus d'informations sur chaque paramètre, consultez Paramètres de l'option du plugin d'audit.

Suppression du plugin d'audit MariaDB

Amazon RDS ne prend pas en charge la désactivation de la journalisation du plugin d'audit MariaDB. Toutefois, vous pouvez supprimer le plugin dans une instance de base de données. Lorsque vous supprimez le plugin d'audit MariaDB, l'instance de base de données est automatiquement redémarrée pour cesser l'audit.

Pour supprimer le plugin d'audit MariaDB d'une instance de base de données, effectuez l'une des actions suivantes :

- Supprimez l'option de plugin d'audit MariaDB du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options
- Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas le plugin. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier

le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Paramètres pour MariaDB

Par défaut, une instance de base de données MariaDB utilise un groupe de paramètres DB qui est spécifique à une base de données MariaDB. Ce groupe de paramètres contient certains, mais pas la totalité, des paramètres du groupe de paramètres de base de données Amazon RDS du moteur de base de données MySQL. Il contient également un certain nombre de nouveaux paramètres propres à MariaDB. Pour de plus amples informations sur l'utilisation des groupes de paramètres et sur la définition des paramètres, veuillez consulter Groupes de paramètres pour (Amazon RDS).

Affichage des paramètres MariaDB

Les paramètres RDS for MariaDB sont définis aux valeurs par défaut du moteur de stockage que vous avez sélectionné. Pour plus d'informations sur les paramètres MariaDB, veuillez consulter la documentation MariaDB. Pour plus d'informations sur les moteurs de stockage MariaDB, veuillez consulter Moteurs de stockage pris en charge pour MariaDB sur Amazon RDS.

Vous pouvez afficher les paramètres disponibles pour une version spécifique de RDS pour MariaDB à l'aide de la console RDS ou de l'AWS CLI. Pour plus d'informations sur l'affichage des paramètres d'un groupe de paramètres MariaDB dans la console RDS, veuillez consulter Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon.

À l'aide de l'AWS CLI, vous pouvez afficher les paramètres d'une version RDS for MariaDB en exécutant la commande `describe-engine-default-parameters`. Spécifiez l'une des valeurs suivantes pour l'option `--db-parameter-group-family` :

- `mariadb10.11`
- `mariadb10.6`
- `mariadb10.5`
- `mariadb10.4`
- `mariadb10.3`

Par exemple, pour afficher les paramètres de RDS for MariaDB version 10.6, exécutez la commande suivante.

```
aws rds describe-engine-default-parameters --db-parameter-group-family mariadb10.6
```

Votre résultat ressemble à ce qui suit.

```
{
    "EngineDefaults": {
        "Parameters": [
            {
                "ParameterName": "alter_algorithm",
                "Description": "Specify the alter table algorithm.",
                "Source": "engine-default",
                "ApplyType": "dynamic",
                "DataType": "string",
                "AllowedValues": "DEFAULT,COPY,INPLACE,NOCOPY,INSTANT",
                "IsModifiable": true
            },
            {
                "ParameterName": "analyze_sample_percentage",
                "Description": "Percentage of rows from the table ANALYZE TABLE will
 sample to collect table statistics.",
                "Source": "engine-default",
                "ApplyType": "dynamic",
                "DataType": "float",
                "AllowedValues": "0-100",
                "IsModifiable": true
            },
            {
                "ParameterName": "aria_block_size",
                "Description": "Block size to be used for Aria index pages.",
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "integer",
                "AllowedValues": "1024-32768",
                "IsModifiable": false
            },
            {
                "ParameterName": "aria_checkpoint_interval",
                "Description": "Interval in seconds between automatic checkpoints.",
                "Source": "engine-default",
                "ApplyType": "dynamic",
                "DataType": "integer",
                "AllowedValues": "0-4294967295",
                "IsModifiable": true
            },
        ...
```

Pour ne lister que les paramètres modifiables de RDS for MariaDB version 10.6, exécutez la commande suivante.

Pour LinuxmacOS, ou Unix :

```
aws rds describe-engine-default-parameters --db-parameter-group-family mariadb10.6 \
   --query 'EngineDefaults.Parameters[?IsModifiable==`true`]'
```

Dans Windows :

```
aws rds describe-engine-default-parameters --db-parameter-group-family mariadb10.6 ^
   --query "EngineDefaults.Parameters[?IsModifiable==`true`]"
```

Paramètres MySQL qui ne sont pas disponibles

Les paramètres MySQL suivants ne sont pas disponibles dans les groupes de paramètres DB spécifiques à MariaDB :

- bind_address
- binlog_error_action
- binlog_gtid_simple_recovery
- binlog_max_flush_queue_time
- binlog_order_commits
- binlog_row_image
- binlog_rows_query_log_events
- binlogging_impossible_mode
- block_encryption_mode
- core_file
- default_tmp_storage_engine
- div_precision_increment
- end_markers_in_json
- enforce_gtid_consistency
- eq_range_index_dive_limit
- explicit_defaults_for_timestamp
- gtid_executed

- gtid-mode
- gtid_next
- gtid_owned
- gtid_purged
- log_bin_basename
- log_bin_index
- log_bin_use_v1_row_events
- log_slow_admin_statements
- log_slow_slave_statements
- log_throttle_queries_not_using_indexes
- master-info-repository
- optimizer_trace
- optimizer_trace_features
- optimizer_trace_limit
- optimizer_trace_max_mem_size
- optimizer_trace_offset
- relay_log_info_repository
- rpl_stop_slave_timeout
- slave_parallel_workers
- slave_pending_jobs_size_max
- slave_rows_search_algorithms
- storage_engine
- table_open_cache_instances
- timed_mutexes
- transaction_allow_batching
- validate-password
- validate_password_dictionary_file
- validate_password_length
- validate_password_mixed_case_count
- validate_password_number_count

- validate_password_policy
- validate_password_special_char_count

Pour de plus amples informations sur les paramètres MySQL, veuillez consulter la documentation MySQL.

Migration des données d'un instantané My SQL DB vers une instance de base de données MariaDB

Vous pouvez migrer un instantané RDS de My SQL DB vers une nouvelle instance de base de données exécutant MariaDB en utilisant AWS Management Console le AWS CLI, le ou Amazon. RDS API Vous devez utiliser un instantané de base de données créé à partir d'une RDS instance de base de données Amazon exécutant My SQL 5.6 ou 5.7. Pour savoir comment créer un instantané RDS pour My SQL DB, consultezCréation d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

La migration de l'instantané n'affecte pas l'instance de base de données d'origine à partir de laquelle l'instantané a été pris. Vous pouvez tester et valider la nouvelle instance de base de données avant d'y détourner le trafic en remplacement de l'instance de base de données d'origine.

Après avoir migré de My SQL vers MariaDB, l'instance de base de données MariaDB est associée au groupe de paramètres de base de données et au groupe d'options par défaut. Après la restauration de l'instantané de base de données, vous pouvez associer un groupe de paramètres de base de données personnalisé à la nouvelle instance de base de données. Toutefois, un groupe de paramètres MariaDB présente un ensemble de variables système configurables différent. Pour plus d'informations sur les différences entre les variables système My SQL et MariaDB, consultez Différences de variables système entre MariaDB et My. SQL Pour en savoir plus sur les groupes de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS). Pour en savoir plus sur les groupes d'options, consultez Utilisation de groupes d'options.

Exécution de la migration

Vous pouvez migrer un instantané RDS de My SQL DB vers une nouvelle instance de base de données MariaDB en utilisant AWS Management Console le, le ou AWS CLI le. RDS API

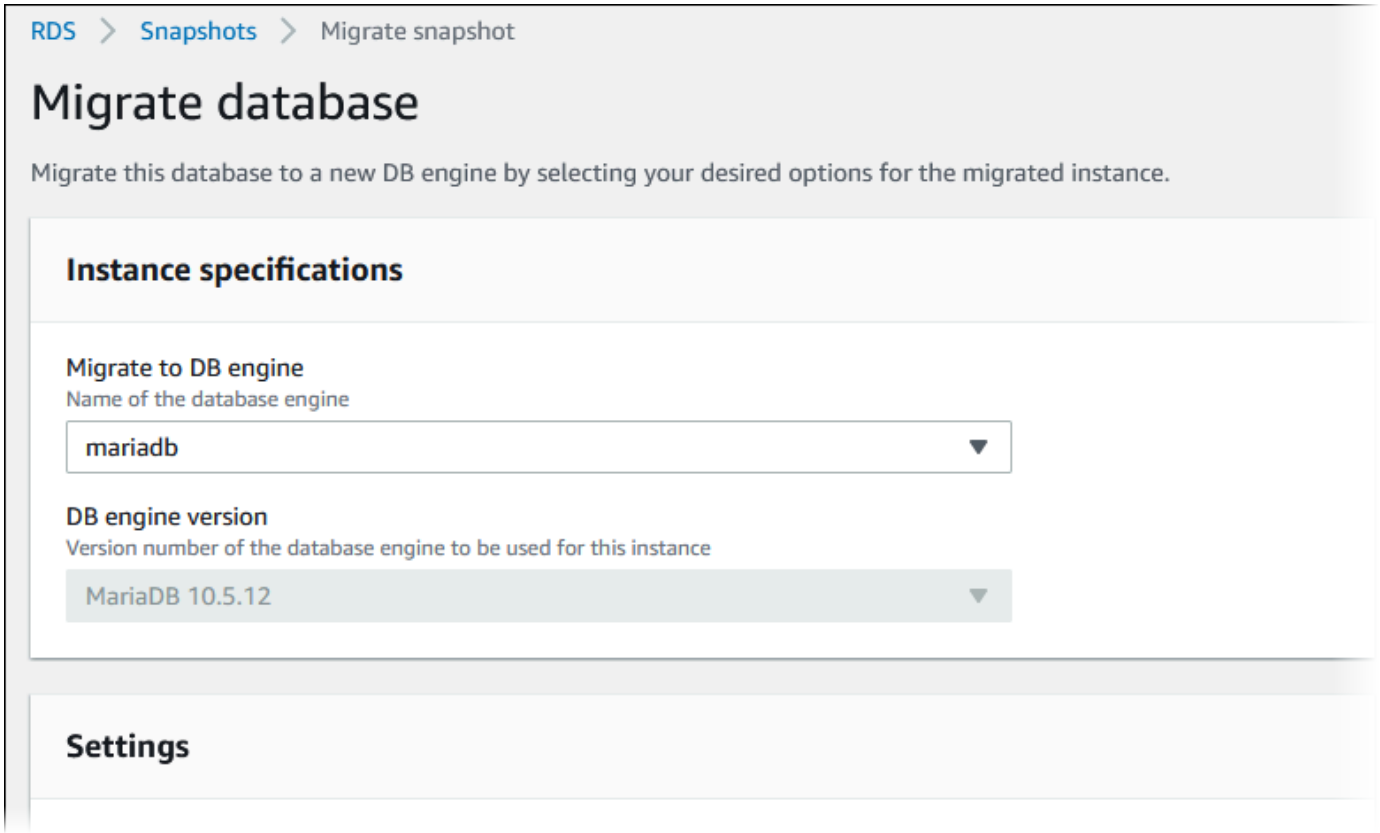
Console

Pour migrer un instantané My SQL DB vers une instance de base de données MariaDB

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Snapshots, puis sélectionnez le snapshot My SQL DB que vous souhaitez migrer.

3. Pour Actions, choisissez Migrate shapshot (Migrer l'instantané). La page Migrate database (Migrer la base de données) apparaît.
4. Pour Migrate to DB Engine (Migrer vers le moteur de base de données), choisissez mariadb.

   Amazon RDS sélectionne automatiquement la version du moteur de base de données. Vous ne pouvez pas modifier la version du moteur de base de données.



5. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.
6. Choisissez Migrate (Migrer).

AWS CLI

Pour migrer les données d'un instantané My SQL DB vers une instance de base de données MariaDB, exécutez AWS CLI `restore-db-instance-from-db-snapshot`la commande avec les options suivantes :

- -- db-instance-identifier — Nom de l'instance de base de données à créer à partir de l'instantané de base de données.

- -- db-snapshot-identifier — Identifiant du snapshot de base de données à partir duquel effectuer la restauration.
- --engine – Moteur de base de données à utiliser pour la nouvelle instance.

Example

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier newmariadbinstance \
    --db-snapshot-identifier mysqlsnapshot \
    --engine mariadb
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier newmariadbinstance ^
    --db-snapshot-identifier mysqlsnapshot ^
    --engine mariadb
```

API

Pour migrer les données d'un instantané My SQL DB vers une instance de base de données MariaDB, appelez l'opération Amazon. RDS API RestoreDBInstanceFromDBSnapshot

Incompatibilités entre MariaDB et My SQL

Les incompatibilités entre My et SQL MariaDB sont les suivantes :

- Vous ne pouvez pas migrer un instantané de base de données créé avec My SQL 8.0 vers MariaDB.
- Si la base de SQL données Ma base de données source utilise un hachage de mot de SHA256 passe, assurez-vous de réinitialiser les mots de passe utilisateur SHA256 hachés avant de vous connecter à la base de données MariaDB. Le code suivant montre comment réinitialiser un mot de passe SHA256 haché.

```
SET old_passwords = 0;
UPDATE mysql.user SET plugin = 'mysql_native_password',
```

```
Password = PASSWORD('new_password')
WHERE (User, Host) = ('master_user_name', %);
FLUSH PRIVILEGES;
```

- Si votre compte utilisateur RDS principal utilise le hachage de mot de passe SHA -256, veillez à réinitialiser le mot de passe à l' AWS Management Console aide de la `modify-db-instance` AWS CLI commande, ou de l'opération M. odifyDBInstance RDS API Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.
- MariaDB ne prend pas en charge le plugin Memcached. Toutefois, les données utilisées par le plugin Memcached sont stockées dans les tables InnoDB. Après avoir migré un instantané My SQL DB, vous pouvez accéder aux données utilisées par le plugin Memcached à l'aide de. SQL Pour de plus amples informations sur la base de données innodb_memcache, consultez la page InnoDB memcached Plugin Internals.

Référence MariaDB sur SQL Amazon RDS

La section suivant décrit les procédures stockées système disponibles pour les instances Amazon RDS exécutant le moteur de base de données MariaDB.

Vous pouvez utiliser les procédures système stockées, qui sont disponibles pour les instances de bases de données MySQL et MariaDB. Ces procédures stockées sont documentées dans RDSpour Ma référence de procédure SQL stockée. Les instances de base de données MariaDB prennent en charge toutes les procédures stockées, à l'exception de `mysql.rds_start_replication_until` et `mysql.rds_start_replication_until_gtid`.

En outre, les procédures stockées système suivantes sont prises en charge uniquement pour les instances de base de données Amazon RDS exécutant MariaDB :

- mysql.rds_replica_status
- mysql.rds_set_external_master_gtid
- mysql.rds_kill_query_id

mysql.rds_replica_status

Affiche l'état de réplication d'un réplica en lecture MariaDB.

Appelez cette procédure sur le réplica en lecture pour afficher les informations d'état sur les paramètres essentiels des threads du réplica.

Syntaxe

```
CALL mysql.rds_replica_status;
```

Notes d'utilisation

Cette procédure est uniquement prise en charge pour les instances de base de données MariaDB exécutant MariaDB version 10.5 et ultérieure.

Cette procédure est l'équivalent de la commande `SHOW REPLICA STATUS`. Cette commande n'est pas prise en charge pour les instances de base de données MariaDB version 10.5 et ultérieures.

Dans les versions antérieures de MariaDB, la commande `SHOW SLAVE STATUS` équivalente exigeait le privilège `REPLICATION SLAVE`. Dans MariaDB version 10.5, elle requiert le privilège

`REPLICATION REPLICA ADMIN`. Pour protéger la gestion RDS des instances de base de données MariaDB 10.5 et versions ultérieures, ce nouveau privilège n'est pas accordé à l'utilisateur principal RDS.

Exemples

L'exemple suivant montre l'état d'un réplica en lecture MariaDB :

```
call mysql.rds_replica_status;
```

La réponse est similaire à ce qui suit :

```
*************************** 1. row ***************************
               Replica_IO_State: Waiting for master to send event
                    Source_Host: XX.XX.XX.XXX
                    Source_User: rdsrepladmin
                    Source_Port: 3306
                  Connect_Retry: 60
                Source_Log_File: mysql-bin-changelog.003988
            Read_Source_Log_Pos: 405
                 Relay_Log_File: relaylog.011024
                  Relay_Log_Pos: 657
          Relay_Source_Log_File: mysql-bin-changelog.003988
             Replica_IO_Running: Yes
            Replica_SQL_Running: Yes
                Replicate_Do_DB:
            Replicate_Ignore_DB:
             Replicate_Do_Table:
         Replicate_Ignore_Table:
mysql.rds_sysinfo,mysql.rds_history,mysql.rds_replication_status
        Replicate_Wild_Do_Table:
    Replicate_Wild_Ignore_Table:
                     Last_Errno: 0
                     Last_Error:
                   Skip_Counter: 0
            Exec_Source_Log_Pos: 405
                Relay_Log_Space: 1016
                Until_Condition: None
                 Until_Log_File:
                  Until_Log_Pos: 0
             Source_SSL_Allowed: No
             Source_SSL_CA_File:
             Source_SSL_CA_Path:
```

```
                Source_SSL_Cert:
              Source_SSL_Cipher:
                 Source_SSL_Key:
          Seconds_Behind_Master: 0
  Source_SSL_Verify_Server_Cert: No
                  Last_IO_Errno: 0
                  Last_IO_Error:
                 Last_SQL_Errno: 0
                 Last_SQL_Error:
    Replicate_Ignore_Server_Ids:
               Source_Server_Id: 807509301
                 Source_SSL_Crl:
             Source_SSL_Crlpath:
                     Using_Gtid: Slave_Pos
                    Gtid_IO_Pos: 0-807509301-3980
        Replicate_Do_Domain_Ids:
    Replicate_Ignore_Domain_Ids:
                  Parallel_Mode: optimistic
                      SQL_Delay: 0
            SQL_Remaining_Delay: NULL
      Replica_SQL_Running_State: Reading event from the relay log
             Replica_DDL_Groups: 15
Replica_Non_Transactional_Groups: 0
    Replica_Transactional_Groups: 3658
1 row in set (0.000 sec)

Query OK, 0 rows affected (0.000 sec)
```

mysql.rds_set_external_master_gtid

Configure la réplication GTID d'une instance MariaDB s'exécutant à l'extérieur de Amazon RDS à une instance de base de données MariaDB. Cette procédure stockée est prise en charge uniquement lorsque l'instance MariaDB externe est à la version 10.0.24 ou ultérieure. Lors de la configuration d'une réplication où une ou les deux instances ne prennent pas en charge les identificateurs de transaction globaux (GTID) MariaDB, utilisez .

L'utilisation de GTID pour la réplication fournit des fonctions de sécurité en cas d'incident non proposées par une réplication de journal binaire. Nous la recommandons donc dans les situations où les instances de réplication assurent une prise en charge.

Syntaxe

```
CALL mysql.rds_set_external_master_gtid(
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , gtid
  , ssl_encryption
);
```

Paramètres

host_name

String. Nom d'hôte ou adresse IP de l'instance MariaDB s'exécutant à l'extérieur de Amazon RDS et qui deviendra l'instance source.

host_port

Entier : Port utilisé par l'instance MariaDB s'exécutant à l'extérieur de Amazon RDS pour une configuration comme instance source. Si votre configuration réseau inclut une réplication de port SSH qui convertit le numéro de port, spécifiez le numéro de port qui est exposé par SSH.

replication_user_name

String. L'ID d'un utilisateur avec les autorisations `REPLICATION SLAVE` de l'instance de base de données MariaDB à configurer comme réplica en lecture.

replication_user_password

String. Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

gtid

String. L'ID de transaction global sur l'instance source à partir de laquelle la réplication doit démarrer.

Vous pouvez utiliser `@@gtid_current_pos` pour obtenir le GTID actuel si l'instance source a été verrouillée pendant que vous configurez la réplication, afin que le journal binaire ne change pas entre les moments où vous obtenez le GTID et celui où la réplication démarre.

Sinon, si vous utilisez `mysqldump` version 10.0.13, ou ultérieure, pour remplir l'instance de réplica avant de démarrer la réplication, vous pouvez obtenir la position GTID dans le résultat en utilisant les options `--master-data` ou `--dump-slave`. Si vous n'utilisez pas `mysqldump`

version 10.0.13 ou ultérieure, vous pouvez exécuter le `SHOW MASTER STATUS` ou les mêmes options `mysqldump` pour obtenir la position et le nom du fichier journal binaire, puis les convertir dans un GTID en exécutant `BINLOG_GTID_POS` sur l'instance MariaDB :

```
SELECT BINLOG_GTID_POS('<binary log file name>', <binary log file position>);
```

Pour plus d'informations sur l'implémentation MariaDB de GTID, accédez à ID de transaction global dans la documentation MariaDB.

ssl_encryption

Valeur indiquant si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. La valeur 1 spécifie d'utiliser le chiffrement SSL, et la valeur 0 de ne pas l'utiliser. La valeur par défaut est 0.

> **Note**
>
> L'option `MASTER_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

Notes d'utilisation

La procédure `mysql.rds_set_external_master_gtid` doit être exécutée par l'utilisateur maître. Elle doit être exécutée sur l'instance de base de données MariaDB que vous configurez comme le réplica d'une instance MariaDB s'exécutant à l'extérieur de Amazon RDS. Avant d'exécuter `mysql.rds_set_external_master_gtid`, vous devez avoir configuré l'instance de MariaDB s'exécutant en dehors de Amazon RDS comme instance source. Pour plus d'informations, consultez Importation de données dans une instance de base de données Amazon RDS for MariaDB.

> **Warning**
>
> N'utilisez pas `mysql.rds_set_external_master_gtid` pour gérer la réplication entre deux instances de base de données Amazon RDS. N'utilisez la procédure que lors de la réplication avec une instance MariaDB s'exécutant à l'extérieur de RDS. Pour plus d'informations sur la gestion de la réplication entre les instances de base de données Amazon RDS, consultez Utilisation des réplicas en lecture d'instance de base de données.

Après avoir appelé `mysql.rds_set_external_master_gtid` pour configurer une instance de base de données Amazon RDS comme réplica en lecture, vous pouvez appeler mysql.rds_start_replication sur le réplica pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Quand la procédure `mysql.rds_set_external_master_gtid` est appelée, Amazon RDS enregistre l'heure, l'utilisateur et une action de « set master » dans les tables `mysql.rds_history` et `mysql.rds_replication_status`.

Exemples

Lorsqu'il est exécuté sur une instance de base de données MariaDB, l'exemple suivant la configure comme le réplica d'une instance de MariaDB s'exécutant à l'extérieur de Amazon RDS.

```
call mysql.rds_set_external_master_gtid
 ('Sourcedb.some.com',3306,'ReplicationUser','SomePassW0rd','0-123-456',0);
```

mysql.rds_kill_query_id

Met fin à une requête exécutée sur le serveur MariaDB afin de mettre fin aux requêtes de longue durée ou problématiques. Vous pouvez identifier l'ID de requête et arrêter efficacement une requête spécifique pour résoudre les problèmes de performances et maintenir un fonctionnement optimal de la base de données.

Syntaxe

```
CALL mysql.rds_kill_query_id(queryID);
```

Paramètres

queryID

Entier : L'identité de la requête à terminer.

Notes d'utilisation

Pour arrêter une requête s'exécutant sur le serveur MariaDB, utilisez la procédure `mysql.rds_kill_query_id` et transmettez-lui l'ID de cette requête. Pour obtenir l'ID de requête, interrogez la table du PROCESSLISTschéma d'informations MariaDB, comme indiqué ci-dessous :

```
SELECT USER, HOST, COMMAND, TIME, STATE, INFO, QUERY_ID FROM
                INFORMATION_SCHEMA.PROCESSLIST WHERE USER = '<user name>';
```

La connexion au serveur MariaDB est conservée.

Exemples

L'exemple suivant termine une requête avec un ID de requête 230040 :

```
call mysql.rds_kill_query_id(230040);
```

Fuseau horaire local pour les instances de base de données MariaDB

Par défaut, le fuseau horaire d'une instance de base de données MariaDB est Universal Time Coordinated (). UTC Vous pouvez à la place définir le fuseau horaire de votre instance de base de données sur le fuseau horaire local de votre application.

Pour définir le fuseau horaire local d'une instance de base de données, définissez le paramètre `time_zone` du groupe de paramètres de votre instance de base de données avec l'une des valeurs prises en charge et répertoriées plus bas dans cette section. Lorsque vous définissez le paramètre `time_zone` d'un groupe de paramètres, toutes les instances de base de données et tous les réplicas en lecture qui ont recours à ce groupe de paramètres sont modifiés de façon à utiliser le nouveau fuseau horaire local. Pour plus d'informations sur la définition des paramètres d'un groupe de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Une fois que vous avez défini le fuseau horaire local, toutes les nouvelles connexions à la base de données reflètent la modification. Si des connexions à votre base de données sont ouvertes lorsque vous modifiez le fuseau horaire local, la mise à jour du fuseau horaire local n'apparaît pas tant que vous n'avez pas fermé la connexion et n'en avez pas ouvert une nouvelle.

Vous pouvez définir un fuseau horaire local différent pour une instance de base de données et un ou plusieurs de ses réplicas en lecture. Pour ce faire, utilisez un autre groupe de paramètres pour l'instance de base de données et les réplicas, et définissez le paramètre `time_zone` de chaque groupe de paramètres avec un autre fuseau horaire local.

Si vous effectuez une réplication Régions AWS, l'instance de base de données source et la réplique en lecture utilisent des groupes de paramètres différents (les groupes de paramètres sont uniques à un Région AWS). Pour que chaque instance utilise le même fuseau horaire local, vous devez définir le paramètre `time_zone` dans les groupes de paramètres de l'instance et du réplica en lecture.

Lorsque vous restaurez une instance de base de données à partir d'un instantané de base de données, le fuseau horaire local est défini surUTC. Vous pouvez mettre à jour le fuseau horaire sur votre fuseau horaire local une fois la restauration terminée. Si vous restaurez une instance de base de données à un instant dans le passé, le fuseau horaire local de l'instance de base de données restaurée est le paramètre de fuseau horaire du groupe de paramètres de l'instance de base de données restaurée.

L'Internet Assigned Numbers Authority (IANA) publie de nouveaux fuseaux horaires sur https://www.iana.org/time-zones plusieurs fois par an. Chaque fois qu'RDSune nouvelle version de

maintenance mineure de MariaDB est publiée, celle-ci est livrée avec les dernières données de fuseau horaire au moment de la publication. Lorsque vous utilisez les dernières versions RDS de MariaDB, vous disposez de données de fuseau horaire récentes provenant de. RDS Pour vous assurer que votre instance de base de données dispose de données de fuseau horaire récentes, nous vous recommandons de passer à une version supérieure du moteur de base de données. Vous pouvez également modifier manuellement les tables de fuseaux horaires dans les instances de base de données MariaDB. Pour ce faire, vous pouvez utiliser des SQL commandes ou exécuter l'outil mysql_tzinfo_to_sql dans un client. SQL Après avoir mis à jour les données de fuseau horaire manuellement, redémarrez votre instance de base de données afin que les modifications prennent effet. RDSne modifie ni ne réinitialise les données de fuseau horaire des instances de base de données en cours d'exécution. Les nouvelles données de fuseau horaire ne sont installées que lorsque vous effectuez une mise à niveau de la version du moteur de base de données.

Vous pouvez définir votre fuseau horaire local avec l'une des valeurs suivantes.

disponibilité	Fuseau horaire
Afrique	Africa/Cairo, Africa/Casablanca, Africa/Harare, Africa/Monrovia, Africa/Nairobi, Africa/Tripoli, Africa/Windhoek
Amérique	America/Araguaina, America/Asuncion, America/Bogota, America/Buenos_Aires, America/Caracas, America/Chihuahua, America/Cuiaba, America/Denver, America/Fortaleza, America/Guatemala, America/Halifax, America/Manaus, America/Matamoros, America/Monterrey, America/Montevideo, America/Phoenix, America/Santiago, America/Tijuana
Asie	Asia/Amman, Asia/Ashgabat, Asia/Baghdad, Asia/Baku, Asia/Bangkok, Asia/Beirut, Asia/Calcutta, Asia/Damascus, Asia/Dhaka, Asia/Irkutsk, Asia/Jerusalem, Asia/Kabul, Asia/Karachi, Asia/Kathmandu, Asia/Krasnoyarsk, Asia/Magadan, Asia/Muscat, Asia/Novosibirsk, Asia/Riyadh, Asia/Seoul, Asia/Shanghai, Asia/Singapore, Asia/Taipei, Asia/Tehran, Asia/Tokyo, Asia/Ulaanbaatar, Asia/Vladivostok, Asia/Yakutsk, Asia/Yerevan
Atlantique	Atlantique/Açores
Australie	Australia/Adelaide, Australia/Brisbane, Australia/Darwin, Australia/Hobart, Australia/Perth, Australia/Sydney

disponibilité	Fuseau horaire
Brésil	Brazil/DeNoronha, Brazil/East
Canada	Canada/Newfoundland, Canada/Saskatchewan, Canada/Yukon
Europe	Europe/Amsterdam, Europe/Athens, Europe/Dublin, Europe/Helsinki, Europe/Istanbul, Europe/Kaliningrad Europe/Moscow, Europe/Paris, Europe/Prague, Europe/Sarajevo
Pacifique	Pacific/Auckland, Pacific/Fiji, Pacific/Guam, Pacific/Honolulu, Pacific/Samoa
ETATS-UNIS	US/Alaska, US/Central, US/East-Indiana, US/Eastern, US/Pacific
UTC	UTC

Limites et problèmes connus pour RDS for MariaDB

Les éléments suivants sont des problèmes et limitations connus liés à l'utilisation de RDS for MariaDB.

> **Note**
>
> Cette liste n'est pas exhaustive.

Rubriques

- Limites de taille des fichiers MariaDB dans Amazon RDS
- Mot réservé InnoDB
- Ports personnalisés
- Performance Insights

Limites de taille des fichiers MariaDB dans Amazon RDS

Pour les instances de base de données MariaDB, la taille maximale d'une table est de 16 To lors de l'utilisation des espaces de table file-per-table InnoDB. Cette limite restreint également l'espace de table du système à une taille maximum de 16 To. Les espaces de table file-per-table InnoDB (avec des tables chacune dans leur propre espace de table) sont définis par défaut pour les instances de base de données MariaDB. Cette limite n'est pas liée à la limite de stockage maximale pour les instances de base de données MariaDB. Pour plus d'informations sur les limites de stockage, veuillez consulter Stockage d'instance de base de données Amazon RDS.

Selon votre application, l'utilisation des espaces de table file-per-table InnoDB présente des avantages et des inconvénients. Pour déterminer l'approche optimale pour votre application, veuillez consulter File-Per-Table Tablespaces dans la documentation MySQL.

Il est déconseillé d'autoriser les tables à dépasser la taille maximale de fichier. En général, une meilleure pratique consiste à partitionner les données en tables plus petites, ce qui peut améliorer la performance et les temps de récupération.

Vous pouvez utiliser l'option de partitionnement pour diviser une grande table en tables plus petites. Le partitionnement répartit des portions de votre grande table en fichiers distincts en fonction des règles que vous spécifiez. Par exemple, si vous stockez des transactions par date, vous pouvez créer

des règles de partitionnement qui répartissent des transactions plus anciennes en fichiers distincts en utilisant le partitionnement. Ensuite, vous pouvez archiver régulièrement les données de transaction historiques qui n'ont pas besoin d'être rapidement utilisables par votre application. Pour de plus amples informations, veuillez consulter Partitionnement dans la documentation MySQL.

Pour déterminer la taille de tous les espaces de table InnoDB

- Utilisez la commande SQL suivante pour déterminer si certaines de vos tables sont trop volumineuses et peuvent faire l'objet d'un partitionnement.

> **Note**
>
> Pour MariaDB 10.6 et versions supérieures, cette requête renvoie également la taille de l'espace de table système InnoDB.
> Pour les versions de MariaDB antérieures à 10.6, vous ne pouvez pas déterminer la taille de l'espace de table système InnoDB en interrogeant les tables système. Nous vous recommandons de procéder à une mise à niveau vers une version ultérieure.

```
SELECT SPACE,NAME,ROUND((ALLOCATED_SIZE/1024/1024/1024), 2)
as "Tablespace Size (GB)"
FROM information_schema.INNODB_SYS_TABLESPACES ORDER BY 3 DESC;
```

Pour déterminer la taille des tables utilisateur non-InnoDB

- Utilisez la commande SQL suivante pour déterminer si certaines de vos tables utilisateur non-InnoDB sont trop volumineuses.

```
SELECT TABLE_SCHEMA, TABLE_NAME, round(((DATA_LENGTH + INDEX_LENGTH+DATA_FREE)
/ 1024 / 1024/ 1024), 2) As "Approximate size (GB)" FROM information_schema.TABLES
WHERE TABLE_SCHEMA NOT IN ('mysql', 'information_schema', 'performance_schema')
and ENGINE<>'InnoDB';
```

Pour activer les espaces de table file-per-table InnoDB

- Définissez le paramètre `innodb_file_per_table` sur 1 dans le groupe de paramètres pour l'instance de base de données.

Pour désactiver les espaces de table file-per-table InnoDB

- Définissez le paramètre `innodb_file_per_table` sur `0` dans le groupe de paramètres pour l'instance de base de données.

Pour plus d'informations sur la mise à jour d'un groupe de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Après avoir activé ou désactivé les espaces de table file-per-table InnoDB, vous pouvez émettre une commande `ALTER TABLE`. Vous pouvez utiliser cette commande pour déplacer une table de l'espace de table global vers son propre espace de table. Vous pouvez également déplacer une table de son propre espace de table vers l'espace de table global. Voici un exemple.

```
ALTER TABLE table_name ENGINE=InnoDB, ALGORITHM=COPY;
```

Mot réservé InnoDB

`InnoDB` est un mot réservé pour RDS for MariaDB. Vous ne pouvez pas utiliser ce nom pour une base de données MariaDB.

Ports personnalisés

Amazon RDS bloque les connexions au port personnalisé 33060 pour le moteur MariaDB. Choisissez un port différent pour votre moteur MariaDB.

Performance Insights

Les compteurs InnoDB ne sont pas visibles dans l'analyse des performances pour RDS for MariaDB version 10.11, car la communauté MariaDB ne les prend plus en charge.

Amazon RDS for Microsoft SQL Server

Amazon RDS prend en charge plusieurs versions et éditions de Microsoft SQL Server. Le tableau suivant indique la version mineure prise en charge la plus récente de chaque version majeure. Pour obtenir la liste complètes des versions, éditions et versions de moteur RDS prises en charge, consultez Versions de Microsoft SQL Server sur Amazon RDS.

Version majeure	Service Pack/ GDR	Mise à jour cumulative	Version mineure	Article de la base de connaissances	Date de parution
SQL Server 2022	Ne s'applique pas	CU20	16,0,4205,1	KB5059390	10 juillet 2025
SQL Server 2019	GDR	CU32 RDA	15,0,4435,7	KB5058722	8 juillet 2025
SQL Server 2017	Ne s'applique pas	CU31 RDA	14,0.3495,9	KB5058714	8 juillet 2025
SQL Server 2016	SP3 RDA	CU14	13,0.7055,9	KB5058717	8 juillet 2025

Pour plus d'informations sur les licences SQL Server;, consultez Gestion des licences Microsoft SQL Server sur Amazon RDS. Pour plus d'informations sur les versions de SQL Server, consultez cet article du support Microsoft sur Où trouver des informations sur les dernières versions de SQL Server.

Avec Amazon RDS, vous pouvez créer des instances de base de données et des instantanés de base de données, des point-in-time restaurations et des sauvegardes automatisées ou manuelles. Les instances de bases de données exécutant SQL Server peuvent être utilisées dans un VPC. Vous pouvez également utiliser le protocole SSL pour vous connecter à une instance de base de données exécutant SQL Server, et vous servir du chiffrement TDE (Transparent Data Encryption) pour chiffrer les données au repos. Amazon RDS prend actuellement en charge les déploiements multi-AZ pour

SQL Server en utilisant SQL Server Database Mirroring (DBM) ou Always On Availability Groups (AGs) comme solution de basculement à haute disponibilité.

Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de bases de données et limite l'accès à certaines tables et procédures système qui requièrent des privilèges avancés. Amazon RDS prend en charge l'accès aux bases de données sur une instance de base de données à l'aide d'application cliente SQL standard telle que Microsoft SQL Server Management Studio. Amazon RDS ne permet pas d'accès d'hôte direct à une instance de base de données via Telnet, Secure Shell (SSH) ou une connexion Bureau à distance Windows. Lorsque vous créez une instance de base de données, le rôle db_owner est attribué à l'utilisateur principal pour toutes les bases de données utilisateur sur cette instance, et cet utilisateur dispose de toutes les autorisations au niveau de la base de données, sauf celles qui sont utilisées pour les sauvegardes. Amazon RDS gère les sauvegardes pour vous.

Avant de créer votre première instance de base de données, vous devez suivre la procédure décrite dans la section du présent guide relative à la configuration. Pour plus d'informations, consultez Configuration de votre environnement Amazon RDS.

Rubriques

- Tâches courantes de gestion pour Microsoft SQL Server sur Amazon RDS
- Limites propres aux instances de bases de données Microsoft SQL Server
- prise en charge des classes d'instance de base de données pour Microsoft SQL Server
- Sécurité SQL de Microsoft Server
- Prise en charge du programme de conformité pour les instances de bases de données Microsoft SQL Server
- Versions de Microsoft SQL Server sur Amazon RDS
- Fonctionnalités SQL de Microsoft Server sur Amazon RDS
- Déploiements multi-AZ à l'aide de la mise en miroir de bases de données ou des groupes de disponibilité AlwaysOn Microsoft SQL Server
- Utilisation de Transparent Data Encryption pour chiffrer les données au repos
- Fonctions et procédures stockées Amazon RDS for Microsoft SQL Server
- Fuseau horaire local pour les instances de base de données Microsoft SQL Server
- Gestion des licences Microsoft SQL Server sur Amazon RDS
- Connexion à votre instance de base de données Microsoft SQL Server
- Utilisation d'Active Directory avec RDS for SQL Server

- Mises à niveau du moteur de base de données Microsoft SQL Server
- Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives
- Utilisation des réplicas en lecture pour Microsoft SQL Server dans Amazon RDS
- Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL
- Fonctionnalités supplémentaires pour Microsoft SQL Server sur Amazon RDS
- Options pour le moteur de base de données Microsoft SQL Server
- Tâches DBA courantes pour Amazon RDS pour Microsoft SQL Server

Tâches courantes de gestion pour Microsoft SQL Server sur Amazon RDS

Vous trouverez ci-dessous les tâches courantes de gestion que vous exécutez avec une instance de base de données Amazon RDS for SQL Server, avec des liens vers la documentation appropriée relative à chaque tâche.

Type de tâche	Description	Documentation utile
Classes d'instance, stockage et PIOPS	Si vous créez une instance de base de données à des fins de production, vous devez comprendre comment les classes d'instance, les types de stockage et les IOPS provisionnées fonctionnent dans Amazon RDS.	prise en charge des classes d'instance de base de données pour Microsoft SQL Server Types de stockage Amazon RDS
Déploiements multi-AZ	Une instance de base de données de production doit utiliser des déploieme nts multi-AZ. Les déploieme nts Multi-AZ améliorent la disponibilité, la durabilité des donnés et la tolérance aux pannes pour les instances	Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS Déploiements multi-AZ à l'aide de la mise en miroir de bases de données ou des groupes

Type de tâche	Description	Documentation utile
	de bases de données. Les déploiements multi-AZ pour SQL Server sont mis en œuvre à l'aide de la technolog ie DBM ou de la technologie native de SQL Server. AGs	de disponibilité AlwaysOn Microsoft SQL Server
Amazon Virtual Private Cloud (VPC)	Si votre AWS compte possède un VPC par défaut, votre instance de base de données est automatiquement créée dans le VPC par défaut. Si votre compte n'a pas de VPC par défaut et que vous voulez que l'instance de base de données soit dans un VPC, vous devez créer le VPC et les groupes de sous-réseaux avant de créer l'instance de base de données.	Utilisation d'un(e) instance de base de données dans un VPC
Groupes de sécurité	Par défaut, les instances de bases de données sont créées avec un pare-feu qui empêche d'y accéder. Vous devez donc créer un groupe de sécurité avec les adresses IP correctes et la configuration réseau permettant d'accéder à l'instance de base de données.	Contrôle d'accès par groupe de sécurité

Type de tâche	Description	Documentation utile
Groupes de paramètres	Si votre instance de base de données doit nécessite r des paramètres de base de données spécifiques, vous devez créer un groupe de paramètres avant de créer l'instance de base de données.	Groupes de paramètres pour (Amazon RDS)
Groupes d'options	Si votre instance de base de données doit nécessiter des options de base de données spécifiques, vous devez créer un groupe d'options avant de créer l'instance de base de données.	Options pour le moteur de base de données Microsoft SQL Server
Connexion à votre instance de base de données	Après avoir créé un groupe de sécurité et l'avoir associé à une instance de base de données, vous pouvez vous connecter à l'instance de base de données en utilisant une application cliente SQL standard quelconque telle que Microsoft SQL Server Management Studio.	Connexion à votre instance de base de données Microsoft SQL Server

Type de tâche	Description	Documentation utile
Sauvegarde et restauration	Lorsque vous créez votre instance de base de données, vous pouvez la configurer pour effectuer des sauvegard es automatiques. Vous pouvez également sauvegard er et restaurer vos bases de données manuellem ent à l'aide des fichiers de sauvegarde complète (fichiers .bak).	Présentation des sauvegardes Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives
Surveillance	Vous pouvez surveiller votre instance de base de données SQL Server à l'aide des métriques, des événements et de la surveillance améliorée d' CloudWatch Amazon RDS.	Afficher les métriques dans la RDS console Amazon Consulter les RDS événement s Amazon
Les fichiers journaux	Vous pouvez accéder aux fichiers journaux de votre instance de base de données SQL Server.	Surveillance des fichiers journaux RDSAmazon Fichiers journaux RDS de base de données Amazon pour Microsoft SQL Server

L'utilisation d'instances de bases de données SQL Server implique également des tâches d'administration avancées. Pour plus d'informations, consultez la documentation suivante :

- Tâches DBA courantes pour Amazon RDS pour Microsoft SQL Server.
- Utilisation d'Active Directory AWS géré avec RDS pour SQL Server
- Accès à la base de données tempdb

Limites propres aux instances de bases de données Microsoft SQL Server

L'implémentation Amazon RDS de Microsoft SQL Server sur une instance de base de données comporte certaines restrictions que vous devez connaître :

- Le nombre maximum de bases de données prises en charge sur une instance de base de données dépend du type de classe d'instance et du mode de disponibilité : mise en miroir de bases de données mono-AZ, mise en miroir de bases de données multi-AZ (DBM) ou groupes de disponibilité multi-AZ (). AGs Les bases de données système Microsoft SQL Server ne sont pas prises en compte dans cette limite.

  Le tableau suivant montre le nombre maximum de bases de données prises en charge pour chacun des types de classe d'instance et des modes de disponibilité. Utilisez ce tableau pour déterminer si vous pouvez passer d'un type de classe d'instance ou d'un mode de disponibilité à un autre. La modification de votre instance de base de données source échoue si celle-ci comporte plus de bases de données que ce qui est pris en charge par le type de classe d'instance ou le mode de disponibilité cible. Vous pouvez voir le statut de votre demande dans le panneau Événements.

Type de classe d'instance	Mono-AZ	Multi-AZ avec DBM	Multi-AZ avec Always On AGs
db.*.micro vers db.*.medium	30	N/A	N/A
db.*.large	30	30	30
db.*.xlarge vers db.*.16xlarge	100	50	75
db.*.24xlarge vers db.*.32xlarge	100	50	100

  * Représente les différents types de classes d'instances.

Par exemple, supposons que votre instance de base de données s'exécute sur un type de classe d'instance db.*.16xlarge avec une seule zone de disponibilité (mono-AZ) et qu'elle comporte 76 bases de données. Vous modifiez l'instance de base de données pour passer à l'utilisation de Multi-AZ Always On AGs. Cette mise à niveau échoue, car votre instance de base de données contient plus de bases de données que ce que votre configuration cible peut prendre en charge. Si vous mettez à niveau votre type de classe d'instance plutôt vers db.*.24xlarge, la modification réussit.

Si la mise à niveau échoue, des événements et des messages similaires à ce qui suit s'affichent :

- Unable to modify database instance class. The instance has 76 databases, but after conversion it would only support 75. (Impossible de modifier la classe d'instance de base de données. L'instance comporte 76 bases de données, mais après la conversion, elle n'en prendrait en charge que 75.
- Unable to convert the DB instance to Multi-AZ: The instance has 76 databases, but after conversion it would only support 75. (Impossible de convertir la classe d'instance de base de données en multi-AZ : L'instance comporte 76 bases de données, mais après la conversion, elle n'en prendrait en charge que 75).

En cas d'échec de la point-in-time restauration ou de la restauration par capture instantanée, des événements et des messages similaires aux suivants s'affichent :

- Database instance put into incompatible-restore. The instance has 76 databases, but after conversion it would only support 75. (Impossible de modifier la classe d'instance de base de données. L'instance comporte 76 bases de données, mais après la conversion, elle n'en prendrait en charge que 75.

- Les ports suivants sont réservés à Amazon RDS, et vous ne pouvez pas les utiliser au moment de créer une instance de base de données : `1234, 1434, 3260, 3343, 3389, 47001,` et `49152-49156`.
- Les connexions client à partir d'adresse IP dans la plage 169.254.0.0/16 ne sont pas autorisées. Il s'agit d'une plage d'adresses IP privées automatiques (APIPA, Automatic Private IP Addressing Range), qui est utilisée pour l'adressage de liens locaux.
- SQL Server Standard Edition n'utilisera qu'un sous-ensemble des processeurs disponibles si l'instance de base de données dispose de plus de processeurs que les limites logicielles (24 cœurs, 4 sockets et 128 Go de RAM). Exemples : les classes d'instances db.m5.24xlarge et db.r5.24xlarge.

Pour plus d'informations, consultez le tableau des limites d'échelle sous Editions and supported features of SQL Server 2019 (15.x) (Éditions et fonctions prises en charge de SQL Server 2019 (15.x) dans la documentation Microsoft.

- Amazon RDS for SQL Server ne prend pas en charge l'importation de données dans la base de données msdb.
- Vous ne pouvez pas renommer les bases de données sur une instance de base de données dans un déploiement multi-AZ SQL Server.
- Assurez-vous d'utiliser ces instructions lorsque vous définissez les paramètres de base de données suivants sur RDS for SQL Server :
  - `max server memory (mb)` >= 256 Mo
  - `max worker threads`>= (nombre de caractères CPUs logiques* 7)

  Pour plus d'informations sur la définition des paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).
- La taille de stockage maximale pour les instances de bases de données SQL Server est la suivante :
  - Stockage à usage général (SSD) – 16 Tio pour toutes les éditions
  - Stockage IOPS provisionné : 64 TiB pour toutes les éditions
  - Stockage magnétique – 1 Tio pour toutes les éditions

  Si vous disposez d'un scénario exigeant un important volume de stockage, vous pouvez utiliser le partitionnement sur plusieurs instances de bases de données pour contourner la limite. Cette approche nécessite une logique de routage dépendant des données dans les applications qui se connectent au système partitionné. Vous pouvez utiliser une infrastructure de partitionnement existante ou écrire du code personnalisé pour activer le partitionnement. Si vous utilisez une infrastructure existante, elle ne peut installer aucun composant sur le même serveur que l'instance de base de données.
- La taille de stockage minimale pour les instances de bases de données SQL Server est la suivante :
  - Stockage à usage général (SSD) – 20 Gio pour les éditions Enterprise, Standard, Web et Express
  - Stockage IOPS provisionnés – 20 Gio pour les éditions Enterprise, Standard, Web et Express
  - Stockage magnétique – 20 Gio pour les éditions Enterprise, Standard, Web et Express

- Amazon RDS ne prend pas en charge l'exécution de ces services sur le même serveur que votre instance de base de données RDS :
  - Data Quality Services
  - Master Data Services

  Pour utiliser ces fonctionnalités, nous vous recommandons d'installer SQL Server sur une EC2 instance Amazon ou d'utiliser une instance SQL Server sur site. Dans ces cas, l' EC2 instance SQL Server agit en tant que serveur principal des services de données pour votre instance de base de données SQL Server sur Amazon RDS. Vous pouvez installer SQL Server sur une EC2 instance Amazon avec le stockage Amazon EBS, conformément aux politiques de licence de Microsoft.
- Du fait de limitations dans Microsoft SQL Server, une restauration à un moment donné avant une exécution réussie de `DROP DATABASE` peut ne pas refléter l'état de cette base de données à ce moment précis. Par exemple, la base de données supprimée est généralement restaurée à l'état qui était le sien jusqu'à 5 minutes avant l'exécution de la commande `DROP DATABASE`. Ce type de restauration signifie que vous ne pouvez pas restaurer les transactions effectuées pendant ces quelques minutes sur votre base de données supprimée. Pour contourner ce problème, vous pouvez réémettre la commande `DROP DATABASE` après que l'opération de restauration est terminée. La suppression d'une base de données supprime les journaux de transaction pour cette base de données.
- Pour SQL Server, vous créez vos bases de données après avoir créé votre instance de base de données. Les noms de base de données suivent les règles de dénomination SQL Server habituelles, avec les différences suivantes :
  - Les noms de base de données ne peuvent pas commencer par `rdsadmin`.
  - Ils ne peuvent pas commencer ni se terminer par un espace ou une tabulation.
  - Ils ne peuvent contenir aucun caractère qui crée une nouvelle ligne.
  - Ils ne peuvent pas contenir de guillemet simple (`'`).
- SQL Server Web Edition vous permet uniquement d'utiliser le modèle Dev/Test lors de la création d'une nouvelle instance de base de données RDS pour SQL Server.

prise en charge des classes d'instance de base de données pour Microsoft SQL Server

Les capacités de calcul et de mémoire d'une instance de base de données sont déterminées par sa classe d'instance de base de données. La classe d'instance de bases de données dont vous avez

besoin varie selon vos exigences en mémoire et en puissance de traitement. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .

La liste suivante des classes d'instances de base de données prises en charge pour Microsoft SQL Server est fournie ici pour votre commodité. Pour la liste la plus récente, consultez la RDS console : https://console.aws.amazon.com/rds/.

Toutes les classes d'instances de base de données ne sont pas disponibles sur toutes les versions mineures SQL du serveur prises en charge. Par exemple, certaines classes d'instance de base de données plus récentes telles que db.r6i ne sont pas disponibles sur les versions mineures antérieures. Vous pouvez utiliser la AWS CLI commande describe-orderable-db-instance-options pour savoir quelles classes d'instances de base de données sont disponibles pour l'édition et la version de votre SQL serveur.

SQLÉdition serveur	Gamme de support 2022	Prise en charge 2019	Prise en charge 2017 et 2016
Enterprise Edition	db.t3.xlarge –db.t3.2xlarge	db.t3.xlarge –db.t3.2xlarge	db.t3.xlarge –db.t3.2xlarge
	db.r5.xlarge –db.r5.24xlarge	db.r5.xlarge –db.r5.24xlarge	db.r5.xlarge –db.r5.24xlarge
	db.r5b.xlarge –db.r5b.24xlarge	db.r5b.xlarge –db.r5b.24xlarge	db.r5b.xlarge –db.r5b.24xlarge
	db.r5d.xlarge –db.r5d.24xlarge	db.r5d.xlarge –db.r5d.24xlarge	db.r5d.xlarge –db.r5d.24xlarge
	db.r6i.xlarge –db.r6i.32xlarge	db.r6i.xlarge –db.r6i.32xlarge	db.r6i.xlarge –db.r6i.32xlarge
	db.m5.xlarge –db.m5.24xlarge	db.m5.xlarge –db.m5.24xlarge	db.m5.xlarge –db.m5.24xlarge

SQLÉdition serveur	Gamme de support 2022	Prise en charge 2019	Prise en charge 2017 et 2016
	db.m5d.xlarge –db.m5d.24xlarge	db.m5d.xlarge –db.m5d.24xlarge	db.m5d.xlarge –db.m5d.24xlarge
	db.m6i.xlarge –db.m6i.32xlarge	db.m6i.xlarge –db.m6i.32xlarge	db.m6i.xlarge –db.m6i.32xlarge
	db.x2iedn.xlarge –db.x2iedn.32xlarge	db.x1.16xlarge –db.x1.32xlarge	db.x1.16xlarge –db.x1.32xlarge
	db.z1d.xlarge –db.z1d.12xlarge	db.x1e.xlarge –db.x1e.32xlarge	db.x1e.xlarge –db.x1e.32xlarge
		db.x2iedn.xlarge –db.x2iedn.32xlarge	db.x2iedn.xlarge –db.x2iedn.32xlarge
		db.z1d.xlarge –db.z1d.12xlarge	db.z1d.xlarge –db.z1d.12xlarge

SQLÉdition serveur	Gamme de support 2022	Prise en charge 2019	Prise en charge 2017 et 2016
Standard	db.t3.xlarge –db.t3.2xlarge db.r5.large –db.r5.24xlarge db.r5b.large –db.r5b.8xlarge db.r5d.large –db.r5d.24xlarge db.r6i.large –db.r6i.8xlarge db.m5.large –db.m5.24xlarge db.m5d.large –db.m5d.24xlarge db.m6i.large –db.m6i.8xlarge db.x2iedn.xlarge –db.x2iedn.8xlarge db.z1d.large –db.z1d.12xlarge	db.t3.xlarge –db.t3.2xlarge db.r5.large –db.r5.24xlarge db.r5b.large –db.r5b.24xlarge db.r5d.large –db.r5d.24xlarge db.r6i.large –db.r6i.8xlarge db.m5.large –db.m5.24xlarge db.m5d.large –db.m5d.24xlarge db.m6i.large –db.m6i.8xlarge db.x1.16xlarge –db.x1.32xlarge db.x1e.xlarge –db.x1e.32xlarge	db.t3.xlarge –db.t3.2xlarge db.r5.large –db.r5.24xlarge db.r5b.large –db.r5b.24xlarge db.r5d.large –db.r5d.24xlarge db.r6i.large –db.r6i.8xlarge db.m5.large –db.m5.24xlarge db.m5d.large –db.m5d.24xlarge db.m6i.large –db.m6i.8xlarge db.x1.16xlarge –db.x1.32xlarge db.x1e.xlarge –db.x1e.32xlarge

SQLÉditi n serveur	Gamme de support 2022	Prise en charge 2019	Prise en charge 2017 et 2016
		db.x2iedn .xlarge –db.x2iedn .32xlarge db.z1d.la rge –db.z1d.12 xlarge	db.x2iedn .xlarge –db.x2iedn .32xlarge db.z1d.la rge –db.z1d.12 xlarge
Web Edi	db.t3.sma ll –db.t3.xlarge db.r5.lar ge –db.r5.4xlarge db.r5b.la rge –db.r5b.4xlarge db.r5d.la rge –db.r5d.4xlarge db.r6i.la rge –db.r6i.4xlarge db.m5.lar ge –db.m5.4xlarge db.m5d.la rge –db.m5d.4xlarge db.m6i.la rge –db.m6i.4xlarge db.z1d.la rge –db.z1d.13 xlarge	db.t3.sma ll –db.t3.2xlarge db.r5.lar ge –db.r5.4xlarge db.r5b.la rge –db.r5b.4xlarge db.r5d.la rge –db.r5d.4xlarge db.r6i.la rge –db.r6i.4xlarge db.m5.lar ge –db.m5.4xlarge db.m5d.la rge –db.m5d.4xlarge db.m6i.la rge –db.m6i.4xlarge db.z1d.la rge –db.z1d.3xlarge	db.t3.sma ll –db.t3.2xlarge db.r5.lar ge –db.r5.4xlarge db.r5b.la rge –db.r5b.4xlarge db.r5d.la rge –db.r5d.4xlarge db.r6i.la rge –db.r6i.4xlarge db.m5.lar ge –db.m5.4xlarge db.m5d.la rge –db.m5d.4xlarge db.m6i.la rge –db.m6i.4xlarge db.z1d.la rge –db.z1d.3xlarge

SQLÉdition serveur	Gamme de support 2022	Prise en charge 2019	Prise en charge 2017 et 2016
Express	db.t3.micro –db.t3.xlarge	db.t3.micro –db.t3.xlarge	db.t3.micro –db.t3.xlarge

Sécurité SQL de Microsoft Server

Le moteur de base de données Microsoft SQL Server utilise une sécurité basée sur les rôles. Le nom d'utilisateur principal que vous spécifiez lorsque vous créez une instance de base de données est un identifiant d'authentification SQL serveur membre des rôles serveur `setupadmin` fixes et est membre des rôles de serveur. `processadmin` `public`

Tout utilisateur qui créé une base de données est affecté au rôle db_owner pour cette base de données et possède toutes les autorisations de niveau base de données, sauf celles utilisées pour les sauvegardes. Amazon RDS gère les sauvegardes pour vous.

Les rôles de niveau serveur suivants ne sont pas disponibles dans Amazon RDS for SQL Server :

- bulkadmin
- dbcreator
- diskadmin
- securityadmin
- serveradmin
- sysadmin

Les autorisations de niveau serveur suivantes ne sont pas disponibles RDS pour les instances de base de données de SQL serveur :

- ALTER ANY DATABASE
- ALTER ANY EVENT NOTIFICATION
- ALTER RESOURCES

- ALTERSETTINGS(vous pouvez utiliser les API opérations de groupe de paramètres de base de données pour modifier les paramètres ; pour plus d'informations, voirGroupes de paramètres pour (Amazon RDS))
- AUTHENTICATE SERVER
- CONTROL_SERVER
- CREATE DDL EVENT NOTIFICATION
- CREATE ENDPOINT
- CREATE SERVER ROLE
- CREATE TRACE EVENT NOTIFICATION
- DROP ANY DATABASE
- EXTERNAL ACCESS ASSEMBLY
- SHUTDOWN(Vous pouvez utiliser l'option de RDS redémarrage à la place)
- UNSAFE ASSEMBLY
- ALTER ANY AVAILABILITY GROUP
- CREATE ANY AVAILABILITY GROUP

SSLprise en charge des instances de base de données Microsoft SQL Server

Vous pouvez l'utiliser SSL pour chiffrer les connexions entre vos applications et vos instances Amazon RDS DB exécutant Microsoft SQL Server. Vous pouvez également forcer l'utilisation de toutes les connexions à votre instance de base de donnéesSSL. Si vous forcez l'utilisation des connexionsSSL, cela se produit de manière transparente pour le client, et le client n'a aucune tâche à effectuer pour l'utiliserSSL.

SSLest pris en charge dans toutes les AWS régions et pour toutes les éditions de SQL serveur prises en charge. Pour de plus amples informations, veuillez consulter Utilisation SSL avec une instance de base de données Microsoft SQL Server.

Utilisation SSL avec une instance de base de données Microsoft SQL Server

Vous pouvez utiliser Secure Sockets Layer (SSL) pour chiffrer les connexions entre vos applications clientes et vos instances Amazon RDS DB exécutant Microsoft SQL Server. SSLle support est disponible dans toutes les AWS régions pour toutes les éditions de SQL serveur prises en charge.

Lorsque vous créez une instance de base de données de SQL serveur, Amazon RDS crée un SSL certificat pour celle-ci. Le SSL certificat inclut le point de terminaison de l'instance de base de données comme nom commun (CN) du SSL certificat afin de se prémunir contre les attaques par usurpation d'identité.

Il existe deux méthodes pour vous connecter SSL à votre instance de base de données de SQL serveur :

- Force SSL pour toutes les connexions : cela se produit de manière transparente pour le client, et le client n'a aucun travail à effectuer pour l'utiliserSSL.

  > **Note**
  >
  > Lorsque vous définissez 1 et utilisez `rds.force_ssl` les SSMS versions 19.3, 20.0 et 20.2, vérifiez les points suivants :
  >
  > - Activez le certificat Trust Server dansSSMS.
  > - Importez le certificat dans votre système.

- Chiffrer des connexions spécifiques : cela permet de configurer une SSL connexion à partir d'un ordinateur client spécifique, et vous devez travailler sur le client pour chiffrer les connexions.

Pour plus d'informations sur la prise en charge de Transport Layer Security (TLS) pour SQL Server, voir Support TLS 1.2 pour Microsoft SQL Server.

Forcer les connexions à votre instance de base de données à utiliser SSL

Vous pouvez forcer l'utilisation de toutes les connexions à votre instance de base de donnéesSSL. Si vous forcez l'utilisation des connexionsSSL, cela se produit de manière transparente pour le client, et le client n'a aucune tâche à effectuer pour l'utiliserSSL.

Si vous voulez forcerSSL, utilisez le `rds.force_ssl` paramètre. Par défaut, le paramètre `rds.force_ssl` est défini sur `0 (off)`. Définissez le `rds.force_ssl` paramètre sur `1 (on)` pour forcer l'utilisation des connexionsSSL. Le paramètre `rds.force_ssl` est statique, donc après avoir modifié la valeur, vous devez redémarrer votre instance de base de données pour que la modification soit effective.

Pour forcer toutes les connexions à votre instance de base de données à utiliser SSL

1. Déterminez le groupe de paramètres attaché à votre instance de base de données :
   a. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
   b. Dans le coin supérieur droit de la RDS console Amazon, choisissez la AWS région de votre instance de base de données.
   c. Dans le panneau de navigation, choisissez Bases de données, puis le nom de votre instance de base de données pour afficher ses détails.
   d. Choisissez l'onglet Configuration. Recherchez le groupe de paramètres dans la section.
2. Le cas échéant, créez un nouveau groupe de paramètres. Si votre instance de base de données utilise le groupe de paramètres par défaut, vous devez créer un nouveau groupe de paramètres. Si votre instance de base de données utilise un groupe de paramètres personnalisé, vous pouvez choisir de modifier le groupe de paramètres existant ou de créer un nouveau groupe de paramètre. Si vous modifiez un groupe de paramètre existant, la modification s'applique à l'ensemble des instances de base de données qui utilisent ce groupe de paramètres.

   Pour créer un nouveau groupe de paramètres, suivez les instructions dans Création d'un groupe de paramètres de base de données dans RDSAmazon.
3. Modifiez le groupe de paramètres nouvellement créé ou existant pour définir le paramètre `rds.force_ssl` sur `true`. Pour modifier le groupe de paramètres, suivez les instructions dans Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).
4. Si vous créez un nouveau groupe de paramètres, modifiez votre instance de base de données à attacher au nouveau groupe de paramètres. Modifiez le paramètre Groupe de paramètres de base de données de l'instance de base de données. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
5. Redémarrez votre instance de base de données. Pour plus d'informations, consultez Redémarrage d'une instance de base de données.

Chiffrement de connexions spécifiques

Vous pouvez forcer l'utilisation de toutes les connexions à votre instance de base de donnéesSSL, ou vous pouvez chiffrer les connexions provenant d'ordinateurs clients spécifiques uniquement. Pour l'utiliser à SSL partir d'un client spécifique, vous devez obtenir des certificats pour l'ordinateur client, importer des certificats sur l'ordinateur client, puis chiffrer les connexions depuis l'ordinateur client.

> **Note**
>
> Toutes les instances de SQL serveur créées après le 5 août 2014 utilisent le point de terminaison de l'instance de base de données dans le champ Nom commun (CN) du SSL certificat. Avant le 5 août 2014, la vérification des SSL certificats n'était pas disponible pour les instances VPC basées sur le SQL serveur. Si vous avez une instance de VPC base de données de SQL serveur créée avant le 5 août 2014 et que vous souhaitez utiliser la vérification des SSL certificats et vous assurer que le point de terminaison de l'instance est inclus en tant que CN pour le SSL certificat de cette instance de base de données, renommez l'instance. Lorsque vous renommez une instance de base de données, un nouveau certificat est déployé et l'instance est redémarrée pour activer le nouveau certificat.

Obtention de certificats pour les ordinateurs clients

Pour chiffrer les connexions entre un ordinateur client et une RDS instance de base de données Amazon exécutant Microsoft SQL Server, vous avez besoin d'un certificat sur votre ordinateur client.

Pour obtenir ce certificat, téléchargez-le sur votre ordinateur client. Vous pouvez télécharger un certificat racine valide pour toutes les régions. Un ensemble de certificats contenant l'ancien certificat racine et le nouveau certificat racine peut également être téléchargé. De plus, vous pouvez télécharger des certificats intermédiaires propres à une région. Pour en savoir plus sur le téléchargement de certificats, consultez .

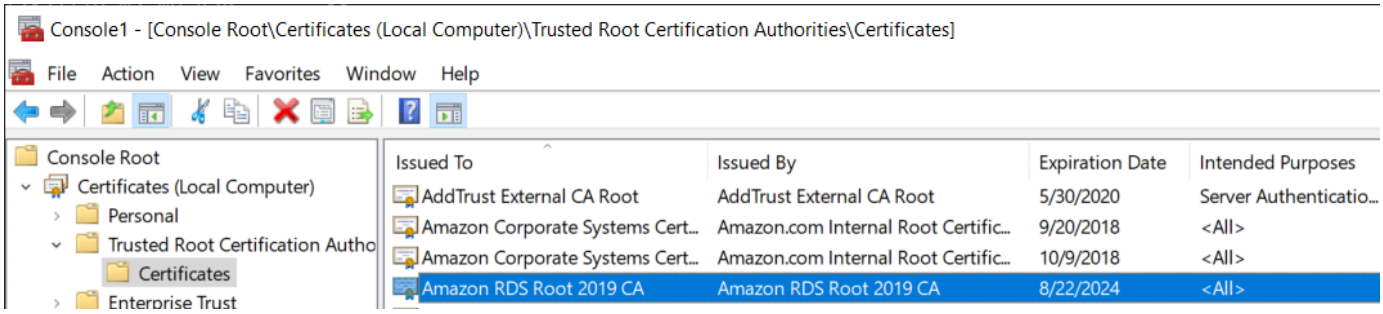
Après avoir téléchargé le certificat approprié, importez-le dans votre système d'exploitation Microsoft Windows en suivant la procédure de la section suivante.

Importation de certificats sur des ordinateurs clients

Vous pouvez utiliser la procédure suivante pour importer votre certificat dans le système d'exploitation Microsoft Windows de votre ordinateur client.

Pour importer le certificat dans votre système d'exploitation Windows :

1. Dans le menu Début, saisissez **Run** dans la zone de recherche et appuyez sur Entrée.
2. Dans la zone Ouvrir, saisissez **MMC** et choisissez OK.
3. Dans le menu Fichier de la MMC console, choisissez Ajouter/Supprimer un composant logiciel enfichable.
4. Dans la boîte de dialogue Add or Remove Snap-ins (Ajouter ou Supprimer les Snap-ins), pour Available snap-ins (Snap-ins disponibles), sélectionnez **Certificates** et choisissez Ajouter.
5. Dans la boîte de dialogue Certificates snap-in (Certificats snap-in), choisissez Computer account (Compte d'ordinateur) et Suivant.
6. Dans la boîte de dialogue Select computer (Sélectionner ordinateur), choisissez Terminer.
7. Dans la boîte de dialogue Add or Remove Snap-ins (Ajouter ou supprimer snap-ins), choisissez OK.
8. Dans la MMC console, développez Certificats, ouvrez le menu contextuel (clic droit) pour les autorités de certification Trusted Root, choisissez Toutes les tâches, puis Importer.
9. Sur la première page de l'assistant d'importation de certificat, choisissez Suivant.
10. Sur la deuxième page de l'assistant d'importation de certificat, choisissez Browse (Naviguer). Dans la fenêtre de navigation, modifiez le type de fichier à Tous les fichiers (*.*), car .pem n'est pas une extension de certificat standard. Recherchez le fichier .pem que vous avez téléchargé précédemment.
11. Choisissez Ouvrir pour sélectionner le fichier de certificat, puis Suivant.
12. Sur la troisième page de l'assistant d'importation de certificat, choisissez Suivant.
13. Sur la quatrième page de l'assistant d'importation de certificat, choisissez Terminer. Une boîte de dialogue apparaît confirmant la réussite de l'importation.
14. Dans la MMC console, développez Certificates, développez Trusted Root Certification Authorities, puis sélectionnez Certificates. Recherchez le certificat pour confirmer son existence, comme illustré ici.



Chiffrement des connexions à une RDS instance de base de données Amazon exécutant Microsoft Server SQL

Après avoir importé un certificat sur votre ordinateur client, vous pouvez chiffrer les connexions entre l'ordinateur client et une RDS instance de base de données Amazon exécutant Microsoft SQL Server.

Pour SQL Server Management Studio, suivez la procédure suivante. Pour plus d'informations sur SQL Server Management Studio, voir Utiliser SQL Server Management Studio.

Pour chiffrer les connexions depuis SQL Server Management Studio

1. Lancez SQL Server Management Studio.
2. Pour Connect to server (Se connecter au serveur), entrez les informations serveur, le mot de passe et le nom d'utilisateur de connexion.
3. Choisissez Options.
4. Sélectionnez Encrypt connection (Chiffrer la connexion).
5. Choisissez Connexion.
6. Vérifiez que votre connexion est chiffrée en exécutant la requête suivante. Vérifiez que la requête renvoie `true` pour `encrypt_option`.

```
select ENCRYPT_OPTION from SYS.DM_EXEC_CONNECTIONS where SESSION_ID = @@SPID
```

Pour tout autre SQL client, suivez la procédure ci-dessous.

Pour chiffrer les connexions provenant d'autres clients SQL

1. Ajoutez `encrypt=true` au début du votre chaîne de connexion. Cette chaîne peut être disponible en option ou en tant que propriété sur la page de connexion dans les GUI outils.

> **Note**
>
> Pour activer SSL le chiffrement pour les clients qui se connectent viaJDBC, vous devrez peut-être ajouter le RDS SQL certificat Amazon au magasin de certificats Java CA (cacerts). Vous pouvez faire cela en utilisant l'utilitaire keytool.

2. Vérifiez que votre connexion est chiffrée en exécutant la requête suivante. Vérifiez que la requête renvoie `true` pour `encrypt_option`.

```
select ENCRYPT_OPTION from SYS.DM_EXEC_CONNECTIONS where SESSION_ID = @@SPID
```

Configuration des protocoles de sécurité et des chiffrements du SQL serveur

Vous pouvez activer et désactiver certains protocoles de sécurité et chiffrements à l'aide des paramètres de base de données. Les paramètres de sécurité que vous pouvez configurer (sauf pour TLS la version 1.2) sont présentés dans le tableau suivant.

Paramètre de base de données	Valeurs autorisées (valeur par défaut en gras)	Description
rds.tls10	par défaut, activé, désactivé	TLS1,0.
rds.tls11	par défaut, activé, désactivé	TLS1,1.
rds.tls12	default	TLS1,2. Vous ne pouvez pas modifier cette valeur.
rds.fips	0, 1	Lorsque vous définissez le paramètre sur 1, RDS force l'utilisation de modules conformes à la norme Federal Information Processing Standard (FIPS) 140-2. Pour plus d'informations, consultez la section Utiliser SQL Server 2016 en mode FIPS compatible 140-2 dans la documentation Microsoft.
rds.rc4	par défaut, activé, désactivé	RC4chiffrement de flux.
rds.diffie-hellman	par défaut, activé, désactivé	Chiffrement d'échange de clés Diffie-Hellman.
rds. diffie-hellman-min-key-longueur en bits	par défaut, 1024, 2048, 3072, 4096	Longueur minimale de bits pour les clés Diffie-Hellman.

Paramètre de base de données	Valeurs autorisées (valeur par défaut en gras)	Description
rds.curve25519	par défaut, activé, désactivé	Chiffrement Curve25519 elliptic-curve. Ce paramètre n'est pas pris en charge pour toutes les versions du moteur.
rds.3des168	par défaut, activé, désactivé	Chiffre de chiffrement Triple Data Encryption Standard (DES) avec une longueur de clé de 168 bits.

> **Note**
>
> Pour les versions mineures du moteur postérieures aux versions 16.00.4120.1, 15.00.4365.2, 14.00.3465.1, 13.00.6435.1 et 12.00.6449.1, le paramètre par défaut pour les paramètres de base de données,,, et est désactivé. *`rds.tls10` `rds.tls11` `rds.rc4` `rds.curve25519` `rds.3des168`* Dans le cas contraire, le paramètre par défaut est activé.
> Pour les versions mineures du moteur postérieures à 16.00.4120.1, 15.00.4365.2, 14.00.3465.1, 13.00.6435.1 et 12.00.6449.1, le paramètre par défaut pour est 3072. `rds.diffie-hellman-min-key-bit-length` Sinon, le paramètre par défaut est 2048.

Utilisez la procédure suivante pour configurer les protocoles de sécurité et les chiffrements :

1. Créez un groupe de paramètres DB personnalisé.
2. Modifiez les paramètres du groupe de paramètres.
3. Associez le groupe de paramètres DB à l'instance de base de données.

Pour plus d'informations sur les groupes de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).

Création du groupe de paramètres liés à la sécurité

Créez un groupe de paramètres pour vos paramètres liés à la sécurité qui correspond à l'édition SQL du serveur et à la version de votre instance de base de données.

Console

La procédure suivante crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans le volet Créer un groupe de paramètres, faites ce qui suit :
   a. Pour Famille de groupes de paramètres, choisissez sqlserver-se-13.0.
   b. Pour Nom du groupe, saisissez un identifiant pour le groupe de paramètres, tel que **`sqlserver-ciphers-se-13`**.
   c. Pour Description, saisissez **`Parameter group for security protocols and ciphers`**.
5. Sélectionnez Create (Créer).

CLI

La procédure suivante crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

- Exécutez une des commandes suivantes :

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-db-parameter-group \
    --db-parameter-group-name sqlserver-ciphers-se-13 \
    --db-parameter-group-family "sqlserver-se-13.0" \
    --description "Parameter group for security protocols and ciphers"
```

  Dans Windows:

```
aws rds create-db-parameter-group ^
```

```
    --db-parameter-group-name sqlserver-ciphers-se-13 ^
    --db-parameter-group-family "sqlserver-se-13.0" ^
    --description "Parameter group for security protocols and ciphers"
```

Modification des paramètres liés à la sécurité

Modifiez les paramètres liés à la sécurité dans le groupe de paramètres correspondant à l'édition et à la version SQL du serveur de votre instance de base de données.

Console

La procédure suivante modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016. Cet exemple désactive TLS la version 1.0.

Pour modifier le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez le groupe de paramètres, par exemple sqlserver-ciphers-se-13.
4. Sous Paramètres, filtrez la liste des paramètres pour **`rds`**.
5. Choisissez Modifier les paramètres.
6. Choisissez rds.tls10.
7. Pour Valeurs, choisissez désactivé.
8. Sélectionnez Enregistrer les modifications.

CLI

La procédure suivante modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016. Cet exemple désactive TLS la version 1.0.

Pour modifier le groupe de paramètres

- Exécutez une des commandes suivantes :

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name sqlserver-ciphers-se-13 \
    --parameters
 "ParameterName='rds.tls10',ParameterValue='disabled',ApplyMethod=pending-reboot"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name sqlserver-ciphers-se-13 ^
    --parameters
 "ParameterName='rds.tls10',ParameterValue='disabled',ApplyMethod=pending-reboot"
```

Association du groupe de paramètres liés à la sécurité à votre instance de base de données

Pour associer le groupe de paramètres à votre instance de base de données, utilisez le AWS Management Console ou le AWS CLI.

Console

Vous pouvez associer le groupe de paramètres à une instance de base de données nouvelle ou existante :

- Pour une nouvelle instance de base de données, associez-la lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-la en modifiant l'instance. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

CLI

Vous pouvez associer le groupe de paramètres à une instance de base de données nouvelle ou existante.

Pour créer une instance de base de données avec le groupe de paramètres

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe de paramètres.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 13.00.5426.0.v1 \
    --allocated-storage 100 \
    --master-user-password secret123 \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --db-parameter-group-name sqlserver-ciphers-se-13
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 13.00.5426.0.v1 ^
    --allocated-storage 100 ^
    --master-user-password secret123 ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --db-parameter-group-name sqlserver-ciphers-se-13
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Pour modifier une instance de base de données et associer le groupe de paramètres

- Exécutez une des commandes suivantes :

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --db-parameter-group-name sqlserver-ciphers-se-13 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --db-parameter-group-name sqlserver-ciphers-se-13 ^
    --apply-immediately
```

Mise à jour des applications pour se connecter aux instances de base de données Microsoft SQL Server à l'aide de nouveaux TLS certificatsSSL/

Depuis le 13 janvier 2023, Amazon RDS a publié de nouveaux certificats d'autorité de certification (CA) pour la connexion à vos RDS instances de base de données à l'aide de Secure Socket Layer ou de Transport Layer Security (SSL/TLS). Vous trouverez ci-après des informations sur la mise à jour de vos applications afin d'utiliser les nouveaux certificats.

Cette rubrique peut vous aider à déterminer si des applications clientes utilisentSSL/TLSpour se connecter à vos instances de base de données. Si tel est le cas, il vous est alors possible de vérifier si ces applications nécessitent une vérification du certificat pour se connecter.

> **Note**
>
> Certaines applications sont configurées pour se connecter aux instances de base de données SQL du serveur uniquement si elles peuvent vérifier avec succès le certificat sur le serveur. Pour ces applications, vous devez mettre à jour les magasins d'approbations des applications clientes afin d'inclure les nouveaux certificats de l'autorité de certification.

Une fois que vous avez mis à jour les certificats de l'autorité de certification dans les magasins d'approbations des applications clientes, vous pouvez soumettre les certificats de vos instances de bases de données à une rotation. Nous vous recommandons vivement de tester ces procédures dans un environnement de développement ou intermédiaire avant de les implémenter dans vos environnements de production.

Pour de plus amples informations sur la rotation de certificats, veuillez consulter Rotation de votre certificat SSL/TLS. Pour en savoir plus sur le téléchargement de certificats, consultez . Pour plus d'informations sur l'utilisation deSSL/TLSavec les instances de base de données Microsoft SQL Server, consultezUtilisation SSL avec une instance de base de données Microsoft SQL Server.

Rubriques

- Déterminer si des applications se connectent à votre instance de base de données Microsoft SQL Server à l'aide de SSL
- Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter
- Mise à jour du magasin d'approbations de votre application

Déterminer si des applications se connectent à votre instance de base de données Microsoft SQL Server à l'aide de SSL

Dans la configuration de l'instance de base de données, vérifiez la valeur du paramètre `rds.force_ssl`. Par défaut, le paramètre `rds.force_ssl` a pour valeur 0 (désactivé). Si le `rds.force_ssl` paramètre est défini sur 1 (activé), les clients doivent utiliserSSL/TLSpour les connexions. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Exécutez la requête suivante afin d'obtenir l'option de chiffrement actuelle pour toutes les connexion ouvertes à une instance de base de données. La colonne `ENCRYPT_OPTION` renvoie `TRUE` si la connexion est chiffrée.

```
select SESSION_ID,
    ENCRYPT_OPTION,
    NET_TRANSPORT,
    AUTH_SCHEME
    from SYS.DM_EXEC_CONNECTIONS
```

Cette requête affiche uniquement les connexions actuelles. Il n'indique pas si les applications qui se sont connectées et déconnectées par le passé ont utiliséSSL.

Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter

Vous pouvez vérifier si différents types de clients requièrent une vérification du certificat pour pouvoir se connecter.

> **Note**
>
> Si vous utilisez des connecteurs autres que ceux répertoriés, veuillez consulter la documentation spécifique au connecteur pour des informations sur leur façon d'appliquer des connexions chiffrées. Pour plus d'informations, consultez la section Modules de connexion pour les SQL bases de données Microsoft dans la documentation Microsoft SQL Server.

SQLStudio de gestion de serveurs

Vérifiez si le chiffrement est appliqué pour les connexions à SQL Server Management Studio :

1. Lancez SQL Server Management Studio.
2. Pour Connect to server (Se connecter au serveur), entrez les informations de serveur, le mot de passe et le nom d'utilisateur de connexion.
3. Choisissez Options.
4. Vérifiez si Encrypt connection (Chiffrer la connexion) est sélectionné sur la page de connexion.

Pour plus d'informations sur SQL Server Management Studio, voir Utiliser SQL Server Management Studio.

sqlcmd

L'exemple suivant avec le `sqlcmd` client montre comment vérifier la connexion au SQL serveur d'un script afin de déterminer si les connexions réussies nécessitent un certificat valide. Pour plus d'informations, consultez la section Connexion avec sqlcmd dans la documentation de Microsoft SQL Server.

Lors de l'utilisation`sqlcmd`, une SSL connexion doit être vérifiée par rapport au certificat du serveur si vous utilisez l'argument de `-N` commande pour chiffrer les connexions, comme dans l'exemple suivant.

```
$ sqlcmd -N -S dbinstance.rds.amazon.com -d ExampleDB
```

> **Note**
>
> Si `sqlcmd` est invoqué avec l'option `-C`, il approuve le certificat de serveur, même s'il ne correspond pas au magasin d'approbations côté client.

ADO.NET

Dans l'exemple suivant, l'application se connecte à l'aide deSSL, et le certificat du serveur doit être vérifié.

```
using SQLC = Microsoft.Data.SqlClient;

...

    static public void Main()
    {
        using (var connection = new SQLC.SqlConnection(
            "Server=tcp:dbinstance.rds.amazon.com;" +
            "Database=ExampleDB;User ID=LOGIN_NAME;" +
            "Password=YOUR_PASSWORD;" +
            "Encrypt=True;TrustServerCertificate=False;"
            ))
        {
            connection.Open();
            ...
        }
```

Java

Dans l'exemple suivant, l'application se connecte à l'aide deSSL, et le certificat du serveur doit être vérifié.

```
String connectionUrl =
    "jdbc:sqlserver://dbinstance.rds.amazon.com;" +
    "databaseName=ExampleDB;integratedSecurity=true;" +
    "encrypt=true;trustServerCertificate=false";
```

Pour activer SSL le chiffrement pour les clients qui se connectent viaJDBC, vous devrez peut-être ajouter le RDS certificat Amazon au magasin de certificats Java CA. Pour obtenir des instructions, consultez la section Configuration du client pour le chiffrement dans la documentation de Microsoft SQL Server. Vous pouvez également fournir directement le nom du fichier du certificat de l'autorité de certification approuvé en ajoutant `trustStore=`*`path-to-certificate-trust-store-file`* à la chaîne de connexion.

> **Note**
>
> Si vous utilisez `TrustServerCertificate=true` (ou son équivalent) dans la chaîne de connexion, le processus de connexion ignore la validation de la chaîne d'approbation. Dans ce cas, l'application se connecte, même lorsque le certificat ne peut pas être vérifié. Utiliser `TrustServerCertificate=false` applique la validation du certificat, en plus d'être une bonne pratique.

Mise à jour du magasin d'approbations de votre application

Vous pouvez mettre à jour le Trust Store pour les applications qui utilisent Microsoft SQL Server. Pour obtenir des instructions, consultez Chiffrement de connexions spécifiques. Consultez également la section Configuration du client pour le chiffrement dans la documentation de Microsoft SQL Server.

Si vous utilisez un système d'exploitation autre que Microsoft Windows, consultez la documentation de distribution logicielle pourSSL/TLSimplementation pour plus d'informations sur l'ajout d'un nouveau certificat d'autorité de certification racine. Par exemple, Open SSL et Gnu TLS sont des options populaires. Utilisez la méthode d'implémentation pour renforcer la confiance au certificat de l'autorité de certification RDS racine. Microsoft fournit des instructions afin de configurer des certificats sur certains systèmes.

Pour plus d'informations sur le téléchargement du certificat racine, consultez .

Pour obtenir des exemples de scripts qui importent des certificats, consultez Exemple de script pour importer les certificats dans votre magasin d'approbations.

> **Note**
>
> Lors de la mise à jour du magasin d'approbations, vous pouvez conserver les certificats plus anciens en complément de l'ajout des nouveaux certificats.

Prise en charge du programme de conformité pour les instances de bases de données Microsoft SQL Server

AWS Les services concernés ont été entièrement évalués par un auditeur tiers et ont donné lieu à une certification, à une attestation de conformité ou à une autorisation d'exploitation (ATO). Pour

de plus amples informations, veuillez consulter les services AWS concernés par le programme de conformité.

Prise en charge de la loi HIPAA pour les instances de bases de données Microsoft SQL Server

Vous pouvez utiliser les bases de données Amazon RDS for Microsoft SQL Server afin de développer des applications conformes à la loi HIPAA. Vous pouvez stocker les informations relatives à la santé, y compris les données de santé protégées (PHI, Protected Health Information), selon les termes d'un accord de partenariat (BAA, Business Associate Agreement) avec AWS. Pour de plus amples informations, veuillez consulter HIPAA compliance (français non garanti).

Amazon RDS for SQL Server prend en charge la loi HIPAA pour les versions et éditions suivantes :

- Éditions SQL Server 2022 Enterprise, Standard et Web
- SQL Server 2019 Enterprise, Standard et Web Editions
- SQL Server 2017 Enterprise, Standard et Web Editions
- SQL Server 2016 Enterprise, Standard et Web Editions

Pour activer la prise en charge de la loi HIPAA sur votre instance de base de données, installez les trois composants suivants.

Composant	Détails
Audit	Pour configurer l'audit, définissez le paramètre `rds.sqlserver_audit` sur la valeur `fedramp_hipaa`. Si votre instance de base de données n'utilise pas déjà un groupe de paramètres de base de données personnalisé, vous devez créer un groupe de paramètres personnalisé et l'attacher à votre instance de base de données avant de pouvoir modifier le paramètre `rds.sqlserver_audit`. Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).
Chiffrement de transport	Pour configurer le chiffrement de transport, forcez toutes les connexions à votre instance de base de données pour qu'elles utilisent le protocole

Composant	Détails
	SSL (Secure Sockets Layer). Pour plus d'informations, consultez Forcer les connexions à votre instance de base de données à utiliser SSL.
Chiffrement au repos	Pour configurer le chiffrement au repos, vous disposez de deux options : 1. Si vous utilisez SQL Server Enterprise Edition 2016—2022 ou Standard Edition 2022, vous pouvez utiliser le chiffrement transparent des données (TDE) pour réaliser le chiffrement au repos. Pour de plus amples informations, veuillez consulter Support pour le chiffrement transparent des données sur le SQL serveur. 2. Vous pouvez configurer le chiffrement au repos à l'aide des clés de chiffrement AWS Key Management Service (AWS KMS). Pour de plus amples informations, veuillez consulter Chiffrement des ressources Amazon RDS.

Versions de Microsoft SQL Server sur Amazon RDS

Vous pouvez spécifier n'importe quelle version de Microsoft SQL Server actuellement prise en charge lorsque vous créez une instance de base de données. Vous pouvez spécifier la version majeure de Microsoft SQL Server (par exemple, Microsoft SQL Server 14.00), puis toute version mineure prise en charge pour la version majeure spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version prise en charge, généralement la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée.

Le tableau suivant indique les versions de SQL Server prises en charge pour toutes les éditions et toutes les AWS régions, sauf indication contraire.

> **Note**
>
> Vous pouvez également utiliser la describe-db-engine-versions AWS CLI commande pour voir la liste des versions prises en charge, ainsi que les valeurs par défaut pour les instances de base de données nouvellement créées. Vous pouvez consulter les versions principales de

vos bases de données SQL Server en exécutant la AWS CLI commande describe-db-major-engine-versions ou en utilisant l'opération d'API DBMajorEngineVersionsDescribe RDS.

Version majeure	Version mineure	API RDS **`EngineVersion`** et CLI **`engine-version`**
SQL Server 2022	16,00.4205,1 (0) CU2	`16.00.4205.1.v1`
	16,00.4195,2 () CU19	`16.00.4195.2.v1`
	16,00.4185,3 () CU18	`16.00.4185.3.v1`
	16,00.4175,1 () CU17	`16.00.4175.1.v1`
	16,00.4165,4 () CU16	`16.00.4165.4.v1`
	16,00.4150,1 () CU15	`16.00.4150.1.v1`
	16.00.4140.3 (RDA) CU14	`16.00.4140.3.v1`
	16,00.4135,4 () CU14	`16.00.4135.4.v1`
	16,00.4131,2 () CU13	`16.00.4131.2.v1`
	16,00.4125,3 () CU13	`16.00.4125.3.v1`
	16.00.4120.1 (RDA) CU12	`16.00.4120.1.v1`
	16,00,415,5 () CU12	`16.00.4115.5.v1`
	16,00.4105,2 () CU11	`16.00.4105.2.v1`
	16,00.4095,4 (0) CU1	`16.00.4095.4.v1`
	16,00.4085,2 () CU9	`16.00.4085.2.v1`
SQL Server 2019	15,00,435,7 () CU32	`15.00.4435.7.v1`
	15,00.4430.1 () CU32	`15.00.4430.1.v1`
	15,00.420,2 () CU31	`15.00.4420.2.v1`

Version majeure	Version mineure	API RDS **EngineVersion** et CLI **engine-version**
	15,00.4415,2 (0) CU3	15.00.4415.2.v1
	15.00.4410.1 (GDR) CU29	15.00.4410.1.v1
	15,00.4395,2 () CU28	15.00.4395.2.v1
	15,00.4390,2 () CU28	15.00.4390.2.v1
	15,00.4385,2 () CU28	15.00.4385.2.v1
	15,00.4382,1 () CU27	15.00.4382.1.v1
	15,00.4375,4 () CU27	15.00.4375.4.v1
	15,00.4365,2 () CU26	15.00.4365.2.v1
	15,00.435,3 () CU25	15.00.4355.3.v1
	15,00.4345,5 () CU24	15.00.4345.5.v1
	15,00.435.1 () CU23	15.00.4335.1.v1
	15,00.4322,2 () CU22	15.00.4322.2.v1
	15,00.4316,3 () CU21	15.00.4316.3.v1
	15,00.4312,2 (0) CU2	15.00.4312.2.v1
	15,00.4236,7 () CU16	15.00.4236.7.v1
	15,00.4198.2 () CU15	15.00.4198.2.v1
	15,00.4153,1 () CU12	15.00.4153.1.v1
	15,00.4073,23 () CU8	15.00.4073.23.v1
	15,00.4043,16 () CU5	15.00.4043.16.v1

Version majeure	Version mineure	API RDS **EngineVersion** et CLI **engine-version**
SQL Server 2017	14.00.3495.9 (RDA) CU31	14.00.3495.9.v1
	14.00.3485.1 (RDA) CU31	14.00.3485.1.v1
	14,00.3480,1 () CU31	14.00.3480.1.v1
	14,00.3475.1 () CU31	14.00.3475.1.v1
	14,00.3471,2 () CU31	14.00.3471.2.v1
	14,00.3465,1 () CU31	14.00.3465.1.v1
	14,00.3460,9 () CU31	14.00.3460.9.v1
	14,00.3451,2 (0) CU3	14.00.3451.2.v1
	14,00.3421,10 () CU27	14.00.3421.10.v1
	14.00.3401.7 () CU25	14.00.3401.7.v1
	14,00.3381,3 () CU23	14.00.3381.3.v1
	14,00.3356,20 () CU22	14.00.3356.20.v1
	14,00.3294,2 (0) CU2	14.00.3294.2.v1
	14,00.3281,6 () CU19	14.00.3281.6.v1

Version majeure	Version mineure	API RDS **EngineVersion** et CLI **engine-version**
SQL Server 2016	13.00.7055.9 (ALLEMAGNE)	`13.00.7055.9.v1`
	13.00.645,2 (ALLEMAGNE)	`13.00.6455.2.v1`
	13.00.6450.1 (ALLEMAGNE)	`13.00.6450.1.v1`
	13.00.645.1 (ALLEMAGNE)	`13.00.6445.1.v1`
	13.00.6441.1 (ALLEMAGNE)	`13.00.6441.1.v1`
	13.00.6435.1 (GDR)	`13.00.6435.1.v1`
	13.00.6430.49 (GDR)	`13.00.6430.49.v1`
	13.00.6419.1 (+ Hotfix) SP3	`13.00.6419.1.v1`
	13,00.6300,2 () SP3	`13.00.6300.2.v1`

Gestion des versions dans Amazon RDS

Amazon RDS comprend une gestion flexible des versions qui vous permet de contrôler quand et comment votre instance de base de données est corrigée ou mise à niveau. Cela vous permet d'effectuer les opérations suivantes pour votre moteur de base de données :

- Conserver la compatibilité avec les versions correctives du moteur de base de données.
- Tester les nouvelles versions correctives pour vérifier qu'elles fonctionnent avec votre application avant de les déployer en production.
- Planifier et effectuer des mises à niveau de version pour répondre à vos contrats de niveau de service et à vos exigences en termes de temps.

Application de correctifs de moteur Microsoft SQL Server dans Amazon RDS

Amazon RDS ajoute régulièrement des correctifs de base de données Microsoft SQL Server officiels à une version de moteur d'instance de base de données spécifique à Amazon RDS. Pour de plus

amples informations sur les correctifs Microsoft SQL Server de chaque version de moteur de base de données, veuillez consulter Versions et fonctions prises en charge sur Amazon RDS.

Actuellement, vous devez effectuer manuellement toutes les mises à niveau de moteur sur votre instance de base de données. Pour plus d'informations, consultez Mises à niveau du moteur de base de données Microsoft SQL Server.

Calendrier d'obsolescence pour les versions de moteur majeures de Microsoft SQL Server sur Amazon RDS

Le tableau suivant affiche le calendrier planifié de l'obsolescence des versions de moteur majeures de Microsoft SQL Server.

Date	Informations
9 juillet 2024	Microsoft arrêtera les mises à jour critiques de correctifs pour SQL Server 2014. Pour consultez Microsoft SQL Server 2014 dans la documentation Microsoft.
1er juin 2024	Amazon RDS prévoit de mettre fin à la prise en charge de Microsoft SQL Server 201 SQL Server. À ce moment-là, toutes les instances restantes seront planifiées pour ur SQL Server 2016 (dernière version mineure disponible). Pour plus d'informations, co prise en charge Amazon RDS for SQL Server pour les versions majeures de SQL Se Pour éviter une mise à niveau automatique depuis Microsoft SQL Server 2014, vous mise à niveau au moment qui vous convient. Pour de plus amples informations, veuil niveau de la version du moteur d'une instance de base de données.
12 juillet 2022	Microsoft arrêtera les mises à jour de correctifs critiques pour SQL Server 2012. Pour consultez Microsoft SQL Server 2012 dans la documentation Microsoft.
1 juin 2022	Amazon RDS prévoit de mettre fin à la prise en charge de Microsoft SQL Server 201 SQL Server. À ce moment-là, toutes les instances restantes seront planifiées pour ur SQL Server 2014 (dernière version mineure disponible). Pour plus d'informations, co prise en charge Amazon RDS for SQL Server pour les versions majeures SQL Serve

Date	Informations
	Pour éviter une mise à niveau automatique depuis Microsoft SQL Server 2012, vous mise à niveau au moment qui vous convient. Pour plus d'informations, consultez Mise du moteur d'une instance de base de données.
1er septembre 2021	Amazon RDS commence à désactiver la création de nouvelles instances de base de Server à l'aide de Microsoft SQL Server 2012. Pour plus d'informations, consultez An charge Amazon RDS for SQL Server pour les versions majeures SQL Server 2012.
12 juillet 2019	L'équipe Amazon RDS a rendu obsolète la prise en charge de Microsoft SQL Server Les instances restantes de Microsoft SQL Server 2008 R2 sont migrées vers SQL Se version mineure disponible). Pour éviter une mise à niveau automatique depuis Microsoft SQL Server 2008 R2, vo mise à niveau au moment qui vous convient. Pour plus d'informations, consultez Mise du moteur d'une instance de base de données.
25 avril 2019	Avant fin avril 2019, vous ne pourrez plus créer de nouvelles instances de base de do for SQL Server à l'aide de Microsoft SQL Server 2008 R2.

Fonctionnalités SQL de Microsoft Server sur Amazon RDS

Les versions de SQL serveur prises en charge sur Amazon RDS incluent les fonctionnalités suivantes. En général, une version inclut également les fonctionnalités des versions précédentes, sauf indication contraire dans la documentation Microsoft.

Rubriques

- Fonctionnalités SQL de Microsoft Server 2022
- Fonctionnalités SQL de Microsoft Server 2019
- Fonctionnalités SQL de Microsoft Server 2017
- Fonctionnalités SQL de Microsoft Server 2016
- Fin du support de Microsoft SQL Server 2014 sur Amazon RDS
- Fin du support de Microsoft SQL Server 2012 sur Amazon RDS
- Fin du support de Microsoft SQL Server 2008 R2 sur Amazon RDS

- Modifier le support de capture de données pour les instances de base de données Microsoft SQL Server
- Fonctions non prises en charge et fonctions avec prise en charge limitée

Fonctionnalités SQL de Microsoft Server 2022

SQLServer 2022 inclut de nombreuses nouvelles fonctionnalités, telles que les suivantes :

- Optimisation du plan sensible aux paramètres : autorise plusieurs plans mis en cache pour une seule instruction paramétrée, ce qui réduit potentiellement les problèmes liés à l'analyse des paramètres.
- SQLServer Ledger : permet de prouver de manière cryptographique que vos données n'ont pas été modifiées sans autorisation.
- Initialisation instantanée des fichiers pour les événements de croissance du journal des transactions : accélère l'exécution des événements de croissance du journal jusqu'à 64 Mo, y compris pour les bases de données activées. TDE
- Améliorations apportées à la simultanéité des loquets de page dans le système : réduit la contention liée au verrouillage de page lors de l'allocation et de la désallocation des pages de données et des extensions, ce qui améliore considérablement les performances pour les charges de travail lourdes. `tempdb`

Pour obtenir la liste complète des fonctionnalités de SQL Server 2022, consultez la section Nouveautés de SQL Server 2022 (16.x) dans la documentation Microsoft.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctions non prises en charge et fonctions avec prise en charge limitée.

Fonctionnalités SQL de Microsoft Server 2019

SQLServer 2019 inclut de nombreuses nouvelles fonctionnalités, telles que les suivantes :

- Restauration accélérée de la base de données (ADR) : réduit le temps de restauration après un crash après un redémarrage ou une annulation de transaction prolongée.
- Traitement intelligent des requêtes (IQP) :
  - Rétroaction d'octroi de mémoire en mode ligne : correction automatique des octrois excessifs entraînant un gaspillage de mémoire et une réduction de la concurrence.

  - Mode batch sur rowstore : permet l'exécution du mode batch pour les charges de travail d'analyse sans nécessiter d'index columnstore.
  - Compilation différée des variables de table : amélioration de la qualité du plan et des performances globales des requêtes faisant référence aux variables de table.
- Performances intelligentes :
  - `OPTIMIZE_FOR_SEQUENTIAL_KEY`Option d'index : amélioration du débit des insertions à forte concurrence dans les index.
  - Amélioration de l'évolutivité des points de contrôle indirects : aide les bases de données à supporter de lourdes charges de travailDML.
  - Mises à jour simultanées de l'espace libre sur les pages (PFS) : permet de les gérer en tant que loquet partagé plutôt qu'en tant que loquet exclusif.
- Surveillance des améliorations :
  - `WAIT_ON_SYNC_STATISTICS_REFRESH`Type d'attente : affiche le temps accumulé au niveau de l'instance consacré aux opérations d'actualisation des statistiques synchrones.
  - Configurations définies par base de données : inclure `LIGHTWEIGHT_QUERY_PROFILING` et `LAST_QUERY_PLAN_STATS`.
  - Fonctions de gestion dynamiques (DMFs) — Incluez `sys.dm_exec_query_plan_stats` et`sys.dm_db_page_info`.
- Avertissements de troncature verbeux : le message d'erreur de troncation des données inclut par défaut les noms de table et de colonne et la valeur tronquée.
- Création d'index en ligne pouvant être reprise : dans SQL Server 2017, seule la reconstruction d'index en ligne pouvant être reprise est prise en charge.

Pour obtenir la liste complète des fonctionnalités de SQL Server 2019, consultez la section Nouveautés de SQL Server 2019 (15.x) dans la documentation Microsoft.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctions non prises en charge et fonctions avec prise en charge limitée.

Fonctionnalités SQL de Microsoft Server 2017

SQLServer 2017 inclut de nombreuses nouvelles fonctionnalités, telles que les suivantes :

- Traitement des requêtes adaptives
- Correction automatique du plan (fonction de réglage automatique)

- GraphDB
- Reconstructions d'index pouvant être interrompues

Pour obtenir la liste complète des fonctionnalités de SQL Server 2017, consultez la section Nouveautés de SQL Server 2017 dans la documentation Microsoft.

Pour obtenir la liste des fonctions non prises en charge, consultez Fonctions non prises en charge et fonctions avec prise en charge limitée.

Fonctionnalités SQL de Microsoft Server 2016

Amazon RDS prend en charge les fonctionnalités suivantes de SQL Server 2016 :

- Toujours chiffré
- JSONSupport
- Analytique opérationnelle
- Magasin de requêtes
- Tables temporelles

Pour obtenir la liste complète des fonctionnalités de SQL Server 2016, consultez la section Nouveautés de SQL Server 2016 dans la documentation Microsoft.

Fin du support de Microsoft SQL Server 2014 sur Amazon RDS

SQLLe support de Server 2014 a atteint la fin du support sur AmazonRDS.

RDSmet à niveau toutes les instances de base de données existantes qui utilisent encore SQL Server 2014 vers la dernière version mineure de SQL Server 2016. Pour de plus amples informations, veuillez consulter Gestion des versions dans Amazon RDS.

Fin du support de Microsoft SQL Server 2012 sur Amazon RDS

SQLLe support de Server 2012 a atteint la fin du support sur AmazonRDS.

RDSmet à niveau toutes les instances de base de données existantes qui utilisent encore SQL Server 2012 vers la dernière version mineure de SQL Server 2016. Pour de plus amples informations, veuillez consulter Gestion des versions dans Amazon RDS.

Fin du support de Microsoft SQL Server 2008 R2 sur Amazon RDS

SQLLe support de Server 2008 R2 a atteint la fin du support sur AmazonRDS.

RDSmet à niveau toutes les instances de base de données existantes qui utilisent encore SQL Server 2008 R2 vers la dernière version mineure de SQL Server 2012. Pour de plus amples informations, veuillez consulter Gestion des versions dans Amazon RDS.

Modifier le support de capture de données pour les instances de base de données Microsoft SQL Server

Amazon RDS prend en charge la capture des données de modification (CDC) pour vos instances de base de données exécutant Microsoft SQL Server. CDCcapture les modifications apportées aux données de vos tables et stocke les métadonnées relatives à chaque modification auxquelles vous pourrez accéder ultérieurement. Pour de plus amples informations, veuillez consulter Modifier la capture de données dans la documentation de Microsoft.

Amazon RDS prend en CDC charge les éditions et versions de SQL serveur suivantes :

- Microsoft SQL Server Enterprise Edition (toutes les versions)
- Édition standard SQL de Microsoft Server :
  - 2022
  - 2019
  - 2017
  - Version 2016 SP1 CU2 13.00.4422.0 et versions ultérieures

Pour les utiliser CDC avec vos instances Amazon RDS DB, activez ou désactivez d'abord CDC au niveau de la base de données à l'aide des procédures stockées RDS fournies. Ensuite, tout utilisateur ayant le `db_owner` rôle pour cette base de données peut utiliser les procédures stockées Microsoft natives pour contrôler cette CDC base de données. Pour de plus amples informations, veuillez consulter Utilisation de la capture des données de modification pour Amazon RDS for SQL Server.

Vous pouvez utiliser CDC et activer AWS Database Migration Service la réplication continue à partir des instances de base de données SQL du serveur.

Fonctions non prises en charge et fonctions avec prise en charge limitée

Les fonctionnalités Microsoft SQL Server suivantes ne sont pas prises en charge sur Amazon RDS :

- Sauvegarde dans Microsoft Azure Blob Storage
- Extension du pool de mémoires tampons
- Politiques de mots de passe personnalisées
- Data Quality Services
- Copie des journaux de transaction de base de données
- Instantanés de base de données (Amazon RDS prend uniquement en charge les instantanés d'instance de base de données)
- Procédures stockées étendues, y compris xp_cmdshell
- Prise en charge de FILESTREAM
- Tables de fichiers
- Machine Learning and R Services (exige un accès de système d'exploitation pour l'installer)
- Plans de maintenance
- Performance Data Collector
- Gestion basée sur la politique
- PolyBase
- Réplication
- Resource Governor

  Dans un environnement multi-tenant, nous vous recommandons de comprendre les exigences et les considérations en matière de performances afin de minimiser les problèmes liés à la concurrence entre les charges de travail pour les ressources.
- Déclencheurs de niveau serveur
- Points de terminaison Service Broker
- Stretch Database
- TRUSTWORTHYpropriété de base de données (nécessite le rôle d'administrateur système)
- SQLPoints de terminaison T (toutes les opérations utilisées CREATE ENDPOINT ne sont pas disponibles)
- Services de données WCF

Les fonctionnalités Microsoft SQL Server suivantes sont prises en charge de manière limitée sur Amazon RDS :

- Requêtes distribuées/serveurs liés. Pour plus d'informations, consultez Implémenter des serveurs liés avec Amazon RDS pour Microsoft SQL Server.
- Langage d'exécution commun (CLR). Activé RDS pour SQL Server 2016 et les versions antérieures, CLR il est pris en charge en SAFE mode et en utilisant uniquement des bits d'assemblage. CLRn'est pas pris en RDS charge sur SQL Server 2017 et les versions ultérieures. Pour plus d'informations, consultez Intégration du Common Runtime Language dans la documentation Microsoft.
- Liez les serveurs OLEDB à Oracle dans Amazon RDS for SQL Server. Pour de plus amples informations, veuillez consulter Prise en charge des serveurs liés avec Oracle OLEDB dans Amazon RDS for SQL Server.

Les fonctionnalités suivantes ne sont pas prises en charge sur Amazon RDS avec SQL Server 2022 :

- Suspendre la base de données pour un
- Source de données externe
- Backup et restauration vers un stockage d'objets compatible S3
- Intégration à un magasin d'objets
- TLS1.3 et TDS MS-8.0
- Backup, compression et déchargement avec QAT
- SQLServices d'analyse de serveurs (SSAS)
- Mise en miroir de bases de données avec déploiements multi-AZ. SQL Server Always On est la seule méthode prise en charge pour les déploiements multi-AZ.

Déploiements multi-AZ à l'aide de la mise en miroir de bases de données ou des groupes de disponibilité AlwaysOn Microsoft SQL Server

Amazon RDS prend en charge les déploiements multi-AZ pour les instances de base de données exécutant Microsoft SQL Server en utilisant SQL Server Database Mirroring (DBM) ou Always On Availability Groups (). AGs Les déploiements Multi-AZ améliorent la disponibilité, la durabilité des

donnés et la tolérance aux pannes pour les instances de bases de données. En cas de maintenance planifiée de la base de données ou d'interruption de service imprévue, Amazon RDS bascule automatiquement vers la réplique up-to-date secondaire afin que les opérations de base de données puissent reprendre rapidement sans intervention manuelle. Les instances principales et secondaires utilisent le même point de terminaison, dont l'adresse réseau physique est transférée vers le réplica secondaire passif dans le cadre du processus de basculement. Vous n'avez pas à reconfigurer votre application lorsqu'un basculement se produit.

Amazon RDS gère le basculement en surveillant activement votre déploiement multi-AZ et en initiant un basculement en cas de problème avec l'élément principal. Le basculement ne se produit que si les instances principales et de secours sont complètement synchronisées. Amazon RDS maintient activement votre déploiement Multi-AZ en réparant automatiquement les instances de base de données défectueuses et en rétablissant une réplication synchrone. Vous ne devez rien gérer. Amazon RDS gère automatiquement les instances principales, les instances témoin et les instances de secours. Lorsque vous configurez déploiements multi-AZ SQL Server, RDS configure des instances secondaires passives pour toutes les bases de données de l'instance.

Pour plus d'informations, consultez Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.

Utilisation de Transparent Data Encryption pour chiffrer les données au repos

Amazon RDS prend en charge le chiffrement TDE (Transparent Data Encryption) de Microsoft SQL Server, qui chiffre les données stockées en toute transparence. Amazon RDS utilise des groupes d'options pour activer et configurer ces fonctions. Pour plus d'informations sur l'option TDE, consultez Support pour le chiffrement transparent des données sur le SQL serveur.

Fonctions et procédures stockées Amazon RDS for Microsoft SQL Server

La liste suivante répertorie les fonctions Amazon RDS et les procédures stockées qui aident à automatiser les tâches SQL Server.

Type de tâche	Procédure ou fonction	Utilisation
Tâches administratives	`rds_drop_database`	Suppression d'une base de données dans une instance de base de données Amazon RDS pour Microsoft SQL Server
	`rds_failover_time`	Déterminer l'heure du dernier basculement d'Amazon RDS for Server SQL
	`rds_manage_view_db_permission`	Refuser ou autoriser l'affichage des noms de base de données pour Amazon RDS for SQL Server
	`rds_modify_db_name`	Modification du nom d'une base de données Amazon RDS for Microsoft SQL Server dans le cadre d'un déploiement multi-AZ
	`rds_read_error_log`	Affichage des journaux des erreurs et des agents
	`rds_set_configuration`	Cette opération permet de définir diverses configurations d'instance de base de données : • Capture de données modifiées (CDC) pour les instances multi-AZ • Configuration de la période de rétention pour les fichiers de trace et de vidage • Compression des fichiers de sauvegarde
	`rds_set_database_online`	Transition d'une base de données Amazon RDS for SQL Server de OFFLINE vers ONLINE
	`rds_set_system_dat`	Activation de la réplication des tâches de l'agent SQL Server

Type de tâche	Procédure ou fonction	Utilisation

	`abase_sync_objects`
	
	`rds_fn_get_system_database_sync_objects`
	
	`rds_fn_server_object_last_sync_time`	

	`rds_show_configuration`	Pour voir les valeurs définies à l'aide de `rds_set_configuration`, consultez ces rubriques : • Capture de données modifiées (CDC) pour les instances multi-AZ • Configuration de la période de rétention pour les fichiers de trace et de vidage
	`rds_shrink_tempdbfile`	Réduction de la base de données tempdb
Capture des données de modification (CDC)	`rds_cdc_disable_db`	Désactivation CDC
	`rds_cdc_enable_db`	Activation de la CDC

Type de tâche	Procédure ou fonction	Utilisation
Messagerie de base de données	`rds_fn_sysmail_allitems`	Affichage des messages, des journaux et des pièces jointes
	`rds_fn_sysmail_event_log`	Affichage des messages, des journaux et des pièces jointes
	`rds_fn_sysmail_mailattachments`	Affichage des messages, des journaux et des pièces jointes
	`rds_sysmail_control`	Cette opération est utilisée pour démarrer et arrêter la file d'attente de messagerie : • Lancement de la file d'attente de messagerie • Arrêt de la file d'attente de messagerie
	`rds_sysmail_delete_mailitems_sp`	Suppression de messages
Sauvegarde et restauration natives	`rds_backup_database`	Sauvegarde d'une base de données
	`rds_cancel_task`	Annulation d'une tâche
	`rds_finish_restore`	Finalisation d'une restauration de base de données

Type de tâche	Procédure ou fonction	Utilisation
	`rds_restore_database`	Restauration d'une base de données
	`rds_restore_log`	Restauration d'un journal
Transfert de fichiers Amazon S3	`rds_delete_from_filesystem`	Supprimer des fichiers sur l'RDSinstance de base de données
	`rds_download_from_s3`	Téléchargement de fichiers depuis un compartiment Amazon S3 vers une instance de base de données de SQL serveur
	`rds_gather_file_details`	Lister les fichiers sur l'RDSinstance de base de données
	`rds_upload_to_s3`	Téléchargement de fichiers depuis une instance de base de données de SQL serveur vers un compartiment Amazon S3
Microsoft Distributed Transaction Coordinator (MSDTC)	`rds_msdtc_transaction_tracing`	Utilisation du suivi des transactions
SQL Server Audit	`rds_fn_get_audit_file`	Consultation des journaux d'audit

Type de tâche	Procédure ou fonction	Utilisation

Transparent Data Encryption	`rds_backup_tde_certificate`
	
	`rds_drop_tde_certificate`
	
	`rds_restore_tde_certificate`
	
	`rds_fn_list_user_tde_certificates`
		Support pour le chiffrement transparent des données sur le SQL serveur

Type de tâche	Procédure ou fonction	Utilisation
Microsoft Business Intelligence (MSBI)	`rds_msbi_task`	Cette opération est utilisée avec SQL Server Analysis Services (SSAS) : • Déploiement SSAS de projets sur Amazon RDS • Ajout d'un utilisateur de domaine en tant qu'administrateur de base de données • Sauvegarde d'une SSAS base de données • Restauration d'une SSAS base de données Cette opération est également utilisée avec SQL Server Integration Services (SSIS) : • Autorisations administratives sur SSISDB • Déploiement d'un projet SSIS Cette opération est également utilisée avec SQL Server Reporting Services (SSRS) : • Octroi de l'accès aux utilisateurs du domaine • Révocation des autorisations de niveau système
	`rds_fn_task_status`	Cette opération affiche l'état des tâches MSBI : • SSAS : Surveillance de l'état d'une tâche de déploiement • SSIS : Surveillance de l'état d'une tâche de déploiement • SSRS : Surveillance du statut d'une tâche
SSIS	`rds_drop_ssis_database`	Suppression de la SSISDB base de données

Type de tâche	Procédure ou fonction	Utilisation
	`rds_sqlagent_proxy`	Création d'un SSIS proxy
SSRS	`rds_drop_ssrs_databases`	Supprimer les SSRS bases de données

Fuseau horaire local pour les instances de base de données Microsoft SQL Server

Le fuseau horaire d'une instance Amazon RDS DB exécutant Microsoft SQL Server est défini par défaut. La valeur par défaut actuelle est le temps universel coordonné (UTC). Vous pouvez définir le fuseau horaire de votre instance de base de données à un fuseau horaire local, correspondant à celui de vos applications.

Vous définissez le fuseau horaire lorsque vous créez votre instance de base de données. Vous pouvez créer votre instance de base de données à l'AWS Management Consoleaide de l'reateDBInstanceaction Amazon RDS API C ou de la AWS CLI create-db-instancecommande.

Si votre instance de base de données fait partie d'un déploiement multi-AZ (à l'aide d'un SQL serveur DBM ouAGs), en cas de basculement, votre fuseau horaire reste le fuseau horaire local que vous avez défini. Pour de plus amples informations, veuillez consulter Déploiements multi-AZ à l'aide de la mise en miroir de bases de données ou des groupes de disponibilité AlwaysOn Microsoft SQL Server.

Lorsque vous demandez une point-in-time restauration, vous spécifiez l'heure à laquelle la restauration doit être effectuée. L'heure est affichée dans votre fuseau horaire local. Pour plus d'informations, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Ci-après les restrictions affectant la définition du fuseau horaire local sur votre instance de base de données :

- Vous ne pouvez pas modifier le fuseau horaire d'une instance de base de données de SQL serveur existante.
- Vous ne pouvez pas restaurer un instantané à partir d'une instance de base de données dans un fuseau horaire dans une instance de base de données d'un autre fuseau horaire.
- Nous vous recommandons vivement de ne pas restaurer de fichier de sauvegarde d'un fuseau horaire dans un autre fuseau horaire. Si vous restaurez un fichier de sauvegarde d'un fuseau horaire dans un autre fuseau horaire, vous devez auditer vos requêtes et vos applications afin de déterminer les effets du changement de fuseau horaire. Pour plus d'informations, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Fuseaux horaires pris en charge

Vous pouvez définir votre fuseau horaire local avec l'une des valeurs du tableau suivant.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Afghanistan	(UTC+ 04H30)	Kaboul	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Alaska	(UTC— 09H00)	Alaska	
Heure normale Aléoutiennes	(UTC— 10H00)	Îles Aléoutiennes	
Heure normale de l'Altaï	(UTC+ 07h00)	Barnaul, Gorno-Altaysk	
Heure normale arabe	(UTC+ 03:00)	Koweït, Riyad	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale des Emirats Arabe Unis	(UTC+ 04:00)	Abou Dhabi, Mascate	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale Arabie saoudite	(UTC+ 03:00)	Bagdad	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Argentine	(UTC— 03H00)	Ville de Buenos Aires	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Astrakhan	(UTC+ 04:00)	Astrakhan, Oulianovsk	
Heure normale de l'Atlantique	(UTC— 04H00)	Heure de l'Atlanti que (Canada)	
AUSHeure normale du Centre	(UTC+ 09H30)	Darwin	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Australie centrale	(UTC+ 08h45)	Eucla	
AUSHeure normale de l'Est	(UTC+ 10 H)	Canberra, Melbourne, Sydney	
Heure normale d'Azerbaïdjan	(UTC+ 04:00)	Bakou	
Heure normale des Açores	(UTC— 01H00)	Açores	
Heure normale de Bahia	(UTC— 03H00)	Salvador	
Heure normale du Banglades h	(UTC+ 06:00)	Dacca	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de Biélorussie	(UTC+ 03:00)	Minsk	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Bougainville	(UTC+ 11 h)	Île de Bougainville	
Heure normale du Canada central	(UTC— 06H00)	Saskatchewan	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Cap-Vert	(UTC— 01H00)	Cap-Vert	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Caucase	(UTC+ 04:00)	Erevan	
Heure normale de l'Australie centrale	(UTC+ 09H30)	Adélaïde	
Heure normale de l'Amérique centrale	(UTC— 06H00)	Amérique centrale	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Asie centrale	(UTC+ 06:00)	Astana	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Brésil central	(UTC— 04H00)	Cuiabá	
Heure normale de l'Europe centrale	(UTC+ 01:00)	Belgrade, Bratislava, Budapest, Ljubljana, Prague	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de l'Europe centrale	(UTC+ 01:00)	Sarajevo, Skopje, Varsovie, Zagreb	
Heure normale du Pacifique central	(UTC+ 11 h)	Îles Salomon, Nouvelle-Calédonie	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Centre	(UTC— 06H00)	Heure centrale (États-Unis et Canada)	
Heure normale du Centre (Mexique)	(UTC— 06H00)	Guadalajara, Mexico, Monterrey	
Heure normale des îles Chatham	(UTC+ 12h45)	Îles Chatham	
Heure normale de Chine	(UTC+ 08h00)	Pékin, Chongqing, Hong Kong, Urumqi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Cuba	(UTC— 05H00)	La Havane	
Heure normale de la ligne de changement de date	(UTC— 12H00)	Ligne de changement de date internationale Ouest	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique de l'Est	(UTC+ 03:00)	Nairobi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Australie de l'Est	(UTC+ 10 H)	Brisbane	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Europe de l'Est	(UTC+ 02:00)	Chi#inău	
Heure normale d'Amérique du Sud est	(UTC— 03H00)	Brasilia	
Heure normale de l'île de Pâques	(UTC— 06H00)	Île de Pâques	
Heure normale de l'Est	(UTC— 05H00)	Heure de l'Est (États-Unis et Canada)	
Heure normale de l'Est (Mexique)	(UTC— 05H00)	Chetumal	
Heure normale de l'Égypte	(UTC+ 02:00)	Le Caire	
Heure normale d'Iekater inbourg	(UTC+ 05:00)	Iekaterinbourg	
Heure normale des Fidji	(UTC+ 12H00)	Fidji	
FLEHeure standard	(UTC+ 02:00)	Helsinki, Kiev, Riga, Sofia, Tallinn, Vilnius	
Heure normale de Géorgie	(UTC+ 04:00)	Tbilisi	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
GMTHeure standard	(UTC)	Dublin, Edimbourg, Lisbonne, Londres	Ce fuseau horaire n'est pas le même que l'heure moyenne de Greenwich (GMT). Ce fuseau horaire respecte l'heure d'été.
Heure normale du Groenland	(UTC— 03H00)	Groenland	
Heure normale de Greenwich	(UTC)	Monrovia, Reykjavik	Ce fuseau horaire ne respecte pas l'heure d'été.
GTBHeure standard	(UTC+ 02:00)	Athènes, Bucarest	
Heure normale d'Haïti	(UTC— 05H00)	Haïti	
Heure normale de Hawaï	(UTC— 10H00)	Hawaii	
Heure normale d'Inde	(UTC+ 05H30)	Chennai, Calcutta, Mumbai, New Delhi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Iran	(UTC+ 03H30)	Téhéran	
Heure normale d'Israël	(UTC+ 02:00)	Jérusalem	
Heure normale de Jordanie	(UTC+ 02:00)	Amman	
Heure normale de Kaliningrad	(UTC+ 02:00)	Kaliningrad	
Heure normale du Kamtchatka	(UTC+ 12H00)	Petropavlovsk-Kamchatsky – Ancienne	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale de Corée	(UTC+ 09h00)	Séoul	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Libye	(UTC+ 02:00)	Tripoli	
Heure normale des îles de la Ligne	(UTC+ 14h00)	Île Christmas	
Heure normale de l'île Lord Howe	(UTC+ 10 H 30)	Île Lord Howe	
Heure normale de Magadan	(UTC+ 11 h)	Magadan	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale en Magallanes	(UTC— 03H00)	Punta Arenas	
Heure normale des Marquises	(UTC— 9 H 30)	Îles Marquises	
Heure normale de Maurice	(UTC+ 04:00)	Port Louis	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Moyen-Orient	(UTC+ 02:00)	Beyrouth	
Heure normale de Montevideo	(UTC— 03H00)	Montevideo	
Heure normale du Maroc	(UTC+ 01:00)	Casablanca	
Heure normale des Rocheuses	(UTC— 07h00)	Heure des Rocheuses (États-Unis et Canada)	
Heure normale des Rocheuses (Mexique)	(UTC— 07h00)	Chihuahua, La Paz, Mazatlán	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Myanmar	(UTC+ 06H30)	Yangon (Rangoun)	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Asie centrale nord	(UTC+ 07h00)	Novossibirsk	
Heure normale de Namibie	(UTC+ 02:00)	Windhoek	
Heure normale du Népal	(UTC+ 05:45)	Katmandou	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Nouvelle-Zélande	(UTC+ 12H00)	Auckland, Wellington	
Heure normale de Terre-Neuve	(UTC— 03H30)	Terre-Neuve	
Heure normale de l'île Norfolk	(UTC+ 11 h)	Île Norfolk	
Heure normale de l'Asie du Nord-Est	(UTC+ 08h00)	Irkoutsk	
Heure normale de l'Asie du Nord	(UTC+ 07h00)	Krasnoïarsk	
Heure normale de la Corée du Nord	(UTC+ 09h00)	Pyongyang	
Heure normale d'Omsk	(UTC+ 06:00)	Omsk	
Heure normale du Pacifique	(UTC— 03H00)	Santiago	
Heure normale du Pacifique	(UTC— 08H00)	Heure du Pacifique (États-Unis et Canada)	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Pacifique (Mexique)	(UTC— 08H00)	Basse-Californie	
Heure normale du Pakistan	(UTC+ 05:00)	Islamabad, Karachi	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Paraguay	(UTC— 04H00)	Asunción	
Heure normale Romance	(UTC+ 01:00)	Bruxelles, Copenhague, Madrid, Paris	
Fuseau horaire 10 Russie	(UTC+ 11 h)	Chokurdakh	
Fuseau horaire 11 Russie	(UTC+ 12H00)	Anadyr, Petropavl ovsk-Kamchatsky	
Fuseau horaire 3 Russie	(UTC+ 04:00)	Izhevsk, Samara	
Heure normale de Russie	(UTC+ 03:00)	Moscou, Saint-Pét ersbourg, Volgograd	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Est AS	(UTC— 03H00)	Cayenne, Fortaleza	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Pacifique	(UTC— 05H00)	Bogota, Lima, Quito, Rio Branco	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Ouest AS	(UTC— 04H00)	Georgetown, La Paz, Manaus, San Juan	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale Saint-Pierre	(UTC— 03H00)	Saint-Pierre-et-Miquelon	
Heure normale de Sakhaline	(UTC+ 11 h)	Sakhaline	
Heure normale des Samoa	(UTC+ 13h00)	Samoa	
Heure normale de Sao Tomé	(UTC+ 01:00)	Sao Tomé	
Heure normale de Saratov	(UTC+ 04:00)	Saratov	
Heure normale de l'Asie du Sud-Est	(UTC+ 07h00)	Bangkok, Hanoï, Djakarta	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Singapour	(UTC+ 08h00)	Kuala Lumpur, Singapour	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique du Sud	(UTC+ 02:00)	Harare, Pretoria	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Sri Lanka	(UTC+ 05H30)	Sri Jayawardenepura	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale du Soudan	(UTC+ 02:00)	Khartoum	
Heure normale de Syrie	(UTC+ 02:00)	Damas	
Heure normale de Taipei	(UTC+ 08h00)	Taipei	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Tasmanie	(UTC+ 10 H)	Hobart	

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale du Tocantins	(UTC— 03H00)	Araguaina	
Heure normale de Tokyo	(UTC+ 09h00)	Osaka, Sapporo, Tokyo	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Tomsk	(UTC+ 07h00)	Tomsk	
Heure normale des Tonga	(UTC+ 13h00)	Nuku'alofa	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de la Transbaïkalie	(UTC+ 09h00)	Tchita	
Heure normale de Turquie	(UTC+ 03:00)	Istanbul	
Heure normale des îles Turques-et-Caïques	(UTC— 05H00)	Turques-et-Caïques	
Heure normale d'Oulan-Bator	(UTC+ 08h00)	Oulan-Bator	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de l'Est	(UTC— 05H00)	Indiana (Est)	
Heure normale des Rocheuses	(UTC— 07h00)	Arizona	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC	UTC	Temps universel coordonné	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
UTC—02	(UTC— 02H00)	Temps universel coordonné–02	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC—08	(UTC— 08H00)	Temps universel coordonné–08	
UTC—09	(UTC— 09H00)	Temps universel coordonné–09	
UTC—11	(UTC— 11 H 00)	Temps universel coordonné–11	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC+12	(UTC+ 12H00)	Temps universel coordonné+12	Ce fuseau horaire ne respecte pas l'heure d'été.
UTC+13	(UTC+ 13h00)	Temps universel coordonné+13	
Heure normale du Venezuela	(UTC— 04H00)	Caracas	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Vladivostok	(UTC+ 10 H)	Vladivostok	
Heure normale de Volgograd	(UTC+ 04:00)	Volgograd	
Heure normale d'Australie de l'Ouest	(UTC+ 08h00)	Perth	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale d'Afrique centrale ouest	(UTC+ 01:00)	Afrique centrale de l'Ouest	Ce fuseau horaire ne respecte pas l'heure d'été.

Fuseau horaire	Décalage horaire standard	Description	Remarques
Heure normale d'Europe de l'ouest	(UTC+ 01:00)	Amsterdam, Berlin, Berne, Rome, Stockholm, Vienne	
Heure normale de Mongolie de l'Ouest	(UTC+ 07h00)	Hovd	
Heure normale de l'Asie de l'Est	(UTC+ 05:00)	Achgabat, Tachkent	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Cisjordanie	(UTC+ 02:00)	Gaza, Hébron	
Heure normale du Pacifique Ouest	(UTC+ 10 H)	Guam, Port Moresby	Ce fuseau horaire ne respecte pas l'heure d'été.
Heure normale de Iakoutsk	(UTC+ 09h00)	Iakoutsk	

Gestion des licences Microsoft SQL Server sur Amazon RDS

Lorsque vous définissez une instance de base de données Amazon RDS pour Microsoft SQL Server, la licence logicielle est incluse.

Cela signifie que vous n'avez pas besoin d'acheter séparément des licences SQL Server. AWS détient la licence du logiciel de base de données SQL Server. La tarification d'Amazon RDS inclut les licences de logiciels, les ressources matérielles sous-jacentes et les capacités de gestion d'Amazon RDS.

Amazon RDS prend en charge les éditions Microsoft SQL Server suivantes :

- Enterprise
- Standard
- Web
- Express

> **Note**
>
> La gestion des licences pour SQL Server Web Edition prend en charge uniquement les pages Web, les sites Web, les applications Web et les services Web bénéficiant d'un accès public et Internet. Ce niveau de prise en charge est obligatoire pour la conformité avec les droits d'utilisation de Microsoft. Pour plus d'informations, consultez Conditions de service AWS.

Amazon RDS prend en charge les déploiements Multi-AZ pour les instances de base de données exécutant Microsoft SQL Server à l'aide de la mise en miroir de bases de données (DBM) ou des groupes de disponibilité (AG) AlwaysOn. Il n'y a aucune exigence de licence supplémentaire pour les déploiements multi-AZ. Pour plus d'informations, consultez Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.

Restauration des instances de bases de données résiliées faute de licence

Amazon RDS prend des instantanés des instances de base de données résiliées faute de licence. Si votre instance est résiliée en raison de problèmes de licences, vous pouvez la restaurer depuis l'instantané vers une nouvelle instance de base de données. Les nouvelles instances de base de données incluent une licence.

Pour plus d'informations, consultez Restauration d'instances de base de données dont la licence est terminée pour Amazon for Server RDS SQL.

Développement et test

En raison des exigences concernant les licences, nous ne pouvons pas fournir l'édition SQL Server Developer dans Amazon RDS. Vous pouvez utiliser l'édition Express à des fins de développement, des tests et pour répondre à des besoins autre que la production. Toutefois, si vous avez besoin de toutes les fonctionnalités d'une installation de SQL Server au niveau de l'entreprise pour le développement, vous pouvez télécharger et installer SQL Server Developer Edition sur RDS Custom for SQL Server à l'aide d'un CEV avec BYOM. Pour plus d'informations, consultez Préparation d'une version CEV à l'aide du modèle Bring Your Own Media (BYOM). Aucune infrastructure dédiée n'est requise pour l'édition Développeur. En utilisant votre propre, vous pouvez également accéder à d'autres fonctions de programmabilité qui ne sont pas accessibles dans Amazon RDS. Pour plus d'informations sur la différence entre les éditions de SQL Server, consultez la section Éditions et fonctionnalités prises en charge de SQL Server 2019 dans la documentation Microsoft.

Connexion à votre instance de base de données Microsoft SQL Server

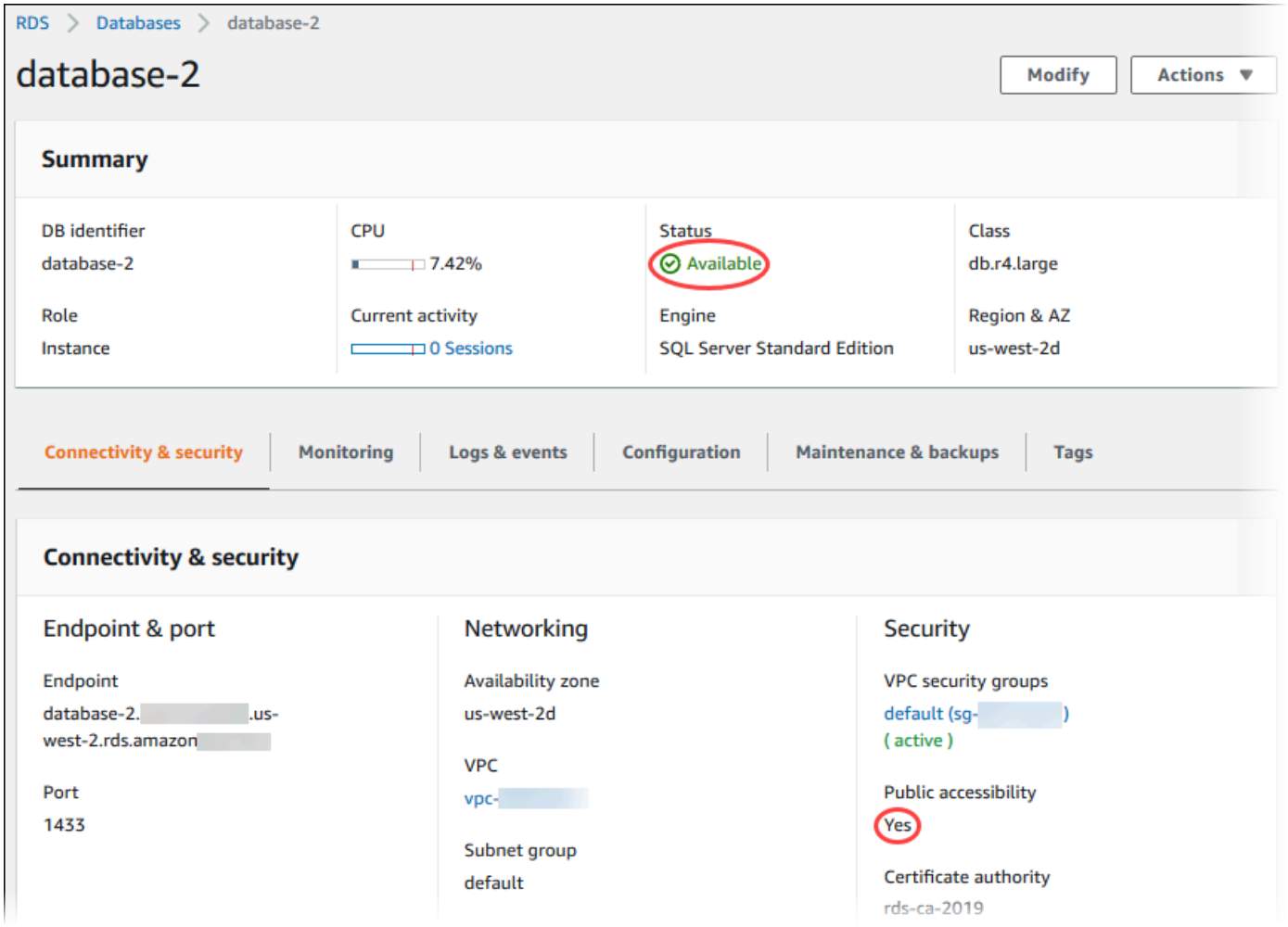
Après qu'Amazon RDS a provisionné votre instance de base de données, vous pouvez utiliser n'importe quelle application cliente SQL standard pour vous connecter à l'instance de base de données. Dans cette rubrique, vous vous connectez à votre instance de base de données à l'aide de Microsoft SQL Server Management Studio (SSMS) ou de SQL Workbench/J.

Pour obtenir un exemple qui vous explique le processus de création et de connexion à un exemple d'instance de base de données, consultez Création et connexion à une instance de base de données Microsoft SQL Server.

Avant de vous connecter

Avant de pouvoir vous connecter à votre instance de base de données, elle doit être disponible et accessible.

1. Assurez-vous que son statut est bien `available`. Vous pouvez le vérifier sur la page de détails de votre instance dans le AWS Management Console ou à l'aide de la describe-db-instances AWS CLI commande.



2. Assurez-vous qu'il est accessible à votre source. Selon votre scénario, il n'est peut-être pas nécessaire qu'il soit accessible au public. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.
3. Assurez-vous que les règles entrantes de votre groupe de sécurité VPC autorisent l'accès à votre instance de base de données. Pour plus d'informations, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Recherche du point de terminaison de l'instance de base de données et du numéro de port

Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.

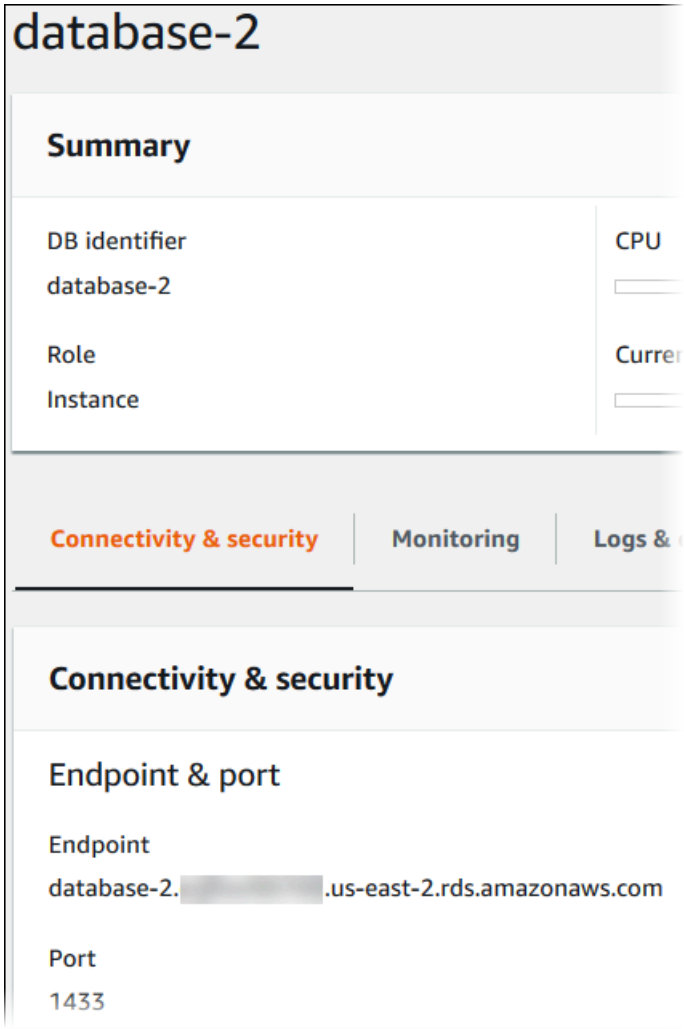
Pour trouver le point de terminaison et le port

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la AWS région de votre instance de base de données.
3. Cherchez le nom DNS (point de terminaison) et le numéro de port de votre instance de base de données :

   a. Ouvrez la console RDS et choisissez Bases de données pour afficher une liste de vos instances de bases de données.

   b. Choisissez le nom de l'instance de base de données SQL Server pour afficher ses détails.

   c. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison.



d. Notez le numéro du port.

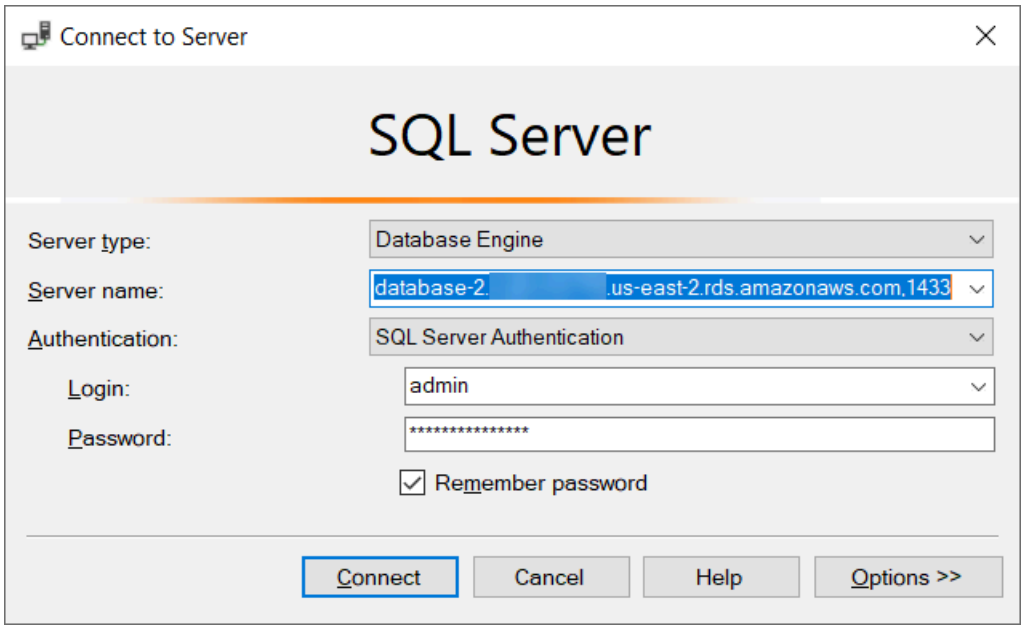
Connexion à votre instance de base de données avec Microsoft SQL Server Management Studio

Dans cette procédure, vous vous connectez à votre exemple d'instance de base de données à l'aide de Microsoft SQL Server Management Studio (SSMS). Pour télécharger une version autonome de cet utilitaire, voir Download SQL Server Management Studio (SSMS) dans la documentation Microsoft.

Pour vous connecter à une instance de base de données à l'aide de SSMS

1. Démarrez SQL Server Management Studio.

   La boîte de dialogue Connect to Server (Se connecter à un serveur) s'affiche.



2. Fournissez les informations relatives à votre instance de base de données :

   a. Pour Server type (Type de serveur), choisissez Database Engine (Moteur de base de données).

   b. Dans Nom du serveur, entrez le DNS nom (point de terminaison) et le numéro de port de votre instance de base de données, séparés par une virgule.

> **Important**
>
> Remplacez les deux-points entre le point de terminaison et le numéro de port par une virgule.

Votre nom de serveur doit ressembler à l'exemple suivant.

```
database-2.cg034itsfake.us-east-1.rds.amazonaws.com,1433
```

c. Pour Authentification, choisissez Authentification SQL du serveur.

d. Pour Login (Connexion), saisissez le nom d'utilisateur principal de votre instance de base de données.

e. Pour Password (Mot de passe), saisissez le mot de passe de votre instance de base de données.

3. Choisissez Connexion.

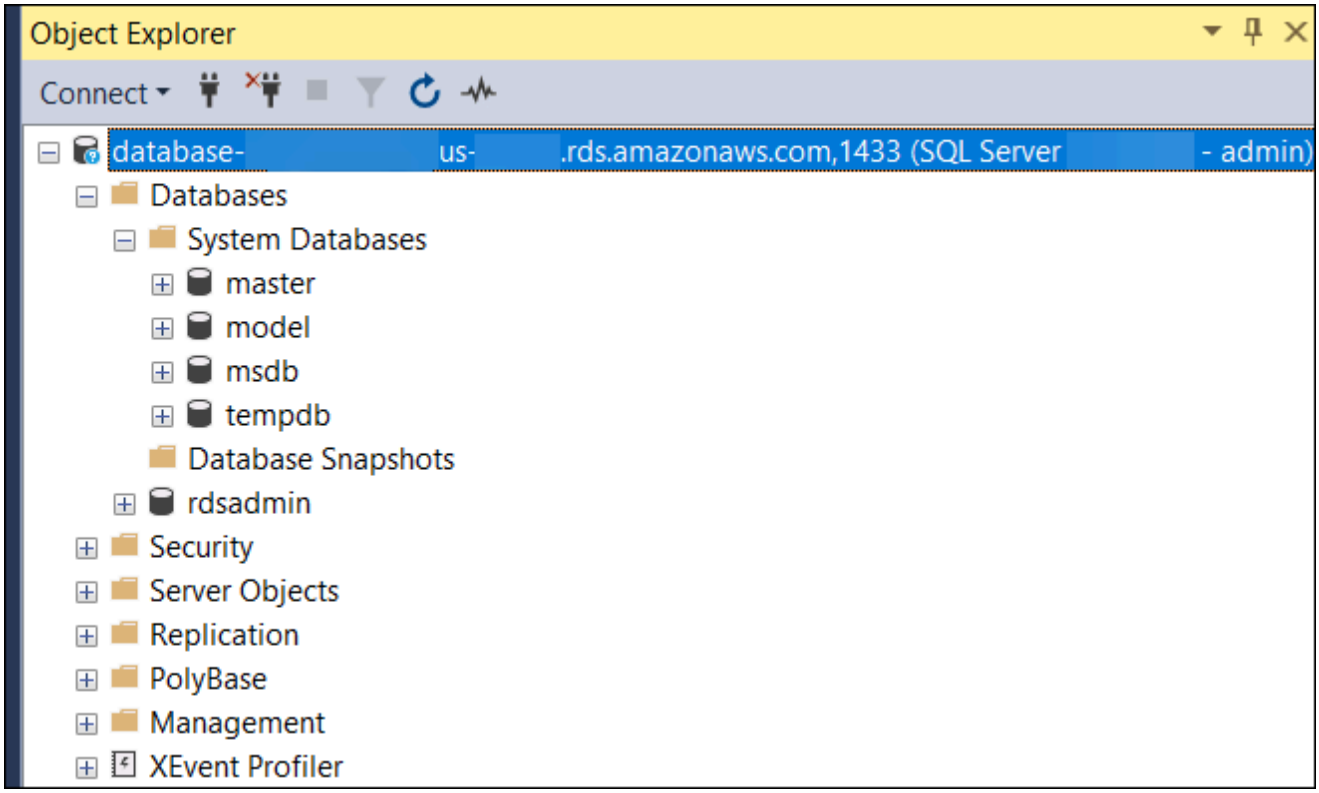
   Après quelques instants, SSMS se connecte à votre instance de base de données.

   Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Considérations relatives aux groupes de sécurité et Résolution des problèmes de connexion à votre instance de base de données de SQL serveur.

4. Votre SQL instance de base de données SQL Server est fournie avec les bases de données système intégrées standard du serveur (`master`,`model`,`msdb`, `tempdb`). Pour explorer les bases de données système, effectuez les opérations suivantes :

   a. Dans SSMS le menu Affichage, choisissez Explorateur d'objets.

   b. Développez votre instance de base de données, développez Databases (Bases de données), puis System Databases (Bases de données système).



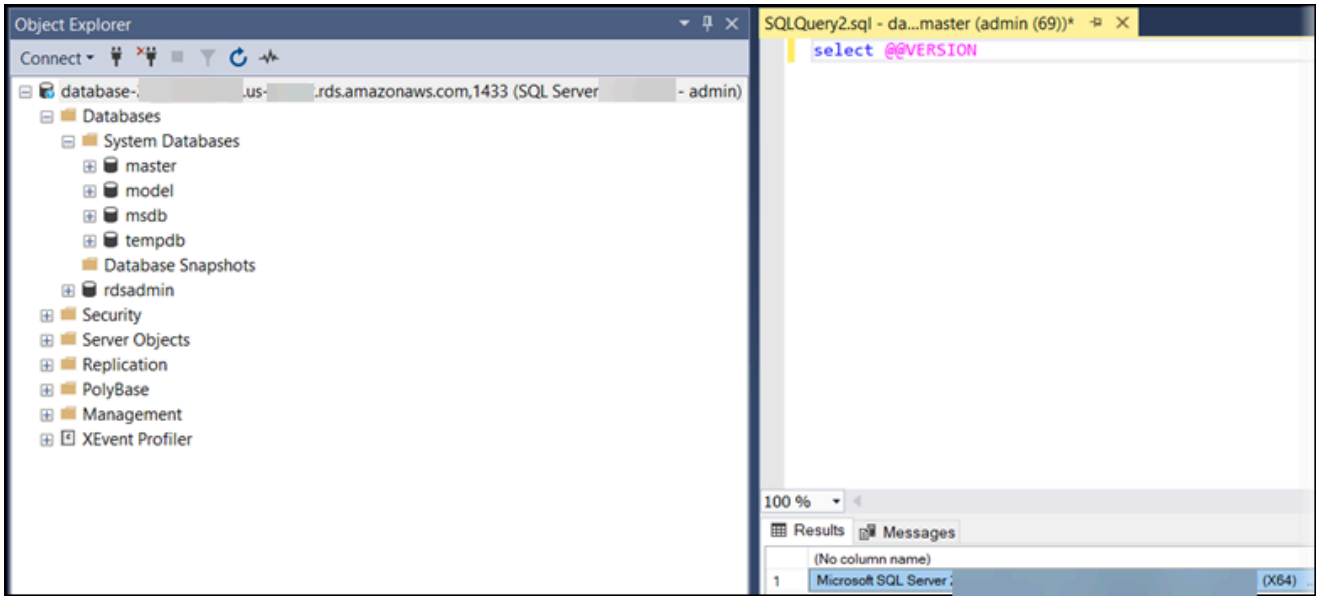
5. Votre instance SQL de base de données Server est également fournie avec une base de données nomméerdsadmin. Amazon RDS utilise cette base de données pour stocker les objets qu'il utilise pour gérer votre base de données. La base de données `rdsadmin` inclut également des procédures stockées que vous pouvez exécuter pour effectuer des tâches avancées. Pour plus d'informations, consultez Tâches DBA courantes pour Amazon RDS pour Microsoft SQL Server.
6. Vous pouvez maintenant commencer à créer vos propres bases de données et à exécuter des requêtes sur votre instance de base de données et vos bases de données comme d'habitude. Pour exécuter une requête de test sur votre instance de base de données, procédez comme suit :

   a. DansSSMS, dans le menu Fichier, pointez sur Nouveau, puis choisissez Requête avec connexion actuelle.

   b. Entrez la SQL requête suivante.

   ```
   select @@VERSION
   ```

   c. Exécutez la requête. SSMSrenvoie la version SQL du serveur de votre RDS instance de base de données Amazon.



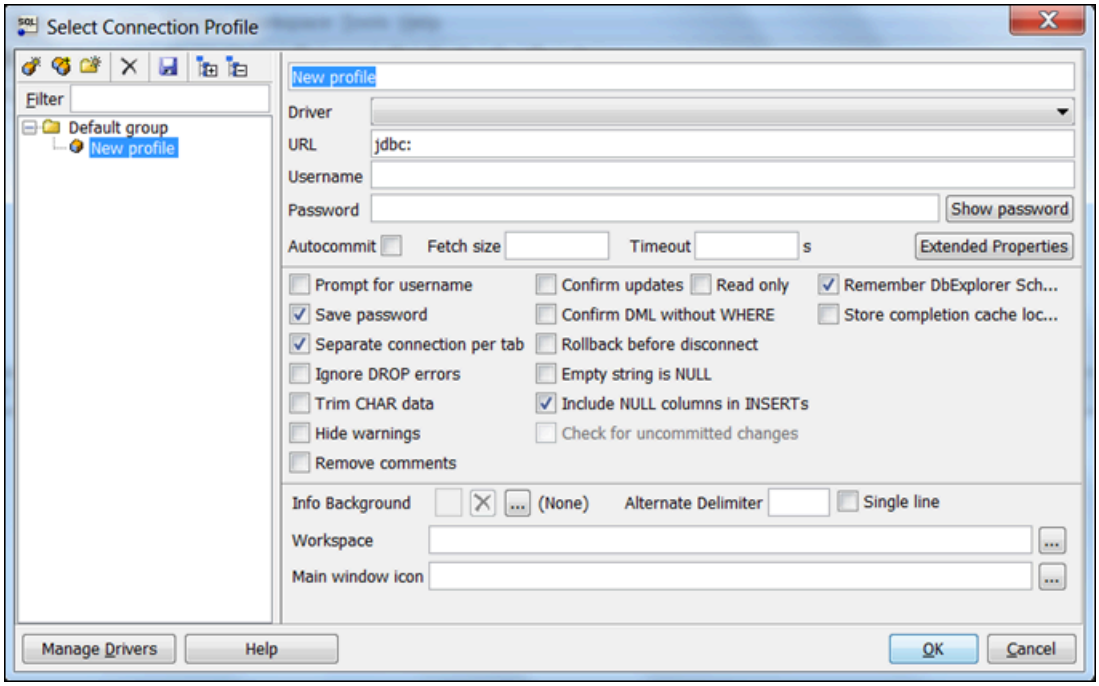
Connexion à votre instance de base de données avec SQL Workbench/J

Cet exemple montre comment se connecter à une instance de base de données exécutant le moteur de base de données Microsoft SQL Server à l'aide du SQLWorkbench/J database tool. To download SQL Workbench/J, voir SQL Workbench/J.

SQL Workbench/J utilise JDBC pour se connecter à votre instance de base de données. Vous avez également besoin du pilote JDBC pour SQL Server. Pour télécharger ce pilote, voir Télécharger le pilote Microsoft JDBC pour SQL Server.

Se connecter à une instance de base de données à l'aide de SQL Workbench/J

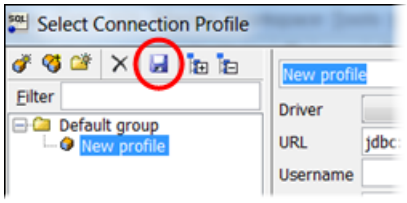
1. Ouvrez SQL Workbench/J. La boîte de dialogue Select Connection Profile (Sélectionner le profil de connexion) s'affiche, comme illustré ci-dessous.



2. Dans la première case en haut de la boîte de dialogue, saisissez un nom pour le profil.
3. Pour Driver (Pilote), sélectionnez **`SQL JDBC 4.0`**.
4. Pour URL, saisissez **`jdbc:sqlserver://`**, puis le point de terminaison de votre instance de base de données. Par exemple, la valeur de l'URL pourrait être la suivante.

```
jdbc:sqlserver://sqlsvr-pdz.abcd12340.us-west-2.rds.amazonaws.com:1433
```

5. Pour Nom d'utilisateur, saisissez le nom d'utilisateur principal de votre instance de base de données.
6. Pour Mot de passe, saisissez le mot de passe pour l'utilisateur principal.
7. Choisissez l'icône d'enregistrement dans la barre d'outils de la boîte de dialogue, comme illustré ci-dessous.

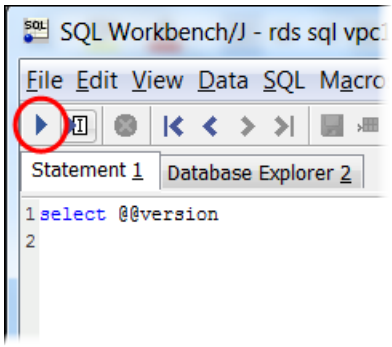


8. Choisissez OK. Après quelques instants, SQL Workbench/J se connecte à votre instance de base de données. Si vous ne pouvez pas vous connecter à votre instance de base de données, consultez Considérations relatives aux groupes de sécurité et Résolution des problèmes de connexion à votre instance de base de données de SQL serveur.

9. Dans le volet de requête, saisissez la requête SQL suivante.

```
select @@VERSION
```

10. Choisissez l'icône `Execute` dans la barre d'outils, comme illustré ci-dessous.



La requête renvoie les informations de version pour votre instance de base de données, comme illustré ci-dessous.

```
Microsoft SQL Server 2017 (RTM-CU22) (KB4577467) - 14.0.3356.20 (X64)
```

Considérations relatives aux groupes de sécurité

Pour vous connecter à votre instance de base de données, cette dernière doit être associée à un groupe de sécurité. Ce groupe de sécurité contient les adresses IP et la configuration réseau que vous utilisez pour accéder à l'instance de base de données. Vous avez peut-être associé votre instance de base de données à un groupe de sécurité correspondant lorsque vous avez créé votre instance de base de données. Si vous avez attribué un groupe de sécurité non configuré par défaut lors de la création de votre instance de base de données, le pare-feu de celle-ci bloque les connexions.

Dans certains cas, vous devrez peut-être créer un groupe de sécurité pour rendre l'accès possible. Pour obtenir des instructions sur la création d'un nouveau groupe de sécurité, consultez Contrôle d'accès par groupe de sécurité. Pour consulter une rubrique expliquant le processus de configuration des règles pour votre groupe VPC de sécurité, consultezTutoriel : Création d'un VPC à utiliser avec un) IPv4 .

Une fois le groupe de sécurité créé, modifiez votre instance de base de données pour l'associer au groupe de sécurité. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Vous pouvez améliorer la sécurité en chiffrant SSL les connexions à votre instance de base de données. Pour de plus amples informations, veuillez consulter Utilisation SSL avec une instance de base de données Microsoft SQL Server.

Résolution des problèmes de connexion à votre instance de base de données de SQL serveur

Le tableau suivant indique les messages d'erreur que vous pouvez rencontrer lorsque vous tentez de vous connecter à votre instance de base de données de SQL serveur.

Problème	Suggestions de dépannage
Impossible d'ouvrir une connexion au SQL serveur — Microsoft SQL Server, erreur : 53	Vérifiez que vous avez spécifié le nom de serveur correct. Dans Nom du serveur, entrez le DNS nom et le numéro de port de votre exemple d'instance de base de données, séparés par une virgule. **Important** Si le DNS nom et le numéro de port sont séparés par deux points, remplacez-les par une virgule. Votre nom de serveur doit ressembler à l'exemple suivant. `sample-instance.cg034itsfake.us-east-1.rds.amazonaws.com,1433`
Aucune connexion n'a pu être établie car la machine cible l'a activement refusée — Microsoft SQL Server, erreur : 10061	Vous avez pu atteindre l'instance de base de données, mais la connexion a été refusée. Ce problème est souvent dû au fait que le nom d'utilisateur ou le mot de passe spécifié est incorrect. Vérifiez le nom d'utilisateur et le mot de passe, puis réessayez.
Une erreur liée au réseau ou spécifique à l'instance s'est produite lors de l'établissement d'une connexion au serveur.	Les règles d'accès appliquées par votre pare-feu local et les adresses IP autorisées à accéder à votre instance de base de données ne correspondent peut-être pas. Le problème est probablement lié aux règles entrantes de votre groupe de sécurité.

Problème	Suggestions de dépannage
SQL The server was not found or was not accessible... L'opération d'attente a expiré — Microsoft SQL Server, erreur : 258	Pour de plus amples informations, veuillez consulter Sécurité dans Amazon RDS.
	Votre instance de base de données doit être accessible au public. Pour s'y connecter depuis l'extérieur deVPC, une adresse IP publique doit être attribuée à l'instance.

> **Note**
>
> Pour de plus amples informations sur les problèmes de connexion, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

Utilisation d'Active Directory avec RDS for SQL Server

Vous pouvez joindre une instance de base de données RDS for SQL Server à un domaine Microsoft Active Directory (AD). Votre domaine AD peut être hébergé sur AWS Managed AD au sein d'AWS, ou sur un annuaire AD autogéré à un emplacement de votre choix, y compris dans les centres de données de votre entreprise, sur AWS EC2 ou auprès d'autres fournisseurs de services cloud.

Vous pouvez authentifier les utilisateurs du domaine à l'aide de l'authentification NTLM avec Active Directory autogéré. Vous pouvez utiliser l'authentification Kerberos et NTLM avec AWS Managed Active Directory.

Dans les sections suivantes, vous trouverez des informations sur l'utilisation d'Active Directory autogéré et d'AWS Managed Active Directory pour Microsoft SQL Server sur Amazon RDS.

Rubriques

- Utilisation d'Active Directory autogéré avec une instance de base de données Amazon RDS for SQL Server
- Utilisation d'Active Directory AWS géré avec RDS pour SQL Server

Utilisation d'Active Directory autogéré avec une instance de base de données Amazon RDS for SQL Server

Vous pouvez joindre vos instances de base de données RDS for SQL Server directement à votre domaine Active Directory (AD) autogéré, quel que soit l'endroit où votre AD est hébergé : dans les centres de données d'entreprise AWS EC2, sur ou auprès d'autres fournisseurs de cloud. Avec AD autogéré, vous utilisez l'NTLMauthentification pour contrôler directement l'authentification des utilisateurs et des services sur vos instances de base de données RDS de SQL serveur sans utiliser de domaines intermédiaires ni d'approbations forestières. Lorsque les utilisateurs s'authentifient avec une instance de base de données RDS for SQL Server jointe à votre domaine AD autogéré, les demandes d'authentification sont transférées vers un domaine AD autogéré que vous spécifiez.

Rubriques

- Disponibilité des régions et des versions
- Prérequis
- Limites
- Vue d'ensemble de la configuration d'Active Directory autogéré
- Configuration d'Active Directory autogéré
- Gestion d'une instance de base de données dans un domaine Active Directory autogéré
- Comprendre l'appartenance à un domaine Active Directory autogéré
- Résolution des problèmes liés à Active Directory autogéré
- Restaurer une instance de base de données de SQL serveur, puis l'ajouter à un domaine Active Directory autogéré

Disponibilité des régions et des versions

Amazon RDS prend en charge l'utilisation d'AD autogéré pour les SQL serveurs NTLM dans tous les domaines Régions AWS.

Prérequis

Assurez-vous de répondre aux exigences suivantes avant de joindre une instance de base de données RDS for SQL Server à votre domaine AD autogéré.

Rubriques

- Configuration de votre annuaire AD sur site
- Configuration de votre connectivité réseau
- Configuration de votre compte de service de domaine AD

Configuration de votre annuaire AD sur site

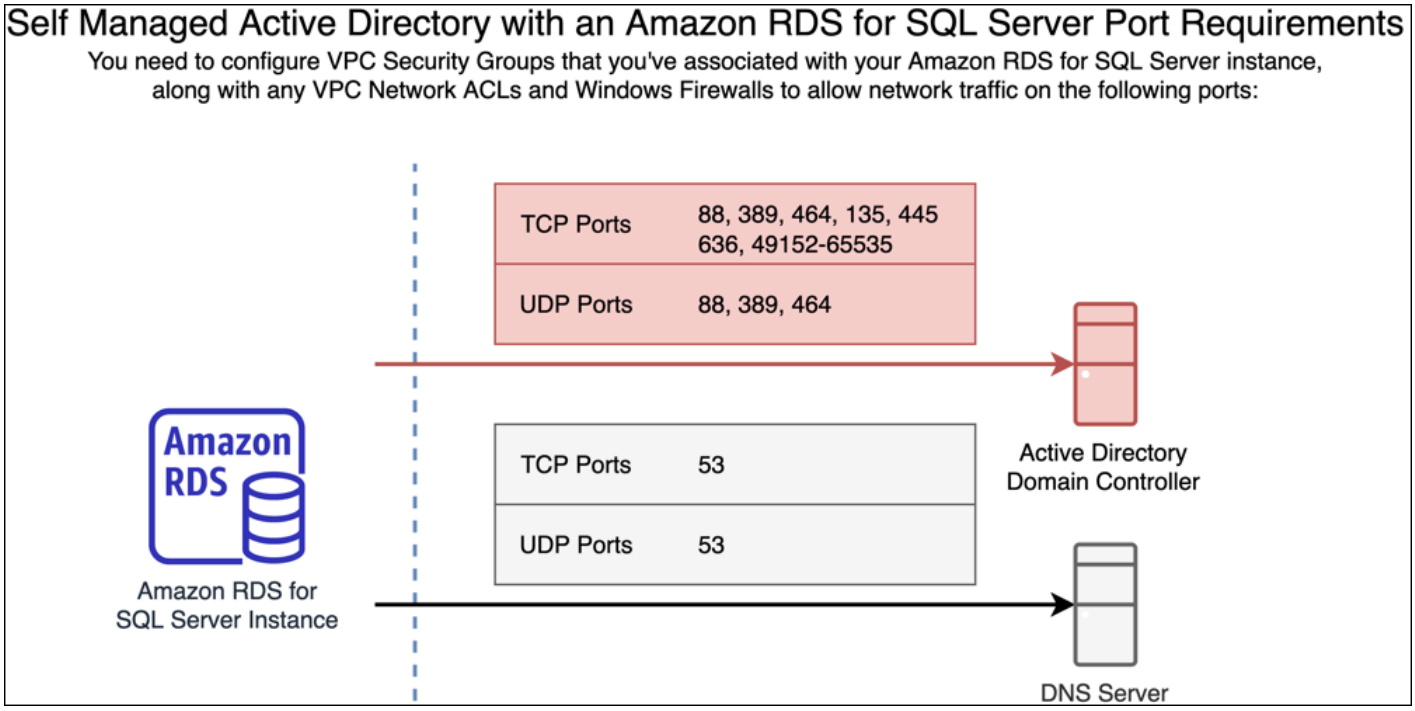
Assurez-vous de disposer d'un Microsoft AD sur site ou d'un autre service autogéré auquel vous pouvez joindre l'instance Amazon RDS for SQL Server. Votre annuaire AD sur site doit avoir la configuration suivante :

- Si vous avez défini des sites Active Directory, assurez-vous que les sous-réseaux VPC associés à votre instance de base de données RDS for SQL Server sont définis dans votre site Active Directory. Vérifiez qu'il n'y a aucun conflit entre les sous-réseaux de votre site AD VPC et ceux de vos autres sites AD.
- Votre contrôleur de domaine AD a un niveau fonctionnel de domaine correspondant à Windows Server 2008 R2 ou supérieur.
- Votre nom de domaine AD ne peut pas être au format Single Label Domain (SLD). RDSfor SQL Server ne prend pas en charge SLD les domaines.
- Le nom de domaine complet (FQDN) de votre annonce ne peut pas dépasser 47 caractères.

Configuration de votre connectivité réseau

Assurez-vous de respecter les configurations réseau suivantes :

- Connectivité configurée entre l'Amazon sur VPC lequel vous souhaitez créer l'instance de base de données RDS for SQL Server et votre Active Directory autogéré. Vous pouvez configurer la connectivité à l'aide de AWS Direct Connect AWS VPN, de VPC peering ou de AWS Transit Gateway.
- Pour les groupes VPC de sécurité, le groupe de sécurité par défaut pour votre Amazon par défaut VPC est déjà ajouté à votre instance de base de données RDS for SQL Server dans la console. Assurez-vous que le groupe de sécurité et le VPC réseau ACLs du ou des sous-réseaux dans lesquels vous créez votre instance de base de données RDS for SQL Server autorisent le trafic sur les ports et dans les directions indiquées dans le schéma suivant.



Le tableau suivant identifie le rôle de chaque port.

Protocole	Ports	Rôle
TCP/UDP	53	Système de noms de domaine (DNS)
TCP/UDP	88	Authentification Kerberos
TCP/UDP	464	Changement/définition de mot de passe
TCP/UDP	389	Protocole léger d'accès aux annuaires (LDAP)
TCP	135	Environnement informatique distribué/Endpoint Mapper (DCE/EPMAP)
TCP	445	Partage de SMB fichiers avec les services d'annuaire

Protocole	Ports	Rôle
TCP	636	Protocole léger d'accès aux répertoires viaTLS/SSL(LDAPS)
TCP	49152 - 65535	Ports éphémères pour RPC

- En général, les DNS serveurs de domaine sont situés dans les contrôleurs de domaine AD. Il n'est pas nécessaire de configurer le jeu d'VPCDHCPoptions pour utiliser cette fonctionnalité. Pour plus d'informations, consultez les ensembles d'DHCPoptions dans le guide de VPC l'utilisateur Amazon.

> **Important**
>
> Si vous utilisez le VPC réseauACLs, vous devez également autoriser le trafic sortant sur les ports dynamiques (49152-65535) depuis votre instance de base de données for Server. RDS SQL Assurez-vous que ces règles de trafic sont également reflétées sur les pare-feux qui s'appliquent à chacun des contrôleurs de domaine AD, aux serveurs et aux instances de base de données RDS de DNS SQL serveurs.
>
> Alors que les groupes de VPC sécurité exigent que les ports soient ouverts uniquement dans le sens où le trafic réseau est initié, la plupart des pare-feux et VPC réseaux Windows ACLs exigent que les ports soient ouverts dans les deux sens.

Configuration de votre compte de service de domaine AD

Assurez-vous de respecter les exigences suivantes pour un compte de service de domaine AD :

- Assurez-vous de disposer d'un compte de service dans votre domaine AD autogéré avec des autorisations déléguées pour joindre des ordinateurs au domaine. Un compte de service de domaine est un compte utilisateur de votre annuaire AD autogéré auquel l'autorisation d'effectuer certaines tâches a été déléguée.
- Les autorisations suivantes doivent être déléguées au compte de service de domaine dans l'unité organisationnelle (UO) à laquelle vous rejoignez votre instance de base de données RDS pour le SQL serveur :
  - Capacité validée d'écrire sur le nom DNS d'hôte
  - Capacité validée d'écrire dans le nom du principal de service

- Création et suppression d'objets informatiques

Il s'agit de l'ensemble minimal d'autorisations requises pour joindre des objets informatiques à votre annuaire Active Directory autogéré. Pour plus d'informations, consultez Erreurs lors d'une tentative visant à joindre des ordinateurs à un domaine dans la documentation de Microsoft Windows Server.

> **Important**
>
> Ne déplacez pas les objets informatiques créés par SQL Server dans l'unité organisationnelle après la création de votre instance de base de données. RDS Le déplacement des objets associés entraînera une mauvaise configuration de votre RDS instance de base de données pour le SQL serveur. Si vous devez déplacer les objets informatiques créés par AmazonRDS, utilisez l'odifyDBInstanceRDSAPIopération M pour modifier les paramètres du domaine en fonction de l'emplacement souhaité pour les objets informatiques.

Limites

Les limitations suivantes s'appliquent à Self Managed AD for SQL Server.

- NTLMest le seul type d'authentification pris en charge. L'authentification Kerberos n'est pas prise en charge. Si vous devez utiliser l'authentification Kerberos, vous pouvez utiliser AWS Managed AD au lieu d'AD autogéré.
- Le service Microsoft Distributed Transaction Coordinator (MSDTC) n'est pas pris en charge, car il nécessite une authentification Kerberos.
- Vos instances de base de données RDS for SQL Server n'utilisent pas le serveur Network Time Protocol (NTP) de votre domaine AD autogéré. Ils utilisent plutôt un AWS NTP service.
- SQLLes serveurs liés au serveur doivent utiliser SQL l'authentification pour se connecter à d'autres RDS instances de base de données de SQL serveur jointes à votre domaine AD autogéré.
- Les paramètres Microsoft Group Policy Object (GPO) de votre domaine AD autogéré ne sont pas appliqués aux instances de base RDS de données de SQL serveur.

Vue d'ensemble de la configuration d'Active Directory autogéré

Pour configurer AD autogéré pour une instance de base de données RDS pour SQL serveur, suivez les étapes suivantes, expliquées plus en détail dans Configuration d'Active Directory autogéré :

Dans votre domaine AD :

- Créez une unité d'organisation (OU).
- Créez un utilisateur de domaine AD.
- Déléguez le contrôle à l'utilisateur du domaine AD.

Depuis le AWS Management Console ou API :

- Créez une AWS KMS clé.
- Créez un secret à l'aide de AWS Secrets Manager.
- Créez ou modifiez une instance de base de données RDS for SQL Server et associez-la à votre domaine AD autogéré.

Configuration d'Active Directory autogéré

Pour configurer AD autogéré, procédez comme suit.

Rubriques

- Étape 1 : Créer une unité d'organisation dans votre annuaire AD
- Étape 2 : Créer un utilisateur de domaine AD dans votre annuaire AD
- Étape 3 : Déléguer le contrôle à l'utilisateur AD
- Étape 4 : Création d'une AWS KMS clé
- Étape 5 : Créez un AWS secret
- Étape 6 : Créer ou modifier une instance de base de données SQL Server
- Étape 7 : Créer des connexions SQL Server pour l'authentification Windows

Étape 1 : Créer une unité d'organisation dans votre annuaire AD

> **Important**
>
> Nous vous recommandons de créer une unité d'organisation dédiée et des informations d'identification de service étendues à cette unité d'organisation pour tout AWS compte propriétaire d'une instance de base de données RDS pour SQL Server jointe à votre domaine AD autogéré. En dédiant une unité d'organisation et des informations d'identification de service, vous pouvez éviter les conflits d'autorisations et suivre le principe de moindre privilège.

Pour créer une unité d'organisation dans votre annuaire AD

1. Connectez-vous à votre domaine AD en tant qu'administrateur de domaine.
2. Ouvrez Utilisateurs et ordinateurs Active Directory et sélectionnez le domaine où vous souhaitez créer votre unité d'organisation.
3. Cliquez avec le bouton droit sur le domaine et choisissez Nouveau, puis Unité d'organisation.
4. Saisissez un nom pour l'unité d'organisation.
5. Laissez la case cochée pour Protéger le conteneur contre la suppression accidentelle.
6. Cliquez sur OK. Votre nouvelle unité d'organisation apparaîtra sous votre domaine.

Étape 2 : Créer un utilisateur de domaine AD dans votre annuaire AD

Les informations d'identification de l'utilisateur du domaine seront utilisées pour le secret dans AWS Secrets Manager.

Pour créer un utilisateur de domaine AD dans votre annuaire AD

1. Ouvrez Utilisateurs et ordinateurs Active Directory et sélectionnez le domaine et l'unité d'organisation où vous souhaitez créer votre utilisateur.
2. Cliquez avec le bouton droit sur l'objet Utilisateurs et choisissez Nouveau, puis Utilisateur.
3. Saisissez le prénom, le nom de famille et le nom de connexion de l'utilisateur. Cliquez sur Next (Suivant).
4. Saisissez un mot de passe pour l'utilisateur. Ne sélectionnez pas « L'utilisateur doit modifier le mot de passe lors de sa prochaine connexion ». Ne sélectionnez pas « Le compte est désactivé ». Cliquez sur Next (Suivant).

5. Cliquez sur OK. Votre nouvel utilisateur apparaîtra sous votre domaine.

Étape 3 : Déléguer le contrôle à l'utilisateur AD

Pour déléguer le contrôle à l'utilisateur du domaine AD dans votre domaine

1. Ouvrez le composant logiciel enfichable MMC Utilisateurs et ordinateurs Active Directory et sélectionnez le domaine où vous souhaitez créer votre utilisateur.
2. Cliquez avec le bouton droit sur l'unité d'organisation que vous avez créée précédemment et choisissez Déléguer le contrôle.
3. Sur la page Assistant Délégation de contrôle, cliquez sur Suivant.
4. Dans la section Utilisateurs ou groupes, cliquez sur Ajouter.
5. Dans la section Sélectionner les utilisateurs, les ordinateurs ou les groupes, entrez l'utilisateur AD que vous avez créé et cliquez sur Vérifier les noms. Si votre vérification de l'utilisateur AD aboutit, cliquez sur OK.
6. Dans la section Utilisateurs ou groupes, confirmez que votre utilisateur AD a été ajouté et cliquez sur Suivant.
7. Dans la section Tâches à déléguer, choisissez Créer une tâche personnalisée à déléguer et cliquez sur Suivant.
8. Dans la section Type d'objet Active Directory :
    a. Choisissez Seulement les objets suivants dans le dossier.
    b. Sélectionnez Objets informatiques.
    c. Sélectionnez Créer les objets sélectionnés dans ce dossier.
    d. Sélectionnez Supprimer les objets sélectionnés dans ce dossier et cliquez sur Suivant.
9. Dans la section Autorisations :
    a. Gardez l'option Général sélectionnée.
    b. Sélectionnez Écriture validée sur le nom d'hôte DNS.
    c. Sélectionnez Écriture validée sur le nom du principal de service et cliquez sur Suivant.
10. Pour Fin de l'Assistant Délégation de contrôle, passez en revue et confirmez vos paramètres, puis cliquez sur Terminer.

Étape 4 : Création d'une AWS KMS clé

La clé KMS est utilisée pour chiffrer votre AWS secret.

Pour créer une AWS KMS clé

> **Note**
>
> Pour la clé de chiffrement, n'utilisez pas la clé KMS AWS par défaut. Assurez-vous de créer la AWS KMS clé dans le même AWS compte qui contient l'instance de base de données RDS pour SQL Server que vous souhaitez joindre à votre AD autogéré.

1. Dans la AWS KMS console, choisissez Create key.
2. Pour Type de clé, choisissez Symétrique.
3. Pour Utilisation de la clé, choisissez Chiffrer et déchiffrer.
4. Pour Options avancées :
   a. Pour Origine des clés, choisissez KMS.
   b. Pour Régionalité, choisissez Clé à région unique et cliquez sur Suivant.
5. Pour Alias, attribuez un nom à la clé KMS.
6. (Facultatif) Pour Description, fournissez une description de la clé KMS.
7. (Facultatif) Pour Balises, spécifiez une balise pour la clé KMS et cliquez sur Suivant.
8. Pour Administrateurs de clé, spécifiez le nom d'un utilisateur IAM et sélectionnez-le.
9. Pour Suppression de clé, laissez la case cochée pour Autoriser les administrateurs de clé à supprimer cette clé et cliquez sur Suivant.
10. Pour Utilisateurs de clé, spécifiez le même utilisateur IAM que celui de l'étape précédente et sélectionnez-le. Cliquez sur Next (Suivant).
11. Passez en revue la configuration.
12. Pour Stratégie de clé, ajoutez ce qui suit à la déclaration de stratégie :

```
{
    "Sid": "Allow use of the KMS key on behalf of RDS",
    "Effect": "Allow",
    "Principal": {
        "Service": [
```

```
            "rds.amazonaws.com"
        ]
    },
    "Action": "kms:Decrypt",
    "Resource": "*"
}
```

13. Cliquez sur Finish.

Étape 5 : Créez un AWS secret

Pour créer un secret

> **Note**
>
> Assurez-vous de créer le secret dans le même AWS compte qui contient l'instance de base de données RDS pour SQL Server que vous souhaitez joindre à votre AD autogéré.

1. Dans AWS Secrets Manager, choisissez Enregistrer un nouveau secret.
2. Pour Secret type (Type de secret), choisissez Other type of secret (Autre type de secret).
3. Pour Paires clé/valeur, ajoutez vos deux clés :
    a. Pour la première clé, entrez `SELF_MANAGED_ACTIVE_DIRECTORY_USERNAME`.
    b. Pour la valeur de la première clé, entrez uniquement le nom d'utilisateur (sans le préfixe de domaine) de l'utilisateur AD. N'incluez pas le nom de domaine car cela entraîne l'échec de la création de l'instance.
    c. Pour la deuxième clé, entrez `SELF_MANAGED_ACTIVE_DIRECTORY_PASSWORD`.
    d. Pour la valeur de la deuxième clé, saisissez le mot de passe que vous avez créé pour l'utilisateur AD sur votre domaine.
4. Pour Clé de chiffrement, saisissez la clé KMS que vous avez créée à une étape précédente et cliquez sur Suivant.
5. Dans Nom du secret, saisissez un nom descriptif qui vous aidera à rechercher votre secret ultérieurement.
6. (Facultatif) Pour Description, saisissez une description du nom du secret.
7. Pour Autorisation des ressources, cliquez sur Modifier.
8. Ajoutez la politique suivante à la politique d'autorisation :

> **Note**
>
> Nous vous recommandons d'utiliser les conditions `aws:sourceAccount` et `aws:sourceArn` dans la politique pour éviter le problème de l'adjoint confus. Utilisez votre Compte AWS for `aws:sourceAccount` et l'ARN de votre instance de base de données RDS pour SQL Server pour`aws:sourceArn`. Pour de plus amples informations, veuillez consulter Prévention des problèmes d'adjoint confus entre services.

JSON

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
            "Effect": "Allow",
            "Principal":
            {
                "Service": "rds.amazonaws.com"
            },
            "Action": "secretsmanager:GetSecretValue",
            "Resource": "*",
            "Condition":
            {
                "StringEquals":
                {
                    "aws:sourceAccount": "123456789012"
                },
                "ArnLike":
                {
                    "aws:sourceArn": "arn:aws:rds:us-
west-2:123456789012:db:*"
                }
            }
        }
    ]
}
```

9. Cliquez sur Enregistrer, puis sur Suivant.
10. Pour Configurer les paramètres de rotation, conservez les valeurs par défaut et choisissez Suivant.
11. Passez en revue les paramètres du secret et cliquez sur Stocker.
12. Choisissez le secret que vous avez créé et copiez la valeur de l'ARN du secret. Il sera utilisé à l'étape suivante pour configurer Active Directory autogéré.

Étape 6 : Créer ou modifier une instance de base de données SQL Server

Vous pouvez utiliser la console, l'interface de ligne de commande ou l'API RDS pour associer une instance de base de données RDS for SQL Server à un domaine AD autogéré. Vous pouvez effectuer cette opération de différentes manières :

- Créez une nouvelle instance de base de données SQL Server à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

  Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données SQL Server existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstanceModify RDS API.

  Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Restaurez une instance de base de données SQL Server à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération DBInstance Restore DBSnapshot From RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.
- Restaurez une instance de base de données SQL Server à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération Restore DBInstance ToPointInTime RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de base de données puisse utiliser le domaine Active Directory autogéré que vous avez créé :

- Pour le paramètre `--domain-fqdn`, utilisez le nom de domaine complet (FQDN) de votre annuaire Active Directory autogéré.
- Pour le paramètre `--domain-ou`, utilisez l'unité d'organisation que vous avez créée dans votre annuaire AD autogéré.
- Pour le paramètre `--domain-auth-secret-arn`, utilisez la valeur de l'ARN du secret que vous avez créé dans une étape précédente.
- Pour le `--domain-dns-ips` paramètre, utilisez les IPv4 adresses principale et secondaire des serveurs DNS de votre AD autogéré. Si vous ne possédez pas d'adresse IP de serveur DNS secondaire, entrez deux fois l'adresse IP principale.

Les exemples de commandes CLI suivants montrent comment créer, modifier et supprimer une instance de base de données RDS for SQL Server avec un domaine AD autogéré.

> **Important**
>
> Si vous modifiez une instance de base de données pour la joindre à un domaine AD autogéré ou pour l'en supprimer, un redémarrage de l'instance de base de données est requis pour que la modification prenne effet. Vous pouvez choisir d'appliquer les modifications immédiatement ou d'attendre la prochaine fenêtre de maintenance. Le choix de l'option Appliquer immédiatement entraînera un temps d'arrêt pour l'instance de base de données mono-AZ. Une instance de base de données multi-AZ effectuera un basculement avant de terminer le redémarrage. Pour de plus amples informations, veuillez consulter Utilisation du paramètre de modification du calendrier.

La commande CLI suivante crée une nouvelle instance de base de données RDS for SQL Server et la joint à un domaine AD autogéré.

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier my-DB-instance \
    --db-instance-class db.m5.xlarge \
    --allocated-storage 50 \
```

```
    --engine sqlserver-se \
    --engine-version 15.00.4043.16.v1 \
    --license-model license-included \
    --master-username my-master-username \
    --master-user-password my-master-password \
    --domain-fqdn my_AD_domain.my_AD.my_domain \
    --domain-ou OU=my-AD-test-OU,DC=my-AD-test,DC=my-AD,DC=my-domain \
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:my-AD-test-secret-123456" \
    --domain-dns-ips "10.11.12.13" "10.11.12.14"
```

Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier my-DB-instance ^
    --db-instance-class db.m5.xlarge ^
    --allocated-storage 50 ^
    --engine sqlserver-se ^
    --engine-version 15.00.4043.16.v1 ^
    --license-model license-included ^
    --master-username my-master-username ^
    --master-user-password my-master-password ^
    --domain-fqdn my-AD-test.my-AD.mydomain ^
    --domain-ou OU=my-AD-test-OU,DC=my-AD-test,DC=my-AD,DC=my-domain ^
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:my-AD-test-secret-123456" \ ^
    --domain-dns-ips "10.11.12.13" "10.11.12.14"
```

La commande CLI suivante modifie une instance de base de données RDS for SQL Server existante afin d'utiliser un domaine Active Directory autogéré.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my-DB-instance \
    --domain-fqdn my_AD_domain.my_AD.my_domain \
    --domain-ou OU=my-AD-test-OU,DC=my-AD-test,DC=my-AD,DC=my-domain \
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:my-AD-test-secret-123456" \
    --domain-dns-ips "10.11.12.13" "10.11.12.14"
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my-DBinstance ^
    --domain-fqdn my_AD_domain.my_AD.my_domain ^
    --domain-ou OU=my-AD-test-OU,DC=my-AD-test,DC=my-AD,DC=my-domain ^
    --domain-auth-secret-arn "arn:aws:secretsmanager:region:account-number:secret:my-AD-test-secret-123456" ^
    --domain-dns-ips "10.11.12.13" "10.11.12.14"
```

La commande CLI suivante supprime une instance de base de données RDS for SQL Server d'un domaine Active Directory autogéré.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my-DB-instance \
    --disable-domain
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my-DB-instance ^
    --disable-domain
```

Étape 7 : Créer des connexions SQL Server pour l'authentification Windows

Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données SQL Server de la même manière qu'à n'importe quelle instance de base de données. Étant donné que l'instance de base de données est jointe au domaine AD autogéré, vous pouvez provisionner des connexions et des utilisateurs SQL Server. Pour ce faire, utilisez l'utilitaire Utilisateurs et groupes AD de votre domaine AD autogéré. Les autorisations pour la base de données sont gérées via des autorisations SQL Server standard accordées et révoquées en fonction des connexions Windows.

Pour qu'un utilisateur AD autogéré puisse s'authentifier à SQL Server, une connexion Windows SQL Server doit exister pour l'utilisateur AD autogéré ou un groupe Active Directory autogéré dont l'utilisateur est membre. Un contrôle précis des accès est géré par l'attribution ou la révocation d'autorisations pour ces connexions SQL Server. Un utilisateur AD autogéré qui n'a pas de connexion SQL Server ou qui n'appartient pas à un groupe AD autogéré avec une telle connexion ne peut pas accéder à l'instance de base de données SQL Server.

L'autorisation ALTER ANY LOGIN est requise pour créer une connexion SQL Server à AD autogéré. Si vous n'avez pas créé de connexion avec cette autorisation, connectez vous en tant qu'utilisateur principal de l'instance de base de données à l'aide de l'authentification SQL Server et créez vos connexions SQL Server à AD autogéré dans le contexte de l'utilisateur principal.

Vous pouvez exécuter une commande DDL (Data Definition Language) telle que la suivante afin de créer une connexion SQL Server pour un utilisateur ou un groupe AD autogéré.

> **Note**
>
> Spécifiez les utilisateurs et les groupes à l'aide du nom de connexion antérieur à Windows 2000 au format *my_AD_domain**my_AD_domain_user*. Vous ne pouvez pas utiliser un nom d'utilisateur principal (UPN) au format *my_AD_domain_user*@*my_AD_domain*.

```
USE [master]
GO
CREATE LOGIN [my_AD_domain\my_AD_domain_user] FROM WINDOWS WITH DEFAULT_DATABASE =
 [master], DEFAULT_LANGUAGE = [us_english];
GO
```

Pour plus d'informations, consultez CREATE LOGIN (Transact-SQL) dans la documentation de Microsoft Developer Network.

Les utilisateurs (personnes et applications) de votre domaine peuvent désormais se connecter à l'instance RDS for SQL Server à partir d'un ordinateur client joint au domaine AD autogéré à l'aide de l'authentification Windows.

Gestion d'une instance de base de données dans un domaine Active Directory autogéré

Vous pouvez utiliser la console ou Amazon RDS API pour gérer votre instance de base de données et sa relation avec votre domaine AD autogéré. AWS CLI Par exemple, vous pouvez déplacer l'instance de base de données dans, hors ou entre des domaines.

Par exemple, en utilisant Amazon RDSAPI, vous pouvez effectuer les opérations suivantes :

- Pour réessayer de joindre un domaine autogéré en cas d'échec d'adhésion, utilisez l'odifyDBInstanceAPIopération M et spécifiez le même ensemble de paramètres :

  - `--domain-fqdn`
  - `--domain-dns-ips`
  - `--domain-ou`
  - `--domain-auth-secret-arn`
- Pour supprimer une instance de base de données d'un domaine autogéré, utilisez l'`ModifyDBInstance`APIopération et spécifiez le paramètre `--disable-domain` de domaine.
- Pour déplacer une instance de base de données d'un domaine autogéré vers un autre, utilisez l'`ModifyDBInstance`APIopération et spécifiez les paramètres de domaine pour le nouveau domaine :
  - `--domain-fqdn`
  - `--domain-dns-ips`
  - `--domain-ou`
  - `--domain-auth-secret-arn`
- Pour répertorier les membres de domaines AD autogérés pour chaque instance de base de données, utilisez l'escribeDBInstancesAPIopération D.

Comprendre l'appartenance à un domaine Active Directory autogéré

Après la création ou la modification de votre instance de base de données, l'instance devient un membre du domaine AD autogéré. La AWS console indique l'état de l'appartenance au domaine Active Directory autogéré pour l'instance de base de données. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- joined : l'instance est membre du domaine AD.
- joining : l'instance est en train de devenir membre du domaine AD.
- pending-join – L'appartenance de l'instance est en attente.
- pending-maintenance-join— AWS tentera de faire de l'instance un membre du domaine AD lors de la prochaine fenêtre de maintenance planifiée.
- pending-removal : la suppression de l'instance du domaine AD est en attente.
- pending-maintenance-removal— AWS tentera de supprimer l'instance du domaine AD lors de la prochaine fenêtre de maintenance planifiée.
- failed : un problème de configuration a empêché de joindre l'instance au domaine AD. Vérifiez et corrigez votre configuration avant d'émettre à nouveau la commande de modification de l'instance.

- removing : la suppression de l'instance du domaine AD autogéré est en cours.

Une demande pour devenir membre d'un domaine AD autogéré peut échouer à cause d'un problème de connectivité réseau. Par exemple, vous pouvez créer une instance de base de données ou modifier une instance existante et faire échouer la tentative pour que l'instance de base de données devienne membre d'un domaine AD autogéré. Dans ce cas, émettez à nouveau la commande pour créer ou modifier l'instance de base de données, ou modifiez l'instance nouvellement créée pour la joindre au domaine AD autogéré.

Résolution des problèmes liés à Active Directory autogéré

Vous pouvez rencontrer les problèmes suivants lors de la configuration ou de la modification d'un annuaire AD autogéré.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 2 / 0x2	Le fichier spécifié est introuvable.	Le format ou l'emplacement de l'unité d'organisation (OU) spécifiée avec le paramètre –`domain-ou` est non valide. Le compte de service de domaine spécifié via AWS Secrets Manager ne dispose pas des autorisations requises pour rejoindre l'unité d'organisation.	Passez en revue le paramètre –`domain-ou` . Assurez-vous que le compte de service de domaine dispose des autorisations appropriées pour accéder à l'unité d'organisation. Pour de plus amples informations, veuillez consulter Configuration de votre compte de service de domaine AD.
Erreur 5 / 0x5	Accès refusé.	Autorisations mal configurées pour le compte de service de domaine, ou le compte d'ordinateur existe déjà dans le domaine.	Vérifiez les autorisations du compte de service de domaine dans le domaine et vérifiez que le compte d'RDSordinateur n'est pas dupliqué dans le domaine.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
			Vous pouvez vérifier le nom du compte d'RDSordinateur en exécutant votre instance de `SELECT @@SERVERNAME` base de données RDS for SQL Server. Si vous utilisez le mode Multi-AZ, essayez de redémarrer avec basculement, puis vérifiez à nouveau que le compte de l'RDSordinateur est bien enregistré. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.
Erreur 87 / 0x57	Le paramètre est incorrect.	Le compte de service de domaine spécifié via AWS Secrets Manager ne dispose pas des autorisations appropriées. Le profil utilisateur est peut-être également endommagé.	Passez en revue les exigences relatives au compte de service de domaine. Pour de plus amples informations, veuillez consulter Configuration de votre compte de service de domaine AD.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 234 / 0xEA	L'unité d'organisation (OU) spécifiée n'existe pas.	L'unité d'organisation spécifiée avec le paramètre –`domain-ou` n'existe pas dans votre annuaire AD autogéré.	Vérifiez le paramètre –`domain-ou` et assurez-vous que l'unité d'organisation spécifiée existe dans votre annuaire AD autogéré.
Erreur 1326 / 0x52E	Le nom d'utilisateur ou le mot de passe est incorrect.	Les informations d'identification du compte de service de domaine fournies dans AWS Secrets Manager contiennent un nom d'utilisateur inconnu ou un mot de passe incorrect. Le compte de domaine peut également être désactivé dans votre annuaire AD autogéré.	Assurez-vous que les informations d'identification fournies dans AWS Secrets Manager sont correctes et que le compte de domaine est activé dans votre Active Directory autogéré.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 1355 / 0x54B	Le domaine spécifié n'existe pas ou n'a pas pu être contacté.	Le domaine est en panne, l'ensemble spécifié DNS IPs est inaccessible ou l'ensemble spécifié FQDN est inaccessible.	Vérifiez les paramètres `–domain-dns-ips` et `–domain-fqdn` pour vous assurer qu'ils sont corrects. Passez en revue la configuration réseau de votre instance de base de données RDS for SQL Server et assurez-vous que votre AD autogéré est accessible. Pour de plus amples informations, veuillez consulter Configuration de votre connectivité réseau.
Erreur 1722/0x6BA	Le RPC serveur n'est pas disponible.	Un problème est survenu lors de l'accès au RPC service de votre domaine AD. Il peut s'agir d'un problème de service ou de réseau.	Vérifiez que le RPC service s'exécute sur vos contrôleurs de domaine et que les TCP ports `135` `49152-65535` sont accessibles sur votre domaine depuis votre instance de base de données RDS for SQL Server.

Code d’erreur	Description	Causes courantes	Suggestions de dépannage
Erreur 2224 / 0x8B0	Le compte d'utilisateur existe déjà.	Le compte d'ordinateur qui tente d'être ajouté à votre annuaire AD autogéré existe déjà.	Identifiez le compte d'ordinateur en l'exécutant `SELECT @@SERVERNAME` sur votre instance de base de données RDS for SQL Server, puis supprimez-le soigneusement de votre AD autogéré.
Erreur 2242 / 0x8c2	Le mot de passe de cet utilisateur a expiré.	Le mot de passe du compte de service de domaine spécifié via AWS Secrets Manager a expiré.	Mettez à jour le mot de passe du compte de service de domaine utilisé pour joindre votre instance de base de données RDS for SQL Server à votre AD autogéré.

Restaurer une instance de base de données de SQL serveur, puis l'ajouter à un domaine Active Directory autogéré

Vous pouvez restaurer un instantané de base de données ou effectuer une point-in-time restauration (PITR) pour une instance de base de données de SQL serveur, puis l'ajouter à un domaine Active Directory autogéré. Une fois l'instance de base de données restaurée, modifiez cette instance à l'aide du processus expliqué dans Étape 6 : Créer ou modifier une instance de base de données SQL Server afin d'ajouter l'instance de base de données à un domaine AD autogéré.

Utilisation d'Active Directory AWS géré avec RDS pour SQL Server

Vous pouvez l'utiliser AWS Managed Microsoft AD pour authentifier les utilisateurs avec l'authentification Windows lorsqu'ils se connectent à votre instance de base de données RDS pour SQL Server. L'instance de base de données fonctionne avec AWS Directory Service for Microsoft Active Directory, également appelée AWS Managed Microsoft AD, pour activer l'authentification Windows. Lorsque les utilisateurs s'authentifient à une instance de base de données SQL Server jointe au domaine d'approbation, les demandes d'authentification sont transmises à l'annuaire de domaine que vous créez avec AWS Directory Service.

Disponibilité des régions et des versions

Amazon RDS prend en charge l'utilisation uniquement AWS Managed Microsoft AD pour l'authentification Windows. RDS ne prend pas en charge l'utilisation AD Connector. Pour plus d'informations, consultez les ressources suivantes :

- Politique de compatibilité des applications pour AWS Managed Microsoft AD
- Politique de compatibilité des applications pour AD Connector

Pour obtenir des informations sur la disponibilité des versions, consultez Authentification Kerberos avec RDS for SQL Server.

Présentation de la configuration de l'authentification Windows

Amazon RDS utilise le mode mixte pour l'authentification Windows. Cette approche signifie que l'utilisateur principal (nom et mot de passe utilisés pour créer votre instance de base de données SQL Server) utilise l'authentification SQL. Étant donné que le compte utilisateur principal comporte des informations d'identification privilégiées, vous devez limiter l'accès à ce compte.

Pour obtenir l'authentification Windows à l'aide d'un compte Microsoft Active Directory sur site ou auto-géré, créez une approbation de forêt. L'approbation peut être unidirectionnelle ou bidirectionnelle. Pour plus d'informations sur la configuration des approbations forestières à l'aide AWS Directory Service de la section Quand créer une relation de confiance dans le Guide d'AWS Directory Service administration.

Pour configurer l'authentification Windows pour une instance de base de données SQL Server, procédez comme suit. Les étapes sont expliquées de façon plus détaillée dans Configuration de l'authentification Windows pour les instances de base de données SQL Server:

1. AWS Managed Microsoft AD Utilisez-le, soit depuis l' AWS Management Console AWS Directory Service API, soit pour créer un AWS Managed Microsoft AD répertoire.
2. Si vous utilisez l'API AWS CLI ou Amazon RDS pour créer votre instance de base de données SQL Server, créez un rôle AWS Identity and Access Management (IAM). Ce rôle utilise la stratégie IAM gérée `AmazonRDSDirectoryServiceAccess` et autorise Amazon RDS à effectuer des appels vers votre annuaire. Si vous utilisez la console pour créer votre instance de base de données SQL Server, AWS crée le rôle IAM pour vous.

   Pour que le rôle autorise l'accès, le point de terminaison AWS Security Token Service (AWS STS) doit être activé dans la AWS région de votre AWS compte. AWS STS les points de terminaison sont actifs par défaut dans toutes les AWS régions, et vous pouvez les utiliser sans autre action. Pour plus d'informations, consultez Gestion de AWS STS dans une Région AWS dans le Guide de l'utilisateur IAM.
3. Créez et configurez des utilisateurs et des groupes dans l' AWS Managed Microsoft AD annuaire à l'aide des outils Microsoft Active Directory. Pour plus d'informations sur la création d'utilisateurs et de groupes dans votre Active Directory, consultez Gérer les utilisateurs et les groupes dans AWS Managed Microsoft AD dans le Guide d'administration AWS Directory Service .
4. Si vous prévoyez de localiser le répertoire et l'instance de base de données dans un emplacement différent VPCs, activez le trafic inter-VPC.
5. Utilisez Amazon RDS pour créer une nouvelle instance de base de données SQL Server à partir de la console ou de l'API Amazon RDS. AWS CLI Dans la demande de création, vous indiquez l'identifiant du domaine (identifiant « `d-*` ») généré lors de la création de votre annuaire, ainsi que le nom du rôle que vous avez créé. Vous pouvez également modifier une instance de base de données SQL Server pour qu'elle utilise l'authentification Windows en configurant les paramètres de domaine et de rôle IAM de l'instance de base de données.
6. Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données SQL Server de la même manière qu'à n'importe quelle instance de base de données. Comme l'instance de base de données est jointe au AWS Managed Microsoft AD domaine, vous pouvez configurer des connexions et des utilisateurs SQL Server à partir des utilisateurs et des groupes Active Directory de leur domaine. (Ceux-ci sont connus sous le nom de connexions SQL Server « Windows ».) Les autorisations pour la base de données sont gérées via des autorisations SQL Server standard accordées et révoquées en fonction des connexions Windows.

Lorsque vous créez une instance de base de données RDS pour SQL Server connectée à un domaine à l'aide de la console Amazon RDS, le rôle IAM est AWS automatiquement créé. `rds-directoryservice-access-role` Ce rôle est essentiel pour gérer les instances connectées au domaine et est requis pour les opérations suivantes :

- Apporter des modifications de configuration aux instances SQL Server connectées à un domaine
- Gestion des paramètres d'intégration d'Active Directory
- Réalisation d'opérations de maintenance sur des instances jointes à un domaine

> **Important**
>
> Si vous supprimez le rôle `rds-directoryservice-access-role` IAM, vous ne pouvez pas apporter de modifications à votre instance SQL Server connectée au domaine via la console ou l'API Amazon RDS. Toute tentative de modification de l'instance entraîne l'affichage d'un message d'erreur indiquant : Vous n'avez pas l'autorisation d'utiliser iam : CreateRole. Pour demander l'accès, copiez le texte suivant et envoyez-le à votre AWS administrateur.
> Cette erreur se produit car Amazon RDS doit recréer le rôle pour gérer la connexion au domaine, mais ne dispose pas des autorisations nécessaires. De plus, cette erreur n'est pas enregistrée CloudTrail, ce qui peut compliquer le dépannage.

Si vous le supprimez accidentellement`rds-directoryservice-access-role`, vous devez être `iam:CreateRole` autorisé à le recréer avant de pouvoir apporter des modifications à votre instance SQL Server connectée au domaine. Pour recréer le rôle manuellement, assurez-vous qu'il est associé `AmazonRDSDirectoryServiceAccess` à la politique gérée et à la relation de confiance appropriée permettant au service RDS d'assumer le rôle.

Création d'un point de terminaison pour l'authentification Kerberos

L'authentification basée sur Kerberos nécessite que le point de terminaison soit le nom d'hôte spécifié par le client, un point, puis le nom de domaine complet (). FQDN Par exemple, l'exemple suivant illustre un point de terminaison à utiliser avec l'authentification basée sur Kerberos. Dans cet exemple, le nom d'hôte de l'instance de base de données du SQL serveur est `ad-test` et le nom de domaine est`corp-ad.company.com`.

```
ad-test.corp-ad.company.com
```

Pour vérifier que votre connexion utilise Kerberos, exécutez la requête suivante :

```
SELECT net_transport, auth_scheme
  FROM sys.dm_exec_connections
 WHERE session_id = @@SPID;
```

Configuration de l'authentification Windows pour les instances de base de données SQL Server

Vous utilisez AWS Directory Service for Microsoft Active Directory, également appelé AWS Managed Microsoft AD, pour configurer l'authentification Windows pour une instance de base de données SQL Server. Pour configurer l'authentification Windows, procédez comme suit.

Étape 1 : créer un répertoire à l'aide du AWS Directory Service for Microsoft Active Directory

AWS Directory Service crée un Microsoft Active Directory entièrement géré dans le AWS cloud. Lorsque vous créez un AWS Managed Microsoft AD annuaire, il AWS Directory Service crée deux contrôleurs de domaine et des serveurs DNS (Domain Name Service) en votre nom. Les serveurs de répertoire sont créés dans deux sous-réseaux sur deux zones de disponibilité différentes avec un VPC. Cette redondance permet de s'assurer que votre répertoire reste accessible y compris en cas de défaillance.

Lorsque vous créez un AWS Managed Microsoft AD répertoire, il AWS Directory Service exécute les tâches suivantes en votre nom :

- Configuration de Microsoft Active Directory dans le VPC.
- Création d'un compte d'administrateur d'annuaire avec le nom d'utilisateur Admin et le mot de passe spécifié. Ce compte est utilisé pour gérer votre annuaire.
- Création d'un groupe de sécurité pour les contrôleurs de l'annuaire.

Lorsque vous lancez un AWS Directory Service for Microsoft Active Directory, AWS crée une unité organisationnelle (UO) qui contient tous les objets de votre répertoire. Cette unité d'organisation, qui porte le nom NetBIOS que vous avez saisi lorsque vous avez créé votre annuaire, est située dans la racine du domaine. La racine du domaine est détenue et gérée par AWS.

Le compte admin qui a été créé avec votre annuaire AWS Managed Microsoft AD dispose des autorisations pour les activités administratives les plus courantes pour votre unité d'organisation :

- Créer, mettre à jour ou supprimer des utilisateurs, des groupes et des ordinateurs

- Ajouter des ressources à votre domaine, comme des serveurs de fichiers ou d'impression, puis attribuer des autorisations pour ces ressources aux utilisateurs et groupes dans votre unité d'organisation.
- Créez OUs des conteneurs supplémentaires.
- Déléguer des autorités.
- Créer et associer des stratégies de groupes.
- Restaurer des objets supprimés de la corbeille Active Directory.
- Exécutez les PowerShell modules Windows AD et DNS sur le service Web Active Directory.

Le compte admin dispose également de droits pour exécuter les activités suivantes au niveau du domaine :

- Gérer les configurations DNS (ajouter, supprimer ou mettre à jour des enregistrements, des zones et des redirecteurs)
- Afficher les journaux d'événements DNS.
- Afficher les journaux d'événements de sécurité.

Pour créer un répertoire avec AWS Managed Microsoft AD

1. Dans le panneau de navigation de la console AWS Directory Service, choisissez Annuaires, puis Configurer un annuaire.
2. Choisissez AWS Managed Microsoft AD. Il s'agit de la seule option prise en charge actuellement pour être utilisée avec Amazon RDS.
3. Choisissez Suivant.
4. Sur la page Enter directory information (Saisir les détails du répertoire), renseignez les informations suivantes :

   Edition

   Choisissez l'édition qui correspond à vos besoins.

   Nom de DNS de l'annuaire

   Nom complet de l'annuaire, par exemple `corp.example.com`. Les noms de plus de 47 caractères ne sont pas pris en charge par SQL Server.

Nom NetBIOS de l'annuaire

Nom court facultatif pour l'annuaire, par exemple `CORP`.

Description de l'annuaire

Description facultative de l'annuaire.

Mot de passe administrateur

Mot de passe de l'administrateur de l'annuaire. Le processus de création d'un annuaire crée un compte d'administrateur avec le nom d'utilisateur Admin et ce mot de passe.

Le mot de passe de l'administrateur de l'annuaire ne peut pas inclure le terme `admin`. Le mot de passe est sensible à la casse et doit comporter entre 8 et 64 caractères. Il doit également contenir au moins un caractère de trois des quatre catégories suivantes :

- Lettres minuscules (a-z)
- Lettres majuscules (A-Z)
- Chiffres (0-9)
- Caractères non alphanumériques (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)

Confirmer le mot de passe

Saisissez à nouveau le mot de passe de l'administrateur.

5. Choisissez Suivant.
6. Sur la page Choose VPC and subnets (Choisir un VPC et des sous-réseaux), indiquez les informations suivantes :

VPC

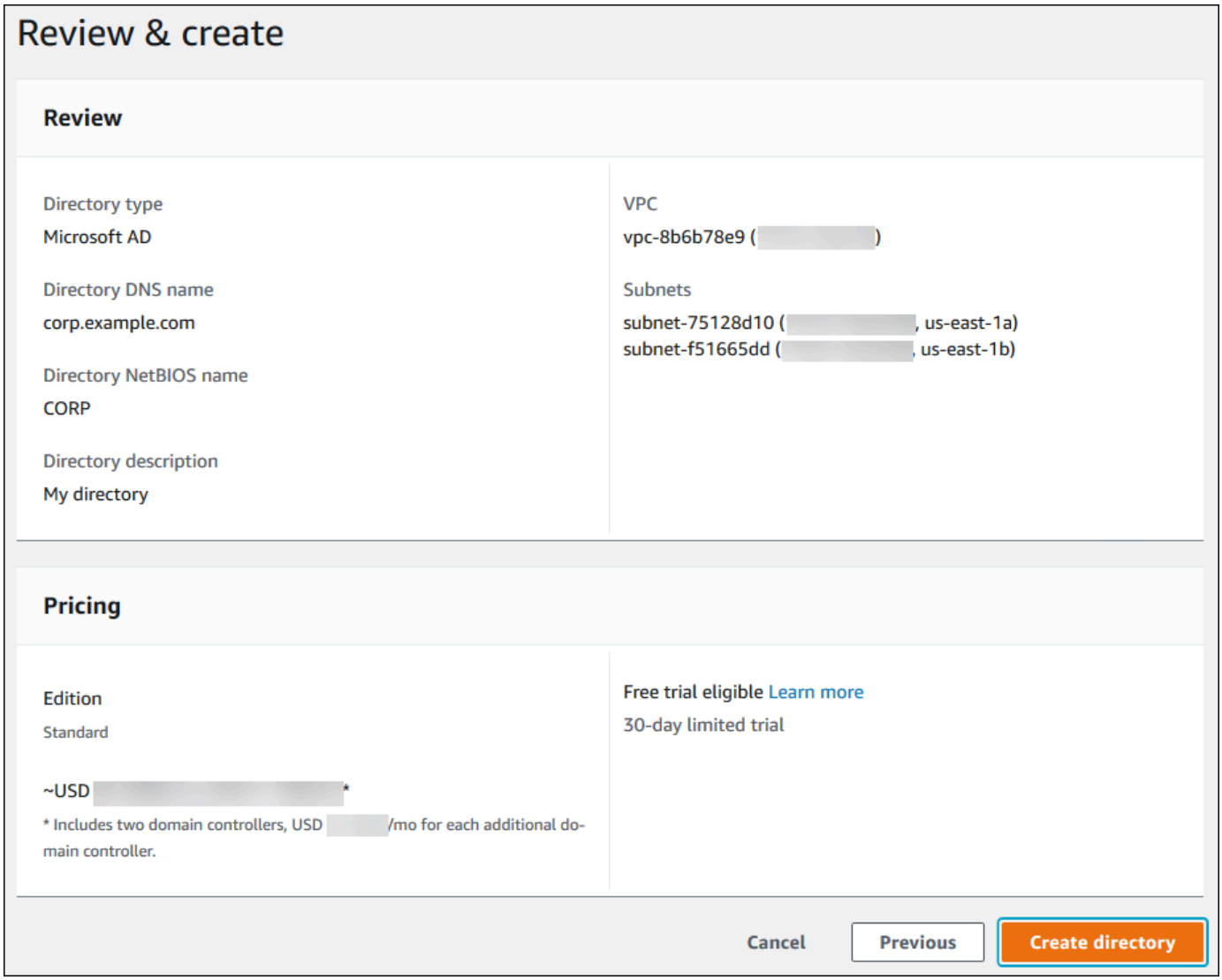
Sélectionnez le VPC pour l'annuaire.

> **Note**
>
> Vous pouvez localiser le répertoire et l'instance de base de données différemment VPCs, mais si c'est le cas, assurez-vous d'activer le trafic inter-VPC. Pour de plus amples informations, veuillez consulter Étape 4 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données.

Sous-réseaux

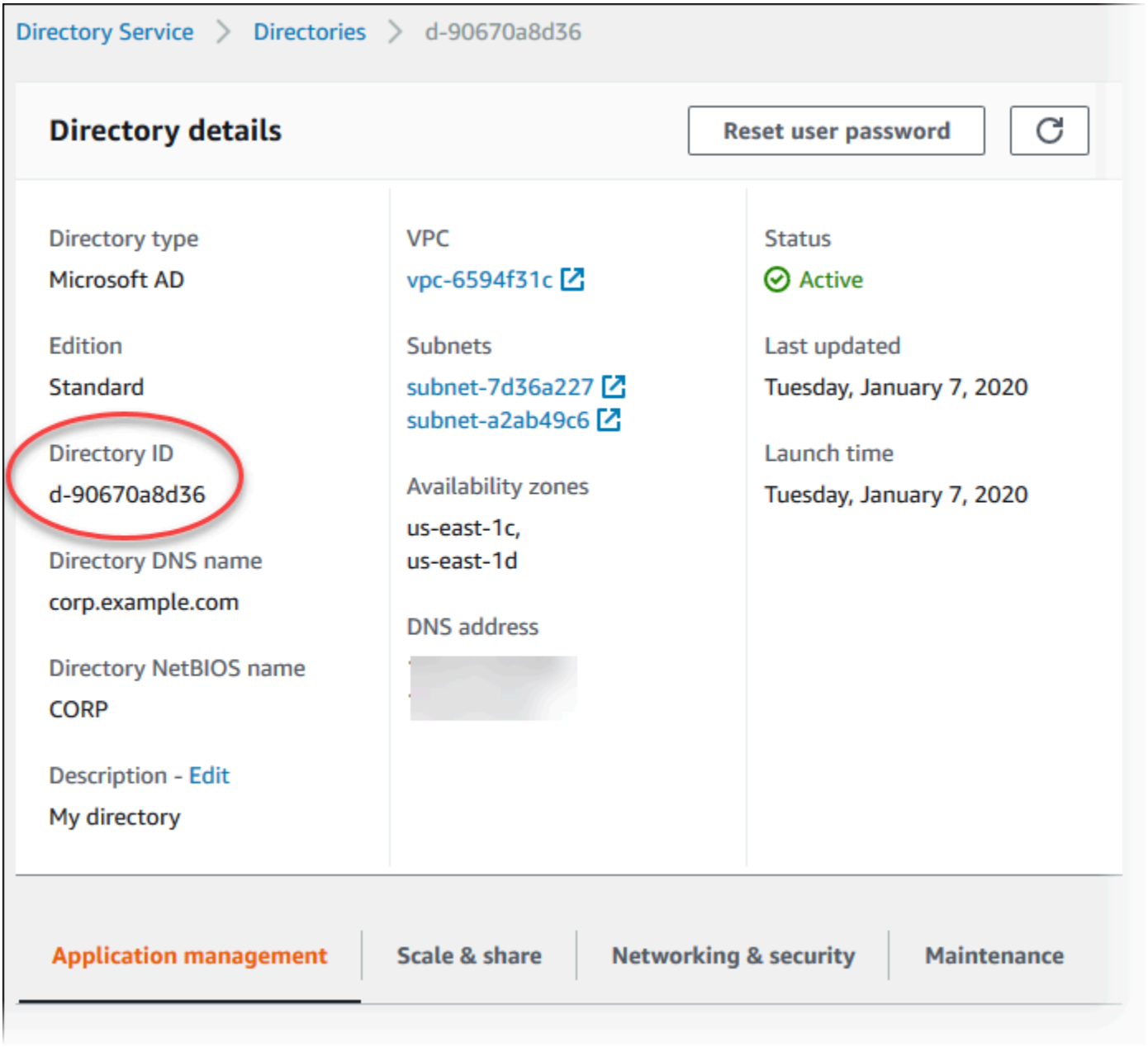
Choisissez les sous-réseaux pour les serveurs d'annuaires. Les deux sous-réseaux doivent être dans des zones de disponibilité différentes.

7. Choisissez Suivant.
8. Vérifiez les informations de l'annuaire. Si vous devez apporter des modifications, choisissez Previous (Précédent). Lorsque les informations sont correctes, choisissez Create directory (Créer l'annuaire).



La création de l'annuaire prend plusieurs minutes. Lorsqu'il est créé, la valeur du champ Statut devient Actif.

Pour consulter les informations relatives à votre annuaire, choisissez l'ID de l'annuaire dans la liste. Notez la valeur de ID de l'annuaire. Vous en aurez besoin pour créer ou modifier votre instance de base de données SQL Server.



Étape 2 : Créer le rôle IAM qui sera utilisé par Amazon RDS

Si vous utilisez la console pour créer votre instance de base de données SQL Server, vous pouvez ignorer cette étape. Si vous utilisez la CLI ou l'API RDS pour créer votre instance de base de données SQL Server, vous devez créer un rôle IAM qui utilise la stratégie IAM gérée `AmazonRDSDirectoryServiceAccess`. Ce rôle permet à Amazon RDS de passer des appels AWS Directory Service pour vous.

Si vous utilisez une politique personnalisée pour rejoindre un domaine, au lieu d'utiliser la `AmazonRDSDirectoryServiceAccess` politique AWS-managed, assurez-vous d'autoriser l'`ds:GetAuthorizedApplicationDetails`action. Cette exigence est effective à partir de juillet 2019, en raison d'une modification de l' AWS Directory Service API.

La stratégie IAM suivante, `AmazonRDSDirectoryServiceAccess`, permet d'accéder à AWS Directory Service.

Example Politique IAM pour fournir l'accès à AWS Directory Service

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
            "ds:DescribeDirectories",
            "ds:AuthorizeApplication",
            "ds:UnauthorizeApplication",
            "ds:GetAuthorizedApplicationDetails"
        ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des relations d'approbation basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.

- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la relation d'approbation, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'Amazon Resource Name (ARN) complet des ressources qui accèdent au rôle. Pour l'authentification Windows, veillez à inclure les instances de base de données, comme illustré dans l'exemple suivant.

Example relation d'approbation avec la clé de contexte de condition globale pour l'authentification Windows

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": [

 "arn:aws:rds:Region:my_account_ID:db:db_instance_identifier"
                    ]
                }
            }
        }
    ]
}
```

Créez un rôle IAM à l'aide de cette politique IAM et de cette relation d'approbation. Pour plus d'informations sur la création de rôles IAM, veuillez consulter Création de stratégies gérées par le client dans le IAM Guide de l'utilisateur.

Étape 3 : Créer et configurer des utilisateurs et des groupes

Vous pouvez créer des utilisateurs et des groupes avec l'outil Utilisateurs et ordinateurs Active Directory. Cet outil fait partie des outils Services AD DS (Active Directory Domain Services) et Services AD LDS (Active Directory Lightweight Directory Services). Les utilisateurs représentent des individus ou des entités individuelles qui ont accès à votre annuaire. Les groupes sont très utiles pour octroyer ou refuser des privilèges à des groupes d'utilisateurs, plutôt que d'appliquer ces privilèges à chaque utilisateur.

Pour créer des utilisateurs et des groupes dans un AWS Directory Service annuaire, vous devez être connecté à une EC2 instance Windows membre de l' AWS Directory Service annuaire. Vous devez également être connecté en tant qu'utilisateur disposant de privilèges pour créer des utilisateurs et des groupes. Pour plus d'informations, consultez la section Ajouter des utilisateurs et des groupes (Simple AD et AWS Managed Microsoft AD) dans le Guide d'AWS Directory Service administration.

Étape 4 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données

Si vous avez l'intention de rechercher l'annuaire et l'instance de base de données dans le même VPC, ignorez cette étape et passez à Étape 5 : Créer ou modifier une instance de base de données SQL Server.

Si vous prévoyez de localiser le répertoire et l'instance de base de données différemment VPCs, configurez le trafic inter-VPC à l'aide du peering VPC ou de Transit Gateway.AWS

La procédure suivante active le trafic entre les utilisateurs de VPCs l'appairage VPC.
Suivez les instructions de Qu'est-ce que l'appairage de VPC ? dans le Guide de l'appairage Amazon Virtual Private Cloud.

Pour activer le trafic entre VPC à l'aide de l'appairage de VPC

1. Configurez les règles de routage de VPC appropriées afin de veiller à ce que le trafic réseau puisse être acheminé dans les deux sens.
2. Assurez-vous que le groupe de sécurité de l'instance de base de données puisse recevoir le trafic entrant depuis le groupe de sécurité de cet annuaire.
3. Assurez-vous qu'il n'existe aucune règle de liste de contrôle d'accès (ACL) pour bloquer le trafic.

Si le répertoire appartient à un autre AWS compte, vous devez le partager.

Pour partager le répertoire entre AWS comptes

1. Commencez à partager le répertoire avec le AWS compte dans lequel l'instance de base de données sera créée en suivant les instructions du Tutoriel : Partage de votre AWS Managed Microsoft AD répertoire pour une jonction de EC2 domaine fluide dans le Guide d'AWS Directory Service administration.
2. Connectez-vous à la AWS Directory Service console à l'aide du compte de l'instance de base de données et assurez-vous que le domaine possède le `SHARED` statut requis avant de continuer.
3. Lorsque vous êtes connecté à la AWS Directory Service console à l'aide du compte de l'instance de base de données, notez la valeur de l'ID du répertoire. Vous utilisez cet ID pour joindre l'instance de base de données au domaine.

Étape 5 : Créer ou modifier une instance de base de données SQL Server

Créez ou modifiez une instance de base de données SQL Server en vue de son utilisation avec votre annuaire. Vous pouvez utiliser la console, la CLI ou l'API RDS pour associer une instance de base de données à un annuaire. Vous pouvez effectuer cette opération de différentes manières :

- Créez une nouvelle instance de base de données SQL Server à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

  Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données SQL Server existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstanceModify RDS API.

  Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Restaurez une instance de base de données SQL Server à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération DBInstance Restore DBSnapshot From RDS API.

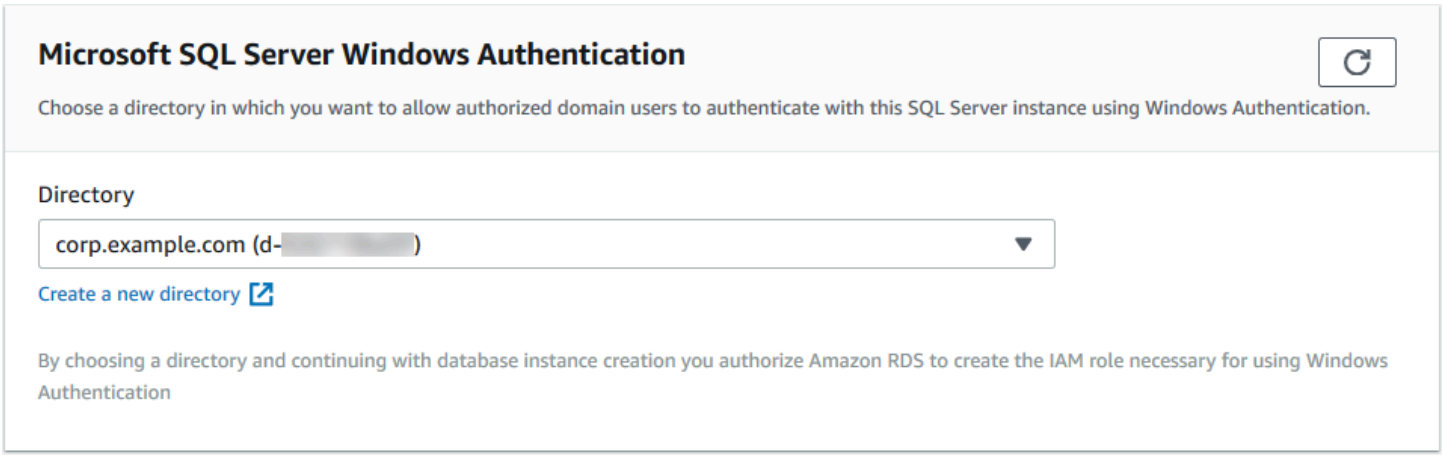
  Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.
- Restaurez une instance de base de données SQL Server à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération Restore DBInstance ToPointInTime RDS API.

Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

L'authentification Windows est uniquement prise en charge pour les instances de base de données SQL dans un VPC.

Pour que l'instance de base de données puisse utiliser l'annuaire de domaine que vous avez créé, les éléments suivants sont nécessaires :

- Pour Annuaire, vous devez choisir l'identifiant du domaine (d-*ID*) généré lors de la création de l'annuaire.
- Assurez-vous que le groupe de sécurité VPC dispose d'une règle sortante qui permet à l'instance de base de données de communiquer avec l'annuaire.



Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de base de données puisse utiliser le répertoire que vous avez créé :

- Pour le paramètre `--domain`, vous devez indiquer l'identifiant du domaine (d-*ID*) généré lors de la création de l'annuaire.
- Pour le paramètre `--domain-iam-role-name`, utilisez le rôle que vous avez créé qui utilise la stratégie IAM gérée `AmazonRDSDirectoryServiceAccess`.

Par exemple, la commande de CLI suivante modifie une instance de base de données de façon à utiliser un annuaire.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --domain d-ID \
    --domain-iam-role-name role-name
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --domain d-ID ^
    --domain-iam-role-name role-name
```

> **Important**
>
> Si vous modifiez une instance de base de données de façon à activer l'authentification Kerberos, redémarrez l'instance de base de données après avoir effectué la modification.

Étape 6 : Créer des connexions SQL Server pour l'authentification Windows

Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données SQL Server de la même manière qu'à n'importe quelle instance de base de données. Comme l'instance de base de données est jointe au AWS Managed Microsoft AD domaine, vous pouvez configurer des connexions et des utilisateurs SQL Server. Vous effectuez cette opération à partir des utilisateurs et groupes Active Directory de votre domaine. Les autorisations pour la base de données sont gérées via des autorisations SQL Server standard accordées et révoquées en fonction des connexions Windows.

Pour qu'un utilisateur Active Directory puisse s'authentifier à SQL Server, une connexion Windows SQL Server doit exister pour l'utilisateur ou un groupe dont l'utilisateur est membre. Un contrôle précis des accès est géré par l'attribution ou la révocation d'autorisations pour ces connexions SQL Server. Un utilisateur qui n'a pas de connexion SQL Server ou qui n'appartient pas à un groupe avec une telle connexion ne peut pas accéder à l'instance de base de données SQL Server.

L'autorisation ALTER ANY LOGIN est requise pour créer une connexion SQL Server Active Directory. Si vous n'avez pas créé de connexion avec cette autorisation, connectez vous en tant qu'utilisateur principal de l'instance de base de données à l'aide de l'authentification SQL Server.

Exécutez une commande DDL (Data Definition Language) telle que l'exemple suivant afin de créer une connexion SQL Server pour un utilisateur ou un groupe Active Directory.

> **Note**
>
> Spécifiez les utilisateurs et les groupes à l'aide du nom de connexion antérieur à Windows 2000 au format *domainName**login_name*. Vous ne pouvez pas utiliser un nom d'utilisateur principal (UPN) au format *login_name*@*DomainName*.
> Vous ne pouvez créer une connexion d'authentification Windows sur une instance RDS pour SQL Server qu'à l'aide d'instructions T-SQL. Vous ne pouvez pas utiliser le studio de gestion SQL Server pour créer une connexion d'authentification Windows.

```
USE [master]
GO
CREATE LOGIN [mydomain\myuser] FROM WINDOWS WITH DEFAULT_DATABASE = [master],
 DEFAULT_LANGUAGE = [us_english];
GO
```

Pour plus d'informations, consultez CREATE LOGIN (Transact-SQL) dans la documentation de Microsoft Developer Network.

Les utilisateurs (personnes et applications) de votre domaine peuvent désormais se connecter à l'instance RDS for SQL Server à partir d'un ordinateur client joint au domaine à l'aide de l'authentification Windows.

Gestion d'une instance de base de données dans un domaine

Vous pouvez utiliser la console ou Amazon RDS API pour gérer votre instance de base de données et sa relation avec votre domaine. AWS CLI Par exemple, vous pouvez déplacer l'instance de base de données dans, hors ou entre des domaines.

Par exemple, à l'aide d'Amazon RDSAPI, vous pouvez effectuer les opérations suivantes :

- Pour réessayer de joindre un domaine en raison d'un échec d'adhésion, utilisez l'odifyDBInstanceAPIopération M et spécifiez l'ID de répertoire de l'adhésion actuelle.
- Pour mettre à jour le nom du IAM rôle pour l'adhésion, utilisez l'`ModifyDBInstanceAPI`opération et spécifiez l'ID de répertoire du membre actuel et le nouveau IAM rôle.

- Pour supprimer une instance de base de données d'un domaine, utilisez l'`ModifyDBInstance`APIopération et `none` spécifiez-la comme paramètre de domaine.
- Pour déplacer une instance de base de données d'un domaine à un autre, utilisez l'`ModifyDBInstance`APIopération et spécifiez l'identifiant de domaine du nouveau domaine comme paramètre de domaine.
- Pour répertorier les membres de chaque instance de base de données, utilisez l'escribeDBInstancesAPIopération D.

Présentation de l'appartenance au domaine

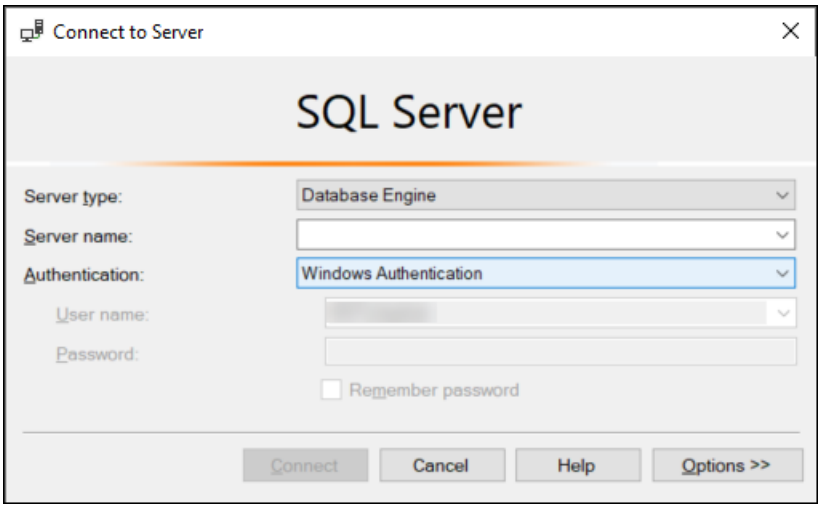
Après la création ou la modification de votre instance de base de données, l'instance devient un membre du domaine. La AWS console indique le statut de l'appartenance au domaine pour l'instance de base de données. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- joined–L'instance est membre du domaine.
- joining–L'instance est en train de devenir membre du domaine.
- pending-join – L'appartenance de l'instance est en attente.
- pending-maintenance-join— AWS tentera de faire de l'instance un membre du domaine lors de la prochaine fenêtre de maintenance planifiée.
- pending-removal–La suppression de l'instance du domaine est en attente.
- pending-maintenance-removal— AWS tentera de supprimer l'instance du domaine lors de la prochaine fenêtre de maintenance planifiée.
- failed–Un problème de configuration a empêché l'instance d'effectuer la jonction du domaine. Vérifiez et corrigez votre configuration avant d'émettre à nouveau la commande de modification de l'instance.
- removing–La suppression de l'instance du domaine est en cours.

Une demande pour devenir membre d'un domaine peut échouer en raison d'un problème de connectivité réseau ou d'un IAM rôle incorrect. Par exemple, vous pouvez créer une instance de base de données ou modifier une instance existante et faire échouer la tentative pour que l'instance de base de données devienne membre d'un domaine. Dans ce cas, émettez à nouveau la commande pour créer ou modifier l'instance de base de données, ou modifiez l'instance nouvellement créée pour rejoindre le domaine.

Connexion au SQL serveur avec authentification Windows

Pour vous connecter au SQL serveur avec l'authentification Windows, vous devez être connecté à un ordinateur joint au domaine en tant qu'utilisateur du domaine. Après avoir lancé SQL Server Management Studio, choisissez Windows Authentication comme type d'authentification, comme indiqué ci-dessous.



Restauration d'une instance de base de données SQL Server puis ajout de cette instance à un domaine

Vous pouvez restaurer un instantané de base de données ou effectuer une point-in-time restauration (PITR) pour une instance de base de données SQL Server, puis l'ajouter à un domaine. Une fois que l'instance de base de données est restaurée, modifiez l'instance à l'aide du processus expliqué dans Étape 5 : Créer ou modifier une instance de base de données SQL Server afin d'ajouter l'instance de base de données à un domaine.

Mises à niveau du moteur de base de données Microsoft SQL Server

Lorsque Amazon RDS prend en charge une nouvelle version d'un moteur de base de données, vous pouvez mettre à niveau vos instances de base de données vers cette nouvelle version. Il existe deux types de mises à niveau pour les instances de base de données SQL Server : les mises à niveau de version majeure et les mises à niveau de version mineure.

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. Par conséquent, vous devez effectuer manuellement les mises à niveau des versions majeures de vos instances de base de données. Vous pouvez lancer une mise à niveau de version majeure en modifiant votre instance de base de données. Cependant, avant d'effectuer une mise à niveau de version majeure, nous vous recommandons de tester la mise à niveau en suivant les étapes décrites dans Test et mise RDS à niveau SQL du serveur.

Les mises à niveau de versions mineures contiennent uniquement des modifications rétrocompatibles avec les applications existantes. Vous pouvez mettre à niveau la version mineure de votre instance de base de données de deux manières :

- Manuellement : modifiez votre instance de base de données pour lancer la mise à niveau
- Automatiquement : activez les mises à niveau automatiques des versions mineures pour votre instance de base de données

Lorsque vous activez les mises à niveau automatiques des versions mineures, RDS pour SQL Server met automatiquement à niveau votre instance de base de données pendant les fenêtres de maintenance planifiées lorsque des mises à jour de sécurité critiques sont disponibles dans une version mineure plus récente.

Pour les versions mineures du moteur après`16.00.4120.1`,`15.00.4365.2`,`14.00.3465.1`,,`13.00.6435.1`,, les protocoles de sécurité suivants sont désactivés par défaut :

- `rds.tls10`(protocole TLS 1.0)
- `rds.tls11`(protocole TLS 1.1)
- `rds.rc4`(RC4 chiffre)
- `rds.curve25519`(Chiffrement Curve25519)

- `rds.3des168`(Chiffrement triple DES)

Pour les versions antérieures du moteur, Amazon RDS active ces protocoles de sécurité par défaut.

```
...

"ValidUpgradeTarget": [
    {
        "Engine": "sqlserver-se",
        "EngineVersion": "14.00.3281.6.v1",
        "Description": "SQL Server 2017 14.00.3281.6.v1",
        "AutoUpgrade": false,
        "IsMajorVersionUpgrade": false
    }
...
```

Pour de plus amples informations sur l'exécution de mises à niveau, veuillez consulter Mise à niveau d'une instance de base de données SQL Server. Pour de plus amples informations sur les versions SQL Server disponibles sur Amazon RDS, veuillez consulter Amazon RDS for Microsoft SQL Server.

Rubriques

- Mises à niveau de version majeures RDS pour for SQL Server
- Considérations relatives aux mises à niveau de SQL Server
- Test et mise RDS à niveau SQL du serveur
- Mise à niveau d'une instance de base de données SQL Server
- Mise à niveau des instances de base de données obsolètes avant la fin de la prise en charge

Mises à niveau de version majeures RDS pour for SQL Server

Amazon prend RDS actuellement en charge les mises à niveau des versions majeures suivantes vers une instance de base de données Microsoft SQL Server.

Vous pouvez mettre à niveau votre instance de base de données existante vers SQL Server 2017 ou 2019 à partir de n'importe quelle version, à l'exception de SQL Server 2008. Pour effectuer une mise à niveau depuis SQL Server 2008, effectuez d'abord une mise à niveau vers l'une des autres versions.

Version actuelle	Versions de mise à niveau prises en charge
SQLServeur 2019	SQLServeur 2022
SQLServeur 2017	SQLServeur 2022 SQLServeur 2019
SQLServeur 2016	SQLServeur 2022 SQLServeur 2019 SQLServeur 2017

Vous pouvez utiliser une AWS CLI requête, telle que l'exemple suivant, pour rechercher les mises à niveau disponibles pour une version de moteur de base de données donnée.

Example

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
    --engine sqlserver-se \
    --engine-version 14.00.3281.6.v1 \
    --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" \
    --output table
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
    --engine sqlserver-se ^
    --engine-version 14.00.3281.6.v1 ^
    --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" ^
    --output table
```

Le résultat indique que vous pouvez mettre à niveau la version 14.00.3281.6 vers les dernières versions disponibles de SQL Server 2017 ou 2019.

```
--------------------------
```

```
|DescribeDBEngineVersions|
+------------------------+
|      EngineVersion     |
+------------------------+
|  14.00.3294.2.v1       |
|  14.00.3356.20.v1      |
|  14.00.3381.3.v1       |
|  14.00.3401.7.v1       |
|  14.00.3421.10.v1      |
|  14.00.3451.2.v1       |
|  15.00.4043.16.v1      |
|  15.00.4073.23.v1      |
|  15.00.4153.1.v1       |
|  15.00.4198.2.v1       |
|  15.00.4236.7.v1       |
+------------------------+
```

Niveau de compatibilité de base de données

Vous pouvez utiliser les niveaux de compatibilité des bases de données Microsoft SQL Server pour ajuster certains comportements de base de données afin d'imiter les versions précédentes de SQL Server. Pour de plus amples informations, veuillez consulter Niveau de compatibilité dans la documentation de Microsoft. Lorsque vous mettez à niveau votre instance de base de données, toutes les bases de données existantes restent à leur niveau de compatibilité initial.

Vous pouvez modifier le niveau de compatibilité d'une base de données à l'aide de la ALTER DATABASE commande. Par exemple, pour modifier une base de données nommée `customeracct` afin qu'elle soit compatible avec SQL Server 2016, exécutez la commande suivante :

```
ALTER DATABASE customeracct SET COMPATIBILITY_LEVEL = 130
```

Considérations relatives aux mises à niveau de SQL Server

Amazon RDS prend deux instantanés de base de données au cours du processus de mise à niveau. Le premier instantané de base de données porte sur l'instance de base de données avant que toute modification de mise à niveau soit apportée. Le second instantané de base de données est pris à la fin de la mise à niveau.

> **Note**
>
> Amazon RDS ne prend des instantanés de base de données que si vous avez défini la période de rétention des sauvegardes de votre instance de base de données sur un nombre supérieur à 0. Pour modifier la période de rétention des sauvegardes, consultez Modification d'une RDS instance de base de données Amazon.

Une fois la mise à niveau terminée, vous ne pouvez pas rétablir la version précédente du moteur de base de données. Si vous souhaitez revenir à la version précédente, restaurez l'instantané de base de données pris avant la mise à niveau pour créer une nouvelle instance de base de données.

Au cours de la mise à niveau d'une version mineure ou majeure de SQL Server, les métriques Espace de stockage disponible et Profondeur de la file d'attente indiquent `-1`. Une fois la mise à niveau terminée, les deux métriques reviennent à la normale.

Avant de mettre à niveau votre instance SQL Server, consultez les informations suivantes.

Rubriques

- Bonnes pratiques avant de lancer une mise à niveau
- Considérations multi-AZ
- Considérations relatives aux réplicas en lecture
- Considérations relatives au groupe d'options
- Considérations relatives au groupe de paramètres

Bonnes pratiques avant de lancer une mise à niveau

Avant de démarrer le processus de mise à niveau, mettez en œuvre les étapes préparatoires suivantes pour optimiser les performances de mise à niveau et minimiser les problèmes potentiels :

Gestion du calendrier et de la charge de travail

- Planifiez des mises à niveau pendant les périodes de faible volume de transactions.
- Minimisez les opérations d'écriture pendant la fenêtre de mise à niveau.

  Cela permet à Amazon RDS d'effectuer les mises à niveau plus rapidement en réduisant le nombre de fichiers de sauvegarde du journal des transactions que RDS doit restaurer lors secondary-to-primary du couplage.

Gestion des transactions

- Identifiez et surveillez les transactions de longue durée.
- Assurez-vous que toutes les transactions critiques sont validées avant de commencer la mise à niveau.
- Empêchez les transactions de longue durée pendant la période de mise à niveau.

Optimisation des fichiers journaux

Passez en revue et optimisez les fichiers journaux de transactions :

- Réduisez les fichiers journaux surdimensionnés.
- Réduisez les modèles de consommation élevée de grumes.
- Gérez les fichiers journaux virtuels (VLFs).
- Maintenez un espace libre suffisant pour les opérations normales.

Considérations multi-AZ

Amazon RDS prend en charge les déploiements multi-AZ pour les instances de base de données exécutant Microsoft SQL Server en utilisant SQL Server Database Mirroring (DBM) ou Always On Availability Groups (). AGs Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.

Dans un déploiement multi-AZ (Mirroring/AlwaysOn), lorsqu'une mise à niveau est demandée, RDS suit une stratégie de mise à niveau progressive pour les instances principales et secondaires. Les mises à niveau continues garantissent qu'au moins une instance est disponible pour les transactions pendant la mise à niveau de l'instance secondaire. La panne ne devrait durer que le temps d'un basculement.

Au cours de la mise à niveau, RDS supprime l'instance secondaire de la configuration multi-AZ, effectue une mise à niveau de l'instance secondaire et restaure toutes les sauvegardes du journal des transactions de l'instance principale effectuées pendant la période de déconnexion. Une fois toutes les sauvegardes de journaux restaurées, RDS joint le secondaire mis à niveau au principal. Lorsque toutes les bases de données sont synchronisées, RDS effectue un basculement vers l'instance secondaire mise à niveau. Une fois le basculement terminé, RDS procède à la mise à niveau de l'ancienne instance principale, restaure toutes les sauvegardes du journal des transactions et l'associe à la nouvelle instance principale.

Pour minimiser cette durée de basculement, nous recommandons d'utiliser le point de terminaison de l'écouteur du groupe de AlwaysOn AGs disponibilité lorsque vous utilisez des bibliothèques clientes

qui prennent en charge l'option de `MultiSubnetFailover` connexion dans la chaîne de connexion. Lors de l'utilisation du point de terminaison du groupe de disponibilité, les temps de basculement sont généralement inférieurs à 10 secondes, mais cette durée n'inclut aucun temps supplémentaire de reprise après incident.

Considérations relatives aux réplicas en lecture

Lors d'une mise à niveau de la version de la base de données, Amazon RDS met à niveau tous vos réplicas en lecture ainsi que l'instance de base de données principale. Amazon RDS ne prend pas en charge les mises à niveau de version de base de données sur les réplicas en lecture séparément. Pour plus d'informations sur les réplicas en lecture, consultez Utilisation des réplicas en lecture pour Microsoft SQL Server dans Amazon RDS.

Lorsque vous effectuez une mise à niveau de la version de la base de données de l'instance de base de donnée principale, tous les réplicas en lecture sont également automatiquement mis à niveau. Amazon RDS mettra à niveau tous les réplicas en lecture simultanément avant de mettre à niveau l'instance de base de données primaire. Les réplicas en lecture peuvent ne pas être disponibles tant que la mise à niveau de la version de la base de données sur l'instance de base de données principale n'est pas terminée.

Considérations relatives au groupe d'options

Si votre instance de base de données utilise un groupe d'options de base de données personnalisé, Amazon RDS ne peut pas toujours attribuer automatiquement un nouveau groupe d'options à votre instance de base de données. Par exemple, lorsque vous procédez à une mise à niveau vers une nouvelle version majeure, vous devez spécifier un nouveau groupe d'options. Nous vous recommandons de créer un nouveau groupe d'options et d'y ajouter les mêmes options qu'à votre groupe d'options personnalisé existant.

Pour plus d’informations, consultez Création d'un groupe d'options ou Copie d'un groupe d'options.

Considérations relatives au groupe de paramètres

Si votre instance de base de données utilise un groupe de paramètres de base de données personnalisé :

- Amazon RDS redémarre automatiquement l'instance de base de données après une mise à niveau.
- Dans certains cas, RDS n'est pas en mesure d'attribuer automatiquement un nouveau groupe de paramètres à votre instance de base de données.

Par exemple, lorsque vous procédez à une mise à niveau vers une nouvelle version majeure, vous devez spécifier un nouveau groupe de paramètres. Nous vous recommandons de créer un nouveau groupe de paramètres et de configurer les mêmes paramètres que ceux de votre groupe de paramètres personnalisé existant.

Pour plus d'informations, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon ou Copier un groupe de paramètres de base de données dans RDSAmazon.

Test et mise RDS à niveau SQL du serveur

Avant d'effectuer une mise à niveau de version majeure sur votre instance de base de données, vous devez tester soigneusement la compatibilité de votre base de données et de toutes les applications qui y accèdent avec la nouvelle version. Nous vous recommandons d'utiliser la procédure suivante.

Pour tester une mise à niveau de version majeure

1. Consultez Upgrade SQL Server dans la documentation Microsoft pour connaître la nouvelle version du moteur de base de données pour voir si des problèmes de compatibilité peuvent affecter votre base de données ou vos applications.
2. Si votre instance de base de données utilise un groupe d'options personnalisé, créez un nouveau groupe d'options compatible avec la version vers laquelle vous procédez à la mise à niveau. Pour plus d'informations, consultez Considérations relatives au groupe d'options.
3. Si votre instance de base de données utilise un groupe de paramètres personnalisé, créez un nouveau groupe de paramètres compatible avec la version vers laquelle vous procédez à la mise à niveau. Pour plus d'informations, consultez Considérations relatives au groupe de paramètres.
4. Créez un instantané de base de données de l'instance de base de données à mettre à niveau. Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
5. Restaurez l'instantané de base de données pour créer une nouvelle instance de base de données de test. Pour plus d'informations, consultez Restauration vers une instance de base de données.
6. Modifiez cette nouvelle instance de base de données de test pour la mettre à niveau vers la nouvelle version, en utilisant l'une des méthodes suivantes :

   - console
   - AWS CLI

- API RDS

7. Évaluez le stockage utilisé par l'instance mise à niveau pour déterminer si la mise à niveau requiert un stockage supplémentaire.
8. Exécutez sur l'instance de base de données mise à niveau autant de tests d'assurance qualité que nécessaire pour garantir que votre base de données et votre application fonctionnent correctement avec la nouvelle version. Implémentez tous les nouveaux tests requis pour évaluer l'impact des éventuels problèmes de compatibilité que vous avez identifiés à l'étape 1. Testez toutes les fonctions et procédures stockées. Dirigez les versions de test de vos applications vers l'instance de base de données mise à niveau.
9. En cas de succès de tous les tests, effectuez la mise à niveau sur votre instance de base de données de production. Nous vous recommandons de ne pas autoriser les opérations d'écriture sur l'instance de base de données tant que vous n'avez pas confirmé que tout fonctionne correctement.

Mise à niveau d'une instance de base de données SQL Server

Pour plus d'informations sur la mise à niveau manuelle ou automatique d'une instance de base de données SQL Server, veuillez consulter les ressources suivantes :

- Mise à niveau de la version du moteur d'une instance de base de données
- Bonnes pratiques de mise à niveau de SQL Server 2008 R2 vers SQL Server 2016 sur Amazon RDS for SQL Server

> **Important**
>
> Si vous avez des instantanés chiffrés à l'aide de ce logiciel AWS KMS, nous vous recommandons de lancer une mise à niveau avant la fin du support.

Mise à niveau des instances de base de données obsolètes avant la fin de la prise en charge

Lorsqu'une version majeure est obsolète, vous ne pouvez pas l'installer sur de nouvelles instances de base de données. RDS va essayer de mettre automatiquement à niveau toutes les instances de base de données existantes.

Si vous devez restaurer une instance de base de données obsolète, vous pouvez effectuer une point-in-time restauration (PITR) ou restaurer un instantané. Cela vous donne un accès temporaire à une instance de base de données qui utilise la version considérée comme obsolète. Cependant, une fois qu'une version majeure est totalement obsolète, ces instances de base de données sont également automatiquement mises à niveau vers une version prise en charge.

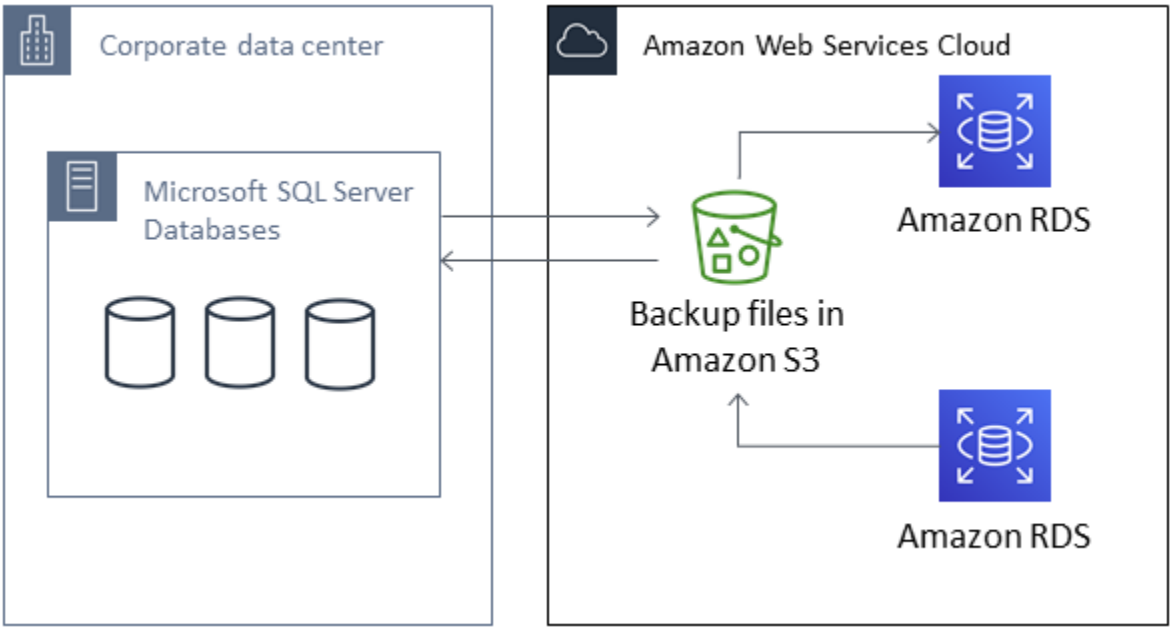
Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives

Amazon RDS prend en charge les sauvegarde et restauration natives pour les bases de données Microsoft SQL Server à l'aide de fichiers de sauvegarde complète (fichiers .bak). Lorsque vous utilisez RDS, vous accédez aux fichiers stockés dans Amazon S3 au lieu d'utiliser le système de fichiers local sur le serveur de base de données.

Par exemple, vous pouvez créer une sauvegarde complète depuis votre serveur local, la stocker sur S3, puis la restaurer sur une instance de base de données Amazon RDS existante. Vous pouvez également créer des sauvegardes à partir de RDS, les stocker sur S3, puis les restaurer chaque fois que vous le souhaitez.

La sauvegarde et la restauration natives sont disponibles dans toutes les AWS régions pour les instances de base de données mono-AZ et multi-AZ, y compris les instances de base de données multi-AZ avec des répliques de lecture. Les sauvegarde et restauration natives sont disponibles pour toutes les éditions de Microsoft SQL Server prises en charge sur Amazon RDS.

Le schéma suivant illustre les scénarios pris en charge.



L'utilisation de fichiers .bak natifs pour sauvegarder et restaurer des bases de données est généralement le moyen le plus rapide de sauvegarder et de restaurer des bases de données. Il existe de nombreux avantages supplémentaires à l'utilisation des sauvegarde et restauration natives. Par exemple, vous pouvez effectuer les opérations suivantes :

- Migrer des bases de données vers ou depuis Amazon RDS.
- Déplacer des bases de données entre des instances de base de données RDS for SQL Server.
- Migrer des données, des schémas, des procédures stockées, des déclencheurs et tout autre code de base de données dans des fichiers .bak.
- Sauvegarder et restaurer des bases de données uniques, au lieu de la totalité d'instances de base de données.
- Créer des copies de bases de données à des fins de développement, de test, de formation et de démonstration.
- Stocker et transférer des fichiers de sauvegarde avec Amazon S3, pour offrir une couche de protection supplémentaire pour la reprise après sinistre.
- Créez des sauvegardes natives de bases de données sur lesquelles Transparent Data Encryption (TDE) est activé, puis restaurez ces sauvegardes sur des bases de données sur site. Pour de plus amples informations, veuillez consulter Support pour le chiffrement transparent des données sur le SQL serveur.
- Restaurez les sauvegardes natives des bases de données sur site sur lesquelles TDE est activé sur des instances de base de données RDS for SQL Server. Pour de plus amples informations, veuillez consulter Support pour le chiffrement transparent des données sur le SQL serveur.

Table des matières

- Limitations et recommandations
- Configuration pour les sauvegarde et restauration natives
  - Création manuelle d'un rôle IAM pour les sauvegarde et restauration natives
- Utilisation des sauvegarde et restauration natives
  - Sauvegarde d'une base de données
    - Utilisation
    - Exemples
  - Restauration d'une base de données
    - Utilisation
    - Exemples
  - Restauration d'un journal
    - Utilisation
    - Exemples

- Finalisation d'une restauration de base de données
  - Utilisation
- Utilisation de bases de données partiellement restaurées
  - Suppression d'une base de données partiellement restaurée
  - Comportement de restauration et point-in-time de restauration des instantanés pour les bases de données partiellement restaurées
- Annulation d'une tâche
  - Utilisation
- Suivi de l'état des tâches
  - Utilisation
  - Exemples
  - Réponse
- Compression des fichiers de sauvegarde
- Résolution des problèmes
- Importation et exportation de données SQL Server à l'aide d'autres méthodes
  - Importation de données dans RDS for SQL Server à l'aide d'un instantané
    - Importer les données
      - Assistant de génération et de publication de scripts
      - Assistant d'importation et d'exportation
      - Copie en bloc
  - Exportation de données depuis RDS for SQL Server
    - Assistant d'importation et d'exportation SQL Server
    - Assistant Générer et publier des scripts et utilitaire bcp
- Utilisation de l'utilitaire BCP de Linux pour importer et exporter des données
  - Prérequis
  - Installation des outils de ligne de commande SQL Server sous Linux
  - Exportation de données depuis RDS for SQL Server
    - Syntaxe d'exportation de base
    - Exemple d'exportation
  - Importation de données vers RDS pour SQL Server

    - Syntaxe d'importation de base
    - Exemple d'importation
- Options BCP courantes
- Bonnes pratiques et considérations
- Résolution des problèmes courants

Limitations et recommandations

Voici quelques limitations quant à l'utilisation des sauvegarde et restauration natives :

- Vous ne pouvez pas effectuer de sauvegarde ou de restauration depuis un compartiment Amazon S3 situé dans une AWS région différente de celle de votre instance de base de données Amazon RDS.
- Vous ne pouvez pas restaurer une base de données qui porte le même nom qu'une base de données existante. Les noms de base de données sont uniques.
- Nous vous recommandons vivement de ne pas restaurer de fichiers de sauvegarde d'un fuseau horaire dans un autre fuseau horaire. Si vous restaurez des sauvegardes d'un fuseau horaire dans un autre, vous devez auditer vos requêtes et vos applications afin de déterminer les effets du changement de fuseau horaire.
- Amazon S3 a une limite de taille de 5 To par fichier. Pour les sauvegardes natives de bases de données plus volumineuses, vous pouvez utiliser la sauvegarde multifichier.
- La taille maximale de la base de données pouvant être sauvegardée sur S3 dépend de la mémoire disponible, du processeur, des I/O et des ressources réseau sur l'instance de base de données. Plus la base de données est grande, plus l'agent de sauvegarde consomme de la mémoire.
- Vous ne pouvez pas effectuer une sauvegarde ou une restauration à partir de plus de 10 fichiers de sauvegarde simultanément.
- Une sauvegarde différentielle est basée sur la dernière sauvegarde complète. Pour que les sauvegardes différentielles fonctionnent, vous ne pouvez prendre un instantané entre la dernière sauvegarde complète et la sauvegarde différentielle. Si vous souhaitez faire une sauvegarde différentielle, mais qu'il existe un instantané manuel ou automatique, créez une autre sauvegarde complète avant de créer la sauvegarde différentielle.
- Les restaurations différentielles et de journaux ne sont pas prises en charge pour les bases de données possédant des fichiers dont l'identifiant unique file_guid est défini sur `NULL`.
- Vous pouvez exécuter jusqu'à deux tâches de sauvegarde ou restauration simultanément.

- Vous ne pouvez pas effectuer de sauvegardes natives de journaux à partir de SQL Server sur Amazon RDS.
- RDS prend en charge les restaurations natives de bases de données jusqu'à 64 TiB. Les restaurations natives des bases de données sur SQL Server Express Edition sont limitées à 10 Go.
- Vous ne pouvez pas sauvegarder une base de données pendant la fenêtre de maintenance, ou à tout moment où Amazon RDS prend un instantané de la base de données. Si une tâche de sauvegarde native se chevauche avec la fenêtre de sauvegarde quotidienne RDS, la tâche de sauvegarde native est annulée.
- Sur des instances de base de données multi-AZ, vous pouvez uniquement restaurer nativement des bases de données sauvegardées en utilisant le modèle de restauration « Full ».
- La restauration à partir de sauvegardes différentielles sur instances multi-AZ n'est pas prise en charge.
- L'appel des procédures RDS pour la sauvegarde/restauration au sein d'une transaction n'est pas pris en charge.
- Utilisez un chiffrement symétrique AWS KMS key pour chiffrer vos sauvegardes. Amazon RDS ne prend pas en charge les clés KMS asymétriques. Pour plus d'informations, consultez Création de clés KMS de chiffrement symétriques dans le Guide du développeur AWS Key Management Service .
- Les fichiers de sauvegarde natifs sont chiffrés avec la clé KMS spécifiée à l'aide du mode cryptographique « Chiffrement seul ». Lorsque vous restaurez des fichiers de sauvegarde chiffrés, gardez à l'esprit qu'ils ont été chiffrés à l'aide du mode cryptographique « Chiffrement seul ».
- Vous ne pouvez pas restaurer une base de données contenant un groupe de fichiers FILESTREAM.
- Le chiffrement côté serveur Amazon S3 avec AWS KMS (SSE-KMS) n'est actuellement pas pris en charge. Lorsque vous fournissez une clé KMS à une procédure stockée, toutes les sauvegardes et restaurations natives sont chiffrées et déchiffrées côté client à l'aide de la clé KMS. AWS stocke les sauvegardes dans le compartiment S3 avec SSE-S3.

Nous vous recommandons d'utiliser les sauvegarde et restauration natives pour migrer votre base de données vers RDS si votre base de données peut être hors connexion pendant que le fichier de sauvegarde est créé, copié et restauré. Si votre base de données locale ne peut pas être hors ligne, nous vous recommandons d'utiliser le AWS Database Migration Service pour migrer votre base de données vers Amazon RDS. Pour plus d'informations, voir Qu'est-ce que c'est AWS Database Migration Service ?

Les sauvegarde et restauration natives ne visent pas à remplacer les fonctionnalités de récupération des données de la fonction de copie d'instantané entre régions. Nous vous recommandons d'utiliser la copie instantanée pour copier l'instantané de votre base de données dans une autre AWS région afin de permettre une reprise après sinistre entre régions sur Amazon RDS. Pour de plus amples informations, veuillez consulter Copier un instantané de base de données pour Amazon RDS.

Configuration pour les sauvegarde et restauration natives

Pour configurer la sauvegarde et la restauration natives, vous avez besoin de trois composants :

1. Un compartiment Amazon S3 pour stocker vos fichiers de sauvegarde.

   Vous devez disposer d'un compartiment S3 pour vos fichiers de sauvegarde et charger les sauvegardes que vous souhaitez migrer vers RDS. Si vous avez déjà un compartiment Amazon S3, vous pouvez l'utiliser. Si vous n'en avez pas, vous pouvez en créer un. Sinon, vous pouvez choisir d'avoir un compartiment créé pour vous quand vous ajoutez l'option `SQLSERVER_BACKUP_RESTORE` à l'aide de AWS Management Console.

   Pour obtenir des informations sur l'utilisation de S3, consultez le Guide de l'utilisateur Amazon Simple Storage Service
2. Rôle AWS Identity and Access Management (IAM) permettant d'accéder au compartiment.

   Si vous avez déjà un rôle IAM, vous pouvez l'utiliser. Vous pouvez choisir d'avoir un nouveau rôle IAM créé pour vous quand vous ajoutez l'option `SQLSERVER_BACKUP_RESTORE` à l'aide de la AWS Management Console. Vous pouvez également en créer un nouveau manuellement.

   Si vous souhaitez créer un nouveau rôle IAM manuellement, optez pour l'approche abordée à la section suivante. Faites de même si vous souhaitez associer des relations d'approbation et des stratégies d'autorisations à un rôle IAM existant.
3. L'option `SQLSERVER_BACKUP_RESTORE` ajoutée à un groupe d'options sur votre instance de base de données.

   Pour activer les sauvegarde et restauration natives sur votre instance de base de données, vous ajoutez l'option `SQLSERVER_BACKUP_RESTORE` à un groupe d'options sur votre instance de base de données. Pour plus d'informations et des instructions, consultez Prise en charge des sauvegarde et restauration natives dans SQL Server.

Création manuelle d'un rôle IAM pour les sauvegarde et restauration natives

Si vous souhaitez créer manuellement un rôle IAM à utiliser avec une sauvegarde native et à restaurer, vous pouvez le faire. Dans ce cas, vous créez un rôle pour déléguer des autorisations depuis le service Amazon RDS vers votre compartiment Amazon S3. Lorsque vous créez un rôle IAM, vous attachez une relation d'approbation et une stratégie d'autorisation. La relation d'approbation permet à RDS d'assumer ce rôle. La politique d'autorisation définit les actions que ce rôle peut exécuter. Pour plus d'informations sur la création d'un rôle, consultez Création d'un rôle pour déléguer des autorisations à un service AWS.

Pour la fonction de sauvegarde et restauration native, utilisez des relations d'approbation et des stratégies d'autorisation similaires aux exemples de cette section. Dans l'exemple suivant, nous utilisons le nom principal de service `rds.amazonaws.com` comme alias de tous les comptes de service. Dans les autres exemples, nous spécifions un ARN (Amazon Resource Name) pour identifier un autre compte, utilisateur ou rôle auquel nous accordons l'accès dans la stratégie d'approbation.

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des relations d'approbation basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la relation d'approbation, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet des ressources qui accèdent au rôle. Pour la sauvegarde et la restauration natives, veillez à inclure à la fois le groupe d'options de base de données et les instances de base de données, comme indiqué dans l'exemple suivant.

Example de relation de confiance avec la clé de contexte de condition globale pour la sauvegarde et la restauration natives

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": [

 "arn:aws:rds:Region:0123456789:db:db_instance_identifier",
                        "arn:aws:rds:Region:0123456789:og:option_group_name",
                  "arn:aws:s3:::amzn-s3-demo-bucket"
                    ],
                    "aws:SourceAccount": "0123456789"
                }
            }
        }
    ]
}
```

L'exemple suivant utilise un ARN pour spécifier une ressource. Pour plus d'informations sur l'utilisationARNs, consultez Amazon resource names (ARNs).

Example de la politique d'autorisation pour la sauvegarde et la restauration natives sans support de chiffrement

JSON

```
{
    "Version": "2012-10-17",
    "Statement":
```

```
    [
        {
        "Effect": "Allow",
        "Action":
            [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
        "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
        },
        {
        "Effect": "Allow",
        "Action":
            [
                "s3:GetObjectAttributes",
                "s3:GetObject",
                "s3:PutObject",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload"
            ],
        "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
        }
    ]
}
```

Example politique d'autorisation pour la sauvegarde et la restauration natives avec prise en charge du chiffrement

Si vous souhaitez chiffrer vos fichiers de sauvegarde, incluez une clé de chiffrement dans votre stratégie d'autorisation. Pour en savoir plus sur les clés de chiffrement, consultez Mise en route dans le Manuel du développeur AWS Key Management Service .

> **Note**
>
> Vous devez utiliser une clé KMS de chiffrement symétrique pour chiffrer vos sauvegardes. Amazon RDS ne prend pas en charge les clés KMS asymétriques. Pour plus d'informations, consultez Création de clés KMS de chiffrement symétriques dans le Guide du développeur AWS Key Management Service .
> Le rôle IAM doit également être un utilisateur de clé et un administrateur de clé pour la clé KMS, c'est-à-dire qu'il doit être spécifié dans la stratégie de clé. Pour plus d'informations,

consultez Création de clés KMS de chiffrement symétriques dans le Guide du développeur AWS Key Management Service .

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowAccessToKey",
      "Effect": "Allow",
      "Action": [
        "kms:DescribeKey",
        "kms:GenerateDataKey",
        "kms:Encrypt",
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:us-east-1:123456789012:key/key-id"
    },
    {
      "Sid": "AllowAccessToS3",
      "Effect": "Allow",
      "Action": [
        "s3:ListBucket",
        "s3:GetBucketLocation"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
    },
    {
      "Sid": "GetS3Info",
      "Effect": "Allow",
      "Action": [
        "s3:GetObjectAttributes",
        "s3:GetObject",
        "s3:PutObject",
        "s3:ListMultipartUploadParts",
        "s3:AbortMultipartUpload"
      ],
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*"
    }
```

```
  ]
}
```

Utilisation des sauvegarde et restauration natives

Une fois que vous avez activé et configuré les sauvegarde et restauration natives, vous pouvez commencer à les utiliser. Tout d'abord, vous vous connectez à votre base de données Microsoft SQL Server, puis vous appelez une procédure stockée Amazon RDS pour faire le travail. Pour plus d'informations sur la connexion à votre base de données, consultez Connexion à votre instance de base de données Microsoft SQL Server.

Certaines procédures stockées nécessitent que vous fournissiez un Amazon Resource Name (ARN) à votre compartiment et votre fichier Amazon S3. Le format pour votre ARN est `arn:aws:s3:::bucket_name/file_name.extension`. Amazon S3 n'a pas besoin de numéro de compte ni de AWS région dans ARNs.

Si vous fournissez également une clé KMS facultative, le format de l'ARN de la clé est `arn:aws:kms:region:account-id:key/key-id`. Pour plus d'informations, consultez les noms des ressources Amazon (ARNs) et les espaces de noms des AWS services. Vous devez utiliser une clé KMS de chiffrement symétrique pour chiffrer vos sauvegardes. Amazon RDS ne prend pas en charge les clés KMS asymétriques. Pour plus d'informations, consultez Création de clés KMS de chiffrement symétriques dans le Guide du développeur AWS Key Management Service .

> **Note**
>
> Que vous utilisiez ou non une clé KMS, les tâches de sauvegarde et de restauration natives activent le chiffrement AES (Advanced Encryption Standard) 256 bits par défaut pour les fichiers téléchargés sur S3.

Pour plus d'informations sur la façon d'appeler chaque procédure stockée, veuillez consulter les rubriques suivantes :

- Sauvegarde d'une base de données
- Restauration d'une base de données
- Restauration d'un journal
- Finalisation d'une restauration de base de données
- Utilisation de bases de données partiellement restaurées
- Annulation d'une tâche
- Suivi de l'état des tâches

Sauvegarde d'une base de données

Pour sauvegarder votre base de données, utilisez la procédure stockée `rds_backup_database`.

> **Note**
>
> Vous ne pouvez pas sauvegarder une base de données pendant la fenêtre de maintenance ou lorsqu'Amazon RDS prend un instantané.

Utilisation

```
exec msdb.dbo.rds_backup_database
 @source_db_name='database_name',
 @s3_arn_to_backup_to='arn:aws:s3:::bucket_name/file_name.extension',
 [@kms_master_key_arn='arn:aws:kms:region:account-id:key/key-id'],
 [@overwrite_s3_backup_file=0|1],
 [@block_size=512|1024|2048|4096|8192|16384|32768|65536],
        [@max_transfer_size=n],
        [@buffer_count=n],
 [@type='DIFFERENTIAL|FULL'],
 [@number_of_files=n];
```

Les paramètres suivants sont obligatoires :

- `@source_db_name` – Nom de la base de données à sauvegarder.
- `@s3_arn_to_backup_to` – L'ARN indique le compartiment Amazon S3 à utiliser pour la sauvegarde, ainsi que le nom du fichier de sauvegarde.

  Le fichier peut avoir n'importe quelle extension mais `.bak` est généralement utilisée.

Les paramètres suivants sont facultatifs :

- `@kms_master_key_arn` – ARN de la clé KMS de chiffrement symétrique à utiliser pour chiffrer l'élément.
  - Vous ne pouvez pas utiliser la clé de chiffrement par défaut. Si vous utilisez la clé par défaut, la base de données n'est pas sauvegardée.
  - Si vous ne spécifiez pas d'identificateur de clé KMS, le fichier de sauvegarde n'est pas chiffré. Pour plus d'informations, veuillez consulter Chiffrer des ressources Amazon RDS.

  - Lorsque vous spécifiez une clé KMS, le chiffrement côté client est utilisé.
  - Amazon RDS ne prend pas en charge les clés KMS asymétriques. Pour plus d'informations, consultez Création de clés KMS de chiffrement symétriques dans le Guide du développeur AWS Key Management Service .
- `@overwrite_s3_backup_file` – Une valeur qui indique si un fichier de sauvegarde existant doit être écrasé.
  - `0` – N'écrase pas un fichier existant. Cette valeur est celle par défaut.

    Si `@overwrite_s3_backup_file` est défini sur 0, une erreur est retournée si le fichier existe déjà.
  - `1` – Écrase le fichier existant qui possède déjà le nom spécifié, même s'il ne s'agit pas d'un fichier de sauvegarde.
- `@type` – Le type de sauvegarde.
  - `DIFFERENTIAL` – Crée une sauvegarde différentielle.
  - `FULL` – Crée une sauvegarde complète. Cette valeur est celle par défaut.

  Une sauvegarde différentielle est basée sur la dernière sauvegarde complète. Pour que les sauvegardes différentielles fonctionnent, vous ne pouvez prendre un instantané entre la dernière sauvegarde complète et la sauvegarde différentielle. Si vous souhaitez faire une sauvegarde différentielle, mais qu'il existe un instantané, alors créez une autre sauvegarde complète avant de créer la sauvegarde différentielle.

  Vous pouvez rechercher la dernière sauvegarde complète ou le dernier instantané à l'aide de l'exemple de requête SQL suivant :

```
select top 1
database_name
,  backup_start_date
,  backup_finish_date
from    msdb.dbo.backupset
where   database_name='mydatabase'
and     type = 'D'
order by backup_start_date desc;
```

- `@number_of_files` – Nombre de fichiers dans lesquels la sauvegarde sera divisée (en morceaux). Le nombre maximum est de 10.

- La sauvegarde en plusieurs fichiers est prise en charge pour les sauvegardes complètes et différentielles.
- Si vous entrez la valeur 1 ou omettez le paramètre, un seul fichier de sauvegarde est créé.

Fournissez le préfixe que les fichiers ont en commun, puis ajoutez un astérisque comme suffixe (*). L'astérisque peut se trouver n'importe où dans la *file_name* partie de l'ARN S3. L'astérisque est remplacé par une série de chaînes alphanumériques dans les fichiers générés, en commençant par `1-of-`*number_of_files*.

Par exemple, si les noms de fichiers dans l'ARN S3 ont pour modèle `backup*.bak` et que vous définissez `@number_of_files=4`, les fichiers de sauvegarde générés auront pour noms `backup1-of-4.bak`, `backup2-of-4.bak`, `backup3-of-4.bak` et `backup4-of-4.bak`.

- Si l'un des noms de fichier existe déjà et que `@overwrite_s3_backup_file` a pour valeur 0, une erreur est renvoyée.
- Les sauvegardes multifichiers ne peuvent comporter qu'un seul astérisque dans la *file_name* partie de l'ARN S3.
- Les sauvegardes à fichier unique peuvent comporter un certain nombre d'astérisques dans la *file_name* partie de l'ARN S3. Les astérisques ne sont pas supprimés du nom de fichier généré.

- `@block_size`— Taille de bloc (en octets) spécifiant la taille de bloc physique pour les opérations de sauvegarde. Les valeurs valides sont 512, 1024, 2048, 4096, 8192, 16384, 32768 et 65536
- `@max_transfer_size`— La taille de transfert maximale indique la limite supérieure du volume de données (en octets) transmis par I/O opération pendant le processus de sauvegarde. Les valeurs valides sont des multiples de 65536 octets (64 Ko) jusqu'à 4194304 octets (4 Mo).
- `@buffer_count`— Nombre total de I/O tampons à utiliser pour le processus de sauvegarde.

Exemples

Example de sauvegarde différentielle

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup1.bak',
@overwrite_s3_backup_file=1,
@type='DIFFERENTIAL';
```

Example de sauvegarde complète avec chiffrement

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup1.bak',
@kms_master_key_arn='arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE',
@overwrite_s3_backup_file=1,
@type='FULL';
```

Example de sauvegarde sur plusieurs fichiers

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@number_of_files=4;
```

Example de sauvegarde différentielle sur plusieurs fichiers

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@type='DIFFERENTIAL',
@number_of_files=4;
```

Example de sauvegarde sur plusieurs fichiers avec chiffrement

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@kms_master_key_arn='arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE',
@number_of_files=4;
```

Example de sauvegarde sur plusieurs fichiers avec écrasement S3

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@overwrite_s3_backup_file=1,
@number_of_files=4;
```

Example de sauvegarde avec taille de bloc

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@block_size=512;
```

Example de sauvegarde multifichiers avec et **@max_transfer_size@buffer_count**

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@number_of_files=4,
@max_transfer_size=4194304,
@buffer_count=10;
```

Example de sauvegarde en un seul fichier avec le paramètre @number_of_files

Cet exemple génère un fichier de sauvegarde nommé `backup*.bak`.

```
exec msdb.dbo.rds_backup_database
@source_db_name='mydatabase',
@s3_arn_to_backup_to='arn:aws:s3:::mybucket/backup*.bak',
@number_of_files=1;
```

Restauration d'une base de données

Pour restaurer votre base de données, appelez la procédure stockée `rds_restore_database`. Amazon RDS crée un instantané initial de la base de données après la fin de la tâche de restauration et l'ouverture de la base de données.

Utilisation

```
exec msdb.dbo.rds_restore_database
 @restore_db_name='database_name',
 @s3_arn_to_restore_from='arn:aws:s3:::bucket_name/file_name.extension',
 @with_norecovery=0|1,
 [@kms_master_key_arn='arn:aws:kms:region:account-id:key/key-id'],
        [@block_size=512|1024|2048|4096|8192|16384|32768|65536],
        [@max_transfer_size=n],
        [@buffer_count=n],
```

```
[@type='DIFFERENTIAL|FULL'];
```

Les paramètres suivants sont obligatoires :

- `@restore_db_name` – Nom de la base de données à restaurer. Les noms de base de données sont uniques. Vous ne pouvez pas restaurer une base de données qui porte le même nom qu'une base de données existante.
- `@s3_arn_to_restore_from` – L'ARN indique le préfixe Amazon S3 et les noms des fichiers de sauvegarde utilisés pour restaurer la base de données.
  - Pour une seule sauvegarde de fichier, fournissez la totalité du nom du fichier.
  - Pour une sauvegarde comportant plusieurs fichiers, fournissez le préfixe que les fichiers ont en commun, puis ajoutez un astérisque comme suffixe (`*`).
  - Si `@s3_arn_to_restore_from` est vide, le message d'erreur suivant est renvoyé : Le préfixe de l'ARN S3 ne peut pas être vide.

Le paramètre suivant est obligatoire pour les différentes restaurations, mais facultatifs pour les restaurations complètes :

- `@with_norecovery` – La clause de restauration à utiliser pour l'opération de restauration.
  - Définissez le paramètre sur `0` pour restaurer avec RECOVERY (RESTAURATION). Dans ce cas, la base de données est en ligne après la restauration.
  - Définissez le paramètre sur 1 pour restaurer avec NORECOVERY (SANS RESTAURATION). Dans ce cas, la base de données reste à l'état RESTORING (EN COURS DE RESTAURATION) après la fin de la tâche de restauration. Grâce à cette approche, vous pouvez procéder à des restaurations différentielles ultérieurement.
  - Pour les restaurations DIFFÉRENTIELLES, spécifiez `0` ou 1.
  - Pour les restaurations `FULL`, cette valeur par défaut est `0`.

Les paramètres suivants sont facultatifs :

- `@kms_master_key_arn` – Clé KMS à utiliser pour déchiffrer le fichier si vous avez chiffré le fichier de sauvegarde.

  Lorsque vous spécifiez une clé KMS, le chiffrement côté client est utilisé.

- `@type` – Le type de restauration. Les types valides sont `DIFFERENTIAL` et `FULL`. La valeur par défaut est `FULL`.
- `@block_size`— Taille de bloc (en octets) spécifiant la taille de bloc physique pour les opérations de sauvegarde. Les valeurs valides sont 512, 1024, 2048, 4096, 8192, 16384, 32768 et 65536
- `@max_transfer_size`— La taille de transfert maximale indique la limite supérieure du volume de données (en octets) transmis par I/O opération pendant le processus de sauvegarde. Les valeurs valides sont des multiples de 65536 octets (64 Ko) jusqu'à 4194304 octets (4 Mo).
- `@buffer_count`— Nombre total de I/O tampons à utiliser pour le processus de sauvegarde.

> **Note**
>
> Pour les restaurations différentielles, la base de données doit se trouver en état RESTORING (EN COURS DE RESTAURATION) ou une tâche de restauration doit déjà exister avec NORECOVERY (SANS RESTAURATION).
> Vous ne pouvez pas restaurer ultérieurement des sauvegardes différentielles tant que la base de données est en ligne.
> Vous ne pouvez pas envoyer de tâche de restauration pour une base de données qui possède déjà une tâche de restauration en attente avec RECOVERY (RESTAURATION).
> Les restaurations complètes avec NORECOVERY ne sont pas prises en charge sur les instances multi-AZ ou les instances avec des répliques en lecture.
> Les restaurations différentielles ne sont prises en charge dans aucune configuration de déploiement multi-AZ.
> Toutes les restaurations natives ne sont pas prises en charge sur les instances dotées de répliques de lecture, y compris les répliques de lecture entre régions.
> Pour les configurations prises en charge, la restauration d'une base de données sur une instance multi-AZ avec des répliques de lecture est similaire à la restauration d'une base de données sur une instance multi-AZ. Vous n'avez pas besoin d'effectuer d'actions supplémentaires pour restaurer une base de données sur un réplica.

Exemples

Example de restauration d'un seul fichier

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
```

```
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak';
```

Example de restauration de plusieurs fichiers

Pour éviter les erreurs lors de la restauration de plusieurs fichiers, assurez-vous que tous les fichiers de sauvegarde ont le même préfixe et qu'aucun autre fichier n'utilise ce préfixe.

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup*';
```

Example de restauration de base de données complète avec RECOVERY (RESTAURATION)

Les trois exemples suivants exécutent la même tâche de restauration complète avec RECOVERY (RESTAURATION).

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak';
```

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
[@type='DIFFERENTIAL|FULL'];
```

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@type='FULL',
@with_norecovery=0;
```

Example de restauration de base de données complète avec chiffrement

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@kms_master_key_arn='arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE';
```

Example de restauration avec taille de bloc

```
exec msdb.dbo.rds_restore_database
```

```
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@block_size=512;
```

Example de restauration multifichiers avec @max_transfer_size et @buffer_count

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup*',
@max_transfer_size=4194304,
@buffer_count=10;
```

Example de restauration de base de données complète avec NORECOVERY (SANS RESTAURATION)

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@type='FULL',
@with_norecovery=1;
```

Example de restauration différentielle avec NORECOVERY (SANS RESTAURATION)

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@type='DIFFERENTIAL',
@with_norecovery=1;
```

Example de restauration différentielle avec RECOVERY (RESTAURATION).

```
exec msdb.dbo.rds_restore_database
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/backup1.bak',
@type='DIFFERENTIAL',
@with_norecovery=0;
```

Restauration d'un journal

Pour restaurer votre journal, appelez la procédure stockée `rds_restore_log`.

Utilisation

```
exec msdb.dbo.rds_restore_log
 @restore_db_name='database_name',
 @s3_arn_to_restore_from='arn:aws:s3:::bucket_name/log_file_name.extension',
 [@kms_master_key_arn='arn:aws:kms:region:account-id:key/key-id'],
 [@with_norecovery=0|1],
 [@stopat='datetime'],
 [@block_size=512|1024|2048|4096|8192|16384|32768|65536],
        [@max_transfer_size=n],
        [@buffer_count=n];
```

Les paramètres suivants sont obligatoires :

- `@restore_db_name` – Le nom de la base de données dont vous souhaitez restaurer le journal.
- `@s3_arn_to_restore_from` – L'ARN indique le préfixe Amazon S3 et le nom du fichier journal utilisé pour restaurer le journal. Le fichier peut avoir n'importe quelle extension mais `.trn` est généralement utilisée.

  Si `@s3_arn_to_restore_from` est vide, le message d'erreur suivant est renvoyé : Le préfixe de l'ARN S3 ne peut pas être vide.

Les paramètres suivants sont facultatifs :

- `@kms_master_key_arn` – Clé KMS à utiliser pour déchiffrer le journal si vous avez chiffré le journal.
- `@with_norecovery` – La clause de restauration à utiliser pour l'opération de restauration. La valeur par défaut est 1.
  - Définissez le paramètre sur `0` pour restaurer avec RECOVERY (RESTAURATION). Dans ce cas, la base de données est en ligne après la restauration. Vous ne pouvez pas restaurer ultérieurement des sauvegardes de journaux tant que la base de données est en ligne.
  - Définissez le paramètre sur 1 pour restaurer avec NORECOVERY (SANS RESTAURATION). Dans ce cas, la base de données reste à l'état RESTORING (EN COURS DE RESTAURATION) après la fin de la tâche de restauration. Grâce à cette approche, vous pouvez procéder à des restaurations de journaux ultérieurement.
- `@stopat` – Une valeur qui spécifie que la base de données est restaurée dans son état à la date et l'heure spécifiées (au format datetime). Seul les enregistrements de journaux de transaction écrits avant la date et l'heure spécifiées sont appliqués à la base de données.

Si ce paramètre n'est pas spécifié (il est NULL), le journal complet est restauré.

- `@block_size`— Taille de bloc (en octets) spécifiant la taille de bloc physique pour les opérations de sauvegarde. Les valeurs valides sont 512, 1024, 2048, 4096, 8192, 16384, 32768 et 65536
- `@max_transfer_size`— La taille de transfert maximale indique la limite supérieure du volume de données (en octets) transmis par I/O opération pendant le processus de sauvegarde. Les valeurs valides sont des multiples de 65536 octets (64 Ko) jusqu'à 4194304 octets (4 Mo).
- `@buffer_count`— Nombre total de I/O tampons à utiliser pour le processus de sauvegarde.

> **Note**
>
> Pour les restaurations de journaux, la base de données doit se trouver en état de restauration ou une tâche de restauration doit déjà exister avec NORECOVERY (SANS RESTAURATION).
> Vous ne pouvez pas restaurer de sauvegardes de journaux tant que la base de données est en ligne.
> Vous ne pouvez pas envoyer de tâche de restauration de journaux sur une base de données qui possède déjà une tâche de restauration en attente avec RECOVERY (RESTAURATION).
> Les restaurations de journaux ne sont pas prises en charge sur les instances multi-AZ.

Exemples

Example de restaurations de journaux

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn';
```

Example de restaurations de journaux avec chiffrement

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn',
@kms_master_key_arn='arn:aws:kms:us-east-1:123456789012:key/AKIAIOSFODNN7EXAMPLE';
```

Example de restaurations de journaux avec NORECOVERY (SANS RESTAURATION)

Les trois exemples suivants exécutent la même tâche de restauration de journaux avec NORECOVERY (SANS RESTAURATION).

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn',
@with_norecovery=1;
```

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn';
```

Example de restauration avec taille de bloc

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn',
@block_size=512;
```

Example de restaurations de journaux avec RECOVERY (RESTAURATION)

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn',
@with_norecovery=0;
```

Example de restaurations de journaux avec clause STOPAT

```
exec msdb.dbo.rds_restore_log
@restore_db_name='mydatabase',
@s3_arn_to_restore_from='arn:aws:s3:::mybucket/mylog.trn',
@with_norecovery=0,
@stopat='2019-12-01 03:57:09';
```

Finalisation d'une restauration de base de données

Si la dernière tâche de restauration sur la base de données a été exécutée à l'aide de `@with_norecovery=1`, la base de données est désormais en état RECOVERY (RESTAURATION).

Ouvrez cette base de données pour exécuter des opérations normales à l'aide de la procédure stockée `rds_finish_restore`.

Utilisation

```
exec msdb.dbo.rds_finish_restore @db_name='database_name';
```

> **Note**
>
> Pour utiliser cette approche, la base de données doit être en état RECOVERY (RESTAURATION) sans aucune tâche de restauration en attente.
> La procédure `rds_finish_restore` n'est pas prise en charge sur les instances multi-AZ.
> Pour finaliser la restauration de la base de données, utilisez l'identifiant principal. Ou utilisez l'identifiant utilisateur qui a permis de restaurer récemment la base de données ou de se connecter avec NORECOVERY (SANS RESTAURATION).

Utilisation de bases de données partiellement restaurées

Suppression d'une base de données partiellement restaurée

Pour supprimer une base de données partiellement restaurée (laissée en état RECOVERY (RESTAURATION), utilisez la procédure stockée `rds_drop_database`.

```
exec msdb.dbo.rds_drop_database @db_name='database_name';
```

> **Note**
>
> Vous ne pouvez pas envoyer de demande de base de données DROP (SUPPRIMER) pour une base de données qui possède déjà une tâche de restauration en attente ou de finalisation de restauration.
> Pour supprimer la base de données, utilisez l'identifiant principal. Ou utilisez l'identifiant utilisateur qui a permis de restaurer récemment la base de données ou de se connecter avec NORECOVERY (SANS RESTAURATION).

Comportement de restauration et point-in-time de restauration des instantanés pour les bases de données partiellement restaurées

Les bases de données partiellement restaurées dans l'instance source (laissées dans l'état RESTORING) sont supprimées de l'instance cible lors de la restauration et de la point-in-time restauration des instantanés.

Annulation d'une tâche

Pour annuler une tâche de sauvegarde ou de restauration, appelez la procédure stockée `rds_cancel_task`.

> **Note**
>
> Vous ne pouvez pas annuler une tâche FINISH_RESTORE.

Utilisation

```
exec msdb.dbo.rds_cancel_task @task_id=ID_number;
```

Les paramètres suivants sont obligatoires :

- `@task_id` – ID de la tâche à annuler. Vous pouvez obtenir l'ID de la tâche en appelant `rds_task_status`.

Suivi de l'état des tâches

Pour suivre l'état de vos tâches de sauvegarde et restauration, appelez la procédure stockée `rds_task_status`. Si vous ne fournissez pas de paramètre, la procédure stockée retourne l'état de toutes les tâches. Le statut des tâches est mis à jour environ toutes les deux minutes. L'historique des tâches est conservé pendant 36 jours.

Utilisation

```
exec msdb.dbo.rds_task_status
 [@db_name='database_name'],
 [@task_id=ID_number];
```

Les paramètres suivants sont facultatifs :

- `@db_name` – Nom de la base de données pour laquelle afficher l'état de la tâche.
- `@task_id` – ID de la tâche pour laquelle afficher l'état de tâche.

Exemples

Example de liste des statuts d'une tâche spécifique

```
exec msdb.dbo.rds_task_status @task_id=5;
```

Example de liste des statuts d'une base de données et d'une tâche spécifiques

```
exec msdb.dbo.rds_task_status
@db_name='my_database',
@task_id=5;
```

Example de liste de toutes les tâches et de leurs statuts sur une base de données spécifique

```
exec msdb.dbo.rds_task_status @db_name='my_database';
```

Example de liste de toutes les tâches et de leurs statuts sur l'instance actuelle

```
exec msdb.dbo.rds_task_status;
```

Réponse

La procédure stockée `rds_task_status` renvoie les colonnes suivantes.

Colonne	Description
`task_id`	ID de la tâche

`task_type`	Le type de tâche dépend des paramètres d'entrée, comme suit :
	• Pour les tâches de sauvegarde :
	  • BACKUP_DB – Sauvegarde de base de données complète
	  •

Colonne	Description

	BACKUP_DB_DIFFERENTIAL – Sauvegarde de base de données différentielle
	• Pour les tâches de restauration :
	  • RESTORE_DB – Restauration de base de données complète avec RECOVERY (RESTAURATION)
	  • RESTORE_DB_NORECOVERY – Restauration de base de données complète avec NORECOVERY (SANS RESTAURATION)
	  • RESTORE_DB_DIFFERENTIAL – Restauration de base de données différentielle avec RECOVERY (RESTAURATION)
	  • RESTORE_DB_DIFFERENTIAL_NORECOVERY – Restauration de base de données différentielle avec NORECOVERY (SANS RESTAURATION)
	  • RESTORE_DB_LOG – Restauration de journaux avec RECOVERY (RESTAURATION)
	  • RESTORE_DB_LOG_NORECOVERY – Restauration de journaux avec NORECOVERY (SANS RESTAURATION)
	• Pour les tâches qui finalisent une restauration :
	  • FINISH_RESTORE – Finalisation de restauration et ouverture de base de données
	
	Amazon RDS crée un instantané initial de la base de données après son ouverture à la fin des tâches de restauration suivantes :
	• RESTORE_DB
	• RESTORE_DB_DIFFERENTIAL

Colonne	Description
	• RESTORE_DB_LOG • FINISH_RESTORE
`database_name`	Nom de la base de données à laquelle est associée à la tâche.
`% complete`	La progression de la tâche sous forme de pourcentage.
`duration (mins)`	Temps consacré à la tâche, en minutes.
`lifecycle`	État de la tâche. Les statuts possibles sont les suivants : • `CREATED` – Dès que vous appelez `rds_backup_database` ou `rds_restore_database` , une tâche est créée et l'état est défini sur `CREATED`. • `IN_PROGRESS` – Après le démarrage d'une tâche de sauvegarde ou de restauration, l'état est défini sur `IN_PROGRESS` . Cela peut prendre jusqu'à 5 minutes pour que l'état change de `CREATED` à `IN_PROGRESS` . • `SUCCESS` – Après l'achèvement d'une tâche de sauvegarde ou de restauration, l'état est défini sur `SUCCESS`. • `ERROR` – En cas d'échec d'une tâche de sauvegarde ou de restauration, l'état est défini sur `ERROR`. Lisez la colonne `task_info` pour plus d'informations sur l'erreur. • `CANCEL_REQUESTED` – Dès que vous appelez `rds_cancel_task` , l'état de la tâche est défini sur `CANCEL_REQUESTED` . • `CANCELLED` – Une fois une tâche annulée avec succès, l'état de la tâche est défini sur `CANCELLED` .

Colonne	Description
`task_info`	Informations supplémentaires sur la tâche. Si une erreur se produit lors de la sauvegarde ou de la restauration d'une base de données, cette colonne contient des informations sur l'erreur. Pour obtenir une liste des erreurs possibles et des stratégies d'atténuation, consultez Résolution des problèmes.
`last_updated`	Date et heure de la dernière mise à jour de l'état de la tâche. Le statut est mis à jour tous les 5 pour cent de progression.
`created_at`	Date et heure de création de la tâche.
`S3_object_arn`	L'ARN indique le préfixe Amazon S3 et le nom du fichier en cours de sauvegarde ou de restauration.
`overwrite_s3_backup_file`	Valeur du paramètre `@overwrite_s3_backup_file` spécifié lorsque vous appelez une tâche de sauvegarde. Pour de plus amples informations, veuillez consulter Sauvegarde d'une base de données.
`KMS_master_key_arn`	L'ARN pour la clé KMS utilisée pour le chiffrement (pour la sauvegarde) et le déchiffrement (pour la restauration).
`filepath`	Non applicable aux tâches de sauvegarde et de restauration natives
`overwrite_file`	Non applicable aux tâches de sauvegarde et de restauration natives

Compression des fichiers de sauvegarde

Pour économiser de l'espace sur votre compartiment Amazon S3, vous pouvez compresser vos fichiers de sauvegarde. Pour plus d'informations sur la compression de fichiers de sauvegarde, consultez Backup Compression (Compression de sauvegarde) dans la documentation Microsoft.

La compression des fichiers sauvegardés est prise en charge pour les éditions de base de données suivantes :

- Configurer SQL Server Enterprise Edition

- Microsoft SQL Server Standard Edition

Pour vérifier l'option de compression de vos fichiers de sauvegarde, exécutez le code suivant :

```
exec rdsadmin.dbo.rds_show_configuration 'S3 backup compression';
```

Pour activer la compression de vos fichiers sauvegardés, exécutez le code suivant :

```
exec rdsadmin.dbo.rds_set_configuration 'S3 backup compression', 'true';
```

Pour désactiver la compression de vos fichiers sauvegardés, exécutez le code suivant :

```
exec rdsadmin.dbo.rds_set_configuration 'S3 backup compression', 'false';
```

Résolution des problèmes

Voici les problèmes que vous risquez de rencontrer lorsque vous utilisez la sauvegarde et la restauration natives.

Problème	Suggestions de dépannage
L'option de sauvegarde/restauration de base de données n'est pas encore activée ou est en cours d'activation. Réessayez ultérieurement.	Vérifiez que vous avez ajouté l'option `SQLSERVER_BACKUP_RESTORE` au groupe d'options de base de données associé à votre instance de base de données. Pour de plus amples informations, veuillez consulter Ajout de l'option de sauvegarde et restauration natives.
L'autorisation EXECUTE a été refusée sur l'objet « *rds_backup_database* », la base de données « msdb » et le schéma « dbo ».	Assurez-vous que vous utilisez l'utilisateur principal lors de l'exécution de la procédure stockée. Si vous rencontrez cette erreur même après avoir été connecté en tant qu'utilisateur principal, cela peut être dû à un mauvais alignement des autorisations de l'utilisateur administrateur. Pour réinitialiser l'utilisateur principal, utilisez le AWS Management Console. Consultez Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL.

Problème	Suggestions de dépannage
L'autorisation EXECUTE a été refusée sur l'objet « *rds_restore_database* », la base de données « msdb » et le schéma « dbo ».	Assurez-vous que vous utilisez l'utilisateur principal lors de l'exécution de la procédure stockée. Si vous rencontrez cette erreur même après avoir été connecté en tant qu'utilisateur principal, cela peut être dû à un mauvais alignement des autorisations de l'utilisateur administrateur. Pour réinitialiser l'utilisateur principal, utilisez le AWS Management Console. Consultez Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL.
Accès refusé	Le processus de sauvegarde ou de restauration ne parvient pas à accéder au fichier de sauvegarde. Cela est généralement provoqué par des problèmes tels que : • Référencement du compartiment incorrect. Référencement du compartiment avec un format incorrect. Référencement d'un nom de fichier sans utiliser l'ARN. • Autorisations incorrectes sur le fichier de compartiment. Par exemple, s'il a été créé par un compte différent qui essaie d'y accéder actuellement, ajoutez les autorisations correctes. • Une stratégie IAM incorrecte ou incomplète. Votre rôle IAM doit inclure tous les éléments nécessaires, y compris, par exemple, la version correcte. Ceux-ci sont mis en évidence dans Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.
BACKUP DATABASE With COMPRESSION n'est pas pris en charge sur l'Édition <edition_name>	La compression des fichiers de sauvegarde est uniquement prise en charge pour Microsoft SQL Server Enterprise Edition. La Standard Edition. Pour plus d'informations, consultez Compression des fichiers de sauvegarde.

Problème	Suggestions de dépannage
La clé <ARN> n'existe pas	Vous avez essayé de restaurer une sauvegarde chiffrée, mais n'avez pas fourni une clé de chiffrement valide. Vérifiez votre clé de chiffrement et réessayez. Pour plus d'informations, consultez Restauration d'une base de données.
Veuillez réémettre la tâche avec le type correct et remplacer la propriété	Si vous essayez de sauvegarder votre base de données et fournissez le nom d'un fichier qui existe déjà, mais que la propriété de remplacement est définie sur false, l'opération de sauvegarde échoue. Pour corriger cette erreur, entrez le nom d'un fichier qui n'existe pas déjà ou définissez la propriété de remplacement sur true. Pour plus d'informations, consultez Sauvegarde d'une base de données. Il est également possible qu'en tentant de restaurer votre base de données, vous ayez accidentellement appelé la procédure stockée `rds_backup_database` . Dans ce cas, appelez la procédure stockée `rds_restore_database` à la place. Pour plus d'informations, consultez Restauration d'une base de données. Si vous aviez pour but de restaurer votre base de données et avez appelé la procédure stockée `rds_restore_database` , assurez-vous que vous avez fourni le nom d'un fichier de sauvegard e valide. Pour plus d'informations, consultez Utilisation des sauvegarde et restauration natives.

Problème	Suggestions de dépannage
Veuillez spécifier un compartiment qui se trouve dans la même région que l'instance RDS	Vous ne pouvez pas effectuer de sauvegarde ou de restauration depuis un compartiment Amazon S3 situé dans une AWS région différente de celle de votre instance de base de données Amazon RDS. Vous pouvez utiliser la réplication Amazon S3 pour copier le fichier de sauvegarde dans la AWS région appropriée. Pour plus d'informations, consultez Réplication entre régions dans la documentation Amazon S3.
Le compartiment spécifié n'existe pas	Vérifiez que vous avez fourni l'ARN correct pour votre compartiment et le fichier, dans le bon format. Pour plus d'informations, consultez Utilisation des sauvegarde et restauration natives.
L’utilisateur <ARN> n’est pas autorisé à exécuter <kms action> sur la ressource <ARN>	Vous avez demandé une opération cryptée, mais vous n'avez pas fourni AWS KMS les autorisations appropriées. Vérifiez que vous avez les autorisations appropriées ou ajoutez-les. Pour plus d'informations, consultez Configuration pour les sauvegarde et restauration natives.
La tâche de restaurat ion n'est pas en mesure de restaurer à partir de plus de 10 fichiers de sauvegarde). Veuillez réduire le nombre de fichiers correspondants et réessayer.	Réduisez le nombre de fichiers que vous utilisez pour la restaurat ion. Si nécessaire, vous pouvez augmenter la taille de chaque fichier.

Problème	Suggestions de dépannage
La base de données *database_name* « » existe déjà. Deux bases de données qui ne diffèrent que par la casse ou l'accentuation ne sont pas autorisées. Choisissez un nom de base de données différent.	Vous ne pouvez pas restaurer une base de données qui porte le même nom qu'une base de données existante. Les noms de base de données sont uniques.

Importation et exportation de données SQL Server à l'aide d'autres méthodes

Par la suite, vous trouverez des informations sur l'utilisation d'instantanés pour importer vos données Microsoft SQL Server vers Amazon RDS. Vous trouverez également des informations sur l'utilisation d'instantanés pour exporter vos données depuis une instance de base de données RDS exécutant SQL Server.

Si votre scénario le prend en charge, il est plus facile transférer des données vers et depuis Amazon RDS à l'aide de la fonctionnalité de sauvegarde et restauration natives. Pour de plus amples informations, veuillez consulter Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

> **Note**
>
> Amazon RDS for Microsoft SQL Server ne prend pas en charge l'importation de données dans la base de données `msdb`.

Importation de données dans RDS for SQL Server à l'aide d'un instantané

Pour importer des données dans une instance de base de données SQL Server à l'aide d'un instantané

1. Créez une instance de base de données. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
2. Arrêtez l'accès des applications à l'instance de base de données de destination.

   Si vous empêchez l'accès à votre instance de base de données pendant l'importation des données, le transfert de données est plus rapide. En outre, vous n'aurez plus à vous inquiéter de conflits pendant le chargement de données si d'autres applications ne peuvent pas écrire sur l'instance de base de données simultanément. En cas de problème, et si vous devez procéder à une restauration vers un instantané de base de données antérieur, les seules modifications que vous perdez sont les données importées. Vous pouvez importer à nouveau ces données après avoir résolu le problème.

   Pour plus d'informations sur le contrôle de l'accès à votre instance de base de données, consultez Contrôle d'accès par groupe de sécurité.

3. Créez un instantané de la base de données cible.

   Si la base de données cible contient déjà des données, nous vous recommandons de prendre un instantané de la base de données avant d'importer les données. En cas de problème d'importation des données ou si vous souhaitez ignorer les modifications, l'instantané vous permet de restaurer l'état précédent de la base de données. Pour de plus amples informations sur les instantanés de base de données, veuillez consulter Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

   > **Note**
   >
   > Lorsque vous prenez un instantané de base de données, les I/O opérations sur la base de données sont suspendues pendant un moment (quelques millisecondes) pendant que la sauvegarde est en cours.

4. Désactivez les sauvegardes automatiques sur la base de données cible.

   La désactivation des sauvegardes automatiques sur l'instance de base de données cible améliore la performance pendant l'importation de vos données, car Amazon RDS ne consigne pas de transactions lorsque les sauvegardes automatiques sont désactivées. Cependant, certains éléments sont à prendre en compte. Des sauvegardes automatisées sont nécessaires pour effectuer une point-in-time restauration. Ainsi, vous ne pouvez pas restaurer la base de données à un instant spécifique dans le passé tant que vous importez des données. En outre, toutes les sauvegardes automatiques qui ont été créées sur l'instance de base de données sont effacées à moins que vous choisissiez de les conserver.

   Le choix de conserver les sauvegardes automatiques peut vous aider à vous protéger contre la suppression accidentelle de données. Amazon RDS enregistre également les propriétés de l'instance de base de données avec chaque sauvegarde automatique afin de faciliter la récupération. L'utilisation de cette option vous permet de restaurer une instance de base de données supprimée à un moment précis pendant la période de conservation des sauvegardes, même après sa suppression. Les sauvegardes automatisées sont automatiquement supprimées à la fin de la plage de sauvegarde spécifiée, comme pour une instance de base de données active.

   Vous pouvez également utiliser des instantanés précédents pour récupérer la base de données. En outre, tous les instantanés que vous avez pris demeureront disponibles. Pour plus d'informations sur les sauvegardes automatiques, consultez Présentation des sauvegardes.

5. Désactivez les contraintes de clé étrangère, le cas échéant.

   Si vous avez besoin de désactiver des contraintes de clé étrangère, vous pouvez utiliser pour cela le script suivant.

```
--Disable foreign keys on all tables
    DECLARE @table_name SYSNAME;
    DECLARE @cmd NVARCHAR(MAX);
    DECLARE table_cursor CURSOR FOR SELECT name FROM sys.tables;

    OPEN table_cursor;
    FETCH NEXT FROM table_cursor INTO @table_name;

    WHILE @@FETCH_STATUS = 0 BEGIN
      SELECT @cmd = 'ALTER TABLE '+QUOTENAME(@table_name)+' NOCHECK CONSTRAINT
 ALL';
      EXEC (@cmd);
      FETCH NEXT FROM table_cursor INTO @table_name;
    END

    CLOSE table_cursor;
    DEALLOCATE table_cursor;

    GO
```

6. Abandonnez les index, le cas échéant.
7. Désactivez les déclencheurs, le cas échéant.

   Si vous avez besoin de désactiver des déclencheurs, vous pouvez utiliser pour cela le script suivant.

```
--Disable triggers on all tables
    DECLARE @enable BIT = 0;
    DECLARE @trigger SYSNAME;
    DECLARE @table SYSNAME;
    DECLARE @cmd NVARCHAR(MAX);
    DECLARE trigger_cursor CURSOR FOR SELECT trigger_object.name trigger_name,
     table_object.name table_name
    FROM sysobjects trigger_object
    JOIN sysobjects table_object ON trigger_object.parent_obj = table_object.id
    WHERE trigger_object.type = 'TR';
```

```
    OPEN trigger_cursor;
    FETCH NEXT FROM trigger_cursor INTO @trigger, @table;

    WHILE @@FETCH_STATUS = 0 BEGIN
      IF @enable = 1
         SET @cmd = 'ENABLE ';
      ELSE
         SET @cmd = 'DISABLE ';

      SET @cmd = @cmd + ' TRIGGER dbo.'+QUOTENAME(@trigger)+' ON
dbo.'+QUOTENAME(@table)+' ';
      EXEC (@cmd);
      FETCH NEXT FROM trigger_cursor INTO @trigger, @table;
    END

    CLOSE trigger_cursor;
    DEALLOCATE trigger_cursor;

    GO
```

8. Interrogez l'instance SQL Server source pour tous les ID de connexion que vous souhaitez importer dans l'instance de base de données de destination.

   SQL Server stocke des ID de connexion et des mots de passe dans la base de données `master`. Du fait que Amazon RDS n'accorde pas l'accès à la base de données `master`, vous ne pouvez pas importer directement des ID de connexion et des mots de passe dans votre instance de base de données de destination. À la place, vous devez interroger la base de données `master` sur l'instance SQL Server source pour générer un fichier Langage de définition de données (DDL). Ce fichier doit inclure tous les identifiants et les mots de passe que vous souhaitez ajouter à l'instance de base de données de destination. Ce fichier doit également inclure des appartenances à un rôle et des autorisations que vous souhaitez transférer.

   Pour plus d'informations sur l'interrogation de la `master` base de données, voir Transférer des identifiants et des mots de passe entre des instances de SQL Server dans la base de connaissances Microsoft.

   Le résultat du script est un script que vous pouvez exécuter sur l'instance de base de données de destination. Le script dans l'article de base de connaissances comporte le code suivant :

```
p.type IN
```

Partout où `p.type` s'affiche, utilisez plutôt le code suivant :

```
p.type = 'S'
```

9. Importez les données à l'aide de la méthode dans Importer les données.
10. Accordez aux applications l'accès à l'instance de base de données cible.

    Lorsque l'importation de vos données est terminée, vous pouvez accorder l'accès à l'instance de base de données aux applications que vous avez bloquées pendant l'importation. Pour plus d'informations sur le contrôle de l'accès à votre instance de base de données, consultez Contrôle d'accès par groupe de sécurité.
11. Activez les sauvegardes automatiques sur l'instance de base de données cible.

    Pour plus d'informations sur les sauvegardes automatiques, consultez Présentation des sauvegardes.
12. Activer les contraintes de clé étrangère.

    Si vous avez désactivé les contraintes de clé étrangère précédemment, vous pouvez à présent les activer avec le script suivant.

```
--Enable foreign keys on all tables
    DECLARE @table_name SYSNAME;
    DECLARE @cmd NVARCHAR(MAX);
    DECLARE table_cursor CURSOR FOR SELECT name FROM sys.tables;

    OPEN table_cursor;
    FETCH NEXT FROM table_cursor INTO @table_name;

    WHILE @@FETCH_STATUS = 0 BEGIN
      SELECT @cmd = 'ALTER TABLE '+QUOTENAME(@table_name)+' CHECK CONSTRAINT ALL';
      EXEC (@cmd);
      FETCH NEXT FROM table_cursor INTO @table_name;
    END

    CLOSE table_cursor;
    DEALLOCATE table_cursor;
```

13. Activez les index, le cas échéant.
14. Activez les déclencheurs, le cas échéant.

Si vous avez désactivé les déclencheurs précédemment, vous pouvez à présent les activer avec le script suivant.

```
--Enable triggers on all tables
    DECLARE @enable BIT = 1;
    DECLARE @trigger SYSNAME;
    DECLARE @table SYSNAME;
    DECLARE @cmd NVARCHAR(MAX);
    DECLARE trigger_cursor CURSOR FOR SELECT trigger_object.name trigger_name,
     table_object.name table_name
    FROM sysobjects trigger_object
    JOIN sysobjects table_object ON trigger_object.parent_obj = table_object.id
    WHERE trigger_object.type = 'TR';

    OPEN trigger_cursor;
    FETCH NEXT FROM trigger_cursor INTO @trigger, @table;

    WHILE @@FETCH_STATUS = 0 BEGIN
      IF @enable = 1
         SET @cmd = 'ENABLE ';
      ELSE
         SET @cmd = 'DISABLE ';

      SET @cmd = @cmd + ' TRIGGER dbo.'+QUOTENAME(@trigger)+' ON
 dbo.'+QUOTENAME(@table)+' ';
      EXEC (@cmd);
      FETCH NEXT FROM trigger_cursor INTO @trigger, @table;
    END

    CLOSE trigger_cursor;
    DEALLOCATE trigger_cursor;
```

Importer les données

Microsoft SQL Server Management Studio est un client SQL Server graphique qui est inclus dans toutes les éditions de Microsoft SQL Server, sauf Express Edition. SQL Server Management Studio Express est disponible auprès de Microsoft en téléchargement gratuit. Pour trouver ce téléchargement, consultez le site Web de Microsoft.

> **Note**
>
> SQL Server Management Studio Express est disponible uniquement en tant qu'application basée sur Windows.

SQL Server Management Studio inclut les outils suivants, qui sont utiles pour l'importation de données dans une instance de base de données SQL Server :

- Assistant de génération et de publication de scripts
- Assistant d'importation et d'exportation
- Copie en bloc

Assistant de génération et de publication de scripts

L'assistant de génération et de publication de scripts crée un script qui contient le schéma d'une base de données, les données elles-mêmes, ou les deux. Vous pouvez générer un script pour une base de données dans votre déploiement SQL Server local. Vous pouvez ensuite exécuter le script pour transférer les informations qu'il contient vers une instance de base de données Amazon RDS.

> **Note**
>
> Pour les bases de données d'1 GiB ou plus, il est plus efficace de scripter uniquement le schéma de la base de données. Ensuite, vous utilisez l'assistant d'importation et d'exportation ou la fonctionnalité de copie en bloc de SQL Server pour transférer les données.

Pour plus d'informations sur l'assistant de génération et de publication de scripts, consultez la documentation Microsoft SQL Server.

Dans l'assistant, prêtez une attention particulière aux options avancées sur la page Set Scripting Options (Définir les options de scripting) pour garantir que tout ce que vous souhaitez que votre script inclue soit sélectionné. Par exemple, par défaut, les déclencheurs de base de données ne sont pas inclus dans le script.

Lorsque le script est généré et enregistré, vous pouvez utiliser SQL Server Management Studio pour vous connecter à votre instance de base de données puis exécuter le script.

Assistant d'importation et d'exportation

L'assistant d'importation et d'exportation crée un package Integration Services spécial que vous pouvez utiliser pour copier des données depuis votre base de données SQL Server locale vers l'instance de base de données de destination. L'assistant peut filtrer les tables et même les tuples dans une table qui sont copiés vers l'instance de base de données de destination.

> **Note**
>
> L'assistant d'importation et d'exportation est efficace pour les jeux de données de grande taille, mais ce n'est peut-être pas la manière la plus rapide d'exporter des données à distance depuis votre déploiement local. Pour une manière encore plus rapide, envisagez la fonction de copie en bloc SQL Server.

Pour plus d'informations sur l'assistant d'importation et d'exportation, consultez la documentation Microsoft SQL Server.

Dans l'assistant, sur la page Choose a Destination (Choisir une destination), procédez comme suit :

- Pour Server Name (Nom de serveur), saisissez le nom du point de terminaison pour votre instance de base de données.
- Pour le mode d'authentification de serveur, sélectionnez Use SQL Server Authentication (Utiliser l'authentification SQL Server).
- Pour Nom d'utilisateur et Mot de passe, entrez les informations d'identification pour l'utilisateur principal que vous avez créé pour l'instance de base de données.

Copie en bloc

La fonction Copie en bloc de SQL Server est un moyen efficace de copier des données depuis une base de données source vers votre instance de base de données. La copie en bloc écrit les données que vous spécifiez vers un fichier de données, tel qu'un fichier ASCII. Vous pouvez ensuite exécuter à nouveau la copie en bloc pour écrire le contenu du fichier dans l'instance de base de données de destination.

Cette section utilise l'utilitaire bcp qui est inclus dans toutes les éditions de SQL Server. Pour plus d'informations sur les opérations d'importation et d'exportation en bloc, consultez la documentation Microsoft SQL Server.

> **Note**
>
> Avant d'utiliser la copie en bloc, vous devez importer votre schéma de base de données vers l'instance de base de données de destination. L'assistant de génération et de publication de scripts, décrit plus tôt dans cette rubrique, est un excellent outil à cette fin.

La commande suivante se connecte à l'instance SQL Server locale. Elle génère un fichier délimité par des tabulations d'une table spécifique dans le répertoire racine C:\ de votre déploiement SQL Server existant. La table est spécifiée par son nom entièrement qualifié, et le fichier texte a le même nom que la table qui est copiée.

```
bcp dbname.schema_name.table_name out C:\table_name.txt -n -S localhost -U username -
P password -b 10000
```

Le code précédent inclut les options suivantes :

- `-n` spécifie que la copie en bloc utilise les types de données natifs des données à copier.
- `-S` spécifie l'instance SQL Server à laquelle l'utilitaire bcp se connecte.
- `-U` spécifie le nom d'utilisateur du compte qui se connecte à l'instance SQL Server.
- `-P` spécifie le mot de passe pour l'utilisateur spécifié par `-U`.
- `-b` spécifie le nombre de lignes par lot de données importées.

> **Note**
>
> Il peut exister d'autres paramètres qui sont importants pour votre situation d'importation. Par exemple, vous pouvez avoir besoin du paramètre `-E` qui concerne les valeurs d'identité. Pour plus d'informations, consultez la description complète de la syntaxe de ligne de commande pour l'utilitaire bcp dans la documentation Microsoft SQL Server.

Prenons l'exemple d'une base de données nommée `store` qui utilise le schéma par défaut, `dbo`, et qui contient une table nommée `customers`. Le compte utilisateur `admin`, avec le mot de passe `insecure`, copie 10 000 lignes de la table `customers` dans un fichier nommé `customers.txt`.

```
bcp store.dbo.customers out C:\customers.txt -n -S localhost -U admin -P insecure -b
 10000
```

Après avoir généré le fichier de données, vous pouvez charger les données sur votre instance de base de données à l'aide d'une commande similaire. Au préalable, créez la base de données et le schéma sur l'instance de base de données cible. Ensuite, utiliser l'argument `in` pour spécifier un fichier d'entrée au lieu de `out` pour spécifier un fichier de sortie. Au lieu d'utiliser l'hôte local pour spécifier l'instance SQL Server locale, vous spécifiez le point de terminaison de votre instance de base de données. Si vous utilisez un port autre que 1433, vous le spécifiez également. Le nom d'utilisateur et le mot de passe sont identiques à l'utilisateur principal et au mot de passe de votre instance de base de données. La syntaxe est la suivante.

```
bcp dbname.schema_name.table_name
     in C:\table_name.txt -n -S endpoint,port -U master_user_name -
P master_user_password -b 10000
```

Pour poursuivre l'exemple précédent, supposons que le nom d'utilisateur maître soit `admin`, et que le mot de passe soit `insecure`. Le point de terminaison pour l'instance de base de données est `rds.ckz2kqd4qsn1.us-east-1.rds.amazonaws.com`, et vous utilisez le port 4080. La commande est la suivante.

```
bcp store.dbo.customers in C:\customers.txt -n -S rds.ckz2kqd4qsn1.us-
east-1.rds.amazonaws.com,4080 -U admin -P insecure -b 10000
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Exportation de données depuis RDS for SQL Server

Vous pouvez choisir l'une des options suivantes pour exporter des données à partir d'une instance de base de données RDS for SQL Server :

- Sauvegarde et restauration natives à l'aide d'un fichier de sauvegarde complète (.bak) – L'utilisation de fichiers .bak pour sauvegarder des bases de données est fortement optimisée et constitue généralement le moyen le plus rapide d'exporter des données. Pour plus d'informations,

consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

- Assistant d'importation et d'exportation SQL Server – Pour en savoir plus, consultez Assistant d'importation et d'exportation SQL Server.
- Assistant de génération et de publication de scripts SQL Server et utilitaire bcp – Pour en savoir plus, consultez Assistant Générer et publier des scripts et utilitaire bcp.

Assistant d'importation et d'exportation SQL Server

Vous pouvez utiliser l'assistant d'importation et d'exportation SQL Server pour copier une ou plusieurs tables, vues ou requêtes depuis votre instance de base de données RDS for SQL Server vers un autre magasin de données. Ce choix est le meilleur si le magasin de données cible n'est pas SQL Server. Pour plus d'informations, consultez la section relative à l'Assistant d'importation et exportation SQL Server dans la documentation Microsoft SQL Server.

L'assistant d'importation et d'exportation SQL Server est disponible dans le cadre de Microsoft SQL Server Management Studio. Ce client SQL Server graphique est inclus dans toutes les éditions de Microsoft SQL Server, sauf Express Edition. SQL Server Management Studio Express est disponible uniquement en tant qu'application basée sur Windows. SQL Server Management Studio Express est disponible auprès de Microsoft en téléchargement gratuit. Pour trouver ce téléchargement, consultez le site Web de Microsoft.

Pour utiliser l'assistant d'importation et d'exportation SQL Server pour exporter des données

1. Dans SQL Server Management Studio, connectez-vous à votre instance de base de données RDS for SQL Server. Pour plus d'informations sur la manière de procéder, consultez Connexion à votre instance de base de données Microsoft SQL Server.
2. Dans Object Explorer (Navigateur d'objet), développez Databases (Bases de données), ouvrez le menu contextuel (clic droit) pour la base de données source, et sélectionnez Tasks (Tâches), puis Export Data (Exporter des données). L'assistant s'affiche.
3. Sur la page Choose a Data Source (Choisir une source de données), procédez comme suit :
    a. Pour Data source (Source de données), choisissez **`SQL Server Native Client 11.0`**.
    b. Vérifiez que la case Server name (Nom du serveur) indique le point de terminaison de votre instance de base de données RDS for SQL Server.

c. Sélectionnez Use SQL Server Authentication (Utiliser l'authentification SQL Server). Pour User name (Nom d'utilisateur) et Password (Mot de passe), saisissez le nom d'utilisateur principal et le mot de passe de votre instance de base de données.

d. Vérifiez que la zone Base de données affiche la base de données depuis laquelle vous souhaitez exporter des données.

e. Choisissez Suivant.

4. Sur la page Choose a Destination (Choisir une destination), procédez comme suit :

a. Pour Destination, choisissez **`SQL Server Native Client 11.0`**.

> **Note**
>
> D'autres sources de données cibles sont disponibles. Elles incluent les suivantes : les fournisseurs de données .NET Framework, les fournisseurs OLE DB, les fournisseurs SQL Server Native Client, les fournisseurs ADO.NET, Microsoft Office Excel, Microsoft Office Access et la source du fichier plat. Si vous choisissez de cibler l'une de ces sources de données, ignorez le reste de l'étape 4. Pour obtenir des détails sur les informations de connexion à fournir ensuite, veuillez consulter Choix d'une destination dans la documentation SQL Server.

b. Pour Server name (Nom de serveur), saisissez le nom du serveur de l'instance de base de données SQL Server cible.

c. Choisissez le type d'authentification qui convient. Saisissez un nom d'utilisateur et un mot de passe si nécessaire.

d. Pour Base de données, choisissez le nom de base de données cible, ou sélectionnez Nouveau pour créer une base de données qui contiendra les données exportées.

Si vous choisissez Nouveau, consultez Créer une base de données dans la documentation SQL Server pour obtenir des renseignements sur les informations de base de données à fournir.

e. Choisissez Suivant.

5. Sur la page Table Copy or Query (Copie ou requête de tableau), choisissez Copy data from one or more tables or views (Copier des données à partir d'un ou plusieurs tableaux ou affichages) ou Write a query to specify the data to transfer (Rédiger une requête pour spécifier les données à transférer). Choisissez Suivant.

6. Si vous avez choisi Write a query to specify the data to transfer (Rédiger une requête pour spécifier les données à transférer), vous voyez la page Provide a Source Query (Fournir une requête source). Saisissez ou collez une requête SQL, puis choisissez Parse (Analyser) pour la vérifier. Après la validation de la requête, choisissez Suivant.
7. Sur la page Select Source Tables and Views (Sélectionner les tableaux et affichages source), procédez comme suit :

    a. Sélectionnez les tables et les affichages que vous souhaitez exporter, ou bien vérifiez que la requête que vous avez fournie est sélectionnée.

    b. Choisissez Edit Mappings (Modifier le mappage) puis spécifiez les informations de mappage de la colonne et de la base de données. Pour de plus amples informations, veuillez consulter Mapping de colonnes (Mappage des colonnes) dans la documentation SQL Server.

    c. (Facultatif) Pour afficher un aperçu des données à exporter, sélectionnez la table, l'affichage ou la requête, puis choisissez Preview (Aperçu).

    d. Choisissez Suivant.
8. Sur la page Run Package (Exécuter le package), vérifiez que l'option Run immediately (Exécuter immédiatement) est sélectionnée. Choisissez Suivant.
9. Sur la page Complete the Wizard (Finaliser l'assistant), vérifiez que les informations d'exportation des données sont telles que vous les attendez. Choisissez Finish (Terminer).
10. Sur la page The execution was successful (L'exécution est réussie), choisissez Close (Fermer).

Assistant Générer et publier des scripts et utilitaire bcp

Vous pouvez utiliser l'assistant de génération et de publication de scripts pour créer des scripts pour une base de données entière ou seulement des objets sélectionnés. Vous pouvez exécuter ces scripts sur une instance de base de données SQL Server pour recréer les objets à base de script. Vous pouvez ensuite utiliser l'utilitaire bcp pour exporter en bloc les données pour les objets sélectionnés vers l'instance de base de données cible. Ce choix est préférable si vous souhaitez déplacer une base de données complète (dont des objets autres que des tables) ou d'importantes quantités de données entre deux instances DB SQL Server. Pour une description complète de la syntaxe de ligne de commande bcp, veuillez consulter la section relative à l'utilitaire bcp dans la documentation Microsoft SQL Server.

L'assistant SQL Server Generate et Publish Scripts est disponible dans le cadre de Microsoft SQL Server Management Studio. Ce client SQL Server graphique est inclus dans toutes les éditions de

Microsoft SQL Server, sauf Express Edition. SQL Server Management Studio Express est disponible uniquement en tant qu'application basée sur Windows. SQL Server Management Studio Express est disponible auprès de Microsoft en téléchargement gratuit.

Pour utiliser l'assistant de génération et de publication de scripts et l'utilitaire bcp pour exporter des données, procédez comme suit :

1. Dans SQL Server Management Studio, connectez-vous à votre instance de base de données RDS for SQL Server. Pour plus d'informations sur la manière de procéder, consultez Connexion à votre instance de base de données Microsoft SQL Server.
2. Dans Object Explorer (Navigateur d'objet), développez le nœud Bases de données et sélectionnez la base de données que vous souhaitez baser sur script.
3. Suivez les instructions dans Assistant générer et publier des scripts dans la documentation SQL Server pour créer un fichier script.
4. Dans SQL Server Management Studio, connectez-vous à votre instance de base de données SQL Server cible.
5. Avec l'instance de base de données SQL Server cible sélectionnée dans Object Explorer (Navigateur d'objet), dans le menu File (Fichier), choisissez Open (Ouvrir), sélectionnez File (Fichier), puis ouvrez le fichier script.
6. Si vous avez scripté la base de données entière, examinez l'instruction CREATE DATABASE dans le script. Veillez à ce que la base de données soit créée dans l'emplacement avec les paramètres de votre choix. Pour plus d'informations, consultez CREATE DATABASE (CRÉER UNE BASE DE DONNÉES) dans la documentation SQL Server.
7. Si vous créez des utilisateurs de base de données dans le script, assurez-vous que les ID de connexion serveur existent sur l'instance de base de données cible pour ces utilisateurs. Sinon, créez des ID de connexion pour ces utilisateurs. Dans le cas contraire, les commandes à base de script pour créer les utilisateurs de base de données échoueront. Pour de plus amples informations, veuillez consulter Créer un compte de connexion dans la documentation SQL Server.
8. Choisissez !Execute dans le menu SQL Editor pour exécuter le fichier script et créer les objets de base de données. Lorsque le script se termine, vérifiez que tous les objets de base de données existent comme prévu.
9. Utilisez l'utilitaire bcp pour exporter des données depuis l'instance de base de données RDS for SQL Server vers des fichiers. Ouvrez une invite de commande et saisissez la commande suivante.

```
bcp database_name.schema_name.table_name out data_file -n -S aws_rds_sql_endpoint -
U username -P password
```

Le code précédent inclut les options suivantes :

- table_name correspond au nom d'une des tables que vous avez recréées dans la base de données cible et que vous voulez à présent remplir avec des données.
- data_file correspond au nom et au chemin d'accès complet du fichier de données à créer.
- `-n` spécifie que la copie en bloc utilise les types de données natifs des données à copier.
- `-S` spécifie l'instance de base de données SQL Server d'origine de l'exportation.
- `-U` spécifie le nom d'utilisateur à utiliser lors de la connexion à l'instance de base de données SQL Server.
- `-P` spécifie le mot de passe pour l'utilisateur spécifié par `-U`.

Ce qui suit présente un exemple de commande .

```
bcp world.dbo.city out C:\Users\JohnDoe\city.dat -n -S sql-jdoe.1234abcd.us-
west-2.rds.amazonaws.com,1433 -U JohnDoe -P ClearTextPassword
```

Répétez cette étape jusqu'à ce que vous ayez des fichiers de données pour toutes les tables que vous souhaitez exporter.

10. Préparez votre instance de base de données cible pour l'importation en bloc de données en suivant les instructions dans Préparer l'importation de données en bloc dans la documentation SQL Server.
11. Décidez du choix d'une méthode d'importation en bloc à utiliser après avoir pris en compte la performance et d'autres problèmes abordés dans A propos des opérations d'exportation et d'importation en bloc dans la documentation SQL Server.
12. Importez en bloc les données depuis les fichiers de données que vous créez à l'aide de l'utilitaire bcp. Pour ce faire, suivez les instructions dans Importer et exporter des données en bloc à l'aide de l'utilitaire bcp ou Importer des données en bloc à l'aide de BULK INSERT ou OPENROWSET(BULK...) dans la documentation SQL Server, selon ce que vous avez décidé à l'étape 11.

Utilisation de l'utilitaire BCP de Linux pour importer et exporter des données

L'utilitaire BCP (Bulk Copy Program) fournit un moyen efficace de transférer de grandes quantités de données entre votre instance de base de données RDS pour SQL Server et les fichiers de données. Vous pouvez utiliser le BCP à partir d'environnements Linux pour effectuer des opérations de données en masse, ce qui le rend utile pour la migration des données, les processus ETL et les transferts de données réguliers.

Le BCP prend en charge à la fois l'importation de données depuis des fichiers vers des tables SQL Server et l'exportation de données depuis des tables SQL Server vers des fichiers. Cela est particulièrement efficace pour le transfert de données structurées dans différents formats, y compris des fichiers texte délimités.

Prérequis

Avant d'utiliser BCP avec votre instance de base de données RDS pour SQL Server depuis Linux, assurez-vous de disposer des éléments suivants :

- Un environnement Linux avec connectivité réseau à votre instance de base de données RDS pour SQL Server
- Outils de ligne de commande Microsoft SQL Server installés sur votre système Linux, notamment :
  - sqlcmd - Outil de requête en ligne de commande SQL Server
  - bcp - Utilitaire du programme Bulk Copy
- Informations d'identification valides pour votre instance de base de données RDS pour SQL Server
- Accès au réseau configuré via des groupes de sécurité pour autoriser les connexions sur le port SQL Server (généralement 1433)
- Autorisations de base de données appropriées pour les opérations que vous souhaitez effectuer

Installation des outils de ligne de commande SQL Server sous Linux

Pour utiliser BCP depuis Linux, vous devez installer les outils de ligne de commande Microsoft SQL Server. Pour obtenir des instructions d'installation détaillées pour votre distribution Linux spécifique, consultez la documentation Microsoft suivante :

- Installez sqlcmd et bcp, les outils de ligne de commande de SQL Server sous Linux
- utilitaire bcp - Référence complète pour l'utilitaire BCP

Après l'installation, assurez-vous que les outils sont disponibles dans votre PATH en exécutant :

```
bcp -v
sqlcmd -?
```

Exportation de données depuis RDS for SQL Server

Vous pouvez utiliser BCP pour exporter les données de votre instance de base de données RDS pour SQL Server vers des fichiers de votre système Linux. Cela est utile pour créer des sauvegardes, analyser des données ou préparer des données pour la migration.

Syntaxe d'exportation de base

La syntaxe de base pour exporter des données à l'aide du protocole BCP est la suivante :

```
bcp database.schema.table out output_file -S server_name -U username -P password
 [options]
```

Où :

- `database.schema.table`- Le nom complet de la table
- `output_file`- Le chemin et le nom du fichier de sortie
- `server_name`- Votre point de terminaison RDS pour SQL Server
- `username`- Nom d'utilisateur de votre base de données
- `password`- Votre mot de passe de base de données

Exemple d'exportation

L'exemple suivant exporte des données depuis une table nommée `customers` dans la `sales` base de données :

```
bcp sales.dbo.customers out /home/user/customers.txt \
    -S mydb.cluster-abc123.us-east-1.rds.amazonaws.com \
    -U admin \
    -P mypassword \
    -c \
    -t "|" \
```

```
    -r "\n"
```

Cette commande :

- Exporte les données de la `customers` table
- Enregistre la sortie dans `/home/user/customers.txt`
- Utilise le format de caractères (`-c`)
- Utilise le tube (|) comme délimiteur de champ () `-t "|"`
- Utilise une nouvelle ligne comme délimiteur de ligne () `-r "\n"`

Importation de données vers RDS pour SQL Server

Vous pouvez utiliser BCP pour importer des données à partir de fichiers de votre système Linux dans votre instance de base de données RDS for SQL Server. Cela est utile pour la migration des données, le chargement des données de test ou les mises à jour régulières des données.

Syntaxe d'importation de base

La syntaxe de base pour importer des données à l'aide du protocole BCP est la suivante :

```
bcp database.schema.table in input_file -S server_name -U username -P password
 [options]
```

Où :

- `database.schema.table`- Le nom complet de la table de destination
- `input_file`- Le chemin et le nom du fichier d'entrée
- `server_name`- Votre point de terminaison RDS pour SQL Server
- `username`- Nom d'utilisateur de votre base de données
- `password`- Votre mot de passe de base de données

Exemple d'importation

L'exemple suivant importe des données d'un fichier dans une table nommée `customers` :

```
bcp sales.dbo.customers in /home/user/customers.txt \
```

```
    -S mydb.cluster-abc123.us-east-1.rds.amazonaws.com \
    -U admin \
    -P mypassword \
    -c \
    -t "|" \
    -r "\n" \
    -b 1000
```

Cette commande :

- Importe des données dans le `customers` tableau
- Lit les données de `/home/user/customers.txt`
- Utilise le format de caractères (`-c`)
- Utilise le tube (|) comme délimiteur de champ () `-t "|"`
- Utilise une nouvelle ligne comme délimiteur de ligne () `-r "\n"`
- Traite les données par lots de 1 000 lignes () `-b 1000`

Options BCP courantes

Le BCP propose de nombreuses options pour contrôler le formatage des données et le comportement de transfert. Le tableau suivant décrit les options couramment utilisées :

Option	Description
`-c`	Utilise le type de données de type caractère pour toutes les colonnes
`-n`	Utilise les types de données natifs des bases
`-t`	Spécifie le délimiteur de champ (la valeur par défaut est tab)
`-r`	Spécifie le délimiteur de ligne (la valeur par défaut est une nouvelle ligne)
`-b`	Spécifie la taille du lot pour les opérations groupées
`-F`	Spécifie la première ligne à exporter ou à importer
`-L`	Spécifie la dernière ligne à exporter ou à importer
`-e`	Spécifie un fichier d'erreur pour capturer les lignes rejetées

Option	Description
`-f`	Spécifie un fichier de format pour le formatage des données
`-q`	Utilise des identifiants entre guillemets pour les noms d'objets

Bonnes pratiques et considérations

Lorsque vous utilisez BCP avec RDS pour SQL Server à partir de Linux, tenez compte des meilleures pratiques suivantes :

- Utiliser le traitement par lots : pour les grands ensembles de données, utilisez l'`-b`option permettant de traiter les données par lots. Cela améliore les performances et permet une meilleure récupération des erreurs.
- Gérez les erreurs avec élégance : utilisez `-e` cette option pour capturer les informations relatives aux erreurs et les lignes rejetées dans un fichier distinct à des fins d'analyse.
- Choisissez les formats de données appropriés : utilisez le format de caractères (`-c`) pour la compatibilité multiplateforme ou le format natif (`-n`) pour de meilleures performances lorsque la source et la destination sont SQL Server.
- Sécurisez vos informations d'identification : évitez de saisir des mots de passe directement dans les lignes de commande. Envisagez d'utiliser des variables d'environnement ou des fichiers de configuration dotés des autorisations appropriées.
- Testez avec de petits ensembles de données : avant de traiter de grandes quantités de données, testez vos commandes BCP avec des ensembles de données plus petits pour vérifier le formatage et la connectivité.
- Surveillez la connectivité réseau : garantissez des connexions réseau stables, en particulier pour les transferts de données importants. Envisagez d'utiliser des outils tels que `screen` ou `tmux` pour des opérations de longue durée.
- Valider l'intégrité des données : après le transfert des données, vérifiez le nombre de lignes et les échantillons de données pour vous assurer que l'opération s'est bien déroulée.

Résolution des problèmes courants

Le tableau suivant décrit les problèmes courants que vous pouvez rencontrer lors de l'utilisation de BCP depuis Linux et leurs solutions :

Problème	Solution
Expiration du délai de connexion ou erreurs réseau	Vérifiez votre point de terminaison Amazon RDS, les paramètres de votre groupe de sécurité et votre connectivité réseau. Assurez-vous que le port SQL Server (généralement 1433) est accessible depuis votre système Linux.
Authentication failures (Échecs d’authentification)	Vérifiez votre nom d'utilisateur et votre mot de passe. Assurez-vous que l'utilisateur de la base de données dispose des autorisations appropriées pour les opérations que vous effectuez.
Erreurs de format de données	Vérifiez vos délimiteurs de champs et de lignes. Assurez-vous que le format des données correspond aux attentes du BCP. Utilisez des fichiers de format pour les structures de données complexes.
Erreurs de type « Autorisation refusée »	Assurez-vous que l'utilisateur de votre base de données dispose `INSERT` des autorisations pour les importations ou `SELECT` des autorisations pour les exportations sur les tables cibles.
Problèmes liés à la gestion de fichiers volumineux	Utilisez le traitement par lots avec l'`-b`option. Envisagez de diviser les fichiers volumineux en petits morceaux pour améliorer les performances et corriger les erreurs.
Problèmes de codage des caractères	Assurez-vous que vos fichiers de données utilisent un codage de caractères compatible. Utilisez l'`-c`option pour le format des caractères ou spécifiez les pages de code appropriées.

Utilisation des réplicas en lecture pour Microsoft SQL Server dans Amazon RDS

Vous utilisez généralement des réplicas en lecture pour configurer la réplication entre instances de base de données Amazon RDS. Pour obtenir des informations générales sur les réplicas en lecture, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Cette section contient des informations spécifiques sur l'utilisation des réplicas en lecture sur Amazon RDS pour SQL Server.

- Synchronisation des utilisateurs et des objets de base de données avec une réplique en lecture SQL sur serveur
- Résolution d'un problème de réplication de lecture sur un SQL serveur

Configuration des réplicas en lecture pour SQL Server

Avant qu'une instance de base de données puisse être utilisée comme instance source pour la réplication, vous devez activer les sauvegardes automatiques sur l'instance de base de données source. Pour cela, vous devez définir la période de rétention des sauvegardes sur une valeur autre que 0. Pour définir ce type de déploiement, l'activation des sauvegardes automatiques doit également être effective.

La création d'un réplica en lecture SQL Server ne nécessite pas d'arrêt de l'instance de base de données principale. Amazon RDS définit les paramètres et autorisations nécessaires pour l'instance de base de données source et le réplica en lecture sans interruption de service. Un instantané de l'instance de base de données source est pris, et devient le réplica en lecture. Aucune interruption de service ne se produit lorsque vous supprimez un réplica en lecture.

Vous pouvez créer jusqu'à 15 réplicas en lecture à partir d'une seule instance de base de données source. Pour que la réplication fonctionne efficacement, nous vous recommandons de configurer chaque réplica en lecture avec la même quantité de ressources de calcul et de stockage que l'instance de base de données source. Si vous mettez à l'échelle l'instance de base de données source, faites-le également pour les réplicas en lecture.

La version du moteur de base de données SQL Server de l'instance de base de données source et de tous ses réplicas en lecture doit être identique. Amazon RDS met à niveau la base de données principale immédiatement après la mise à niveau des réplicas en lecture, indépendamment de la fenêtre de maintenance. Pour de plus amples informations sur la mise à niveau de la version du

moteur de base de données, veuillez consulter la section Mises à niveau du moteur de base de données Microsoft SQL Server.

Pour qu'un réplica en lecture reçoive et applique les modifications de la source, il doit disposer de ressources de calcul et de stockage suffisantes. Si un réplica en lecture atteint sa capacité en ce qui concerne les ressources de calcul, de réseau ou de stockage, il arrête de recevoir ou d'appliquer les modifications provenant de sa source. Vous pouvez modifier les ressources de stockage et d'UC d'un réplica en lecture indépendamment de sa source et d'autres réplicas en lecture.

Pour plus d'informations sur la création d'une réplique en lecture, consultezCréation d'un réplica en lecture.

Limites des réplicas en lecture avec SQL Server

Les limites suivantes s'appliquent aux réplicas en lecture SQL Server sur Amazon RDS :

- Les réplicas en lecture sont uniquement disponibles sur le moteur SQL Server Enterprise Edition (EE).
- Les répliques en lecture sont disponibles pour les versions de SQL Server 2016—2022.
- Vous pouvez créer jusqu'à 15 réplicas en lecture à partir d'une seule instance de base de données source. La réplication peut être retardée lorsque votre instance de base de données source possède plus de 5 répliques de lecture.
- Les répliques de lecture ne sont disponibles que pour les instances de base de données exécutées sur des classes d'instances de base de données avec quatre v. ou plus. CPUs
- Une réplique en lecture prend en charge jusqu'à 100 bases de données selon le type de classe d'instance et le mode de disponibilité. Vous devez créer des bases de données sur l'instance de base de données source pour les répliquer automatiquement dans les répliques de lecture. Vous ne pouvez pas choisir des bases de données individuelles à répliquer. Pour de plus amples informations, veuillez consulter Limites propres aux instances de bases de données Microsoft SQL Server.
- Vous ne pouvez pas supprimer une base de données d'une réplique lue. Pour supprimer une base de données, supprimez-la de l'instance de base de données source avec la procédure `rds_drop_database` stockée. Pour de plus amples informations, veuillez consulter Suppression d'une base de données dans une instance de base de données Amazon RDS pour Microsoft SQL Server.
- Si l'instance de base de données source utilise le chiffrement transparent des données (TDE) pour chiffrer les données, la réplique en lecture configure également automatiquement le TDE.

Si l'instance de base de données source utilise une clé KMS pour chiffrer les données, les répliques de lecture dans la même région utilisent la même clé KMS. Pour les répliques de lecture entre régions, vous devez spécifier une clé KMS provenant de la région de la réplique de lecture lors de la création de la réplique de lecture. Vous ne pouvez pas modifier la clé KMS d'une réplique en lecture.

- Les répliques en lecture ont le même fuseau horaire et le même classement que l'instance de base de données source, quel que soit le fuseau de disponibilité dans lequel elles ont été créées.
- Les éléments suivants ne sont pas pris en charge sur Amazon RDS pour SQL Server :
  - Rétention des sauvegardes des réplicas en lecture
  - Point-in-time restauration à partir de répliques lues
  - Instantanés manuels de réplicas en lecture
  - Réplicas en lecture multi-AZ
  - Création de réplicas en lecture à partir de réplicas en lecture
  - Synchronisation des connexions utilisateur pour lire les réplicas en lecture
- Amazon RDS pour SQL Server n'intervient pas pour atténuer un retard de réplica élevé entre une instance de base de données source et ses réplicas en lecture. Assurez-vous que l'instance de base de données et ses réplicas en lecture ont une taille appropriée, en termes de puissance de calcul et de stockage, afin de pouvoir répondre aux besoins de la charge opérationnelle.
- Vous pouvez effectuer une réplication entre les régions AWS GovCloud (USA Est) et AWS GovCloud (USA Ouest), mais pas à l'intérieur ou à l'extérieur. AWS GovCloud (US) Regions

Considérations relatives aux options pour les réplicas RDS for SQL Server

Avant de créer un réplica RDS for SQL Server, tenez compte des exigences, restrictions et recommandations suivantes :

- Si votre réplica SQL Server se trouve dans la même région que son instance de base de données source, assurez-vous qu'il appartient au même groupe d'options que l'instance de base de données source. Les modifications apportées au groupe d'options source ou à l'appartenance au groupe d'options source sont propagées aux réplicas. Ces modifications sont appliquées aux réplicas immédiatement après leur application à l'instance de base de données source, quelle que soit la fenêtre de maintenance du réplica.

  Pour plus d'informations sur les groupes d'options, consultez Utilisation de groupes d'options.

- Lorsque vous créez un réplica SQL Server entre régions, Amazon RDS crée un groupe d'options qui lui est dédié.

  Vous ne pouvez pas supprimer un réplica SQL Server entre régions du groupe d'options qui lui est dédié. Aucune autre instance de base de données ne peut utiliser le groupe d'options dédié à un réplica SQL Server entre régions.

  Les options suivantes sont des options répliquées. Pour ajouter des options répliquées à un réplica SQL Server entre régions, ajoutez-le au groupe d'options de l'instance de base de données source. L'option est également installée sur tous les réplicas de l'instance de base de données source.

  - `TDE`

  Les options suivantes sont des options non répliquées. Vous pouvez ajouter ou supprimer des options non répliquées dans un groupe d'options dédié.

  - `MSDTC`
  - `SQLSERVER_AUDIT`
  - Pour activer l'option `SQLSERVER_AUDIT` sur le réplica en lecture entre régions, ajoutez l'option `SQLSERVER_AUDIT` sur le groupe d'options dédié sur le réplica en lecture entre régions et dans le groupe d'options de l'instance source. En ajoutant l'option `SQLSERVER_AUDIT` sur l'instance source du réplica en lecture SQL Server entre régions, vous pouvez créer un objet d'audit au niveau du serveur et des spécifications d'audit au niveau du serveur sur chacun des réplicas en lecture entre régions de l'instance source. Pour autoriser l'accès aux réplicas en lecture entre régions afin de charger les journaux d'audit complets dans un compartiment Amazon S3, ajoutez l'option `SQLSERVER_AUDIT` au groupe d'options dédié et configurez les paramètres des options. Le compartiment Amazon S3 que vous utilisez comme cible pour les fichiers d'audit doit se trouver dans la même région que le réplica en lecture entre régions. Vous pouvez modifier le paramètre de l'option pour chaque réplica en lecture entre régions indépendamment afin que chacun puisse accéder à un compartiment Amazon S3 dans sa région respective.

  Les options suivantes ne sont pas prises en charge pour les répliques en lecture.

  - `SSRS`
  - `SSAS`
  - `SSIS`

  Les options suivantes sont partiellement prises en charge pour les réplicas en lecture entre régions.

- SQLSERVER_BACKUP_RESTORE
- L'instance de base de données source d'un réplica SQL Server entre régions peut avoir l'option SQLSERVER_BACKUP_RESTORE, mais vous ne pouvez pas effectuer de restaurations natives sur l'instance de base de données source tant que vous n'avez pas supprimé tous ses réplicas entre régions. Toutes les tâches de restauration natives existantes seront annulées lors de la création d'un réplica entre régions. Vous ne pouvez pas ajouter l'option SQLSERVER_BACKUP_RESTORE à un groupe d'options dédié.

  Pour plus d'informations sur la sauvegarde et la restauration natives, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Lorsque vous promouvez un réplica en lecture SQL Server entre régions, le réplica promu se comporte de la même façon que d'autres instances de base de données SQL Server, y compris pour la gestion de ses options. Pour plus d'informations sur les groupes d'options, consultez Utilisation de groupes d'options.

Synchronisation des utilisateurs et des objets de base de données avec une réplique en lecture SQL sur serveur

Tous les identifiants, rôles de serveur personnalisés, tâches d'SQLagent ou autres objets de niveau serveur existant dans l'instance de base de données principale au moment de la création d'une réplique de lecture devraient être présents dans la réplique de lecture nouvellement créée. Toutefois, les objets de niveau serveur qui sont créés dans l'instance de base de données principale après la création du réplica en lecture ne sont pas répliqués automatiquement. Vous devez les créer manuellement dans le réplica en lecture.

Les utilisateurs de base de données sont automatiquement répliqués à partir de l'instance de base de données principale dans le réplica en lecture. Comme la base de données répliquée en lecture seule est en mode lecture seule, l'identifiant de sécurité (SID) de l'utilisateur de la base de données ne peut pas être mis à jour dans la base de données. Par conséquent, lors de la création de SQL connexions dans la réplique de lecture, il est essentiel SID de s'assurer que la connexion correspond à celle SID de la SQL connexion correspondante dans l'instance de base de données principale. Si vous ne synchronisez pas SIDs les SQL connexions, elles ne pourront pas accéder à la base de données dans la réplique en lecture. Les connexions authentifiées Windows Active Directory (AD) ne rencontrent pas ce problème car le SQL serveur les obtient SID auprès d'Active Directory.

Pour synchroniser une SQL connexion entre l'instance de base de données principale et la réplique en lecture

1. Connectez-vous à l'instance de base de données principale.
2. Créez une nouvelle SQL connexion dans l'instance de base de données principale.

```
USE [master]
GO
CREATE LOGIN TestLogin1
WITH PASSWORD = 'REPLACE WITH PASSWORD';
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

3. Créez un nouvel utilisateur de base de données pour SQL vous connecter à la base de données.

```
USE [REPLACE WITH YOUR DB NAME]
GO
CREATE USER TestLogin1 FOR LOGIN TestLogin1;
GO
```

4. Vérifiez le nom SID de SQL connexion nouvellement créé dans l'instance de base de données principale.

```
SELECT name, sid FROM sys.server_principals WHERE name =  TestLogin1;
```

5. Connectez-vous au réplica en lecture. Créez le nouveau SQL login.

```
CREATE LOGIN TestLogin1 WITH PASSWORD = 'REPLACE WITH PASSWORD', SID=[REPLACE WITH
 sid FROM STEP #4];
```

Comme alternative, si vous avez accès à la base de données de réplica en lecture, vous pouvez corriger l'utilisateur orphelin comme suit :

1. Connectez-vous au réplica en lecture.
2. Identifiez les utilisateurs orphelins dans la base de données.

```
USE [REPLACE WITH YOUR DB NAME]
GO
EXEC sp_change_users_login 'Report';
GO
```

3. Créez un nouvel SQL identifiant pour l'utilisateur de base de données orphelin.

```
CREATE LOGIN TestLogin1 WITH PASSWORD = 'REPLACE WITH PASSWORD', SID=[REPLACE WITH
 sid FROM STEP #2];
```

Exemple :

```
CREATE LOGIN TestLogin1 WITH PASSWORD = 'TestPa$$word#1',
 SID=[0x1A2B3C4D5E6F7G8H9I0J1K2L3M4N5O6P];
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Résolution d'un problème de réplication de lecture sur un SQL serveur

Vous pouvez surveiller le délai de réplication sur Amazon CloudWatch en consultant la RDS `ReplicaLag` métrique Amazon. Pour de plus amples informations sur la durée du retard de réplication, veuillez consulter Supervision de la réplication en lecture.

Si le retard de réplication est trop long, vous pouvez utiliser la requête suivante pour obtenir des informations sur le retard

```
SELECT AR.replica_server_name
     , DB_NAME (ARS.database_id) 'database_name'
     , AR.availability_mode_desc
     , ARS.synchronization_health_desc
     , ARS.last_hardened_lsn
     , ARS.last_redone_lsn
     , ARS.secondary_lag_seconds
FROM sys.dm_hadr_database_replica_states ARS
INNER JOIN sys.availability_replicas AR ON ARS.replica_id = AR.replica_id
```

```
--WHERE DB_NAME(ARS.database_id) = 'database_name'
ORDER BY AR.replica_server_name;
```

Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL

Les déploiements Multi-AZ améliorent la disponibilité, la durabilité des donnés et la tolérance aux pannes pour les instances de bases de données. En cas de maintenance planifiée de la base de données ou d'interruption de service imprévue, Amazon RDS bascule automatiquement vers l'instance de base de données up-to-date secondaire. Cette fonctionnalité permet aux opérations de la base de données de reprendre rapidement sans intervention manuelle. Les instances principales et de secours utilisent le même point de terminaison, l'adresse réseau physique de celui-ci étant transférée vers le réplica secondaire dans le cadre du processus de basculement. Vous n'avez pas à reconfigurer votre application lorsqu'un basculement se produit.

Amazon RDS prend en charge les déploiements multi-AZ pour Microsoft SQL Server en utilisant soit la mise en miroir des bases de données SQL du serveur (DBM), soit les groupes de disponibilité Always On (). AGs Amazon RDS surveille et assure le bon fonctionnement de votre déploiement multi-AZ. En cas de problème, répare RDS automatiquement les instances de base de données défectueuses, rétablit la synchronisation et initie les basculements. Le basculement n'a lieu que si les instances de secours et principales sont parfaitement synchronisées. Vous ne devez rien gérer.

Lorsque vous configurez SQL Server Multi-AZ, configure RDS automatiquement toutes les bases de données de l'instance à utiliser DBM ou. AGs Amazon RDS gère l'instance de base de données principale, témoin et secondaire pour vous. La configuration étant automatique, elle RDS sélectionne DBM ou Always On AGs en fonction de la version du SQL serveur que vous déployez.

Amazon RDS prend en charge le mode multi-AZ avec Always AGs On pour les versions et éditions de SQL serveur suivantes :

- SQLServeur 2022 :
  - Standard Edition
  - Enterprise Edition
- SQLServeur 2019 :
  - Standard Edition 15.00.4073.23 et versions ultérieures
  - Enterprise Edition
- SQLServeur 2017 :
  - Standard Edition 14.00.3401.7 et versions ultérieures
  - Enterprise Edition 14.00.3049.1 et versions ultérieures

- SQLServer 2016 : Enterprise Edition 13.00.5216.0 et versions ultérieures

Amazon RDS prend en charge le mode multi-AZ DBM pour les versions et éditions de SQL serveur suivantes, à l'exception des versions indiquées précédemment :

- SQLServer 2019 : édition standard 15.00.4043.16
- SQLServer 2017 : éditions Standard et Enterprise
- SQLServer 2016 : éditions Standard et Enterprise

Vous pouvez utiliser la SQL requête suivante pour déterminer si votre instance de base de données de SQL serveur est mono-AZ, multi-AZ avec DBM ou multi-AZ avec Always On. AGs

```
SELECT CASE WHEN dm.mirroring_state_desc IS NOT NULL THEN 'Multi-AZ (Mirroring)'
    WHEN dhdrs.group_database_id IS NOT NULL THEN 'Multi-AZ (AlwaysOn)'
    ELSE 'Single-AZ'
    END 'high_availability'
FROM sys.databases sd
LEFT JOIN sys.database_mirroring dm ON sd.database_id = dm.database_id
LEFT JOIN sys.dm_hadr_database_replica_states dhdrs ON sd.database_id =
 dhdrs.database_id AND dhdrs.is_local = 1
WHERE DB_NAME(sd.database_id) = 'rdsadmin';
```

La sortie est semblable à la suivante :

```
high_availability
Multi-AZ (AlwaysOn)
```

Ajout de Multi-AZ à une instance de base de données Microsoft SQL Server

Lorsque vous créez une nouvelle instance de base de données de SQL serveur à l'aide de AWS Management Console, vous pouvez ajouter Multi-AZ avec Database Mirroring (DBM) ou Always On. AGs Pour ce faire, définissez Déploiement multi-AZ sur Oui (Mise en miroir/Always On). Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

Lorsque vous modifiez une instance de base de données de SQL serveur existante à l'aide de la console, vous pouvez ajouter le mode Multi-AZ avec DBM ou AGs en choisissant Oui (Mirroring/Always On) dans Déploiement multi-AZ sur la page Modifier l'instance de base de données. Pour de

plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

> **Note**
>
> Si votre instance de base de données exécute Database Mirroring (DBM), et non Always On Availability Groups (AGs), vous devrez peut-être désactiver l'optimisation en mémoire avant d'ajouter Multi-AZ. Désactivez l'optimisation en mémoire DBM avant d'ajouter Multi-AZ si votre instance de base de données exécute SQL Server 2016 ou 2017 Enterprise Edition et si l'optimisation en mémoire est activée.
> Si votre instance de base de données est en cours d'exécutionAGs, cette étape n'est pas requise.

Suppression de Multi-AZ d'une instance de base de données Microsoft SQL Server

Lorsque vous modifiez une instance de base de données de SQL serveur existante à l'aide de AWS Management Console, vous pouvez supprimer Multi-AZ avec DBM ouAGs. Vous pouvez le faire en choisissant No (Mirroring / Always On) (Non (Mise en miroir / toujours activée) dans le déploiement Multi-AZ sur la page Modify DB instance (Modifier l'instance de base de données). Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Notes, limitations et recommandations concernant le déploiement multi-AZ de Microsoft SQL Server

Voici quelques limitations lorsque vous travaillez avec des déploiements multi-AZ pour les instances de bases de données RDS for SQL Server :

- Le Multi-AZ entre régions n'est pas pris en charge.
- L'arrêt d'une instance de base de données RDS for SQL Server dans un déploiement multi-AZ n'est pas pris en charge.
- Vous ne pouvez pas configurer l'instance de base de données secondaire pour accepter l'activité de lecture de base de données.
- Le mode Multi-AZ avec Always On Availability Groups (AGs) prend en charge l'optimisation en mémoire.

- Le mode Multi-AZ avec groupes de disponibilité Always On (AGs) ne prend pas en charge l'authentification Kerberos pour l'écouteur des groupes de disponibilité. En effet, l'écouteur ne possède pas de nom de principal de service (SPN).
- Vous ne pouvez pas renommer une base de données sur une instance de base de données SQL Server située dans un déploiement multi-AZ SQL Server. Si vous devez renommer une base de données sur une telle instance, vous devez d'abord désactiver le déploiement multi-AZ pour l'instance de base de données, puis renommer la base de données. Réactivez ensuite le déploiement multi-AZ pour l'instance de base de données.
- Vous pouvez uniquement restaurer des instances de base de données multi-AZ sauvegardées en utilisant le modèle de récupération « Full ».
- Les déploiements multi-AZ sont limités à 10 000 tâches SQL Server Agent.

  Si vous avez besoin d'une limite plus élevée, demandez une augmentation en contactant Support. Ouvrez la page du Centre AWS Support, connectez-vous si nécessaire, puis choisissez Create case (Créer une demande de support). Sélectionnez Service Limit increase (Augmentation des limites de service). Remplissez et envoyez le formulaire.
- Vous ne pouvez pas avoir de base de données hors ligne sur une instance de base de données SQL Server faisant partie d'un déploiement SQL Server Multi-AZ.
- RDS pour SQL Server ne réplique pas les autorisations de base de données MSDB sur l'instance secondaire. Si vous avez besoin de ces autorisations sur l'instance secondaire, vous devez les recréer manuellement.

Voici quelques notes sur l'utilisation des déploiements multi-AZ pour les instances de bases de données RDS for SQL Server :

- Amazon RDS expose le point de terminaison Always On AGs Availability Group Listener. Le point de terminaison est visible dans la console et est renvoyé par l'opération d'API `DescribeDBInstances` en tant qu'entrée dans le champ des points de terminaison.
- Amazon RDS prend en charge les basculements à plusieurs sous-réseaux du groupe de disponibilité.
- Pour utiliser les déploiements multi-AZ SQL Server avec une instance de base de données SQL Server dans un cloud privé virtuel (VPC), créez d'abord un groupe de sous-réseaux de base de données comportant des sous-réseaux dans au moins deux zones de disponibilité distinctes. Affectez ensuite le groupe de sous-réseaux de base de données au réplica principal de l'instance de base de données SQL Server.

- Quand une instance de base de données est transformée en déploiement multi-AZ, pendant la modification, son état est modifying (modification). Amazon RDS crée l'instance de secours et effectue une sauvegarde de l'instance de base de données principale. Une fois le processus terminé, le statut de l'instance de base de données principale devient disponible.
- Les déploiements Multi-AZ maintiennent toutes les bases de données sur le même nœud. Si une base de données sur l'hôte principal bascule, toutes vos bases de données SQL Server basculent en tant qu'unité atomique vers votre hôte de secours. Amazon RDS approvisionne un nouvel hôte sain et remplace l'hôte non sain.
- Multi-AZ avec DBM ou AGs prend en charge une seule réplique de secours.
- Les utilisateurs, les ID de connexion et les autorisations sont automatiquement répliqués pour vous sur l'instance secondaire. Vous n'avez pas besoin de les recréer. Les rôles de serveur définis par l'utilisateur ne sont répliqués que dans les instances de base de données qui utilisent Always On AGs pour les déploiements multi-AZ.
- Dans les déploiements multi-AZ, RDS pour SQL Server crée des connexions SQL Server pour autoriser Always On ou la mise en miroir de bases de données. AGs RDS crée des connexions selon le modèle suivant, `db_<dbiResourceId>_node1_logindb_<dbiResourceId>_node2_login`, et. `db_<dbiResourceId>_witness_login`
- RDS for SQL Server crée un identifiant SQL Server pour autoriser l'accès aux répliques en lecture. RDS crée une connexion avec le modèle suivant,`db_<readreplica_dbiResourceId>_node_login`.
- Dans les déploiements Multi-AZ, les tâches de l'agent SQL Server sont répliquées de l'hôte principal vers l'hôte secondaire lorsque la fonction de réplication des tâches est activée. Pour de plus amples informations, veuillez consulter Activation de la réplication des tâches de l'agent SQL Server.
- Il est possible que vous observiez des temps de latence élevés par rapport à un déploiement d'instance de base de données standard (dans une zone de disponibilité unique) en raison de la réplication synchrone des données.
- Les délais de basculement sont affectés par le temps nécessaire à la réalisation du processus de récupération. Le délai de basculement est allongé pour les transactions de volume important.
- Dans les déploiements multi-AZ SQL Server, le redémarrage avec basculement redémarre uniquement l'instance de base de données principale. Après le basculement, l'instance de base de données principale devient la nouvelle instance de base de données secondaire. Les paramètres peuvent ne pas être mis à jour pour les instances multi-AZ. Pour le redémarrage sans basculement, les instances de base de données principale et secondaire redémarrent, et les

paramètres sont mis à jour après le redémarrage. Si l'instance de base de données ne répond pas, nous vous recommandons de procéder à un redémarrage sans basculement.

Voici quelques recommandations sur l'utilisation des déploiements multi-AZ pour les instances de bases de données RDS for Microsoft SQL Server :

- Pour les bases de données utilisées en production ou préproduction, nous vous recommandons les options suivantes :
  - Déploiements multi-AZ pour une haute disponibilité
  - IOPS provisionnés pour des performances rapides et cohérentes
  - « Mémoire optimisée » plutôt que « Usage général »
- Vous ne pouvez pas sélectionner la zone de disponibilité (AZ) pour l'instance secondaire. Tenez-en compte lorsque vous déployez les hôtes d'application. Votre base de données peut basculer vers une autre zone de disponibilité et les hôtes d'application peuvent ne pas se trouver dans la même zone de disponibilité que la base de données. Pour cette raison, nous vous recommandons d'équilibrer les hôtes de vos applications AZs dans l'ensemble de la AWS région donnée.
- Pour de meilleures performances, n'activez pas la mise en miroir de base de données ou Always On AGs lors d'une opération de chargement de données volumineuse. Pour optimiser la vitesse de chargement de vos données, terminez le chargement des données avant de convertir votre instance de base de données en déploiement multi-AZ.
- Les applications qui ont accès aux bases de données SQL Server doit disposer d'une gestion des exceptions qui identifie les erreurs de connexion. L'exemple de code suivant montre un try/catch bloc qui détecte une erreur de communication. Dans cet exemple, l'instruction `break` quitte la boucle `while` si la connexion réussit, mais relance jusqu'à 10 tentatives si une exception est déclenchée.

```
int RetryMaxAttempts = 10;
int RetryIntervalPeriodInSeconds = 1;
int iRetryCount = 0;
while (iRetryCount < RetryMaxAttempts)
{
   using (SqlConnection connection = new SqlConnection(DatabaseConnString))
   {
      using (SqlCommand command = connection.CreateCommand())
      {
         command.CommandText = "INSERT INTO SOME_TABLE VALUES ('SomeValue');";
         try
```

```
         {
            connection.Open();
            command.ExecuteNonQuery();
            break;
         }
         catch (Exception ex)
         {
            Logger(ex.Message);
            iRetryCount++;
         }
         finally {
            connection.Close();
         }
      }
   }
   Thread.Sleep(RetryIntervalPeriodInSeconds * 1000);
}
```

- N'utilisez pas la commande `Set Partner Off` lorsque vous travaillez avec des instances multi-AZ. Par exemple, n'effectuez pas les opérations suivantes :

```
--Don't do this
ALTER DATABASE db1 SET PARTNER off
```

- Ne définissez pas le mode de récupération sur `simple`. Par exemple, n'effectuez pas les opérations suivantes :

```
--Don't do this
ALTER DATABASE db1 SET RECOVERY simple
```

- N'utilisez pas le paramètre `DEFAULT_DATABASE` lors de la création de nouveaux ID de connexion sur des instances de bases de données multi-AZ car ces paramètres ne peuvent pas être appliqués au miroir de secours. Par exemple, n'effectuez pas les opérations suivantes :

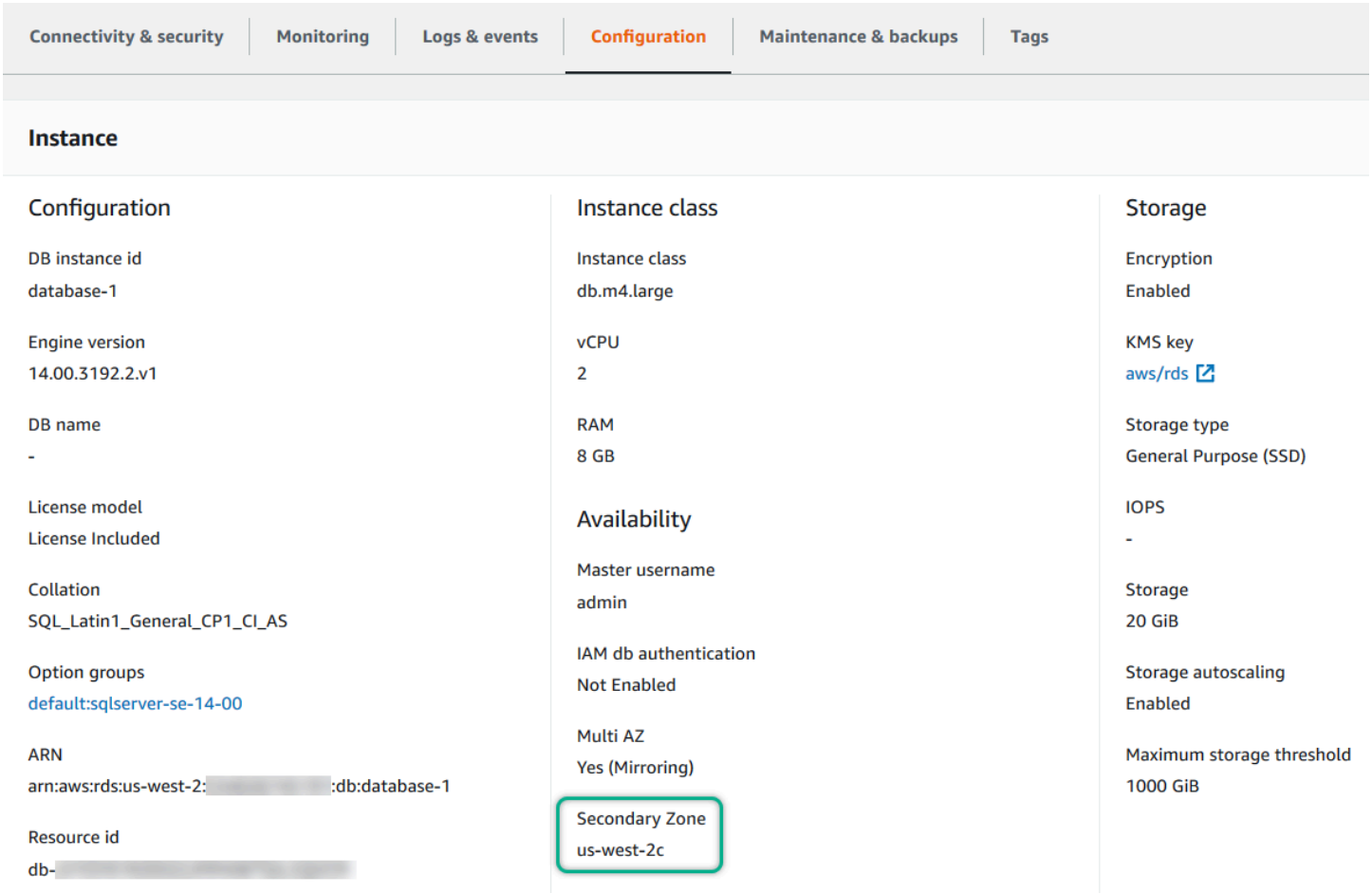
```
--Don't do this
CREATE LOGIN [test_dba] WITH PASSWORD=foo, DEFAULT_DATABASE=[db2]
```

  En outre, n'effectuez pas les opérations suivantes :

```
--Don't do this
ALTER LOGIN [test_dba] WITH DEFAULT_DATABASE=[db3]
```

Détermination de l'emplacement du réplica secondaire

Vous pouvez déterminer l'emplacement du réplica secondaire à l'aide d AWS Management Console. Vous devez connaître l'emplacement de l'instance secondaire si vous configurez votre instance de base de données principale dans unVPC.



Vous pouvez également afficher la zone de disponibilité du secondaire à l'aide de la AWS CLI commande `describe-db-instances` ou de RDS API l'opération`DescribeDBInstances`. Le résultat indique la zone de disponibilité secondaire où se situe le miroir de secours.

Migration de la mise en miroir de bases de données (DBM) vers les groupes de disponibilité AlwaysOn

Dans la version 14.00.3049.1 de SQL Microsoft Server Enterprise Edition, les groupes de disponibilité Always On AGs () sont activés par défaut.

Pour migrer de Database Mirroring (DBM) versAGs, vérifiez d'abord votre version. Lorsque vous utilisez une instance de base de données de version antérieure à 13.00.5216.0, modifiez l'instance

pour la faire passer à la version 13.00.5216.0 ou une version ultérieure. Lorsque vous utilisez une instance de base de données de version antérieure à 14.00.3049.1, modifiez-la pour la faire passer à la version 14.00.3049.1 ou une version ultérieure.

Si vous souhaitez mettre à niveau une instance de base de données en miroir à utiliserAGs, exécutez d'abord la mise à niveau, modifiez l'instance pour supprimer Multi-AZ, puis modifiez-la à nouveau pour ajouter Multi-AZ. Cela convertit votre instance pour qu'elle utilise Always OnAGs.

Fonctionnalités supplémentaires pour Microsoft SQL Server sur Amazon RDS

Dans les sections suivantes, vous trouverez des informations sur l'augmentation des RDS instances Amazon exécutant le moteur de base de données Microsoft SQL Server.

Rubriques

- Utilisation de la politique de mot de passe pour les connexions à SQL Server sur RDS pour SQL Server
- Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3
- Utilisation de Database Mail sur Amazon RDS for SQL Server
- Prise en charge du stockage d'instance pour la base de données tempdb sur Amazon RDS for SQL Server
- Utilisation d'événements étendus avec Amazon RDS pour Microsoft SQL Server
- Accès aux sauvegardes des journaux de transactions avec RDS for SQL Server

Utilisation de la politique de mot de passe pour les connexions à SQL Server sur RDS pour SQL Server

Amazon RDS vous permet de définir la politique de mot de passe pour votre instance de base de données Amazon RDS exécutant Microsoft SQL Server. Utilisez-le pour définir les exigences de complexité, de longueur et de verrouillage pour les connexions qui utilisent l'authentification SQL Server pour s'authentifier auprès de votre instance de base de données.

Termes clés

Connexion

Dans SQL Server, un principal au niveau du serveur capable de s'authentifier auprès d'une instance de base de données est appelé connexion. D'autres moteurs de base de données peuvent désigner ce principal en tant qu'utilisateur. Dans RDS pour SQL Server, un identifiant peut s'authentifier à l'aide de l'authentification SQL Server ou de l'authentification Windows.

Connexion à SQL Server

Un identifiant qui utilise un nom d'utilisateur et un mot de passe pour s'authentifier à l'aide de l'authentification SQL Server est un identifiant SQL Server. La politique de mot de passe que vous configurez via les paramètres de base de données s'applique uniquement aux connexions SQL Server.

Connexion à Windows

Un identifiant basé sur un principal Windows et authentifié à l'aide de l'authentification Windows est un identifiant Windows. Vous pouvez configurer la politique de mot de passe pour vos connexions Windows dans Active Directory. Pour plus d'informations, consultez Utilisation d'Active Directory avec RDS for SQL Server.

Politique d'activation et de désactivation pour chaque connexion

Chaque connexion à SQL Server comporte des indicateurs pour `CHECK_POLICY` et`CHECK_EXPIRATION`. Par défaut, les nouvelles connexions sont créées avec `CHECK_POLICY` set to `ON` et `CHECK_EXPIRATION` set to`OFF`.

S'il `CHECK_POLICY` est activé pour une connexion, RDS pour SQL Server valide le mot de passe en fonction des exigences de complexité et de longueur minimale. Les politiques de

verrouillage s'appliquent également. Exemple d'instruction T-SQL pour activer `CHECK_POLICY` et `CHECK_EXPIRATION` :

```
ALTER LOGIN [master_user] WITH CHECK_POLICY = ON, CHECK_EXPIRATION = ON;
```

Si cette option `CHECK_EXPIRATION` est activée, les mots de passe sont soumis aux politiques relatives à l'âge des mots de passe. L'instruction T-SQL pour vérifier si `CHECK_POLICY` et `CHECK_EXPIRATION` sont définis :

```
SELECT name, is_policy_checked, is_expiration_checked FROM sys.sql_logins;
```

Paramètres de politique de mot de passe

Tous les paramètres de politique de mot de passe sont dynamiques et ne nécessitent pas de redémarrage de la base de données pour être pris en compte. Le tableau suivant répertorie les paramètres de base de données que vous pouvez définir pour modifier la politique de mot de passe pour les connexions à SQL Server :

Paramètre de base de données	Description	Valeurs autorisées	Valeur par défaut	
rds.password_complexity_enabled	Les exigences de complexité des mots de passe doivent être satisfaites lors de la création ou de la modification de mots de passe pour les connexions à SQL Server. Les contraintes suivantes doivent être respectées :	0,1	0	

Paramètre de base de données	Description	Valeurs autorisées	Valeur par défaut	

	• Le mot de passe doit inclure des caractères appartenant à trois des catégories suivantes :
	  • lettre minuscule latine (a à z)
	  • Lettre majuscule latine (A à Z)
	  • Caractères non alphanumériques tels que : point d'exclamation (!) , signe dollar ($), signe numérique (#) ou pourcentage (%).
	• Le mot de passe ne contient pas le nom de			

Paramètre de base de données	Description	Valeurs autorisées	Valeur par défaut	
	compte de l'utilisateur.			
rds.password_min_length	Le nombre minimum de caractères requis dans un mot de passe pour une connexion à SQL Server.	0-14	0	
rds.password_min_age	Nombre minimum de jours pendant lesquels un mot de passe de connexion à SQL Server doit être utilisé avant que l'utilisateur puisse le modifier. Les mots de passe peuvent être modifiés immédiatement lorsqu'ils sont définis sur 0.	0-998	0	

Paramètre de base de données	Description	Valeurs autorisées	Valeur par défaut	
rds.password_max_age	Nombre maximal de jours pendant lesquels un mot de passe de connexion à SQL Server peut être utilisé, après lequel l'utilisateur doit le modifier. Les mots de passe n'expirent jamais lorsqu'ils sont définis sur 0.	0-999	42	
rds.password_lockout_threshold	Nombre de tentatives de connexion infructueuses consécutives qui ont entraîné le verrouillage d'une connexion à SQL Server.	0-999	0	
rds.password_lockout_duration	Le nombre de minutes pendant lesquelles une connexion SQL Server verrouillée doit attendre avant d'être déverrouillée.	1 à 60	10	

Paramètre de base de données	Description	Valeurs autorisées	Valeur par défaut	
rds.password_lockout_reset_counter_after	Le nombre de minutes qui doivent s'écouler après l'échec d'une tentative de connexion avant que le compteur de tentatives de connexion échouées soit remis à 0.	1 à 60	10	

> **Note**
>
> Pour plus d'informations sur la politique de mot de passe de SQL Server, consultez la section Stratégie de mot de passe.
> Les politiques relatives à la complexité et à la longueur minimale des mots de passe s'appliquent également aux utilisateurs de bases de données contenues dans les bases de données. Pour plus d'informations, consultez la section Bases de données contenues.

Les contraintes suivantes s'appliquent aux paramètres de la politique de mot de passe :

- Le `rds.password_min_age` paramètre doit être inférieur à`rds.password_max_age parameter`, sauf s'`rds.password_max_age`il est défini sur 0
- Le `rds.password_lockout_reset_counter_after` paramètre doit être inférieur ou égal au `rds.password_lockout_duration` paramètre.
- `rds.password_lockout_threshold`Il est défini sur 0 `rds.password_lockout_duration` et `rds.password_lockout_reset_counter_after` ne s'applique pas.

Considérations relatives aux connexions existantes

Une fois la politique de mot de passe modifiée sur une instance, les mots de passe existants pour les connexions ne sont pas évalués rétroactivement par rapport aux nouvelles exigences en matière de complexité et de longueur des mots de passe. Seuls les nouveaux mots de passe sont validés par rapport à la nouvelle politique.

SQL Server évalue les mots de passe existants en fonction de l'âge requis.

Il est possible que les mots de passe expirent immédiatement une fois qu'une politique de mots de passe est modifiée. Par exemple, si un identifiant est `CHECK_EXPIRATION` activé et que son mot de passe a été modifié pour la dernière fois il y a 100 jours et que vous définissez le `rds.password_max_age` paramètre sur 5 jours, le mot de passe expire immédiatement et l'identifiant doit changer son mot de passe lors de sa prochaine tentative de connexion.

> **Note**
>
> RDS pour SQL Server ne prend pas en charge les politiques relatives à l'historique des mots de passe. Les politiques d'historique empêchent les connexions de réutiliser les mots de passe précédemment utilisés.

Considérations sur les déploiements multi-AZ

Le compteur de tentatives de connexion échouées et l'état de verrouillage des instances multi-AZ ne se répliquent pas entre les nœuds. En cas de verrouillage d'une connexion lors du basculement d'une instance multi-AZ, il est possible que la connexion soit déjà déverrouillée sur le nouveau nœud.

Considérations relatives au mot de passe pour la connexion principale

Lorsque vous créez une instance de base de données RDS pour SQL Server, le mot de passe de l'utilisateur principal n'est pas évalué par rapport à la politique de mot de passe. Un nouveau mot de passe principal n'est pas non plus évalué par rapport au mot de passe lorsque vous effectuez des opérations sur l'utilisateur principal, `MasterUserPassword` en particulier lors de la configuration de la `ModifyDBInstance` commande. Dans les deux cas, vous pouvez définir un mot de passe pour l'utilisateur principal qui ne correspond pas à votre politique en matière de mots de passe, et l'opération réussit tout de même. Si la politique n'est pas respectée, RDS tente de déclencher un événement RDS, en recommandant de définir un mot de passe fort. Veillez à n'utiliser que des mots de passe forts pour l'utilisateur principal.

RDS tente de générer les messages d'événement suivants lorsque le mot de passe de l'utilisateur principal ne répond pas aux exigences de la politique de mot de passe :

- L'utilisateur principal a été créé, mais le mot de passe ne répond pas à la longueur minimale requise par votre politique en matière de mots de passe. Envisagez d'utiliser un mot de passe plus fort.
- L'utilisateur principal a été créé, mais le mot de passe ne répond pas aux exigences de complexité de votre politique de mots de passe. Envisagez d'utiliser un mot de passe plus fort.
- Le mot de passe de l'utilisateur principal a été réinitialisé, mais il ne répond pas à la longueur minimale requise par votre politique en matière de mots de passe. Envisagez d'utiliser un mot de passe plus fort.
- Le mot de passe de l'utilisateur principal a été réinitialisé, mais il ne répond pas aux exigences de complexité de votre politique en matière de mots de passe. Envisagez d'utiliser un mot de passe plus fort.

Par défaut, l'utilisateur principal est créé avec `CHECK_POLICY` et `CHECK_EXPIRATION` défini sur`OFF`. Pour appliquer la politique de mot de passe à l'utilisateur principal, vous devez activer manuellement ces indicateurs pour l'utilisateur principal après la création de l'instance de base de données. Après avoir activé ces indicateurs, modifiez le mot de passe de l'utilisateur principal directement dans SQL Server (par exemple via des instructions T-SQL ou SSMS) pour valider le nouveau mot de passe par rapport à la politique de mot de passe.

> **Note**
>
> Si l'utilisateur principal est verrouillé, vous pouvez le déverrouiller en réinitialisant le mot de passe de l'utilisateur principal à l'aide de la `ModifyDBInstance` commande.

Modification du mot de passe utilisateur principal

Vous pouvez modifier le mot de passe de l'utilisateur principal à l'aide de la commande ModifyDBInstance.

> **Note**
>
> Lorsque vous réinitialisez le mot de passe de l'utilisateur principal, RDS réinitialise diverses autorisations pour l'utilisateur principal et celui-ci peut perdre certaines autorisations. La

réinitialisation du mot de passe de l'utilisateur principal déverrouille également l'utilisateur principal, s'il a été verrouillé.

RDS valide le nouveau mot de passe de l'utilisateur principal et tente d'émettre un événement RDS si le mot de passe ne satisfait pas à la politique. RDS définit le mot de passe même s'il ne satisfait pas à la politique de mot de passe.

Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3

Vous pouvez transférer des fichiers entre une instance de base de données exécutant Amazon RDS for SQL Server et un compartiment Amazon S3. Ainsi, vous pouvez utiliser Amazon S3 avec des fonctionnalités SQL, telles que BULK INSERT. Par exemple, vous pouvez télécharger des fichiers .csv, .xml, .txt, etc. depuis Amazon S3 vers l'hôte de l'instance de base de données, puis importer les données depuis `D:\S3\` dans la base de données. Tous les fichiers sont stockés dans `D:\S3\` sur l'instance de base de données.

Les limites suivantes s'appliquent :

> **Note**
>
> Le trafic entre l'hôte RDS et S3 est acheminé via les points de terminaison VPC dans RDS en VPCs interne pour toutes les fonctionnalités de SQL Server qui utilisent S3. Ce trafic n'utilise pas le point de terminaison de l'instance RDS ENI. Les politiques relatives aux compartiments S3 ne peuvent pas restreindre le trafic RDS en fonction des conditions du réseau.

- Les fichiers du dossier `D:\S3` sont supprimés du réplica de secours après un basculement sur des instances Multi-AZ. Pour de plus amples informations, veuillez consulter Limitations Multi-AZ pour l'intégration S3.
- L'instance de base de données et le compartiment S3 doivent se trouver dans la même AWS région.
- Si vous exécutez plusieurs tâches d'intégration S3 simultanément, les tâches s'exécutent de manière séquentielle et non en parallèle.

> **Note**
>
> Les tâches d'intégration S3 partagent la même file d'attente que les tâches de sauvegarde et de restauration natives. Vous pouvez uniquement disposer de deux tâches maximum en cours à tout moment dans cette file d'attente. Par conséquent, deux tâches de sauvegarde et de restauration natives en cours d'exécution bloquent toutes les tâches d'intégration S3.

- Vous devez réactiver la fonctionnalité d'intégration S3 sur les instances restaurées. Elle ne se propage pas depuis l'instance source vers l'instance restaurée. Sur une instance restaurée, les fichiers situés sous `D:\S3` sont supprimés.
- Le téléchargement vers l'instance de base de données est limité à 100 fichiers. En d'autres termes, il ne peut pas y avoir plus de 100 fichiers dans `D:\S3\`.
- Seuls les fichiers sans extension ou qui possèdent une des extensions suivantes sont pris en charge pour le téléchargement : .abf, .asdatabase, .bcp, .configsettings, .csv, .dat, .deploymentoptions, .deploymenttarget et .xmla.
- Le compartiment S3 doit avoir le même propriétaire que le rôle AWS Identity and Access Management (IAM) associé. Par conséquent, l'intégration S3 entre comptes n'est pas prise en charge.
- Le compartiment S3 ne peut pas être ouvert au public.
- La taille des chargements de RDS vers S3 est limitée à 50 Go par fichier.
- La taille des fichiers téléchargés de S3 vers RDS est limitée au maximum pris en charge par S3.

Rubriques

- Prérequis pour l'intégration de RDS for SQL Server avec S3
- Activation RDS de l'intégration SQL du serveur avec S3
- Transfert de fichiers entre RDS for SQL Server et Amazon S3
- Lister les fichiers sur l'RDSinstance de base de données
- Supprimer des fichiers sur l'RDSinstance de base de données
- Surveillance du statut d'une tâche de transfert de fichiers
- Annulation d'une tâche
- Limitations Multi-AZ pour l'intégration S3
- Désactivation RDS pour l'intégration SQL du serveur à S3

Pour plus d'informations sur l'utilisation des fichiers dans Amazon S3, veuillez consulter Démarrer avec Amazon Simple Storage Service.

Prérequis pour l'intégration de RDS for SQL Server avec S3

Avant de commencer, recherchez ou créez le compartiment S3 que vous souhaitez utiliser. De plus, ajoutez des autorisations afin que l'instance de base de données RDS puisse accéder au compartiment S3. Pour configurer cet accès, vous créez une stratégie et un rôle IAM.

console

Pour créer une stratégie IAM afin d'accéder à Amazon S3

1. Dans la console IAM Management Console, choisissez Stratégie dans le panneau de navigation.
2. Créez une nouvelle stratégie et utilisez l'onglet Éditeur visuel pour exécuter les étapes suivantes.
3. Pour Service, saisissez **S3** et choisissez le service S3.
4. Pour Actions, choisissez les actions suivantes pour accorder l'accès nécessaire à votre instance de base de données :

   - `ListAllMyBuckets` – Obligatoire
   - `ListBucket` – Obligatoire
   - `GetBucketAcl` – Obligatoire
   - `GetBucketLocation` – Obligatoire
   - `GetObject` – Obligatoire pour télécharger des fichiers depuis S3 vers `D:\S3\`
   - `PutObject` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3
   - `ListMultipartUploadParts` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3
   - `AbortMultipartUpload` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3
5. Pour Ressources, les options qui s'affichent dépendent des actions choisies dans l'étape précédente. Vous pourrez voir des options pour compartiment, objet ou les deux. Pour chacune d'entre elles, ajoutez l'Amazon Resource Name (ARN) approprié.

   Pour compartiment, ajoutez l'ARN du compartiment que vous souhaitez utiliser. Par exemple, si votre compartiment est nommé *`amzn-s3-demo-bucket`*, définissez l'ARN sur `arn:aws:s3:::`*`amzn-s3-demo-bucket`*.

   Pour objet, saisissez l'ARN pour le compartiment, puis choisissez l'une des options suivantes :

   - Pour accorder l'accès à tous les fichiers d'un compartiment spécifié, choisissez Tous pour le Bucket name (Nom du compartiment) et le Object name (Nom de l'objet).

- Pour accorder l'accès à des fichiers ou à des dossiers spécifiques du bucket, ARNs indiquez les buckets et les objets spécifiques auxquels vous souhaitez que SQL Server accède.

6. Suivez les instructions dans la console jusqu'à la création de la stratégie.

   Ce qui précède est un guide abrégé pour configurer une stratégie. Pour obtenir des instructions détaillées sur la création de stratégies IAM, veuillez consulter Création de stratégies IAM dans le Guide de l'utilisateur IAM.

Pour créer un rôle IAM utilisant la stratégie IAM de la procédure précédente.

1. Dans la console IAM Management Console, choisissez Rôles dans le panneau de navigation.
2. Créez un rôle IAM et choisissez les options suivantes à mesure qu'elles s'affichent dans la console :

   - AWS web
   - RDS
   - RDS – Ajoutez un rôle à la base de données

   Ensuite, choisissez Suivant : Autorisations en bas.
3. Pour Attach permissions policies (Attacher des stratégies d'autorisations), saisissez le nom de la stratégie IAM précédemment créée. Ensuite, choisissez la stratégie dans la liste.
4. Suivez les instructions dans la console jusqu'à la création du rôle.

   Ce qui précède est un guide abrégé pour configurer un rôle. Pour obtenir des instructions plus détaillées sur la création des rôles, veuillez consulter Rôles IAM dans le Guide de l'utilisateur IAM.

AWS CLI

Pour accorder à Amazon RDS l'accès à un compartiment Amazon S3, procédez comme suit :

1. Créez une stratégie IAM qui accorde à Amazon RDS l'accès à un compartiment S3.
2. Créez un rôle IAM qu'Amazon RDS peut endosser en votre nom pour accéder à vos compartiments S3.

Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

3. Attachez la politique IAM que vous avez créée au rôle IAM que vous venez de créer.

Pour créer la stratégie IAM

Ajoutez les actions appropriées pour accorder l'accès nécessaire à votre instance de base de données :

- `ListAllMyBuckets` – Obligatoire
- `ListBucket` – Obligatoire
- `GetBucketAcl` – Obligatoire
- `GetBucketLocation` – Obligatoire
- `GetObject` – Obligatoire pour télécharger des fichiers depuis S3 vers `D:\S3\`
- `PutObject` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3
- `ListMultipartUploadParts` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3
- `AbortMultipartUpload` – Obligatoire pour charger des fichiers depuis `D:\S3\` vers S3

1. La AWS CLI commande suivante crée une politique IAM nommée `rds-s3-integration-policy` avec ces options. Elle accorde un accès à un compartiment nommé *`amzn-s3-demo-bucket`*.

   Example

   Pour LinuxmacOS, ou Unix :

```
aws iam create-policy \
  --policy-name rds-s3-integration-policy \
  --policy-document '{
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Action": "s3:ListAllMyBuckets",
                "Resource": "*"
            },
            {
```

```
                "Effect": "Allow",
                "Action": [
                    "s3:ListBucket",
                    "s3:GetBucketAcl",
                    "s3:GetBucketLocation"
                ],
                "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
            },
            {
                "Effect": "Allow",
                "Action": [
                    "s3:GetObject",
                    "s3:PutObject",
                    "s3:ListMultipartUploadParts",
                    "s3:AbortMultipartUpload"
                ],
                "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/key_prefix/*"
            }
        ]
    }'
```

Dans Windows :

Veillez à remplacer les fins de ligne par celles prises en charge par votre interface (^ au lieu de \). De plus, dans Windows, vous devez utiliser une séquence d'échappement sur tous les guillemets doubles avec un \. Pour éviter d'utiliser une séquence d'échappement sur tous les guillemets dans le JSON, vous pouvez l'enregistrer dans un fichier et le transmettre en tant que paramètre.

Tout d'abord, créez le fichier `policy.json` avec la stratégie d'autorisation suivante :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": "*"
        },
```

```
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketACL",
                "s3:GetBucketLocation"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:GetObject",
                "s3:PutObject",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/key_prefix/*"
        }
    ]
}
```

Ensuite, utilisez la commande suivante pour créer la stratégie :

```
aws iam create-policy ^
     --policy-name rds-s3-integration-policy ^
     --policy-document file://file_path/assume_role_policy.json
```

2. Après avoir créé la stratégie, notez son ARN (Amazon Resource Name). Vous aurez besoin de l'ARN lors d'une étape ultérieure.

Pour créer le rôle IAM

- La AWS CLI commande suivante crée le rôle `rds-s3-integration-role` IAM à cette fin.

  Example

  Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
    --role-name rds-s3-integration-role \
```

```
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
             "Service": "rds.amazonaws.com"
           },
          "Action": "sts:AssumeRole"
        }
      ]
    }'
```

Dans Windows :

Veillez à remplacer les fins de ligne par celles prises en charge par votre interface (^ au lieu de \). De plus, dans Windows, vous devez utiliser une séquence d'échappement sur tous les guillemets doubles avec un \. Pour éviter d'utiliser une séquence d'échappement sur tous les guillemets dans le JSON, vous pouvez l'enregistrer dans un fichier et le transmettre en tant que paramètre.

Tout d'abord, créez le fichier `assume_role_policy.json` avec la stratégie suivante :

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "rds.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

Ensuite, utilisez la commande suivante pour créer le rôle IAM :

```
aws iam create-role ^
     --role-name rds-s3-integration-role ^
     --assume-role-policy-document file://file_path/assume_role_policy.json
```

Example d'utiliser la clé de contexte de condition globale pour créer le rôle IAM

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des politiques basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction de politique.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la politique, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'Amazon Resource Name (ARN) complet des ressources qui accèdent au rôle. Pour l'intégration S3, assurez-vous d'inclure l'instance de base de données ARNs, comme indiqué dans l'exemple suivant.

Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
    --role-name rds-s3-integration-role \
    --assume-role-policy-document '{
      "Version": "2012-10-17",
      "Statement": [
        {
          "Effect": "Allow",
          "Principal": {
             "Service": "rds.amazonaws.com"
```

```
          },
         "Action": "sts:AssumeRole",
               "Condition": {
                   "StringEquals": {

"aws:SourceArn":"arn:aws:rds:Region:my_account_ID:db:db_instance_identifier"
                   }
               }
       }
     ]
   }'
```

Dans Windows :

Ajoutez la clé de contexte de condition globale à `assume_role_policy.json`.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": [
                    "rds.amazonaws.com"
                ]
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {

 "aws:SourceArn":"arn:aws:rds:Region:my_account_ID:db:db_instance_identifier"
                }
            }
        }
    ]
}
```

Pour attacher la politique IAM à un rôle IAM

- La AWS CLI commande suivante associe la politique au rôle nommérds-s3-integration-role. Remplacez *your-policy-arn* par l'ARN de stratégie que vous avez noté lors d'une étape précédente.

  Example

  Pour LinuxmacOS, ou Unix :

  ```
  aws iam attach-role-policy \
      --policy-arn your-policy-arn \
      --role-name rds-s3-integration-role
  ```

  Dans Windows :

  ```
  aws iam attach-role-policy ^
      --policy-arn your-policy-arn ^
      --role-name rds-s3-integration-role
  ```

Activation RDS de l'intégration SQL du serveur avec S3

Dans la section suivante, vous trouverez comment activer l'intégration d'Amazon S3 avec Amazon RDS for SQL Server. Pour fonctionner avec l'intégration S3, votre instance de base de données doit être associée au IAM rôle que vous avez créé précédemment avant d'utiliser le paramètre S3_INTEGRATION feature-name.

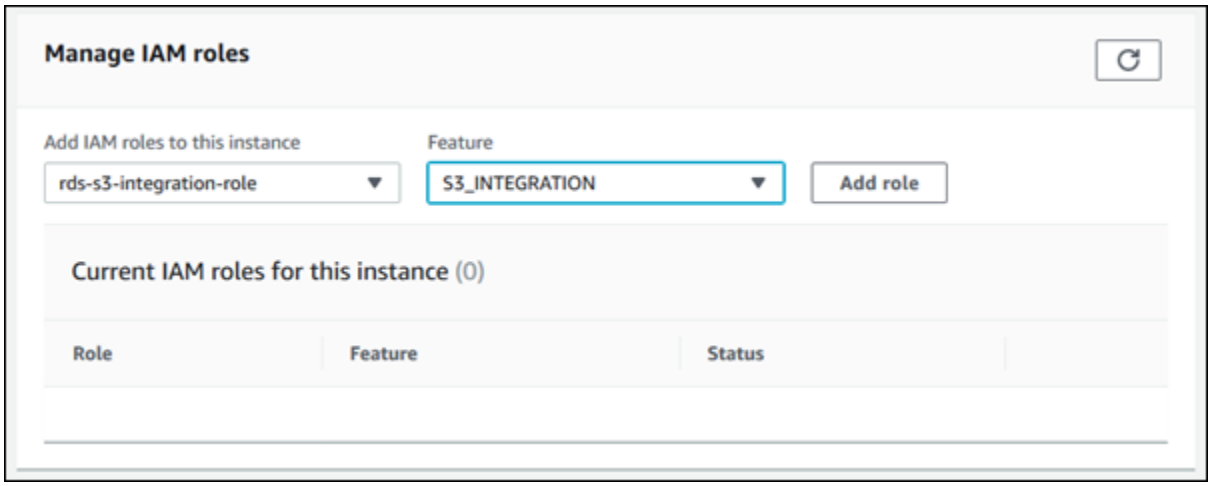
> **Note**
>
> Pour ajouter un IAM rôle à une instance de base de données, le statut de l'instance de base de données doit être disponible.

Console

Pour associer votre IAM rôle à votre instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Choisissez le nom RDS de l'instance de base de données for SQL Server pour afficher ses détails.
3. Dans l'onglet Connectivité et sécurité, dans la section Gérer les IAM rôles, choisissez le IAM rôle à ajouter pour Ajouter IAM des rôles à cette instance.
4. Dans Feature, choisissez S3_ INTEGRATION.



5. Choisissez Add role (Ajouter un rôle).

AWS CLI

Pour ajouter le IAM rôle à l'instance de base de données RDS for SQL Server

- La AWS CLI commande suivante ajoute votre IAM rôle à une instance de base de données RDS for SQL Server nommée*mydbinstance*.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds add-role-to-db-instance \
    --db-instance-identifier mydbinstance \
    --feature-name S3_INTEGRATION \
    --role-arn your-role-arn
```

  Dans Windows:

```
aws rds add-role-to-db-instance ^
    --db-instance-identifier mydbinstance ^
```

```
    --feature-name S3_INTEGRATION ^
    --role-arn your-role-arn
```

*your-role-arn*Remplacez-le par le rôle ARN que vous avez noté à l'étape précédente. `S3_INTEGRATION`doit être spécifié pour l'`--feature-name`option.

Transfert de fichiers entre RDS for SQL Server et Amazon S3

Vous pouvez utiliser les procédures RDS stockées Amazon pour télécharger et charger des fichiers entre Amazon S3 et votre RDS instance de base de données. Vous pouvez également utiliser les procédures RDS stockées d'Amazon pour répertorier et supprimer des fichiers sur l'RDSinstance.

Les fichiers que vous téléchargez depuis/chargez vers S3 sont stockés dans le dossier `D:\S3`. Il s'agit du seul dossier que vous pouvez utiliser pour accéder à vos fichiers. Vous pouvez organiser vos fichiers en sous-dossiers, qui sont créés pour vous lorsque vous incluez le dossier de destination lors du téléchargement.

Certaines procédures stockées nécessitent que vous fournissiez un Amazon Resource Name (ARN) à votre compartiment et à votre fichier S3. Le format pour votre ARN est`arn:aws:s3:::`*amzn-s3-demo-bucket*`/file_name`. Amazon S3 n'a pas besoin de numéro de compte ni de AWS région dansARNs.

Les tâches d'intégration S3 s'exécutent de manière séquentielle, partagent la même file d'attente que la sauvegarde native et restaurent des tâches. Vous pouvez uniquement disposer de deux tâches maximum en cours à tout moment dans cette file d'attente. Le traitement de la tâche peut mettre jusqu'à cinq minutes avant de commencer.

Téléchargement de fichiers depuis un compartiment Amazon S3 vers une instance de base de données de SQL serveur

Pour télécharger des fichiers depuis un compartiment S3 vers une instance de base de données RDS for SQL Server, utilisez la procédure RDS stockée Amazon `msdb.dbo.rds_download_from_s3` avec les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`@s3_arn_of_file`	NVARCHAR	–	Obligatoire	Le S3 ARN du fichier à télécharger, par

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				exemple : `arn:aws:s3::: amzn-s3-demo-bucket /mydata.csv`
`@rds_file_path`	NVARCHAR	–	Facultatif	Le chemin du fichier pour l'RDSinstance. Si aucun n'est spécifié, le chemin du fichier est `D:\S3\<filename in s3>`. RDSprend en charge les chemins absolus et les chemins relatifs. Si vous souhaitez créer un sous-dossier, incluez-le dans le chemin du fichier.
`@overwrite_file`	INT	0	Facultatif	Écraser le fichier existant : 0 = Ne pas écraser 1 = Écraser

Vous pouvez télécharger des fichiers sans extension de fichier et des fichiers avec les extensions de fichier suivantes : .bcp, .csv, .dat, .fmt, .info, .lst, .tbl, .txt et .xml.

> **Note**
>
> Les fichiers portant l'extension de fichier .ispac peuvent être téléchargés lorsque SQL Server Integration Services est activé. Pour plus d'informations sur l'activationSSIS, consultezSQL Server Integration Services.
> Les fichiers portant les extensions de fichier suivantes peuvent être téléchargés lorsque SQL Server Analysis Services est

activé : .abf, .asdatabase, .configsettings, .deploymentoptions, .deploymenttargets et .xmla. Pour plus d'informations sur l'activationSSAS, consultezSQLServices d'analyse de serveurs.

L'exemple suivant illustre la procédure stockée pour télécharger des fichiers depuis S3.

```
exec msdb.dbo.rds_download_from_s3
     @s3_arn_of_file='arn:aws:s3:::amzn-s3-demo-bucket/bulk_data.csv',
     @rds_file_path='D:\S3\seed_data\data.csv',
     @overwrite_file=1;
```

L'exemple `rds_download_from_s3` créé un dossier nommé `seed_data` in `D:\S3\`, si le dossier n'existe pas encore. Ensuite, l'exemple télécharge le fichier source `bulk_data.csv` depuis S3 vers un nouveau fichier nommé `data.csv` sur l'instance de base de données. Si le fichier existait déjà, il est écrasé car le paramètre `@overwrite_file` est défini sur 1.

Téléchargement de fichiers depuis une instance de base de données de SQL serveur vers un compartiment Amazon S3

Pour charger des fichiers depuis une instance de base de données RDS for SQL Server vers un compartiment S3, utilisez la procédure RDS stockée Amazon `msdb.dbo.rds_upload_to_s3` avec les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`@s3_arn_of_file`	NVARCHAR	–	Obligatoire	Le S3 ARN du fichier à créer dans S3, par exemple : `arn:aws:s3::: amzn-s3-demo-bucket /mydata.csv`
`@rds_file_path`	NVARCHAR	–	Obligatoire	Le chemin du fichier à charger sur S3. Les chemins absolus et relatifs sont pris en charge.
`@overwrite_file`	INT	–	Facultatif	

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				Écraser le fichier existant : 0 = Ne pas écraser 1 = Écraser

L'exemple suivant télécharge le fichier nommé `data.csv` depuis l'emplacement spécifié dans `D:\S3\seed_data\` un fichier `new_data.csv` du compartiment S3 spécifié par leARN.

```
exec msdb.dbo.rds_upload_to_s3
  @rds_file_path='D:\S3\seed_data\data.csv',
  @s3_arn_of_file='arn:aws:s3:::amzn-s3-demo-bucket/new_data.csv',
  @overwrite_file=1;
```

Si le fichier existait déjà dans S3, il est écrasé car le paramètre @overwrite_file est défini sur 1.

Lister les fichiers sur l'RDSinstance de base de données

Pour répertorier les fichiers disponibles sur l'instance de base de données, utilisez une procédure stockée et une fonction. Tout d'abord, exécutez la procédure stockée suivante pour récupérer les détails depuis les fichiers dans `D:\S3\`.

```
exec msdb.dbo.rds_gather_file_details;
```

La procédure stockée retourne l'ID de la tâche. À l'instar d'autres tâches, cette procédure stockée s'exécute de manière asynchrone. Dès que le statut de la tâche est `SUCCESS`, vous pouvez utiliser l'ID de tâche dans la fonction `rds_fn_list_file_details` pour répertorier les fichiers et répertoires existants dans D:\S3\, comme illustré ci-dessous.

```
SELECT * FROM msdb.dbo.rds_fn_list_file_details(TASK_ID);
```

La fonction `rds_fn_list_file_details` retourne un tableau avec les colonnes suivantes.

Paramètre de sortie	Description
`filepath`	Chemin absolu du fichier (par exemple, `D:\S3\mydata.csv`)
`size_in_bytes`	Taille du fichier (en octets)
`last_modified_utc`	Date et heure de dernière modification dans le UTC format
`is_directory`	Option qui indique si l'objet est un annuaire (`true`/`false`)

Supprimer des fichiers sur l'RDSinstance de base de données

Pour supprimer les fichiers disponibles sur l'instance de base de données, utilisez la procédure RDS stockée Amazon `msdb.dbo.rds_delete_from_filesystem` avec les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`@rds_file_path`	NVARCHAR	–	Obligatoire	Le chemin du fichier à supprimer. Les chemins absolus et relatifs sont pris en charge.
`@force_delete`	INT	0	Facultatif	Pour supprimer un annuaire, cet indicateur doit être inclus et défini sur 1. 1 = Supprimer un annuaire Ce paramètre est ignoré si vous supprimez un fichier.

Pour supprimer un annuaire, `@rds_file_path` doit se terminer par une barre oblique inverse (\) et `@force_delete` doit être défini sur `1`.

L'exemple suivant supprimer le fichier `D:\S3\delete_me.txt`.

```
exec msdb.dbo.rds_delete_from_filesystem
    @rds_file_path='D:\S3\delete_me.txt';
```

L'exemple suivant supprime l'annuaire `D:\S3\example_folder\`.

```
exec msdb.dbo.rds_delete_from_filesystem
    @rds_file_path='D:\S3\example_folder\',
    @force_delete=1;
```

Surveillance du statut d'une tâche de transfert de fichiers

Pour suivre le statut de votre tâche d'intégration S3, appelez la fonction `rds_fn_task_status`. Deux paramètres sont nécessaires. Le premier paramètre doit toujours être `NULL` car il ne s'applique pas à l'intégration S3. Le second paramètre accepte l'ID de tâche.

Pour consulter une liste de toutes les tâches, définissez le premier paramètre sur `NULL` et le second sur `0`, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,0);
```

Pour obtenir une tâche spécifique, définissez le premier paramètre sur `NULL` et le second sur l'ID de tâche, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,42);
```

La fonction `rds_fn_task_status` retourne les informations suivantes.

Paramètre de sortie	Description
`task_id`	ID de la tâche
`task_type`	Pour l'intégration S3, des tâches peuvent avoir les types suivants :

Paramètre de sortie	Description
	• DOWNLOAD_FROM_S3 • UPLOAD_TO_S3 • LIST_FILES_ON_DISK • DELETE_FILES_ON_DISK
database_name	Non applicable aux tâches d'intégration S3.
% complete	Progression de la tâche sous forme de pourcentage.
duration(mins)	Temps consacré à la tâche, en minutes.

Paramètre de sortie	Description
`lifecycle`	État de la tâche. Les statuts possibles sont les suivants : • `CREATED` – Après que vous avez appelé une des procédures stockées d'intégration S3, une tâche est créée et le statut est défini sur `CREATED`. • `IN_PROGRESS` – Après le démarrage d'une tâche, le statut est défini sur `IN_PROGRESS`. Le passage du statut `CREATED` à `IN_PROGRESS` peut prendre jusqu'à cinq minutes. • `SUCCESS` – Lorsqu'une tâche est terminée, le statut est défini sur `SUCCESS`. • `ERROR` – Si une tâche échoue, le statut est défini sur `ERROR`. Lisez la colonne `task_info` pour plus d'informations sur l'erreur. • `CANCEL_REQUESTED` – Après que vous avez appelé `rds_cancel_task`, le statut de la tâche est défini sur `CANCEL_REQUESTED`. • `CANCELLED` – Une fois une tâche annulée avec succès, l'état de la tâche est défini sur `CANCELLED`.
`task_info`	Informations supplémentaires sur la tâche. Si une erreur se produit pendant le traitement, cette colonne contient des informations sur l'erreur.
`last_updated`	Date et heure de la dernière mise à jour de l'état de la tâche.

Paramètre de sortie	Description
`created_at`	Date et heure de création de la tâche.
`S3_object_arn`	ARNL'objet S3 téléchargé depuis ou vers.
`overwrite_S3_backup_file`	Non applicable aux tâches d'intégration S3.
`KMS_master_key_arn`	Non applicable aux tâches d'intégration S3.
`filepath`	Le chemin du fichier sur l'RDSinstance de base de données.
`overwrite_file`	Option qui indique si un fichier existant a été écrasé.
`task_metadata`	Non applicable aux tâches d'intégration S3.

Annulation d'une tâche

Pour annuler une tâche d'intégration S3, utilisez la procédure stockée `msdb.dbo.rds_cancel_task` avec le paramètre `task_id`. Les tâches de suppression et la liste des tâches en cours ne peuvent pas être annulées. L'exemple suivant illustre une demande d'annulation d'une tâche.

```
exec msdb.dbo.rds_cancel_task @task_id = 1234;
```

Pour obtenir une vue d'ensemble de toutes les tâches et de leur tâcheIDs, utilisez la `rds_fn_task_status` fonction décrite dansSurveillance du statut d'une tâche de transfert de fichiers.

Limitations Multi-AZ pour l'intégration S3

Sur des instances Multi-AZ, les fichiers du dossier `D:\S3` sont supprimés du réplica de secours après un basculement. Un basculement peut être planifié, par exemple, lors de modifications apportées à une instance de base de données telles que la modification de la classe d'instance ou la mise à niveau de la version du moteur. La planification d'un basculement peut être annulée, par exemple en cas d'arrêt de l'instance principale.

> **Note**
>
> Nous ne recommandons pas d'utiliser le dossier `D:\S3` pour le stockage de fichiers. La bonne pratique consiste à charger des fichiers créés dans Amazon S3 afin de les rendre durables, et à télécharger les fichiers lorsque vous avez besoin d'importer des données.

Pour déterminer l'heure du dernier basculement, vous pouvez utiliser la procédure stockée `msdb.dbo.rds_failover_time`. Pour plus d'informations, consultez Déterminer l'heure du dernier basculement d'Amazon RDS for Server SQL.

Example d'Aucun basculement récent

Cet exemple illustre la sortie lorsqu'il n'y a pas de basculement récent dans les journaux d'erreurs. Aucun basculement ne s'est produit depuis le 29/04/2020 à 23:59:00.01.

Par conséquent, tous les fichiers téléchargés après cette date et cette heure et qui n'ont pas été supprimés avec la procédure stockée `rds_delete_from_filesystem` sont toujours accessibles sur l'hôte actuel. Les fichiers téléchargés avant cette date et cette heure peuvent également être disponibles.

errorlog_available_from	recent_failover_time
2020-04-29 23:59:00.0100000	null

Example de Basculement récent

Cet exemple illustre la sortie lorsqu'un basculement récent est détecté dans les journaux d'erreurs. Le basculement le plus récent a eu lieu le 05/05/2020 à 18:57:51.89.

Tous les fichiers téléchargés après cette date et cette heure et qui n'ont pas été supprimés avec la procédure stockée `rds_delete_from_filesystem` sont toujours accessibles sur l'hôte actuel.

errorlog_available_from	recent_failover_time
2020-04-29 23:59:00.0100000	2020-05-05 18:57:51.8900000

Désactivation RDS pour l'intégration SQL du serveur à S3

Vous trouverez ci-dessous comment désactiver l'intégration d'Amazon S3 à Amazon RDS for SQL Server. Les fichiers stockés dans `D:\S3\` ne sont pas supprimés lors de la désactivation de l'intégration S3.

> **Note**
>
> Pour supprimer un IAM rôle d'une instance de base de données, le statut de l'instance de base de données doit être`available`.

Console

Pour dissocier votre IAM rôle de votre instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez le nom RDS de l'instance de base de données for SQL Server pour afficher ses détails.
3. Dans l'onglet Connectivité et sécurité, dans la section Gérer IAM les rôles, choisissez le IAM rôle à supprimer.
4. Sélectionnez Delete (Supprimer).

AWS CLI

Pour supprimer le IAM rôle de l'instance de base de données RDS for SQL Server

- La AWS CLI commande suivante supprime le IAM rôle d'une instance de base de données RDS for SQL Server nommée*mydbinstance*.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds remove-role-from-db-instance \
    --db-instance-identifier mydbinstance \
    --feature-name S3_INTEGRATION \
    --role-arn your-role-arn
```

Dans Windows:

```
aws rds remove-role-from-db-instance ^
    --db-instance-identifier mydbinstance ^
    --feature-name S3_INTEGRATION ^
    --role-arn your-role-arn
```

Remplacez *your-role-arn* par le IAM rôle approprié ARN pour l'`--feature-name`option.

Utilisation de Database Mail sur Amazon RDS for SQL Server

Vous pouvez utiliser Database Mail pour envoyer des e-mails aux utilisateurs depuis votre instance de base de données Amazon RDS on SQL Server. Les messages peuvent contenir des fichiers et des résultats de requête. Database Mail comprend les éléments suivants :

- Objets de configuration et de sécurité – Ces objets créent des profils et des comptes, et sont stockés dans la base de données `msdb`.
- Objets de messagerie – Ces objets incluent la procédure stockée sp_send_dbmail utilisée pour envoyer des messages, ainsi que des structures de données contenant des informations sur les messages. Ils sont stockés dans la base de données `msdb`.
- Objets de journalisation et d'audit – Database Mail écrit les informations de journalisation dans la base de données `msdb` et dans le journal des événements de l'application Microsoft Windows.
- Le fichier exécutable de Database Mail – `DatabaseMail.exe` lit le contenu d'une file d'attente de la base de données `msdb` et envoie les e-mails.

RDSprend en charge Database Mail pour toutes les versions de SQL serveur sur le Web, les éditions Standard et Enterprise.

Limites

Les limites suivantes s'appliquent à l'utilisation de Database Mail sur votre instance de base de données de SQL serveur :

- Database Mail n'est pas pris en charge pour SQL Server Express Edition.
- La modification des paramètres de configuration de Database Mail n'est pas prise en charge. Pour afficher les valeurs prédéfinies (par défaut), vous devez utiliser la procédure stockée sysmail_help_configure_sp.
- Les pièces jointes ne sont pas entièrement prises en charge. Pour plus d'informations, consultez Utilisation de pièces jointes.
- La taille maximale des pièces jointes est de 1 Mo.
- Database Mail requiert une configuration supplémentaire sur les instances de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Considérations sur les déploiements multi-AZ.
- La configuration de l'agent SQL serveur pour envoyer des messages électroniques à des opérateurs prédéfinis n'est pas prise en charge.

Activation de Database Mail

Procédez comme suit pour activer Database Mail sur votre instance de base de données :

1. Créez un groupe de paramètres.
2. Modifiez le groupe de paramètres de manière à définir le paramètre `database mail xps` sur 1.
3. Associez le groupe de paramètres à l'instance de base de données.

Création du groupe de paramètres pour Database Mail

Créez un groupe de paramètres pour le `database mail xps` paramètre correspondant à l'édition et à la version SQL du serveur de votre instance de base de données.

> **Note**
>
> Vous pouvez également modifier un groupe de paramètres existant. Suivez la procédure décrite dans Modification du paramètre qui active Database Mail.

Console

L'exemple suivant crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans le volet Créer un groupe de paramètres, faites ce qui suit :
   a. Pour Famille de groupes de paramètres, choisissez sqlserver-se-13.0.
   b. Pour Nom du groupe, saisissez un identifiant pour le groupe de paramètres, tel que **`dbmail-sqlserver-se-13`**.
   c. Pour Description, saisissez **`Database Mail XPs`**.
5. Sélectionnez Create (Créer).

CLI

L'exemple suivant crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-db-parameter-group \
    --db-parameter-group-name dbmail-sqlserver-se-13 \
    --db-parameter-group-family "sqlserver-se-13.0" \
    --description "Database Mail XPs"
```

  Dans Windows:

```
aws rds create-db-parameter-group ^
    --db-parameter-group-name dbmail-sqlserver-se-13 ^
    --db-parameter-group-family "sqlserver-se-13.0" ^
    --description "Database Mail XPs"
```

Modification du paramètre qui active Database Mail

Modifiez le `database mail xps` paramètre dans le groupe de paramètres qui correspond à l'édition et à la version SQL du serveur de votre instance de base de données.

Pour activer Database Mail, définissez le paramètre `database mail xps` sur 1.

Console

L'exemple suivant modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.

3. Choisissez le groupe de paramètres, par exemple dbmail-sqlserver-se-13.
4. Sous Paramètres, filtrez la liste des paramètres pour **`mail`**.
5. Choisissez database mail xps.
6. Choisissez Modifier les paramètres.
7. Saisissez **1**.
8. Sélectionnez Enregistrer les modifications.

CLI

L'exemple suivant modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name dbmail-sqlserver-se-13 \
    --parameters "ParameterName='database mail
 xps',ParameterValue=1,ApplyMethod=immediate"
```

  Dans Windows:

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name dbmail-sqlserver-se-13 ^
    --parameters "ParameterName='database mail
 xps',ParameterValue=1,ApplyMethod=immediate"
```

Association du groupe de paramètres à l'instance de base de données

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour associer le groupe de paramètres Database Mail à l'instance de base de données.

Console

Vous pouvez associer le groupe de paramètres Database Mail à une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, associez-la lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-la en modifiant l'instance. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

CLI

Vous pouvez associer le groupe de paramètres Database Mail à une instance de base de données nouvelle ou existante.

Pour créer une instance de base de données avec le groupe de paramètres Database Mail

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe de paramètres.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 13.00.5426.0.v1 \
    --allocated-storage 100 \
    --manage-master-user-password \
    --master-username admin \
    --storage-type gp2 \
    --license-model li
    --db-parameter-group-name dbmail-sqlserver-se-13
```

  Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --db-instance-class db.m5.2xlarge ^
```

```
    --engine sqlserver-se ^
    --engine-version 13.00.5426.0.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --db-parameter-group-name dbmail-sqlserver-se-13
```

Pour modifier une instance de base de données et associer le groupe de paramètres Database Mail

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --db-parameter-group-name dbmail-sqlserver-se-13 \
    --apply-immediately
```

  Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --db-parameter-group-name dbmail-sqlserver-se-13 ^
    --apply-immediately
```

Configuration de Database Mail

Pour configurer Database Mail, procédez comme suit :

1. Créez le profil Database Mail.
2. Créez le compte Database Mail.
3. Ajoutez le compte Database Mail au profil Database Mail.
4. Ajoutez des utilisateurs au profil Database Mail.

> **Note**
>
> Pour configurer Database Mail, assurez-vous que vous disposez des autorisations `execute` requises sur les procédures stockées de la base de données `msdb`.

Création du profil Database Mail

Pour créer le profil Database Mail, vous devez utiliser la procédure stockée sysmail_add_profile_sp. L'exemple suivant crée un profil nommé `Notifications`.

Pour créer le profil

- Utilisez la SQL déclaration suivante.

```
USE msdb
GO

EXECUTE msdb.dbo.sysmail_add_profile_sp
    @profile_name         = 'Notifications',
    @description          = 'Profile used for sending outgoing notifications using
 Amazon SES.';
GO
```

Création du compte Database Mail

Pour créer le compte Database Mail, vous devez utiliser la procédure stockée sysmail_add_account_sp. L'exemple suivant crée un compte nommé `SES` sur une instance de base de données RDS for SQL Server dans un environnement privéVPC, à l'aide d'Amazon Simple Email Service.

L'utilisation d'Amazon SES nécessite les paramètres suivants :

- `@email_address`— Une identité SES vérifiée par Amazon. Pour plus d'informations, consultez Identités vérifiées sur Amazon SES.
- `@mailserver_name`— Un point de SES SMTP terminaison Amazon. Pour plus d'informations, consultez Connexion à un point de SES SMTP terminaison Amazon.
- `@username`— Un nom SES SMTP d'utilisateur Amazon. Pour plus d'informations, consultez la section Obtention des SES SMTP informations d'identification Amazon.

N'utilisez pas de nom d' AWS Identity and Access Management utilisateur.

- `@password`— Un SES SMTP mot de passe Amazon. Pour plus d'informations, consultez la section Obtention des SES SMTP informations d'identification Amazon.

Pour créer le compte

- Utilisez la SQL déclaration suivante.

```
USE msdb
GO

EXECUTE msdb.dbo.sysmail_add_account_sp
    @account_name        = 'SES',
    @description         = 'Mail account for sending outgoing notifications.',
    @email_address       = 'nobody@example.com',
    @display_name        = 'Automated Mailer',
    @mailserver_name     = 'vpce-0a1b2c3d4e5f-01234567.email-smtp.us-
west-2.vpce.amazonaws.com',
    @port                = 587,
    @enable_ssl          = 1,
    @username            = 'Smtp_Username',
    @password            = 'Smtp_Password';
GO
```

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

Ajout du compte Database Mail au profil Database Mail

Pour ajouter le compte Database Mail au profil Database Mail, vous devez utiliser la procédure stockée sysmail_add_profileaccount_sp. L'exemple suivant ajoute le compte SES au profil `Notifications`.

Pour ajouter le compte au profil

- Utilisez la SQL déclaration suivante.

```
USE msdb
GO

EXECUTE msdb.dbo.sysmail_add_profileaccount_sp
    @profile_name        = 'Notifications',
    @account_name        = 'SES',
    @sequence_number     = 1;
GO
```

Ajout d'utilisateurs au profil Database Mail

Pour autoriser un principal de base de données `msdb` à utiliser un profil Database Mail, vous devez utiliser la procédure stockée sysmail_add_principalprofile_sp. Un principal est une entité qui peut demander des ressources au SQL serveur. Le principal de base de données doit être mappé à un utilisateur d'authentification SQL serveur, à un utilisateur d'authentification Windows ou à un groupe d'authentification Windows.

L'exemple suivant accorde un accès public au profil `Notifications`.

Pour ajouter un utilisateur au profil

- Utilisez la SQL déclaration suivante.

```
USE msdb
GO

EXECUTE msdb.dbo.sysmail_add_principalprofile_sp
    @profile_name       = 'Notifications',
    @principal_name     = 'public',
    @is_default         = 1;
GO
```

Procédures et fonctions RDS stockées par Amazon pour Database Mail

Microsoft fournit des procédures stockées pour utiliser Database Mail, comme la création, la présentation en listes, la mise à jour et la suppression de comptes et de profils. En outre, RDS fournit les procédures stockées et les fonctions pour Database Mail indiquées dans le tableau suivant.

Procédure/Fonction	Description
rds_fn_sysmail_allitems	Affiche les messages envoyés, y compris ceux envoyés par d'autres utilisateurs.
rds_fn_sysmail_event_log	Affiche les événements, y compris ceux des messages envoyés par d'autres utilisateurs.
rds_fn_sysmail_mailattachments	Affiche les pièces jointes, y compris celles des messages envoyés par d'autres utilisateurs.
rds_sysmail_control	Démarre et arrête la file d'attente de courrier (processus DatabaseMail .exe).
rds_sysmail_delete_mailitems_sp	Supprime des tables internes de Database Mail les e-mails envoyés par l'ensemble des utilisateurs.

Envoi d'e-mails à l'aide de Database Mail

Pour envoyer des e-mails à l'aide de Database Mail, vous devez utiliser la procédure stockée sp_send_dbmail.

Utilisation

```
EXEC msdb.dbo.sp_send_dbmail
@profile_name = 'profile_name',
@recipients = 'recipient1@example.com[; recipient2; ... recipientn]',
@subject = 'subject',
@body = 'message_body',
[@body_format = 'HTML'],
[@file_attachments = 'file_path1; file_path2; ... file_pathn'],
[@query = 'SQL_query'],
[@attach_query_result_as_file = 0|1]';
```

Les paramètres suivants sont obligatoires :

- `@profile_name` – Nom du profil Database Mail à partir duquel envoyer le message doit être envoyé.

- `@recipients` – Liste des adresses e-mail, délimitées par des points-virgules, auxquelles le message doit être envoyé.
- `@subject` – Objet du message.
- `@body` – Corps du message. Vous pouvez également utiliser une variable déclarée comme corps.

Les paramètres suivants sont facultatifs :

- `@body_format`— Ce paramètre est utilisé avec une variable déclarée pour envoyer un e-mail au HTML format.
- `@file_attachments` – Liste des pièces jointes de message délimitées par des points-virgules. Les chemins d'accès aux fichiers doivent être des chemins absolus.
- `@query`— Une SQL requête à exécuter. Les résultats de la requête peuvent être joints sous forme de fichier ou inclus dans le corps du message.
- `@attach_query_result_as_file` – Permet d'indiquer si les résultats de la requête doivent être joints sous forme de fichier. À définir sur 0 pour si la réponse est négative et sur 1 si elle est positive. La valeur par défaut est 0.

Exemples

Les exemples suivants montrent comment envoyer des e-mails.

Example envoi d'un message à un seul destinataire

```
USE msdb
GO

EXEC msdb.dbo.sp_send_dbmail
     @profile_name       = 'Notifications',
     @recipients         = 'nobody@example.com',
     @subject            = 'Automated DBMail message - 1',
     @body               = 'Database Mail configuration was successful.';
GO
```

Example envoi d'un message à plusieurs destinataires

```
USE msdb
GO
```

```
EXEC msdb.dbo.sp_send_dbmail
     @profile_name       = 'Notifications',
     @recipients         = 'recipient1@example.com;recipient2@example.com',
     @subject            = 'Automated DBMail message - 2',
     @body               = 'This is a message.';
GO
```

Example de l'envoi d'un résultat de SQL requête sous forme de pièce jointe

```
USE msdb
GO

EXEC msdb.dbo.sp_send_dbmail
     @profile_name       = 'Notifications',
     @recipients         = 'nobody@example.com',
     @subject            = 'Test SQL query',
     @body               = 'This is a SQL query test.',
     @query              = 'SELECT * FROM abc.dbo.test',
     @attach_query_result_as_file = 1;
GO
```

Example de l'envoi d'un message au HTML format

```
USE msdb
GO

DECLARE @HTML_Body as NVARCHAR(500) = 'Hi, <h4> Heading </h4> </br> See the report. <b>
 Regards </b>';

EXEC msdb.dbo.sp_send_dbmail
     @profile_name       = 'Notifications',
     @recipients         = 'nobody@example.com',
     @subject            = 'Test HTML message',
     @body               = @HTML_Body,
     @body_format        = 'HTML';
GO
```

Example envoi d'un message à l'aide d'un déclencheur lorsqu'un événement spécifique se produit dans la base de données

```
USE AdventureWorks2017
```

```
GO
IF OBJECT_ID ('Production.iProductNotification', 'TR') IS NOT NULL
DROP TRIGGER Purchasing.iProductNotification
GO

CREATE TRIGGER iProductNotification ON Production.Product
   FOR INSERT
   AS
   DECLARE @ProductInformation nvarchar(255);
   SELECT
   @ProductInformation = 'A new product, ' + Name + ', is now available for $' +
 CAST(StandardCost AS nvarchar(20)) + '!'
   FROM INSERTED i;

EXEC msdb.dbo.sp_send_dbmail
     @profile_name       = 'Notifications',
     @recipients         = 'nobody@example.com',
     @subject            = 'New product information',
     @body               = @ProductInformation;
GO
```

Affichage des messages, des journaux et des pièces jointes

Vous utilisez des procédures RDS stockées pour consulter les messages, les journaux d'événements et les pièces jointes.

Pour afficher tous les e-mails

- Utilisez la SQL requête suivante.

```
SELECT * FROM msdb.dbo.rds_fn_sysmail_allitems(); --WHERE sent_status='sent' or
 'failed' or 'unsent'
```

Pour afficher les journaux d'événements des tous les e-mails

- Utilisez la SQL requête suivante.

```
SELECT * FROM msdb.dbo.rds_fn_sysmail_event_log();
```

Pour afficher toutes les pièces jointes

- Utilisez la SQL requête suivante.

```
SELECT * FROM msdb.dbo.rds_fn_sysmail_mailattachments();
```

Suppression de messages

Pour supprimer des messages, vous devez utiliser la procédure stockée `rds_sysmail_delete_mailitems_sp`.

> **Note**
>
> RDSsupprime automatiquement les éléments de la table de messagerie lorsque les données d'DBMailhistorique atteignent 1 Go, avec une période de conservation d'au moins 24 heures. Si vous souhaitez conserver les éléments plus longtemps, vous pouvez les archiver. Pour plus d'informations, consultez la section Créer une tâche d'agent SQL serveur pour archiver les messages électroniques de base de données et les journaux d'événements dans la documentation Microsoft.

Pour supprimer tous les e-mails

- Utilisez la SQL déclaration suivante.

```
DECLARE @GETDATE datetime
SET @GETDATE = GETDATE();
EXECUTE msdb.dbo.rds_sysmail_delete_mailitems_sp @sent_before = @GETDATE;
GO
```

Pour supprimer tous les e-mails dotés d'un état particulier

- Utilisez l'SQLinstruction suivante pour supprimer tous les messages ayant échoué.

```
DECLARE @GETDATE datetime
SET @GETDATE = GETDATE();
EXECUTE msdb.dbo.rds_sysmail_delete_mailitems_sp @sent_status = 'failed';
GO
```

Démarrage et arrêt de la file d'attente de courrier

Suivez les instructions suivantes pour démarrer et arrêter la file d'attente de courrier de la base de données :

Rubriques

- Lancement de la file d'attente de messagerie
- Arrêt de la file d'attente de messagerie

Lancement de la file d'attente de messagerie

Pour lancer le processus Database Mail, vous devez utiliser la procédure stockée `rds_sysmail_control`.

> **Note**
>
> L'activation de Database Mail lance automatiquement la file d'attente de messagerie.

Pour lancer la file d'attente de messagerie

- Utilisez la SQL déclaration suivante.

```
EXECUTE msdb.dbo.rds_sysmail_control start;
GO
```

Arrêt de la file d'attente de messagerie

Pour arrêter le processus Database Mail, vous devez utiliser la procédure stockée `rds_sysmail_control`.

Pour arrêter la file d'attente de messagerie

- Utilisez la SQL déclaration suivante.

```
EXECUTE msdb.dbo.rds_sysmail_control stop;
GO
```

Utilisation de pièces jointes

Les extensions de pièce jointe suivantes ne sont pas prises en charge dans les messages électroniques de base de données envoyés RDS sur le SQL serveur : .ade, .adp, .apk, .appx, .appxbundle, .bat, .bak, .cab, .chm, .cmd, .com, .cpl, .dll, .dmg, .exe, .hta, .ins, .ldf, .lib, .lnk, .mde, .mdf, .msc, .msi, .msix, .msixbundle, .msp, .mst, .nsh, .pif, .ps, .ps1, .psc1, .reg, .rgs, .sct, .shb, .shs, .u3p, .vb, .vbe, .vbs, .vbscript, .vxd, .ws, .wsc, .wsf et .wsh.

Database Mail utilise le contexte de sécurité Microsoft Windows de l'utilisateur actuel pour contrôler l'accès aux fichiers. Les utilisateurs qui se connectent avec l'authentification SQL du serveur ne peuvent pas joindre de fichiers à l'aide du `@file_attachments` paramètre associé à la procédure `sp_send_dbmail` stockée. Windows n'autorise pas le SQL serveur à fournir les informations d'identification d'un ordinateur distant à un autre ordinateur distant. Par conséquent, Database Mail ne peut pas joindre de fichiers à partir d'un partage réseau lorsque la commande est exécutée depuis un ordinateur autre que celui qui exécute le SQL serveur.

Vous pouvez toutefois utiliser les tâches de l'agent SQL serveur pour joindre des fichiers. Pour plus d'informations sur l'agent SQL serveur, consultez la section Utilisation de l'agent SQL Server pour Amazon RDS et l'agent SQL serveur dans la documentation Microsoft.

Considérations sur les déploiements multi-AZ

Lorsque vous configurez Database Mail sur une instance de base de données multi-AZ, la configuration n'est pas automatiquement propagée vers la zone secondaire. Nous vous recommandons de convertir l'instance multi-AZ en instance mono-AZ, de configurer Database Mail, puis de reconvertir l'instance de base de données en instance multi-AZ. Les nœuds principal et secondaire disposeront ensuite de la configuration de Database Mail.

Si vous créez une réplique en lecture à partir de votre instance Multi-AZ sur laquelle Database Mail est configuré, la réplique hérite de la configuration, mais sans le mot de passe du serveur. SMTP Mettez à jour le compte Database Mail avec le mot de passe.

Suppression de la restriction SMTP (port 25)

Par défaut, AWS bloque le trafic sortant sur SMTP (port 25) RDS pour les instances de base de données SQL du serveur. Ceci est fait pour empêcher le spam conformément aux politiques du propriétaire de l'interface elastic network. Vous pouvez supprimer cette restriction si nécessaire. Pour plus d'informations, consultez Comment supprimer la restriction sur le port 25 de mon EC2 instance Amazon ou de ma fonction Lambda ? .

Prise en charge du stockage d'instance pour la base de données tempdb sur Amazon RDS for SQL Server

Un stockage d'instance fournit un stockage temporaire de niveau bloc pour votre instance de base de données. Le stockage réside sur les disques physiquement attachés à l'ordinateur hôte. Ces disques disposent d'un stockage d'instance NVMe (Non-Volatile Memory Express) basé sur des disques SSD. Ce stockage est optimisé pour une faible latence, des hautes performances d'I/O aléatoires et un débit de lecture séquentielle élevé.

En plaçant des fichiers de données `tempdb` et des fichiers journaux `tempdb` sur le stockage d'instance, vous pouvez réduire les latences de lecture et d'écriture par rapport au stockage standard basé sur Amazon EBS.

> **Note**
>
> Les fichiers de base de données et les fichiers journaux de base de données SQL Server ne sont pas placés dans le stockage d'instance.

Activation du stockage d'instance

Lorsque RDS met en service des instances de base de données avec l'une des classes d'instance suivantes, la base de données `tempdb` est automatiquement placée dans le stockage d'instance :

- db.m5d
- db.r5d
- db.x2iedn

Pour activer le stockage d'instance, effectuez l'une des opérations suivantes :

- Créez une instance de base de données SQL Server à l'aide de l'un de ces types d'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données SQL Server existante pour utiliser l'une d'elles. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Le stockage d'instance est disponible dans toutes les régions AWS où un ou plusieurs de ces classes d'instance sont prises en charge. Pour plus d'informations sur les classes d'instance

db.m5d et db.r5d, reportez-vous à la section Classes d'instances de base de données . Pour plus d'informations sur les classes d'instance prises en charge par Amazon RDS pour SQL Server, reportez-vous à la section prise en charge des classes d'instance de base de données pour Microsoft SQL Server.

Considérations relatives à l'emplacement et à la taille des fichiers

Sur les instances sans stockage d'instance, RDS stocke les fichiers de données et les fichiers journaux tempdb dans le répertoire D:\rdsdbdata\DATA. Les deux fichiers commencent à 8 Mo par défaut.

Sur les instances avec un stockage d'instance, RDS stocke les fichiers de données et les fichiers journaux tempdb dans le répertoire T:\rdsdbdata\DATA.

Quand tempdb n'a qu'un seul fichier de données (tempdb.mdf) et un seul fichier journal (templog.ldf), templog.ldf commence à 8 Mo par défaut et tempdb.mdf commence à 80 % ou plus de la capacité de stockage de l'instance. Vingt pour cent de la capacité de stockage ou 200 Go, selon le moins élevé des deux, est gardé libre pour le démarrage. Des fichiers de données tempdb multiples partagent 80 % de l'espace disque uniformément, tandis que les fichiers journaux ont toujours une taille initiale de 8 Mo.

Par exemple, si vous modifiez votre classe d'instance de base de données de db.m5.2xlarge à db.m5d.2xlarge , la taille des fichiers de données tempdb passe de 8 Mo chacun à 234 Go au total.

> **Note**
>
> En plus des fichiers de données et des fichiers journaux tempdb dans le magasin d'instance (T:\rdsdbdata\DATA), vous pouvez continuer de créer des fichiers de données et des fichiers journaux tempdb sur le volume de données (D:\rdsdbdata\DATA). Ces fichiers ont toujours une taille initiale de 8 Mo.

Considérations relatives à la sauvegarde

Vous devrez peut-être conserver les sauvegardes pendant longtemps, ce qui entraîne des coûts au fil du temps. Les blocs de données et de journaux de tempdb peuvent changer très souvent en fonction de la charge de travail. Cela peut augmenter considérablement la taille de l'instantané de base de données.

Lorsque `tempdb` est sur le stockage d'instance, les instantanés n'incluent pas de fichiers temporaires. Cela signifie que les tailles d'instantanés sont plus petites et consomment moins de l'allocation de sauvegarde libre par rapport au stockage EBS uniquement.

Erreurs de disque plein

Si vous utilisez tout l'espace disponible dans le stockage d'instance, vous pourriez recevoir des erreurs comme celles-ci :

- The transaction log for database 'tempdb' is full due to 'ACTIVE_TRANSACTION' (Le journal des transactions pour la base de données 'tempdb' est plein en raison de 'ACTIVE_TRANSACTION').
- Could not allocate space for object 'dbo.SORT temporary run storage: 140738941419520' in database 'tempdb' because the 'PRIMARY' filegroup is full (Impossible d'allouer de l'espace pour l'objet 'dbo.SORT temporary run storage: 140738941419520' dans la base de données 'tempdb' car le groupe de fichiers 'PRIMARY' est plein. Créez de l'espace disque en supprimant des fichiers inutiles, en supprimant des objets dans le groupe de fichiers, en ajoutant des fichiers supplémentaires au groupe de fichiers ou en réglant la croissance automatique pour les fichiers existants du groupe de fichiers.

Vous pouvez effectuer une ou plusieurs des opérations suivantes lorsque le stockage d'instance est plein :

- Ajustez votre charge de travail ou votre façon d'utiliser `tempdb`.
- Mise à l'échelle pour utiliser une classe d'instance de base de données avec plus de stockage NVMe.
- Arrêtez d'utiliser le stockage d'instance et utilisez une classe d'instance avec un stockage EBS uniquement.
- Utilisez un mode mixte en ajoutant des données secondaires ou des fichiers journaux pour `tempdb` sur le volume EBS.

Suppression du stockage d'instance

Pour supprimer le stockage d'instances, modifiez votre instance de base de données SQL Server de sorte qu'elle utilise un type d'instance qui ne prend pas en charge le stockage d'instances, tel que db.m5, db.r5 ou db.x1e.

> **Note**
>
> Lorsque vous supprimez le stockage d'instance, les fichiers temporaires sont déplacés vers le répertoire `D:\rdsdbdata\DATA` et leur taille est réduite à 8 Mo.

Utilisation d'événements étendus avec Amazon RDS pour Microsoft SQL Server

Vous pouvez utiliser des événements étendus dans Microsoft SQL Server pour recueillir des informations de débogage et de dépannage pour Amazon RDS for SQL Server. Les événements étendus remplacent SQL Trace et Server Profiler, qui ont été déconseillés par Microsoft. Les événements étendus sont similaires aux traces du profileur, mais avec un contrôle plus granulaire sur les événements suivis. Les événements étendus sont pris en charge pour les versions de SQL serveur 2016 et ultérieures sur AmazonRDS. Pour plus d'informations, consultez Présentation des événements étendus dans la documentation Microsoft.

Les événements étendus sont automatiquement activés pour les utilisateurs disposant de privilèges d'utilisateur principal dans Amazon RDS for SQL Server.

Rubriques

- Limitations et recommandations
- Configuration des événements étendus sur RDS le SQL serveur
- Considérations sur les déploiements multi-AZ
- Interrogation de fichiers d'événements étendus

Limitations et recommandations

Lorsque vous utilisez des événements étendus sur RDS for SQL Server, les limites suivantes s'appliquent :

- Les événements étendus ne sont pris en charge que pour les éditions Enterprise et Standard.
- Vous ne pouvez pas modifier les sessions d'événements étendus par défaut.
- Assurez-vous de définir le mode de partition de mémoire de session sur `NONE`.
- Le mode de rétention d'événement de session peut être `ALLOW_SINGLE_EVENT_LOSS` ou `ALLOW_MULTIPLE_EVENT_LOSS`.
- Le suivi des événements pour les cibles Windows (ETW) n'est pas pris en charge.
- Assurez-vous que les cibles de fichiers se trouvent dans le répertoire `D:\rdsdbdata\log`.
- Pour les cibles correspondant aux paire, définissez la propriété `respond_to_memory_pressure` sur 1.
- La mémoire cible de la mémoire tampon Ring ne peut pas être supérieure à 4 Mo.

- Les actions suivantes ne sont pas prises en charge :
  - `debug_break`
  - `create_dump_all_threads`
  - `create_dump_single_threads`
- L'événement `rpc_completed` est pris en charge sur les versions suivantes et ultérieures : 15.0.4083.2, 14.0.3370.1, 13.0.5865.1, 12.0.6433.1, 11.0.7507.2.

Configuration des événements étendus sur RDS le SQL serveur

Sur RDS for SQL Server, vous pouvez configurer les valeurs de certains paramètres des sessions d'événements étendues. Le tableau suivant décrit les paramètres configurables.

Nom du paramètre	Description
`xe_session_max_memory`	Spécifie la quantité maximale de mémoire à allouer à la s événements. Cette valeur correspond au paramètre `max_`
`xe_session_max_event_size`	Spécifie la taille de mémoire maximale autorisée pour les paramètre `max_event_size` de la session d'événemer
`xe_session_max_dispatch_latency`	Spécifie la durée pendant laquelle les événements sont m cibles de session d'événements étendus. Cette valeur co de la session d'événement.
`xe_file_target_size`	Spécifie la taille maximale de la cible du fichier. Cette val cible du fichier.
`xe_file_retention`	Spécifie la durée de conservation en jours pour les fichier d'événements.

> **Note**
>
> La valeur zéro `xe_file_retention` entraîne la suppression automatique des fichiers .xel une fois le verrouillage de ces fichiers déverrouillé par le SQL serveur. Le verrouillage est libéré chaque fois qu'un fichier .xel atteint la limite de taille définie dans `xe_file_target_size`.

Vous pouvez utiliser la procédure `rdsadmin.dbo.rds_show_configuration` stockée pour afficher les valeurs actuelles de ces paramètres. Par exemple, utilisez l'SQLinstruction suivante pour afficher le paramètre actuel de`xe_session_max_memory`.

```
exec rdsadmin.dbo.rds_show_configuration 'xe_session_max_memory'
```

Vous pouvez utiliser la procédure stockée `rdsadmin.dbo.rds_set_configuration` pour les modifier. Par exemple, utilisez l'SQLinstruction suivante pour définir une `xe_session_max_memory` valeur de 4 Mo.

```
exec rdsadmin.dbo.rds_set_configuration 'xe_session_max_memory', 4
```

Considérations sur les déploiements multi-AZ

Lorsque vous créez une session d'événements étendus sur une instance de base de données principale, elle ne se propage pas au réplica de secours. Vous pouvez basculer et créer la session d'événements étendus sur la nouvelle instance de base de données principale. Vous pouvez également supprimer et lire la configuration multi-AZ pour propager la session d'événements étendus au réplica de secours. RDSarrête toutes les sessions d'événements prolongées autres que celles par défaut sur la réplique de secours, afin que ces sessions ne consomment pas de ressources sur la réplique de secours. Pour cette raison, après qu'un réplica de secours devient l'instance de base de données principale, veillez à démarrer manuellement les sessions d'événements étendus sur la nouvelle instance principale.

> **Note**
>
> Cette approche s'applique aux groupes de disponibilité Always On et à la mise en miroir de bases de données.

Vous pouvez également utiliser une tâche d'agent SQL serveur pour suivre la réplique de secours et démarrer les sessions si la réplique de secours devient la réplique principale. Par exemple, utilisez la requête suivante dans votre étape de travail de l'agent SQL serveur pour redémarrer les sessions d'événements sur une instance de base de données principale.

```
BEGIN
    IF (DATABASEPROPERTYEX('rdsadmin','Updateability')='READ_WRITE'
    AND DATABASEPROPERTYEX('rdsadmin','status')='ONLINE'
```

```
    AND (DATABASEPROPERTYEX('rdsadmin','Collation') IS NOT NULL OR
 DATABASEPROPERTYEX('rdsadmin','IsAutoClose')=1)
    )
    BEGIN
        IF NOT EXISTS (SELECT 1 FROM sys.dm_xe_sessions WHERE name='xe1')
            ALTER EVENT SESSION xe1 ON SERVER STATE=START
        IF NOT EXISTS (SELECT 1 FROM sys.dm_xe_sessions WHERE name='xe2')
            ALTER EVENT SESSION xe2 ON SERVER STATE=START
    END
END
```

Cette requête redémarre les sessions d'événements `xe1` et `xe2` sur une instance de base de données principale si ces sessions sont à l'état arrêté. Vous pouvez également ajouter une planification avec un intervalle pratique à cette requête.

Interrogation de fichiers d'événements étendus

Vous pouvez utiliser SQL Server Management Studio ou la `sys.fn_xe_file_target_read_file` fonction pour afficher les données d'événements étendus utilisant des fichiers cibles. Pour plus d'informations sur cette fonction, consultez sys.fn_xe_file_target_read_file (Transact -) dans la documentation Microsoft. SQL

Les cibles de fichiers d'événements étendus peuvent uniquement écrire des fichiers `D:\rdsdbdata\log` dans le RDS répertoire SQL du serveur.

Par exemple, utilisez la SQL requête suivante pour répertorier le contenu de tous les fichiers des sessions d'événements étendues dont le nom commence par`xe`.

```
SELECT * FROM sys.fn_xe_file_target_read_file('d:\rdsdbdata\log\xe*', null,null,null);
```

Accès aux sauvegardes des journaux de transactions avec RDS for SQL Server

Avec accès aux sauvegardes des journaux de transactions pour RDS for SQL Server, vous pouvez répertorier les fichiers de sauvegarde des journaux de transactions pour une base de données et les copier dans un compartiment Amazon S3 cible. En copiant les sauvegardes des journaux de transactions dans un compartiment Amazon S3, vous pouvez les utiliser en combinaison avec des sauvegardes de base de données complètes et différentielles pour effectuer des restaurations de base de données ponctuelles. Vous utilisez les procédures stockées RDS pour configurer l'accès aux sauvegardes des journaux de transactions, répertorier les sauvegardes des journaux de transactions disponibles et les copier dans votre compartiment Amazon S3.

L'accès aux sauvegardes des journaux de transactions offre les fonctionnalités et avantages suivants :

- Répertoriez et affichez les métadonnées des sauvegardes des journaux de transactions disponibles pour une base de données sur une instance de base de données RDS for SQL Server.
- Copiez les sauvegardes disponibles des journaux de transactions depuis RDS for SQL Server vers un compartiment Amazon S3 cible.
- Effectuez des point-in-time restaurations de bases de données sans avoir à restaurer une instance de base de données complète. Pour plus d'informations sur la restauration d'un cluster de bases de données à un instant dans le passé, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Disponibilité et prise en charge

L'accès aux sauvegardes du journal des transactions est pris en charge dans toutes les AWS régions. L'accès aux sauvegardes des journaux de transactions est disponible pour toutes les éditions et versions de Microsoft SQL Server prises en charge sur Amazon RDS.

Prérequis

Avant d'autoriser l'accès aux sauvegardes des journaux de transactions, les exigences suivantes doivent être satisfaites :

- Les sauvegardes automatisées doivent être activées sur l'instance de base de données et la conservation des sauvegardes doit être définie sur une valeur d'un ou plusieurs jours. Pour plus

d'informations sur l'activation des sauvegardes automatisées et la configuration d'une politique de conservation, consultez Activation des sauvegardes automatiques.

- Un compartiment Amazon S3 doit exister dans le même compte et la même région que l'instance de base de données source. Avant d'autoriser l'accès aux sauvegardes des journaux de transactions, choisissez un compartiment Amazon S3 existant ou créez un nouveau compartiment à utiliser pour les fichiers de sauvegarde de vos journaux de transactions.
- Une politique d'autorisations pour les compartiments Amazon S3 doit être configurée comme suit pour permettre à Amazon RDS d'y copier des fichiers journaux de transactions :
  1. Définissez la propriété d'appartenance du compte d'objet sur le compartiment sur Bucket Owner Preferred (Propriétaire du compartiment préféré).
  2. Ajoutez la politique suivante. Il n'y aura pas de politique par défaut. Par conséquent, utilisez les listes de contrôle d'accès (ACL) des compartiments pour modifier la politique des compartiments et l'ajouter.

  L'exemple suivant utilise un ARN pour spécifier une ressource. Nous vous recommandons d'utiliser les clés de contexte de condition globale `SourceArn` et `SourceAccount` dans des relations d'approbation basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. Pour plus d'informations sur l'utilisation ARNs, consultez Amazon resource names (ARNs) etNoms des ressources Amazon (ARNs) dans Amazon RDS.

  Example d'une politique d'autorisations Amazon S3 pour l'accès aux sauvegardes des journaux de transactions

  JSON

```
   {
   "Version": "2012-10-17",
   "Statement": [
       {
           "Sid": "Only allow writes to my bucket with bucket owner full
control",
           "Effect": "Allow",
           "Principal": {
               "Service": "backups.rds.amazonaws.com"
           },
           "Action": "s3:PutObject",
           "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/{customer_path}/*",
```

```
            "Condition": {
                "StringEquals": {
                    "s3:x-amz-acl": "bucket-owner-full-control",
                    "aws:sourceAccount": "{customer_account}",
                    "aws:sourceArn": "{db_instance_arn}"
                }
            }
        }
    ]
}
```

- Rôle AWS Identity and Access Management (IAM) permettant d'accéder au compartiment Amazon S3. Si vous avez déjà un rôle IAM, vous pouvez l'utiliser. Vous pouvez choisir d'avoir un nouveau rôle IAM créé pour vous quand vous ajoutez l'option SQLSERVER_BACKUP_RESTORE à l'aide de la AWS Management Console. Vous pouvez également en créer un nouveau manuellement. Pour plus d'informations sur la création et la configuration d'un rôle IAM avec SQLSERVER_BACKUP_RESTORE, consultez Création manuelle d'un rôle IAM pour les sauvegarde et restauration natives.
- L'option SQLSERVER_BACKUP_RESTORE doit être ajoutée à un groupe d'options sur votre instance de base de données. Pour plus d'informations sur l'ajout de l'option SQLSERVER_BACKUP_RESTORE, consultez Prise en charge des sauvegarde et restauration natives dans SQL Server.

> **Note**
>
> Si le chiffrement du stockage est activé sur votre instance de base de données, les actions AWS KMS (KMS) et la clé doivent être fournies dans le rôle IAM fourni dans le groupe d'options de sauvegarde et de restauration natif.

Éventuellement, si vous avez l'intention d'utiliser la procédure stockée `rds_restore_log` pour effectuer des restaurations de base de données ponctuelles, nous vous recommandons d'utiliser le même chemin Amazon S3 pour le groupe d'options de sauvegarde et de restauration natives et d'accéder aux sauvegardes des journaux de transactions. Cette méthode garantit que quand Amazon RDS assume le rôle du groupe d'options pour exécuter les fonctions de restauration des journaux, il a accès à la récupération des sauvegardes des journaux de transactions à partir du même chemin Amazon S3.

- Si l'instance de base de données est chiffrée, quel que soit le type de chiffrement (cléAWS gérée ou clé gérée par le client), vous devez fournir une clé KMS gérée par le client dans le rôle IAM et dans la procédure `rds_tlog_backup_copy_to_S3` stockée.

Limitations et recommandations

L'accès aux sauvegardes des journaux de transactions comporte les limites et recommandations suivantes :

- Vous pouvez répertorier et copier jusqu'aux sept derniers jours de sauvegarde des journaux de transactions pour toute instance de base de données dont la conservation des sauvegardes est configurée entre 1 et 35 jours.
- Le compartiment Amazon S3 utilisé pour accéder aux sauvegardes des journaux de transactions doit exister dans le même compte et la même région que l'instance de base de données source. La copie intercompte et entre régions n'est pas prise en charge.
- Un seul compartiment Amazon S3 peut être configuré comme cible pour y copier les sauvegardes des journaux de transactions. Vous pouvez choisir un nouveau compartiment Amazon S3 cible à l'aide de la procédure stockée `rds_tlog_copy_setup`. Pour plus d'informations sur le choix d'un nouveau compartiment Amazon S3 cible, consultez Configuration de l'accès aux sauvegardes des journaux de transactions.
- Vous ne pouvez pas spécifier la clé KMS lorsque vous utilisez la procédure stockée `rds_tlog_backup_copy_to_S3` si votre instance RDS n'est pas activée pour le chiffrement du stockage.
- La copie multi-compte n'est pas prise en charge. Le rôle IAM utilisé pour la copie autorise uniquement l'accès en écriture aux compartiments Amazon S3 au sein du compte de propriétaire de l'instance de base de données.
- Seules deux tâches simultanées de type quelconque peuvent être exécutées sur une instance de base de données RDS for SQL Server.
- Une seule tâche de copie peut être exécutée à la fois pour une base de données. Si vous souhaitez copier des sauvegardes des journaux de transactions pour plusieurs bases de données sur l'instance de base de données, utilisez une tâche de copie distincte pour chaque base de données.
- Si vous copiez une sauvegarde des journaux de transactions qui existe déjà avec le même nom dans le compartiment Amazon S3, la sauvegarde existante des journaux de transactions sera remplacée.

- Vous ne pouvez exécuter que les procédures stockées qui disposent d'un accès aux sauvegardes des journaux de transactions sur l'instance de base de données principale. Vous ne pouvez pas exécuter ces procédures stockées sur un réplica en lecture RDS for SQL Server ou sur une instance secondaire d'un cluster de bases de données multi-AZ.
- Si l'instance de base de données RDS for SQL Server est redémarrée alors que la procédure stockée `rds_tlog_backup_copy_to_S3` est en cours d'exécution, la tâche redémarre automatiquement depuis le début quand l'instance de base de données est remise en ligne. Toutes les sauvegardes des journaux de transactions qui ont été copiées dans le compartiment Amazon S3 pendant l'exécution de la tâche avant le redémarrage seront remplacées.
- Les bases de données système Microsoft SQL Server et la base de données `RDSAdmin` ne peuvent pas être configurées pour accéder aux sauvegardes des journaux de transactions.
- La copie vers des compartiments chiffrés par SSE-KMS n'est pas prise en charge.

Configuration de l'accès aux sauvegardes des journaux de transactions

Pour configurer l'accès aux sauvegardes des journaux de transactions, complétez la liste des exigences de la section Prérequis, puis exécutez la procédure stockée `rds_tlog_copy_setup`. La procédure permettra d'accéder à la fonctionnalité de sauvegarde des journaux de transactions au niveau de l'instance de base de données. Vous n'avez pas besoin de l'exécuter pour chaque base de données individuelle sur l'instance de base de données.

> **Important**
>
> L'utilisateur de base de données doit disposer du `db_owner` rôle dans le SQL serveur de chaque base de données pour configurer et utiliser la fonctionnalité d'accès aux sauvegardes du journal des transactions.

Example d'utilisation :

```
exec msdb.dbo.rds_tlog_copy_setup
@target_s3_arn='arn:aws:s3:::amzn-s3-demo-bucket/myfolder';
```

Les paramètres suivants sont obligatoires :

- `@target_s3_arn`— Le ARN compartiment Amazon S3 cible dans lequel copier les fichiers de sauvegarde du journal des transactions.

Example de définition d'un compartiment cible Amazon S3 :

```
exec msdb.dbo.rds_tlog_copy_setup @target_s3_arn='arn:aws:s3:::amzn-s3-demo-logging-bucket/mytestdb1';
```

Pour valider la configuration, appelez la procédure stockée `rds_show_configuration`.

Example de validation de la configuration :

```
exec rdsadmin.dbo.rds_show_configuration @name='target_s3_arn_for_tlog_copy';
```

Pour modifier l'accès aux sauvegardes des journaux de transactions afin de pointer vers un autre compartiment Amazon S3, vous pouvez consulter la valeur actuelle du compartiment Amazon S3 et réexécuter la procédure stockée `rds_tlog_copy_setup` en utilisant une nouvelle valeur pour `@target_s3_arn`.

Example d'affichage du compartiment Amazon S3 existant configuré pour accéder aux sauvegardes des journaux de transactions

```
exec rdsadmin.dbo.rds_show_configuration @name='target_s3_arn_for_tlog_copy';
```

Example de mise à jour vers un nouveau compartiment Amazon S3 cible

```
exec msdb.dbo.rds_tlog_copy_setup @target_s3_arn='arn:aws:s3:::amzn-s3-demo-logging-bucket1/mynewfolder';
```

Répertorier les sauvegardes disponibles des journaux de transactions

Avec RDS for SQL Server, les sauvegardes du journal des transactions sont automatiquement activées pour les bases de données configurées pour utiliser le modèle de restauration complète et une durée de conservation des sauvegardes d'instance de base de données définie sur un ou

plusieurs jours. En activant l'accès aux sauvegardes des journaux de transactions, vous pouvez copier jusqu'à sept jours de ces sauvegardes dans votre compartiment Amazon S3.

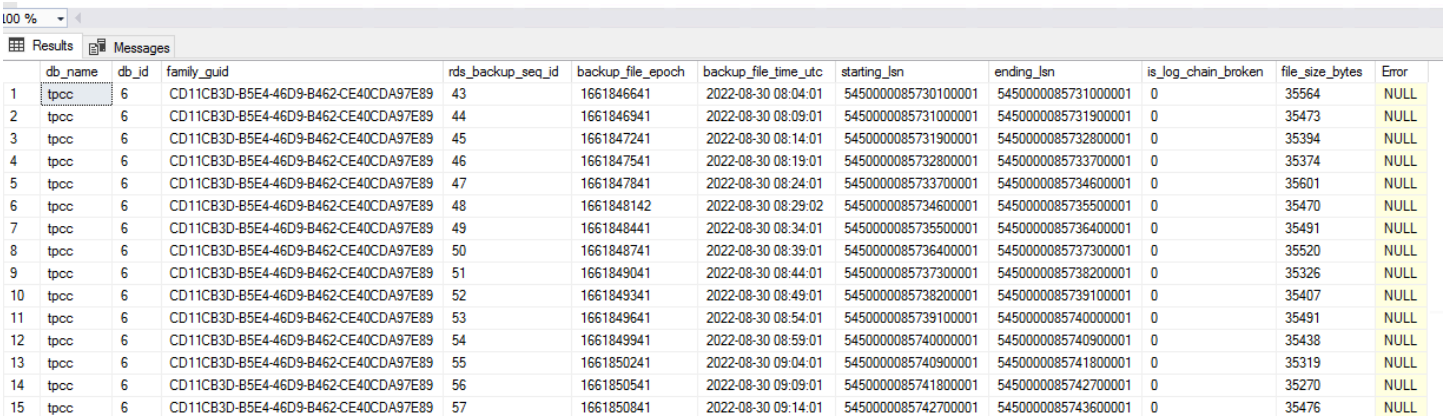
Une fois que vous avez activé l'accès aux sauvegardes des journaux de transactions, vous pouvez commencer à l'utiliser pour répertorier et copier les fichiers de sauvegarde des journaux de transactions disponibles.

Liste des sauvegardes des journaux de transactions

Pour répertorier toutes les sauvegardes des journaux de transactions disponibles pour une base de données individuelle, appelez la fonction `rds_fn_list_tlog_backup_metadata`. Vous pouvez utiliser une clause `ORDER BY` ou `WHERE` lorsque vous appelez la fonction.

Example visant à répertorier et filtrer les fichiers de sauvegarde des journaux de transactions disponibles

```
SELECT * from msdb.dbo.rds_fn_list_tlog_backup_metadata('mydatabasename');
SELECT * from msdb.dbo.rds_fn_list_tlog_backup_metadata('mydatabasename') WHERE
 rds_backup_seq_id = 3507;
SELECT * from msdb.dbo.rds_fn_list_tlog_backup_metadata('mydatabasename') WHERE
 backup_file_time_utc > '2022-09-15 20:44:01' ORDER BY backup_file_time_utc DESC;
```



100 %

Results Messages

	db_name	db_id	family_guid	rds_backup_seq_id	backup_file_epoch	backup_file_time_utc	starting_lsn	ending_lsn	is_log_chain_broken	file_size_bytes	Error
1	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	43	1661846641	2022-08-30 08:04:01	5450000085730100001	5450000085731000001	0	35564	NULL
2	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	44	1661846941	2022-08-30 08:09:01	5450000085731000001	5450000085731900001	0	35473	NULL
3	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	45	1661847241	2022-08-30 08:14:01	5450000085731900001	5450000085732800001	0	35394	NULL
4	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	46	1661847541	2022-08-30 08:19:01	5450000085732800001	5450000085733700001	0	35374	NULL
5	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	47	1661847841	2022-08-30 08:24:01	5450000085733700001	5450000085734600001	0	35601	NULL
6	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	48	1661848142	2022-08-30 08:29:02	5450000085734600001	5450000085735500001	0	35470	NULL
7	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	49	1661848441	2022-08-30 08:34:01	5450000085735500001	5450000085736400001	0	35491	NULL
8	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	50	1661848741	2022-08-30 08:39:01	5450000085736400001	5450000085737300001	0	35520	NULL
9	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	51	1661849041	2022-08-30 08:44:01	5450000085737300001	5450000085738200001	0	35326	NULL
10	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	52	1661849341	2022-08-30 08:49:01	5450000085738200001	5450000085739100001	0	35407	NULL
11	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	53	1661849641	2022-08-30 08:54:01	5450000085739100001	5450000085740000001	0	35491	NULL
12	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	54	1661849941	2022-08-30 08:59:01	5450000085740000001	5450000085740900001	0	35438	NULL
13	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	55	1661850241	2022-08-30 09:04:01	5450000085740900001	5450000085741800001	0	35319	NULL
14	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	56	1661850541	2022-08-30 09:09:01	5450000085741800001	5450000085742700001	0	35270	NULL
15	tpcc	6	CD11CB3D-B5E4-46D9-B462-CE40CDA97E89	57	1661850841	2022-08-30 09:14:01	5450000085742700001	5450000085743600001	0	35476	NULL

La fonction `rds_fn_list_tlog_backup_metadata` renvoie la sortie suivante.

Nom de la colonne	Type de données	Description
db_name	sysname	Nom de base de données fourni pour lequel répertorier les sauvegardes des journaux de transactions.

Nom de la colonne	Type de données	Description
db_id	int	Identifiant de base de données interne pour le paramètre d'entrée db_name.
family_guid	uniqueidentifier	ID unique de la base de données d'origine à sa création. Cette valeur reste la même quand la base de données est restaurée, même avec un nom de base de données différent.
rds_backup_seq_id	int	ID RDS utilisé en interne pour conserver un numéro de séquence pour chaque fichier de sauvegarde du journal des transactions.
backup_file_epoch	bigint	Heure epoch à laquelle un fichier de sauvegarde de transactions a été généré.
backup_file_time_utc	datetime	La valeur UTC convertie dans le temps pour la backup_file_epoch valeur.
starting_lsn	numeric(25,0)	Numéro de séquence de journal du premier ou du plus ancien enregistrement de journal d'un fichier de sauvegarde de journaux de transactions.
ending_lsn	numeric(25,0)	Numéro de séquence de journal du dernier ou du prochain enregistrement de journal d'un fichier de sauvegarde des journaux de transactions.
is_log_chain_broken	bit	Valeur booléenne indiquant si la chaîne de journaux est interrompue entre le fichier de sauvegarde actuel des journaux de transactions et le fichier de sauvegarde précédent des journaux de transactions.
file_size_bytes	bigint	Taille de la sauvegarde transactionnelle définie en octets.

Nom de la colonne	Type de données	Description
`Error`	varchar(4000)	Message d'erreur si la fonction `rds_fn_list_tlog_backup_metadata` lève une exception. NULLs'il n'y a pas d'exception.

Copie des sauvegardes de journaux de transactions

Pour copier un ensemble de sauvegardes disponibles des journaux de transactions pour une base de données individuelle dans votre compartiment Amazon S3, appelez la procédure stockée `rds_tlog_backup_copy_to_S3`. La procédure stockée `rds_tlog_backup_copy_to_S3` lancera une nouvelle tâche pour copier les sauvegardes de journaux de transactions.

> **Note**
>
> La procédure stockée `rds_tlog_backup_copy_to_S3` copiera les sauvegardes de journaux de transactions sans les valider par rapport à l'attribut `is_log_chain_broken`. Pour cette raison, vous devez confirmer manuellement une chaîne de journaux ininterrompue avant d'exécuter la procédure stockée `rds_tlog_backup_copy_to_S3`. Pour une explication approfondie, consultez Validation de la chaîne de sauvegarde des journaux de transactions.

Example d'utilisation de la procédure stockée **`rds_tlog_backup_copy_to_S3`**

```
exec msdb.dbo.rds_tlog_backup_copy_to_S3
 @db_name='mydatabasename',
 [@kms_key_arn='arn:aws:kms:region:account-id:key/key-id'],
 [@backup_file_start_time='2022-09-01 01:00:15'],
 [@backup_file_end_time='2022-09-01 21:30:45'],
 [@starting_lsn=149000000112100001],
 [@ending_lsn=149000000120400001],
 [@rds_backup_starting_seq_id=5],
 [@rds_backup_ending_seq_id=10];
```

Les paramètres d'entrée suivants sont disponibles :

Paramètre	Description
`@db_name`	Nom de la base de données fournie pour laquelle copier les sauvegardes de journaux de transactions.
`@kms_key_arn`	Une KMS clé gérée par le client. Si vous chiffrez votre instance de base de données avec une KMS clé AWS gérée, vous devez créer une clé gérée par le client. Si vous chiffrez votre instance de base de données avec une clé gérée par le client, vous pouvez utiliser la même KMS cléARN.
`@backup_file_start_time`	L'UTChorodatage tel que fourni par la `[backup_file_time_utc]` colonne de la `rds_fn_list_tlog_backup_metadata` fonction.
`@backup_file_end_time`	L'UTChorodatage tel que fourni par la `[backup_file_time_utc]` colonne de la `rds_fn_list_tlog_backup_metadata` fonction.
`@starting_lsn`	Le numéro de séquence log (LSN) tel que fourni dans la `[starting_lsn]` colonne de la `rds_fn_list_tlog_backup_metadata` fonction
`@ending_lsn`	Le numéro de séquence du journal (LSN) tel qu'il est indiqué `[ending_lsn]` dans la colonne de la `rds_fn_list_tlog_backup_metadata` fonction.
`@rds_backup_starting_seq_id`	ID de séquence tel que fourni dans la colonne `[rds_backup_seq_id]` de la fonction `rds_fn_list_tlog_backup_metadata`
`@rds_backup_ending_seq_id`	ID de séquence tel que fourni dans la colonne `[rds_backup_seq_id]` de la fonction `rds_fn_list_tlog_backup_metadata`

Vous pouvez spécifier un ensemble de paramètres d'heure ou LSN d'ID de séquence. Un seul ensemble de paramètres est requis.

Vous pouvez également spécifier un seul paramètre dans l'un quelconque des ensembles. Par exemple, en fournissant une valeur uniquement pour le paramètre `backup_file_end_time`, tous les fichiers de sauvegarde des journaux de transactions disponibles avant cette date, dans la limite de sept jours, seront copiés dans votre compartiment Amazon S3.

Les combinaisons de paramètres d'entrée valides pour la procédure stockée `rds_tlog_backup_copy_to_S3` sont fournies ci-dessous.

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name = 'testdb1', @backup_file_start_time='2022-08-23 00:00:00', @backup_file_end_time='2022-08-30 00:00:00';`	Copie les sauvegardes des journaux de transactions des sept derniers jours et se situe dans la plage fournie `backup_file_start_time` et `backup_file_end_time` . Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions qui ont été générées entre le 23/08/2022 à 00:00:00 et le 30/08/2022 à 00:00:00.
`exec msdb.dbo.rds_tlog_backup_copy_to_S3`	Copie les sauvegardes des journaux de transactions des

Paramètres fournis	Résultat attendu
`@db_name = 'testdb1', @backup_file_start_time='2022-08-23 00:00:00';`	sept derniers jours et commençant à partir de la valeur `backup_file_start_time` fournie. Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions depuis le 23/08/2022 à 00:00:00 jusqu'à la dernière sauvegarde des journaux de transactions.

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name = 'testdb1', @backup_file_end_time='2022-08-30 00:00:00';`	Copie les sauvegardes des journaux de transactions des sept derniers jours jusqu'à la valeur `backup_file_end_time` fournie. Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions depuis le 23/08/2022 à 00:00:00 jusqu'au 30/08/2022 à 00:00:00.

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @starting_lsn =14900000 00040007, @ending_lsn = 149000000 0050009;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours et qui se situent dans la plage fournie de `starting_lsn` et `ending_lsn` . Dans cet exemple, la procédure stockée copiera les sauvegardes du journal des transactions des sept derniers jours avec une LSN plage comprise entre 1490000000040007 et 149000000 0050009.

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @starting_lsn =1490000000040007;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours, en commençant par le paramètre `starting_lsn` fourni. Dans cet exemple, la procédure stockée copiera les sauvegardes du journal des transactions depuis LSN 1490000000040007 jusqu'à la dernière sauvegarde du journal des transactions.

Paramètres fournis	Résultat attendu	
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @ending_lsn=14900000000050009;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours, jusqu'au paramètre `ending_lsn` fourni. Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions commençant à partir des sept derniers jours jusqu'au LSN 149000000 0050009.	

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @rds_backup_starting_seq_id= 2000, @rds_backup_ending_seq_id= 5000;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours et qui se situent dans la plage fournie de `rds_backup_starting_seq_id` à `rds_backup_ending_seq_id` . Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions commençant à partir des sept derniers jours et figurant dans la plage fournie des identifiants de séquence de sauvegarde RDS, de seq_id 2000 à seq_id 5000.

Paramètres fournis	Résultat attendu	
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @rds_backup_starting_seq_id=2000;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours, en commençant par le paramètre `rds_backup_starting_seq_id` fourni. Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions commençant à seq_id 2000 jusqu'à la dernière sauvegarde des journaux de transactions.	

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @rds_backup_ending_seq_id= 5000;`	Copie les sauvegardes des journaux de transactions qui sont disponibles depuis les sept derniers jours, jusqu'au paramètre `rds_backup_ending_seq_id` fourni. Dans cet exemple, la procédure stockée copiera les sauvegardes des journaux de transactions commençant à partir des sept derniers jours, jusqu'à seq_id 5000.

Paramètres fournis	Résultat attendu
`exec msdb.dbo.rds_tlog_backup_copy_to_S3 @db_name='testdb1', @rds_backup_starting_seq_id= 2000; @rds_backup_ending_seq_id= 2000;`	Copie une sauvegarde unique des journaux de transactions avec l'ID `rds_backup_starting_seq_id` fourni, si elle est disponible au cours des sept derniers jours. Dans cet exemple, la procédure stockée copiera une sauvegarde unique des journaux de transactions dont le seq_id est 2000, si elle existe au cours des sept derniers jours.

Validation de la chaîne de sauvegarde des journaux de transactions

La conservation automatisée des sauvegardes doit être activée pour les bases de données configurées pour l'accès aux sauvegardes des journaux de transactions. La conservation automatisée des sauvegardes définit les bases de données sur l'instance de base de données sur le modèle de récupération FULL. Pour prendre en charge la restauration ponctuelle d'une base de données, évitez de modifier le modèle de récupération de base de données, car cela peut entraîner une rupture de la chaîne de journaux. Nous vous recommandons de conserver la base de données configurée sur le modèle de récupération FULL.

Pour valider manuellement la chaîne de journaux avant de copier les sauvegardes des journaux de transactions, appelez la fonction `rds_fn_list_tlog_backup_metadata` et passez en revue les valeurs de la colonne `is_log_chain_broken`. La valeur « 1 » indique que la chaîne de journaux a été interrompue entre la sauvegarde de journaux en cours et la sauvegarde de journaux précédente.

L'exemple suivant montre une chaîne de journaux interrompue dans la sortie de la procédure stockée `rds_fn_list_tlog_backup_metadata`.

rds_sequence_id	first_lsn	last_lsn	is_log_chain_broken
43	90023	90457	0
44	90457	90985	0
45	90987	92034	1

Dans une chaîne de log normale, la valeur du numéro de séquence journal (LSN) pour first_lsn pour un rds_sequence_id donné doit correspondre à la valeur de last_lsn dans le rds_sequence_id précédent. Dans l'image, un rds_sequence_id de 45 possède une valeur first_lsn de 90987, qui ne correspond pas à la valeur last_lsn de 90985 du rds_sequence_id 44 précédent.

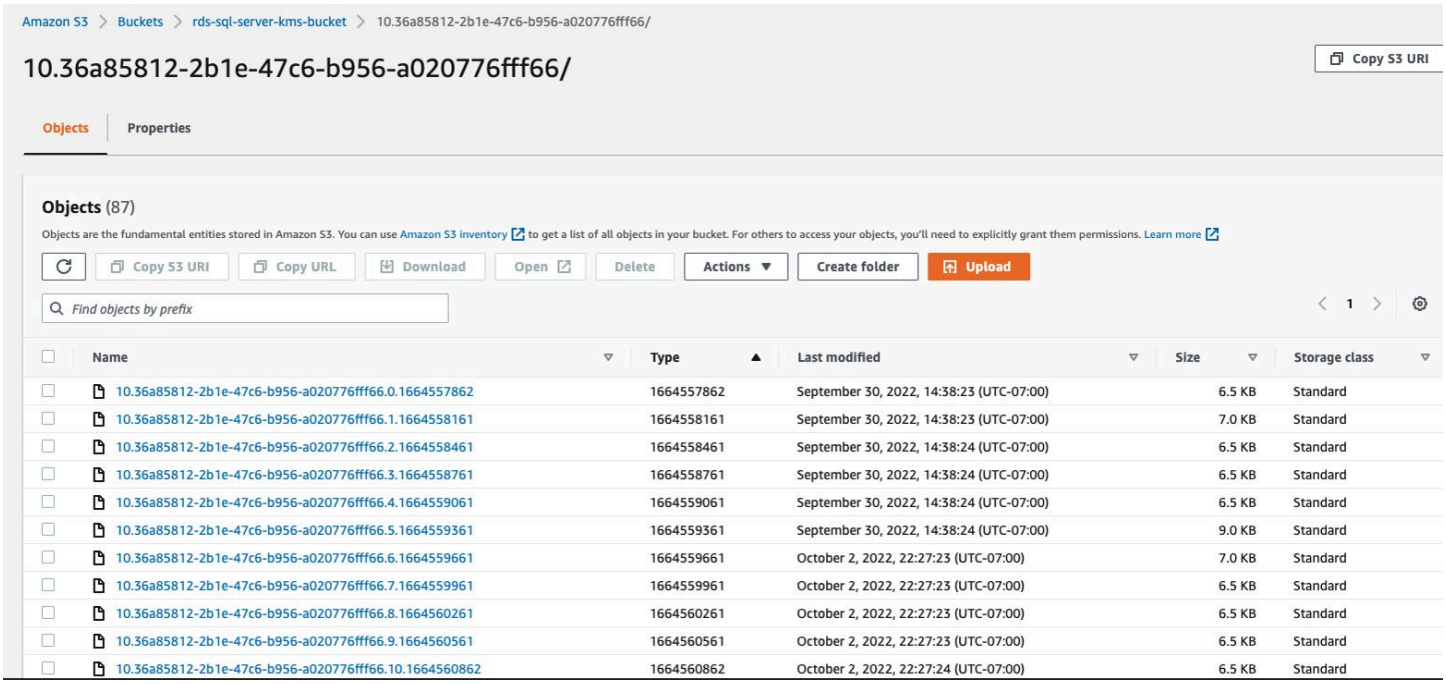
Pour plus d'informations sur l'architecture du journal des transactions SQL du serveur et les numéros de séquence des journaux, consultez la section Architecture logique du journal des transactions dans la documentation de Microsoft SQL Server.

Structure de dossiers et de fichiers d'un compartiment Amazon S3

Les sauvegardes des journaux de transactions présentent la structure standard et la convention de dénomination suivantes au sein d'un compartiment Amazon S3 :

- Un nouveau dossier est créé sous le chemin `target_s3_arn` de chaque base de données avec la structure de dénomination `{db_id}.{family_guid}`.
- Dans le dossier, les sauvegardes des journaux de transactions présentent la structure de noms de fichiers `{db_id}.{family_guid}.{rds_backup_seq_id}.{backup_file_epoch}`.
- Vous pouvez afficher les détails de `family_guid,db_id,rds_backup_seq_id and backup_file_epoch` avec la fonction `rds_fn_list_tlog_backup_metadata`.

L'exemple suivant montre la structure de dossiers et de fichiers d'un ensemble de sauvegardes des journaux de transactions dans un compartiment Amazon S3.



Suivi de l'état des tâches

Pour suivre le statut de vos tâches de copie, appelez la procédure stockée `rds_task_status`. Si vous ne fournissez pas de paramètre, la procédure stockée retourne l'état de toutes les tâches.

Example d'utilisation :

```
exec msdb.dbo.rds_task_status
  @db_name='database_name',
  @task_id=ID_number;
```

Les paramètres suivants sont facultatifs :

- `@db_name` – Nom de la base de données pour laquelle afficher l'état de la tâche.
- `@task_id` – ID de la tâche pour laquelle afficher l'état de tâche.

Example visant à répertorier le statut pour un ID de tâche spécifique :

```
exec msdb.dbo.rds_task_status @task_id=5;
```

Example visant à répertorier le statut pour une base de données et une tâche spécifiques :

```
exec msdb.dbo.rds_task_status@db_name='my_database',@task_id=5;
```

Example visant à répertorier toutes les tâches et leur statut pour une base de données spécifique :

```
exec msdb.dbo.rds_task_status @db_name='my_database';
```

Example visant à répertorier toutes les tâches et leur statut sur l'instance de base de données actuelle

```
exec msdb.dbo.rds_task_status;
```

Annulation d'une tâche

Pour annuler une tâche en cours d'exécution, appelez la procédure stockée `rds_cancel_task`.

Example d'utilisation :

```
exec msdb.dbo.rds_cancel_task @task_id=ID_number;
```

Les paramètres suivants sont obligatoires :

- `@task_id` – ID de la tâche à annuler. Vous pouvez consulter l'ID de la tâche en appelant la procédure stockée `rds_task_status`.

Pour plus d'informations sur l'affichage et l'annulation des tâches en cours, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Résolution des problèmes liés à l'accès aux sauvegardes des journaux de transactions

Les problèmes que vous pouvez rencontrer lorsque vous utilisez les procédures stockées pour accéder aux sauvegardes des journaux de transactions sont répertoriés ci-dessous.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	Les sauvegardes sont désactivées sur cette instance de base de données. Activez les sauvegardes des instances de base de données avec une conservation d'au moins « 1 » et réessayez.	Les sauvegardes automatisées ne sont pas activées pour l'instance de base de données.	La conservation des sauvegardes des instances de base de données doit être activée avec une durée de conservation d'au moins un jour. Pour plus d'informations sur l'activation des sauvegardes automatisées et la configuration de la conservation des sauvegardes, consultez Période de rétention des sauvegardes.
rds_tlog_copy_setup	Erreur lors de l'exécution de la procédure stockée rds_tlog_copy_setup. Reconnectez-vous au RDS terminal et réessayez.	Une erreur interne s’est produite.	Reconnectez-vous au RDS terminal et réexécutez la procédure `rds_tlog_copy_setup` stockée.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	L'exécution de la procédure stockée rds_tlog_backup_copy_setup dans une transaction n'est pas prise en charge. Vérifiez qu'aucune transaction n'est ouverte dans la session et réessayez.	La procédure stockée a été tentée dans le cadre d'une transaction en utilisant `BEGIN` et `END`.	Évitez d'utiliser `BEGIN` et `END` lors de l'exécution de la procédure stockée `rds_tlog_copy_setup` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	Le nom du compartiment S3 pour le paramètre d'entrée `@target_s3_arn` doit contenir au moins un caractère autre qu'un espace.	Une valeur incorrecte a été fournie pour le paramètre d'entrée `@target_s3_arn` .	Assurez-vous que le paramètre d'entrée `@target_s3_arn` spécifie le compartiment Amazon S3 completARN.
rds_tlog_copy_setup	L'option `SQLSERVER_BACKUP_RESTORE` n'est pas activée ou est en cours d'activation. Activez l'option ou réessayez ultérieurement.	L'option `SQLSERVER_BACKUP_RESTORE` n'est pas activée sur l'instance de base de données ou a simplement été activée et est en attente d'une activation interne.	Activez l'option `SQLSERVER_BACKUP_RESTORE` comme indiqué dans la section Exigences. Attendez quelques minutes, puis réexécutez la procédure stockée `rds_tlog_copy_setup` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	L'ARN S3 cible pour le paramètre d'entrée `@target_s3_arn` ne peut pas être vide ou null.	Une valeur `NULL` a été fournie pour le paramètre d'entrée `@target_s3_arn` , ou cette valeur n'a pas été fournie.	Assurez-vous que le paramètre d'entrée `@target_s3_arn` spécifie le compartiment Amazon S3 completARN.
rds_tlog_copy_setup	L'ARN S3 cible pour le paramètre d'entrée `@target_s3_arn` doit commencer par arn:aws.	Le paramètre d'entrée `@target_s3_arn` a été fourni sans `arn:aws` sur le devant.	Assurez-vous que le paramètre d'entrée `@target_s3_arn` spécifie le compartiment Amazon S3 completARN.
rds_tlog_copy_setup	La cible S3 ARN est déjà définie sur la valeur fournie.	La procédure `rds_tlog_copy_setup` stockée s'est précédemment exécutée et a été configurée avec un compartiment Amazon S3ARN.	Pour modifier la valeur du compartiment Amazon S3 afin d'accéder aux sauvegardes des journaux de transactions, fournissez un autre `target S3 ARN`.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	Impossible de générer des informations d'identification pour activer l'accès aux sauvegardes des journaux de transactions. Vérifiez le chemin S3 ARN fourni avec `rds_tlog_copy_setup` , puis réessayez ultérieurement.	Une erreur non spécifiée s'est produite lors de la génération des informations d'identification pour permettre l'accès aux sauvegardes des journaux de transactions.	Passez en revue votre configuration d'installation et réessayez.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_copy_setup	Vous ne pouvez pas exécuter la procédure stockée rds_tlog_copy_setup tant que des tâches sont en attente. Attendez que les tâches en attente soient terminées et réessayez.	Seules deux tâches peuvent être exécutées à la fois. Certaines tâches sont en attente d'achèvement.	Affichez les tâches en attente et attendez qu'elles se terminent . Pour plus d'informations sur la surveillance du statut des tâches, consultez Suivi de l'état des tâches.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Une tâche de copie du fichier de sauvegarde T-log a déjà été lancée pour la base de données : %s avec l'ID de tâche : %d. Réessayez ultérieurement.	Une seule tâche de copie peut être exécutée à la fois pour une base de données spécifiée. Une tâche de copie est en attente d'achèvement.	Affichez les tâches en attente et attendez qu'elles se terminent . Pour plus d'informations sur la surveillance du statut des tâches, consultez Suivi de l'état des tâches.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Au moins l'un de ces trois ensembles de paramètres doit être fourni. SET-1 : (@backup_file_start_time, @backup_file_end_time) | SET -2 : (@starting_lsn, @ending_lsn) | SET -3 : (@rds_backup_starting_seq_id, @rds_backup_ending_seq_id)	Aucun de ces trois ensembles de paramètres n'a été fourni, ou un paramètre obligatoire est manquant dans un jeu de paramètres fourni.	Vous pouvez spécifier les paramètres d'heure, de LSN ou d'ID de séquence. Un de ces trois ensembles de paramètres est requis. Pour plus d'informations sur les paramètres requis, consultez Copie des sauvegardes de journaux de transactions.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Les sauvegardes sont désactivées sur cette instance. Activez les sauvegardes et réessayez dans un certain temps.	Les sauvegardes automatisées ne sont pas activées pour l'instance de base de données.	Pour plus d'informations sur l'activation des sauvegardes automatisées et la configuration de la conservation des sauvegardes, consultez Période de rétention des sauvegardes.
rds_tlog_backup_copy_to_S3	Impossible de trouver la base de données spécifiée %s.	La valeur fournie pour le paramètre d'entrée `@db_name` ne correspond pas à un nom de base de données sur l'instance de base de données.	Utilisez le nom de base de données correct. Pour répertorier toutes les bases de données par nom, exécutez `SELECT * from sys.databases`

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Impossible d'exécuter la procédure stockée RDS_Tlog_Backup_Copy_to_S3 pour les bases de données du système du serveur ou pour la base de données rdsadmin. SQL	La valeur fournie pour le paramètre d'entrée `@db_name` correspond au nom de la base de données SQL du système serveur ou à la RDSAdmin base de données.	Les bases de données suivantes ne peuvent pas être utilisées pour accéder aux sauvegardes des journaux de transactions : `master, model, msdb, tempdb, RDSAdmin.`
rds_tlog_backup_copy_to_S3	Le nom de la base de données pour le paramètre d'entrée @db_name ne peut pas être vide ou null.	La valeur fournie pour le paramètre d'entrée `@db_name` était vide ou `NULL`.	Utilisez le nom de base de données correct. Pour répertorier toutes les bases de données par nom, exécutez `SELECT * from sys.databases`

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	La période de conservation des sauvegardes des instances de base de données doit être définie sur au moins 1 pour exécuter la procédure stockée rds_tlog_backup_copy_setup.	Les sauvegardes automatisées ne sont pas activées pour l'instance de base de données.	Pour plus d'informations sur l'activation des sauvegardes automatisées et la configuration de la conservation des sauvegardes, consultez Période de rétention des sauvegardes.
rds_tlog_backup_copy_to_S3	Erreur lors de l'exécution de la procédure stockée rds_tlog_backup_copy_to_S3. Reconnectez-vous au RDS terminal et réessayez.	Une erreur interne s’est produite.	Reconnectez-vous au RDS terminal et réexécutez la procédure `rds_tlog_backup_copy_to_S3` stockée.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Seul l'un de ces trois ensembles de paramètres peut être fourni. SET-1 : (@backup_file_start_time, @backup_file_end_time) | SET -2 : (@starting_lsn, @ending_lsn) | SET -3 : (@rds_backup_starting_seq_id, @rds_backup_ending_seq_id)	Plusieurs ensembles de paramètres ont été fournis.	Vous pouvez spécifier les paramètres d'heure, de LSN ou d'ID de séquence. Un de ces trois ensembles de paramètres est requis. Pour plus d'informations sur les paramètres requis, consultez Copie des sauvegardes de journaux de transactions.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	L'exécution de la procédure stockée rds_tlog_backup_copy_to_S3 dans une transaction n'est pas prise en charge. Vérifiez qu'aucune transaction n'est ouverte dans la session et réessayez.	La procédure stockée a été tentée dans le cadre d'une transaction en utilisant `BEGIN` et `END`.	Évitez d'utiliser `BEGIN` et `END` lors de l'exécution de la procédure stockée `rds_tlog_backup_copy_to_S3` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Les paramètres fournis ne sont pas inclus dans la période de conservation des journaux de sauvegarde des transactions. Pour répertorier les fichiers de sauvegarde des journaux de transactions disponibles, exécutez la fonction rds_fn_list_tlog_backup_metadata.	Aucune sauvegarde des journaux de transactions n'est disponible pour les paramètres d'entrée fournis qui correspondent à la fenêtre de conservation des copies.	Réessayez avec un ensemble de paramètres valide. Pour plus d'informations sur les paramètres requis, consultez Copie des sauvegardes de journaux de transactions.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Une erreur d'autorisations s'est produite lors du traitement de la demande. Assurez-vous que le compartiment se trouve dans le même compte et la même région que l'instance de base de données, et confirmez les autorisations de la politique de compartiment S3 par rapport au modèle figurant dans la documentation publique.	Un problème a été détecté avec le compartiment S3 fourni ou ses autorisations de politique.	Confirmez que votre configuration d'accès aux sauvegardes des journaux de transactions est correcte. Pour plus d'informations sur les exigences de configuration de votre compartiment S3, consultez Prérequis.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	L'exécution de la procédure `rds_tlog_backup_copy_to_S3` stockée sur une instance de réplication en RDS lecture n'est pas autorisée.	La procédure stockée a été tentée sur une instance de réplique en RDS lecture.	Connectez-vous à l'instance de base de données RDS principale pour exécuter la procédure `rds_tlog_backup_copy_to_S3` stockée.
rds_tlog_backup_copy_to_S3	La valeur LSN pour le paramètre d'entrée `@starting_lsn` doit être inférieure à`@ending_lsn` .	La valeur fournie pour le paramètre d'entrée `@starting_lsn` était supérieure à la valeur fournie pour le paramètre d'entrée `@ending_lsn` .	Veillez à ce que la valeur fournie pour le paramètre d'entrée `@starting_lsn` soit inférieure à la valeur fournie pour le paramètre d'entrée `@ending_lsn` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	La procédure stockée `rds_tlog_backup_copy_to_S3` ne peut être exécutée que par les membres du rôle `db_owner` dans la base de données source.	Le rôle `db_owner` n'a pas été accordé au compte qui tente d'exécuter la procédure stockée `rds_tlog_backup_copy_to_S3` sur la base de données `db_name` fournie.	Veillez à ce que le compte exécutant la procédure stockée soit autorisé avec le rôle `db_owner` pour la base de données `db_name` fournie.
rds_tlog_backup_copy_to_S3	L'ID de séquence pour le paramètre d'entrée `@rds_backup_starting_seq_id` doit être inférieur ou égal à `@rds_backup_ending_seq_id` .	La valeur fournie pour le paramètre d'entrée `@rds_backup_starting_seq_id` était supérieure à la valeur fournie pour le paramètre d'entrée `@rds_backup_ending_seq_id` .	Veillez à ce que la valeur fournie pour le paramètre d'entrée `@rds_backup_starting_seq_id` soit inférieure à la valeur fournie pour le paramètre d'entrée `@rds_backup_ending_seq_id` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	L'RESTORE option SQLSERVER BACKUP _ _ n'est pas activée ou est en cours d'activation. Activez l'option ou réessayez ultérieurement.	L'option SQLSERVER _BACKUP_RESTORE n'est pas activée sur l'instance de base de données ou a simplement été activée et est en attente d'une activation interne.	Activez l'option SQLSERVER _BACKUP_RESTORE comme indiqué dans la section Exigences. Attendez quelques minutes, puis réexécutez la procédure stockée `rds_tlog_backup_copy_to_S3` .
rds_tlog_backup_copy_to_S3	L'heure de début du paramètre d'entrée `@backup_file_start_time` doit être inférieure à `@backup_file_end_time` .	La valeur fournie pour le paramètre d'entrée `@backup_file_start_time` était supérieure à la valeur fournie pour le paramètre d'entrée `@backup_file_end_time` .	Veillez à ce que la valeur fournie pour le paramètre d'entrée `@backup_file_start_time` soit inférieure à la valeur fournie pour le paramètre d'entrée `@backup_file_end_time` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Nous n'avons pas pu traiter la demande en raison d'un manque d'accès. Vérifiez votre configuration et vos autorisations pour cette fonctionnalité.	Il se peut qu'il y ait un problème avec les autorisations du compartiment Amazon S3, ou que le compartiment Amazon S3 fourni se trouve dans un autre compte ou une autre région.	Assurez-vous que les autorisations de la politique du compartiment Amazon S3 sont autorisées pour autoriser RDS l'accès. Veillez à ce que le compartiment Amazon S3 se trouve dans le même compte et la même région que l'instance de base de données.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Vous ne pouvez pas fournir de KMS clé ARN comme paramètre d'entrée à la procédure stockée pour les instances qui ne sont pas cryptées au niveau du stockage.	Quand le chiffrement de stockage n'est pas activé sur l'instance de base de données, le paramètre d'entrée `@kms_key_arn` ne doit pas être fourni.	Ne fournissez aucun paramètre d'entrée pour `@kms_key_arn` .

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Vous devez fournir une KMS clé ARN comme paramètre d'entrée à la procédure stockée pour le stockage des instances chiffrées.	Quand le chiffrement de stockage est activé sur l'instance de base de données, le paramètre d'entrée `@kms_key_arn` doit être fourni.	Fournissez un paramètre d'entrée pour `@kms_key_arn` avec une valeur correspondant à celle ARN du compartiment Amazon S3 à utiliser pour les sauvegardes du journal des transactions.

Procédure stockée	Message d'erreur	Problème	Suggestions de dépannage
rds_tlog_backup_copy_to_S3	Vous devez exécuter la procédure stockée `rds_tlog_copy_setup` et définir le `@target_s3_arn` , avant d'exécuter la procédure stockée `rds_tlog_backup_copy_to_S3` .	La procédure de configuration de l'accès aux sauvegardes des journaux de transactions n'était pas terminée avant la tentative d'exécution de la procédure stockée `rds_tlog_backup_copy_to_S3` .	Exécutez la procédure stockée `rds_tlog_copy_setup` avant d'exécuter la procédure stockée `rds_tlog_backup_copy_to_S3` . Pour plus d'informations sur l'exécution de la procédure de configuration pour accéder aux sauvegardes des journaux de transactions, consultez Configuration de l'accès aux sauvegardes des journaux de transactions.

Options pour le moteur de base de données Microsoft SQL Server

Dans cette section, vous trouverez des descriptions pour les options disponibles pour les instances Amazon RDS exécutant le moteur de base de données Microsoft SQL Server. Pour activer ces options, vous les ajoutez à un groupe d'options, puis associer celui-ci à votre instance de base de données. Pour plus d'informations, consultez Utilisation de groupes d'options.

Si vous recherchez des fonctions facultatives qui ne sont pas ajoutées via les groupes d'options RDS (par exemple, SSL, authentification Microsoft Windows et intégration Amazon S3), consultez Fonctionnalités supplémentaires pour Microsoft SQL Server sur Amazon RDS.

Amazon RDS prend en charge les options suivantes pour les instances de base de données Microsoft SQL Server.

Option	ID d'option	Editions de moteur
Serveurs liés avec Oracle OLEDB	`OLEDB_ORACLE`	SQL Server Enterprise Edition SQL Server Standard Edition
Sauvegarde et restauration natives	`SQLSERVER_BACKUP_RESTORE`	SQL Server Enterprise Edition SQL Server Standard Edition SQL Server Web Edition SQL Server Express Edition
Transparent Data Encryption	`TRANSPARENT_DATA_ENCRYPTION` (console RDS)	SQL Server Édition Enterprise 2016—2022 Édition standard de SQL Server 2022

Option	ID d'option	Editions de moteur
	TDE(AWS CLI et API RDS)	
SQLAudit du serveur	SQLSERVER_AUDIT	Dans RDS, à partir de SQL Server 2016, toutes les éditions de SQL Server prennent en charge les audits au niveau du serveur, et l'édition Enterprise prend également en charge les audits au niveau de la base de données. À partir de SQL Server SQL Server 2016 (13.x) SP1, toutes les éditions prennent en charge les audits au niveau du serveur et au niveau de la base de données. Pour de plus amples informations, consultez SQL Server Audit (moteur de base de données) dans la documentation SQL Server.

Option	ID d'option	Editions de moteur
SQLServices d'analyse de serveurs	SSAS	SQL Server Enterprise Edition SQL Server Standard Edition
SQL Server Integration Services	SSIS	SQL Server Enterprise Edition SQL Server Standard Edition
SQLServices de rapports sur les serveurs	SSRS	SQL Server Enterprise Edition SQL Server Standard Edition
Microsoft Distributed Transaction Coordinator	MSDTC	Dans RDS, à partir de SQL Server 2016, toutes les éditions de SQL Server prennent en charge les transactions distribuées.

Liste des options disponibles pour les versions et éditions de SQL Server

Vous pouvez utiliser la `describe-option-group-options` AWS CLI commande pour répertorier les options disponibles pour les versions et éditions de SQL Server, ainsi que les paramètres de ces options.

L'exemple suivant illustre les options et les paramètres d'options pour SQL Server 2019 Enterprise Edition. L'option `--engine-name` est obligatoire.

```
aws rds describe-option-group-options --engine-name sqlserver-ee --major-engine-version
 15.00
```

La sortie est semblable à la suivante :

```
{
    "OptionGroupOptions": [
        {
            "Name": "MSDTC",
            "Description": "Microsoft Distributed Transaction Coordinator",
            "EngineName": "sqlserver-ee",
            "MajorEngineVersion": "15.00",
            "MinimumRequiredMinorEngineVersion": "4043.16.v1",
            "PortRequired": true,
            "DefaultPort": 5000,
            "OptionsDependedOn": [],
            "OptionsConflictsWith": [],
            "Persistent": false,
            "Permanent": false,
            "RequiresAutoMinorEngineVersionUpgrade": false,
            "VpcOnly": false,
            "OptionGroupOptionSettings": [
                {
                    "SettingName": "ENABLE_SNA_LU",
                    "SettingDescription": "Enable support for SNA LU protocol",
                    "DefaultValue": "true",
                    "ApplyType": "DYNAMIC",
                    "AllowedValues": "true,false",
                    "IsModifiable": true,
                    "IsRequired": false,
                    "MinimumEngineVersionPerAllowedValue": []
                },
        ...

        {
            "Name": "TDE",
            "Description": "SQL Server - Transparent Data Encryption",
            "EngineName": "sqlserver-ee",
            "MajorEngineVersion": "15.00",
            "MinimumRequiredMinorEngineVersion": "4043.16.v1",
            "PortRequired": false,
            "OptionsDependedOn": [],
            "OptionsConflictsWith": [],
```

```
            "Persistent": true,
            "Permanent": false,
            "RequiresAutoMinorEngineVersionUpgrade": false,
            "VpcOnly": false,
            "OptionGroupOptionSettings": []
        }
    ]
}
```

Prise en charge des serveurs liés avec Oracle OLEDB dans Amazon RDS for SQL Server

Les serveurs liés au fournisseur Oracle pour OLEDB sur RDS for SQL Server vous permettent d'accéder à des sources de données externes sur une base de données Oracle. Vous pouvez lire des données provenant de sources de données Oracle distantes et exécuter des commandes sur des serveurs de base de données Oracle distants en dehors de votre instance de base de données RDS for SQL Server. Grâce aux serveurs liés avec Oracle OLEDB, vous pouvez :

- Accéder directement à des sources de données autres que SQL Server
- Exécuter des requêtes sur diverses sources de données Oracle à l'aide de la même requête sans déplacer les données
- Émettre des requêtes, des mises à jour, des commandes et des transactions distribuées sur des sources de données au sein d'un écosystème d'entreprise
- Intégrer des connexions à une base de données Oracle depuis la suite Microsoft Business Intelligence (SSIS, SSRS, SSAS)
- Migrer d'une base de données Oracle vers RDS for SQL Server

Vous pouvez activer un ou plusieurs serveurs liés pour Oracle sur une instance de base de données RDS for SQL Server existante ou nouvelle. Vous pouvez ensuite intégrer des sources de données Oracle externes à votre instance de base de données.

Table des matières

- Versions et régions prises en charge
- Limitations et recommandations
- Activation de serveurs liés avec Oracle
  - Création du groupe d'options pour OLEDB_ORACLE
  - Ajout de l'option OLEDB_ORACLE au groupe d'options
  - Modification de l'option de OLEDB_ORACLE version vers une autre version
  - Association du groupe d'options à votre instance de base de données
- Modification des propriétés du fournisseur OLEDB
- Modification des propriétés du pilote OLEDB
- Désactivation de serveurs liés avec Oracle

Versions et régions prises en charge

RDS for SQL Server prend en charge les serveurs liés avec Oracle OLEDB dans toutes les régions pour SQL Server éditions Standard et Enterprise sur les versions suivantes :

- SQL Server 2022, toutes les versions
- SQL Server 2019, toutes les versions
- SQL Server 2017, toutes les versions

Les serveurs liés avec Oracle OLEDB sont pris en charge pour les versions Oracle Database suivantes :

- Oracle Database 21c, toutes les versions
- Oracle Database 19c, toutes les versions
- Oracle Database 18c, toutes les versions

Les serveurs liés à Oracle OLEDB sont pris en charge pour les versions de pilote Oracle OLEDB suivantes :

- 21,7
- 21,16

Limitations et recommandations

Gardez à l'esprit les limites et recommandations suivantes, qui s'appliquent aux serveurs liés avec Oracle OLEDB :

- Autorisez le trafic réseau en ajoutant le port TCP applicable dans le groupe de sécurité pour chaque instance de base de données RDS for SQL Server. Par exemple, si vous configurez un serveur lié entre une instance de base de données EC2 Oracle et une instance de base de données RDS pour SQL Server, vous devez autoriser le trafic provenant de l'adresse IP de l'instance de base de données EC2 Oracle. Vous devez également autoriser le trafic sur le port utilisé par SQL Server pour écouter les communications de base de données. Pour plus d'informations sur les groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité.
- Redémarrez l'instance de base de données RDS for SQL Server après avoir activé, désactivé ou modifié l'option `OLEDB_ORACLE` dans votre groupe d'options. Le statut du groupe d'options affiche

`pending_reboot` pour ces événements et est obligatoire. Pour les instances RDS for SQL Server Multi-AZ dont l'option de mise en miroir est activée AlwaysOn ou dont l'option de mise en miroir est activée, un basculement est attendu lorsque l'instance est redémarrée après la création ou la restauration de la nouvelle instance.

- Seule une authentification simple est prise en charge avec un nom d'utilisateur et un mot de passe pour la source de données Oracle.
- Les pilotes Open Database Connectivity (ODBC) ne sont pas pris en charge. Seules les versions du pilote OLEDB répertoriées ci-dessus sont prises en charge.
- Les transactions distribuées (XA) sont prises en charge. Pour activer les transactions distribuées, activez l'option `MSDTC` dans le groupe d'options pour votre instance de base de données et veillez à ce que les transactions XA soient activées. Pour de plus amples informations, veuillez consulter Support pour Microsoft Distributed Transaction Coordinator dans RDS for SQL Server.
- La création de noms de sources de données (DSNs) à utiliser comme raccourci pour une chaîne de connexion n'est pas prise en charge.
- Le suivi des pilotes OLEDB n'est pas pris en charge. Vous pouvez utiliser les événements étendus SQL Server pour suivre les événements OLEDB. Pour plus d'informations, consultez Set up Extended Events in RDS for SQL Server (Configuration d'événements étendus dans RDS for SQL Server).
- L'accès au dossier des catalogues d'un serveur lié Oracle n'est pas pris en charge avec SQL Server Management Studio (SSMS).

Activation de serveurs liés avec Oracle

Activez les serveurs liés avec Oracle en ajoutant l'option `OLEDB_ORACLE` à votre instance de base de données RDS for SQL Server. Utilisez la procédure suivante :

1. Créez un groupe d'options ou choisissez un groupe d'options existant.
2. Ajoutez l'option `OLEDB_ORACLE` au groupe d'options.
3. Choisissez une version du pilote OLEDB à utiliser.
4. Associez le groupe d'options à l'instance de base de données.
5. Redémarrez l'instance de la base de données.

Création du groupe d'options pour OLEDB_ORACLE

Pour utiliser des serveurs liés avec Oracle, créez ou modifiez un groupe d'options correspondant à l'édition et à la version de SQL Server de l'instance de base de données que vous prévoyez d'utiliser. Pour terminer cette procédure, utilisez le AWS Management Console ou le AWS CLI.

console

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2019.

Pour créer le groupe d'options

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
    a. Dans Nom, entrez un nom unique au sein de votre AWS compte pour le groupe d'options, tel que**`oracle-oledb-se-2019`**. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
    b. Pour Description, saisissez une brève description du groupe d'options, par exemple **`OLEDB_ORACLE option group for SQL Server SE 2019`**. La description est utilisée à des fins d'affichage.
    c. Pour Moteur, choisissez sqlserver-se.
    d. Pour Major engine version (Version majeure du moteur), choisissez 15.00.
5. Choisissez Créer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2019.

Pour créer le groupe d'options

- Exécutez une des commandes suivantes :

    Example

    Pour LinuxmacOS, ou Unix :

```
aws rds create-option-group \
    --option-group-name oracle-oledb-se-2019 \
    --engine-name sqlserver-se \
    --major-engine-version 15.00 \
    --option-group-description "OLEDB_ORACLE option group for SQL Server SE 2019"
```

Dans Windows :

```
aws rds create-option-group ^
    --option-group-name oracle-oledb-se-2019 ^
    --engine-name sqlserver-se ^
    --major-engine-version 15.00 ^
    --option-group-description "OLEDB_ORACLE option group for SQL Server SE 2019"
```

Ajout de l'option **OLEDB_ORACLE** au groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'`OLEDB_ORACLE`option à votre groupe d'options.

console

Pour ajouter l'option OLEDB_ORACLE

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options que vous venez de créer, à savoir oracle-oledb-se-2019 dans cet exemple.
4. Sélectionnez Ajouter une option.
5. Sous Option details (Détails de l'option), choisissez OLEDB_ORACLE pour Option name (Nom de l'option).
6. Sous Version, choisissez la version du pilote Oracle OLEDB que vous souhaitez installer.
7. Sous Scheduling (Planification), choisissez si vous souhaitez ajouter l'option immédiatement ou lors du créneau de maintenance suivant.
8. Sélectionnez Ajouter une option.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour ajouter l'option OLEDB_ORACLE

- Ajoutez l'option `OLEDB_ORACLE` au groupe d'options.

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name oracle-oledb-se-2019 \
    --options OptionName=OLEDB_ORACLE, OptionVersion=21.16 \
    --apply-immediately
```

  Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name oracle-oledb-se-2019 ^
    --options OptionName=OLEDB_ORACLE, OptionVersion=21.16 ^
    --apply-immediately
```

Modification de l'option de **`OLEDB_ORACLE`** version vers une autre version

Pour modifier la version de l'`OLEDB_ORACLE`option vers une autre version, utilisez le AWS Management Console ou le AWS CLI.

console

Pour modifier l'option OLEDB_ORACLE

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'`OLEDB_ORACLE`option (oracle-oledb-se-2019 dans l'exemple précédent).
4. Choisissez Modify option (Modifier l'option).
5. Sous Option details (Détails de l'option), choisissez OLEDB_ORACLE pour Option name (Nom de l'option).

6. Sous Version, choisissez la version du pilote Oracle OLEDB que vous souhaitez utiliser.
7. Sous Planification, choisissez de modifier l'option immédiatement ou lors de la fenêtre de maintenance suivante.
8. Choisissez Modify option (Modifier l'option).

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour modifier la version de l'`OLEDB_ORACLE`option, utilisez la `rds add-option-to-option-group`AWS CLI commande avec le groupe d'options et la version d'option que vous souhaitez utiliser.

Pour modifier l'option OLEDB_ORACLE

- Example

  Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name oracle-oledb-se-2019 \
    --options OptionName=OLEDB_ORACLE, OptionVersion=21.7 \
    --apply-immediately
```

  Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name oracle-oledb-se-2019 ^
    --options OptionName=OLEDB_ORACLE, OptionVersion=21.7 ^
    --apply-immediately
```

Association du groupe d'options à votre instance de base de données

Pour associer le groupe d'`OLEDB_ORACLE`options et le groupe de paramètres à votre instance de base de données, utilisez le AWS Management Console ou AWS CLI

console

Pour terminer l'activation de serveurs liés pour Oracle, associez votre groupe d'options `OLEDB_ORACLE` à une instance de base de données nouvelle ou existante :

- Pour une nouvelle instance de base de données, associez-les lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-les en modifiant l'instance. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Vous pouvez associer le groupe d'options et le groupe de paramètres `OLEDB_ORACLE` à une instance de base de données nouvelle ou existante.

Pour créer une instance avec le groupe d'options et le groupe de paramètres **`OLEDB_ORACLE`**

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier mytestsqlserveroracleoledbinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 15.0.4236.7.v1 \
    --allocated-storage 100 \
    --manage-master-user-password \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --domain-iam-role-name my-directory-iam-role \
    --domain my-domain-id \
    --option-group-name oracle-oledb-se-2019 \
    --db-parameter-group-name my-parameter-group-name
```

  Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier mytestsqlserveroracleoledbinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
```

```
    --engine-version 15.0.4236.7.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --domain-iam-role-name my-directory-iam-role ^
    --domain my-domain-id ^
    --option-group-name oracle-oledb-se-2019 ^
    --db-parameter-group-name my-parameter-group-name
```

Pour modifier une instance et lui associer le groupe d'options **OLEDB_ORACLE**

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mytestsqlserveroracleoledbinstance \
    --option-group-name oracle-oledb-se-2019 \
    --db-parameter-group-name my-parameter-group-name \
    --apply-immediately
```

  Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mytestsqlserveroracleoledbinstance ^
    --option-group-name oracle-oledb-se-2019 ^
    --db-parameter-group-name my-parameter-group-name ^
    --apply-immediately
```

Modification des propriétés du fournisseur OLEDB

Vous pouvez afficher et modifier les propriétés du fournisseur OLEDB. Seul l'utilisateur `master` peut effectuer cette tâche. Tous les serveurs liés pour Oracle qui sont créés sur l'instance de base de données utilisent les mêmes propriétés de ce fournisseur OLEDB. Appelez la procédure stockée `sp_MSset_oledb_prop` pour modifier les propriétés du fournisseur OLEDB.

Pour modifier les propriétés du fournisseur OLEDB

```
USE [master]
GO
EXEC sp_MSset_oledb_prop N'OraOLEDB.Oracle', N'AllowInProcess', 1
EXEC sp_MSset_oledb_prop N'OraOLEDB.Oracle', N'DynamicParameters', 0
GO
```

Les propriétés suivantes peuvent être modifiées :

Nom de la propriété	Valeur recommandée (1 = Activé, 0 = Désactivé)	Description
`Dynamic parameter`	1	Autorise les espaces réservés SQL (représentés par « ? ») dans des requêtes paramétrées.
`Nested queries`	1	Autorise les instructions `SELECT` imbriquées dans la clause `FROM`, telles que des sous-requêtes.
`Level zero only`	0	Seules des interfaces OLEDB de niveau de base sont appelées par rapport au fournisseur.
`Allow inprocess`	1	Si cette option est activée, Microsoft SQL Server permet d'instancier le fournisseur en tant que serveur en cours de processus. Définissez cette propriété sur 1 pour utiliser des serveurs liés Oracle.
`Non transacted updates`	0	Si la valeur est différente de zéro, SQL Server autorise les mises à jour.
`Index as access path`	False	Si la valeur est différente de zéro, SQL Server tente d'utiliser les index du fournisseur pour récupérer des données.
`Disallow adhoc access`	False	Si cette option est définie, SQL Server n'autorise pas l'exécution de requêtes directes sur le fournisseur OLEDB. Cette option peut être cochée,

Nom de la propriété	Valeur recommandée (1 = Activé, 0 = Désactivé)	Description
		mais il est parfois approprié d'exécuter des requêtes directes.
`Supports LIKE operator`	1	Indique que le fournisseur prend en charge les requêtes utilisant le mot clé LIKE.

Modification des propriétés du pilote OLEDB

Vous pouvez afficher et modifier les propriétés du pilote OLEDB lors de la création d'un serveur lié pour Oracle. Seul l'utilisateur `master` peut effectuer cette tâche. Les propriétés du pilote définissent la manière dont le pilote OLEDB gère les données lorsqu'il travaille avec une source de données Oracle distante. Les propriétés du pilote sont spécifiques à chaque serveur lié Oracle créé sur l'instance de base de données. Appelez la procédure stockée `master.dbo.sp_addlinkedserver` pour modifier les propriétés du pilote OLEDB.

Exemple : pour créer un serveur lié et modifier la propriété `FetchSize` du pilote OLEDB

```
EXEC master.dbo.sp_addlinkedserver
@server = N'Oracle_link2',
@srvproduct=N'Oracle',
@provider=N'OraOLEDB.Oracle',
@datasrc=N'my-oracle-test.cnetsipka.us-west-2.rds.amazonaws.com:1521/ORCL',
@provstr='FetchSize=200'
GO
```

```
EXEC master.dbo.sp_addlinkedsrvlogin
@rmtsrvname=N'Oracle_link2',
@useself=N'False',
@locallogin=NULL,
@rmtuser=N'master',
@rmtpassword='Test#1234'
GO
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Désactivation de serveurs liés avec Oracle

Pour désactiver des serveurs liés avec Oracle, supprimez l'option `OLEDB_ORACLE` de son groupe d'options.

> **Important**
>
> La suppression de cette option ne supprime pas les configurations de serveur lié existantes sur l'instance de base de données. Vous devez les supprimer manuellement pour les supprimer de l'instance de base de données.
> Vous pouvez réactiver l'option `OLEDB_ORACLE` après la suppression pour réutiliser les configurations de serveurs liés qui étaient précédemment configurées sur l'instance de base de données.

console

La procédure suivante supprime l'option `OLEDB_ORACLE`.

Pour supprimer l'option OLEDB_ORACLE de son groupe d'options

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `OLEDB_ORACLE` (`oracle-oledb-se-2019` dans les exemples précédents).
4. Choisissez Supprimer une option.
5. Sous Deletion options (Options de suppression), choisissez OLEDB_ORACLE pour Options to delete (Options à supprimer).
6. Sous Apply immediately (Appliquer immédiatement), choisissez Yes (Oui) pour supprimer l'option immédiatement, ou No (Non) pour la supprimer lors du prochain créneau de maintenance.

7. Sélectionnez Delete (Supprimer).

INTERFACE DE LIGNE DE COMMANDE (CLI)

La procédure suivante supprime l'option `OLEDB_ORACLE`.

Pour supprimer l'option OLEDB_ORACLE de son groupe d'options

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

  ```
  aws rds remove-option-from-option-group \
      --option-group-name oracle-oledb-se-2019 \
      --options OLEDB_ORACLE \
      --apply-immediately
  ```

  Dans Windows :

  ```
  aws rds remove-option-from-option-group ^
      --option-group-name oracle-oledb-se-2019 ^
      --options OLEDB_ORACLE ^
      --apply-immediately
  ```

Serveurs liés avec Teradata ODBC dans RDS pour SQL Server

Support des serveurs liés avec le pilote ODBC Teradata sur RDS for SQL Server vous permettant d'accéder à des sources de données externes sur une base de données Teradata. Vous pouvez lire des données et exécuter des commandes depuis des serveurs de base de données Teradata distants en dehors de votre instance RDS for SQL Server. Utilisez des serveurs liés avec Teradata ODBC pour activer les fonctionnalités suivantes :

- Accédez directement à des sources de données autres que SQL Server.
- Interrogez différentes sources de données Teradata avec la même requête sans déplacer les données.
- Émettez des requêtes, des mises à jour, des commandes et des transactions distribuées sur les sources de données d'un écosystème d'entreprise.
- Intégrez des connexions à une base de données Teradata depuis la suite Microsoft Business Intelligence (SSIS, SSRS, SSAS).
- Migrez d'une base de données Teradata vers RDS for SQL Server.

Vous pouvez choisir d'activer un ou plusieurs serveurs liés pour Teradata sur une instance de base de données RDS for SQL Server existante ou nouvelle. Vous pouvez ensuite intégrer des sources de données Teradata externes à votre instance de base de données.

Rubriques

- Versions et régions prises en charge
- Limitations et recommandations
- Considérations relatives au déploiement multi-AZ
- Activation de serveurs liés avec Teradata
- Création de serveurs liés avec Teradata
- Désactivation des serveurs liés à Teradata

Versions et régions prises en charge

RDS for SQL Server prend en charge les serveurs liés avec Teradata ODBC dans tous les cas Régions AWS pour SQL Server Standard et Enterprise Edition pour les versions suivantes :

- SQL Server 2022, toutes les versions

- SQL Server 2019, toutes les versions
- SQL Server 2017, toutes les versions

Les versions de base de données Teradata suivantes prennent en charge la liaison avec RDS pour SQL Server

- Teradata 17.20, toutes les versions

Limitations et recommandations

Les limitations suivantes s'appliquent aux serveurs liés à Teradata ODBC :

- RDS for SQL Server ne prend en charge que l'authentification simple avec un nom d'utilisateur et un mot de passe pour la source Teradata.
- RDS pour SQL Server prend uniquement en charge la version 17.20.0.33 du pilote ODBC Teradata.
- RDS pour SQL Server ne prend pas en charge la création de noms de sources de données (DSNs) à utiliser comme raccourcis pour une chaîne de connexion.
- RDS pour SQL Server ne prend pas en charge le suivi des pilotes ODBC. Utilisez SQL Server Extended Events pour suivre les événements ODBC. Pour plus d'informations, consultez Set up Extended Events in RDS for SQL Server (Configuration d'événements étendus dans RDS for SQL Server).
- RDS for SQL Server ne prend pas en charge l'accès au dossier des catalogues pour un serveur lié à Teradata lors de l'utilisation de SQL Server Management Studio (SSMS).

Tenez compte des recommandations suivantes lorsque vous utilisez des serveurs liés avec Teradata ODBC :

- Autorisez le trafic réseau en ajoutant le port TCP applicable dans le groupe de sécurité pour chaque instance de base de données RDS for SQL Server. Si vous configurez un serveur lié entre une instance de base de données EC2 Teradata et une instance de base de données RDS for SQL Server, vous devez autoriser le trafic provenant de l'adresse IP de l'instance de base de EC2 données Teradata. Vous devez également autoriser le trafic sur le port que l'instance de base de données RDS pour SQL Server utilise pour écouter les communications de base de données. Pour plus d'informations sur les groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité.

- Les transactions distribuées (XA) sont prises en charge. Pour activer les transactions distribuées, activez l'`MSDTC`option dans le groupe d'options de votre instance de base de données et assurez-vous que les transactions XA sont activées. Pour de plus amples informations, veuillez consulter Support pour Microsoft Distributed Transaction Coordinator dans RDS for SQL Server.
- Linked Teradata ODBC prend en charge le protocole SSL/TLS tant qu'il est configuré sur le serveur Teradata. Pour plus d'informations, consultez Activer la connectivité TLS sur Teradata Vantage.

Considérations relatives au déploiement multi-AZ

RDS for SQL Server ne réplique actuellement pas les serveurs liés vers le serveur de base de données en miroir (ou le serveur secondaire du groupe de disponibilité Always-On) dans le cadre d'un déploiement multi-AZ. Si les serveurs liés sont ajoutés avant que la configuration ne soit modifiée pour ajouter la mise en miroir ou le mode Always-On, les serveurs liés sont copiés pour les serveurs liés existants.

Vous pouvez également créer les serveurs liés sur l'instance principale, basculer vers l'instance de serveur haute disponibilité, puis créer à nouveau les serveurs liés afin qu'ils se trouvent sur les deux instances RDS pour SQL Server.

Activation de serveurs liés avec Teradata

Activez les serveurs liés avec Teradata en ajoutant l'`ODBC_TERADATA`option à votre instance de base de données RDS for SQL Server. Utilisez la procédure suivante :

Rubriques

- Création du groupe d'options pour ODBC_TERADATA
- Ajout de l'option ODBC_TERADATA au groupe d'options
- Associer l'ODBC_TERADATAoption à votre instance de base de données

Création du groupe d'options pour **`ODBC_TERADATA`**

Pour travailler avec des serveurs liés avec Teradata, créez un groupe d'options ou modifiez un groupe d'options correspondant à l'édition et à la version de SQL Server de l'instance de base de données que vous prévoyez d'utiliser. Pour terminer cette procédure, utilisez le AWS Management Console ou le AWS CLI.

console

Utilisez la procédure suivante pour créer un groupe d'options pour SQL Server Standard Edition 2019.

Pour créer le groupe d'options

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
    a. Dans Nom, entrez un nom unique au sein de votre groupe d'options Compte AWS, tel que`teradata-odbc-se-2019`. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
    b. Dans Description, entrez une brève description du groupe d'options.
    c. Pour Moteur, choisissez sqlserver-se.
    d. Pour Major engine version (Version majeure du moteur), choisissez 15.00.
5. Sélectionnez Créer.

AWS CLI

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2019.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-option-group \
    --option-group-name teradata-odbc-se-2019 \
    --engine-name sqlserver-se \
    --major-engine-version 15.00 \
    --option-group-description "ODBC_TERADATA option group for SQL Server SE 2019"
```

Example

Dans Windows:

```
aws rds create-option-group ^
    --option-group-name teradata-odbc-se-2019 ^
    --engine-name sqlserver-se ^
    --major-engine-version 15.00 ^
    --option-group-description "ODBC_TERADATA option group for SQL Server SE 2019"
```

Ajout de l'option **ODBC_TERADATA** au groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'`ODBC_Teradata`option à votre groupe d'options.

console

Utilisez la procédure suivante pour créer un groupe d'options pour SQL Server Standard Edition 2019.

Pour ajouter l'**ODBC_TERADATA**option

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez votre nouveau groupe d'options.
4. Sélectionnez Ajouter une option.
5. Sous Détails de l'option :
    a. Choisissez ODBC_TERADATA pour le nom de l'option.
    b. `17.20.33.00`Pour la version Option.
6. Dans la section Planification, choisissez d'ajouter l'option immédiatement ou lors de la fenêtre de maintenance suivante.
7. Sélectionnez Ajouter une option.

AWS CLI

La procédure suivante ajoute l'`ODBC_TERADATA`option à votre groupe d'options.

Example

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name teradata-odbc-se-2019 \
    --options "OptionName=ODBC_TERADATA,OptionVersion=17.20.33.00" \
    --apply-immediately
```

Example

Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name teradata-odbc-se-2019 ^
    --options "OptionName=ODBC_TERADATA,OptionVersion=17.20.33.00" ^
    --apply-immediately
```

Associer l'**ODBC_TERADATA**option à votre instance de base de données

Pour associer le groupe d'ODBC_TERADATAoptions à votre instance de base de données, utilisez le AWS Management Console ou AWS CLI.

console

Pour terminer l'activation des serveurs liés pour Teradata, associez votre groupe d'options à une instance de base de données nouvelle ou existante :

- Pour une nouvelle instance de base de données, associez-la lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-la en modifiant l'instance. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

AWS CLI

Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mytestsqlserverteradataodbcinstance \
    --db-instance-class db.m5.2xlarge \
```

```
    --engine sqlserver-se \
    --engine-version 15.00 \
    --license-model license-included \
    --allocated-storage 100 \
    --master-username admin \
    --master-user-password password \
    --storage-type gp2 \
    --option-group-name teradata-odbc-se-2019
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mytestsqlserverteradataodbcinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 15.00 ^
    --license-model license-included ^
    --allocated-storage 100 ^
    --master-username admin ^
    --master-user-password password ^
    --storage-type gp2 ^
    --option-group-name teradata-odbc-se-2019
```

Pour modifier une instance et associer le nouveau groupe d'options :

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mytestsqlserverteradataodbcinstance \
    --option-group-name teradata-odbc-se-2019 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mytestsqlserverteradataodbcinstance ^
    --option-group-name teradata-odbc-se-2019 ^
    --apply-immediately
```

Création de serveurs liés avec Teradata

Pour créer un serveur lié avec Teradata, exécutez les commandes suivantes :

```
EXECUTE master.dbo.sp_addlinkedserver
    @server = N'LinkedServer_NAME',
    @srvproduct=N'',
    @provider=N'MSDASQL',
    @provstr=N'"PROVIDER=MSDASQL;DRIVER={Teradata Database ODBC Driver 17.20};
                DBCName=Server;UID=user_name;PWD=user_password;
                UseDataEncryption=YES/NO;SSLMODE=PREFER/ALLOW/DISABLE>;"',
    @catalog='database'
```

```
EXECUTE master.dbo.sp_addlinkedsrvlogin
    @rmtsrvname = N'LinkedServer_NAME',
    @locallogin = NULL ,
    @useself = N'False',
    @rmtuser = N'user_name',
    @rmtpassword = N'user_password'
```

Voici un exemple des commandes ci-dessus :

```
EXECUTE master.dbo.sp_addlinkedserver
    @server = N'LinkedServerToTeradata',
    @srvproduct=N'',
    @provider=N'MSDASQL',
    @provstr=N'"PROVIDER=MSDASQL;DRIVER={Teradata Database ODBC Driver 17.20};
                DBCName=my-teradata-test.cnetsipka.us-west-2.rds.amazonaws.com;
                UID=master;
                PWD=Test#1234;
                UseDataEncryption=YES;
                SSLMODE=PREFER;"',
    @catalog='MyTestTeradataDB'

EXECUTE master.dbo.sp_addlinkedsrvlogin
    @rmtsrvname = N'LinkedServerToTeradata',
    @locallogin = NULL ,
    @useself = N'False',
    @rmtuser = N'master',
    @rmtpassword = N'Test#1234'
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Désactivation des serveurs liés à Teradata

Pour désactiver les serveurs liés à Teradata, supprimez l'`ODBC_TERADATA`option de son groupe d'options.

> **Important**
>
> La suppression de l'option ne supprime pas les configurations de serveur liées sur l'instance de base de données. Vous devez les supprimer manuellement pour les supprimer de l'instance de base de données.
> Vous pouvez le réactiver `ODBC_TERADATA` après la suppression pour réutiliser les configurations de serveur liées précédemment configurées sur l'instance de base de données.

console

Pour supprimer l'`ODBC_TERADATA`option du groupe d'options

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options contenant l'`ODBC_TERADATA`option.
4. Choisissez Supprimer.
5. Sous Options de suppression, choisissez `ODBC_TERADATA` Options à supprimer.
6. Sous Apply immediately (Appliquer immédiatement), choisissez Yes (Oui) pour supprimer l'option immédiatement, ou No (Non) pour la supprimer lors du prochain créneau de maintenance.
7. Choisissez Supprimer.

AWS CLI

Les commandes suivantes suppriment l'`ODBC_TERADATA`option.

Dans Linux, macOS, ou Unix:

```
aws rds remove-option-from-option-group \
```

```
    --option-group-name teradata-odbc-se-2019 \
    --options ODBC_TERADATA \
    --apply-immediately
```

Dans Windows:

```
aws rds remove-option-from-option-group ^
    --option-group-name teradata-odbc-se-2019 ^
    --options ODBC_TERADATA ^
    --apply-immediately
```

Prise en charge des sauvegarde et restauration natives dans SQL Server

La fonction de sauvegarde et restauration natives pour les bases de données SQL Server vous permet de créer une sauvegarde différentielle ou complète de votre base de données sur site et de stocker les fichiers de sauvegarde sur Amazon S3. Vous pouvez ensuite effectuer la restauration sur une instance de base de données Amazon RDS existante exécutant SQL Server. Vous pouvez également sauvegarder une base de données RDS for SQL Server, la stocker sur Amazon S3, puis la restaurer à d'autres emplacements. En outre, vous pouvez restaurer la sauvegarde sur un serveur sur site ou sur une autre instance de base de données Amazon RDS exécutant SQL Server. Pour plus d'informations, consultez Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.

Amazon RDS prend en charge l'option de sauvegarde et restauration natives pour les bases de données Microsoft SQL Server à l'aide de fichiers de sauvegarde différentielle et complète (fichiers .bak).

Ajout de l'option de sauvegarde et restauration natives

Le processus général d'ajout de l'option de sauvegarde et restauration natives à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option `SQLSERVER_BACKUP_RESTORE` au groupe d'options.
3. Associez un rôle AWS Identity and Access Management (IAM) à l'option. Le rôle IAM doit avoir accès à un compartiment S3 pour stocker les sauvegardes de bases de données.

   Cela signifie que l'option doit avoir un Amazon Resource Name (ARN) valide au format `arn:aws:iam::`*`account-id`*`:role/`*`role-name`*. Pour plus d'informations, consultez Amazon Resource Names (ARN) dans le document Références générales AWS.

   Le rôle IAM doit également avoir une relation d'approbation et une politique d'autorisations attachée. La relation d'approbation permet à RDS d'assumer le rôle, tandis que la politique d'autorisations définit les actions que le rôle peut effectuer. Pour plus d'informations, consultez Création manuelle d'un rôle IAM pour les sauvegarde et restauration natives.
4. Associez le groupe d'options à l'instance de base de données.

Après que vous avez ajouté l'option de sauvegarde et restauration natives, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, vous pouvez commencer à sauvegarder et restaurer immédiatement.

Console

Pour ajouter l'option de sauvegarde et restauration natives

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Créez un groupe d'options ou utilisez un groupe d'options existant. Pour de plus amples informations sur la création d'un groupe d'options de base de données personnalisé, veuillez consulter Création d'un groupe d'options.

   Pour utiliser un groupe d'options existant, passez à l'étape suivante.
4. Ajoutez l'option SQLSERVER_BACKUP_RESTORE au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
5. Effectuez l'une des actions suivantes :

   - Pour utiliser un rôle IAM existant et des paramètres Amazon S3, choisissez un rôle IAM existant pour Rôle IAM. Si vous utilisez un rôle IAM existant, RDS utilise les paramètres Amazon S3 que vous avez configurés pour ce rôle.
   - Pour créer un nouveau rôle et configurer des paramètres Amazon S3, procédez comme suit :
     1. Pour Rôle IAM, choisissez Créer un rôle.
     2. Pour S3 bucket name (Nom du compartiment S3), choisissez un compartiment S3 dans la liste.
     3. Pour S3 prefix (optional) (Préfixe du chemin de dossier S3 (facultatif)), saisissez un préfixe à utiliser pour les fichiers stockés dans votre compartiment Amazon S3.

        Ce préfixe peut inclure un chemin de fichier mais cela n'est pas obligatoire. Si vous fournissez un préfixe, RDS l'attache à tous les fichiers de sauvegarde. RDS utilise alors le préfixe durant une restauration pour identifier les fichiers connexes et ignorer les fichiers non concernés. Par exemple, vous pouvez utiliser le compartiment S3 pour d'autres choses que le stockage de fichiers de sauvegarde. Dans ce cas, vous pouvez utiliser le préfixe pour que RDS effectue une sauvegarde et restauration natives uniquement sur un dossier particulier et ses sous-dossiers.

Si vous laissez le préfixe vide, RDS n'utilise pas de préfixe pour identifier les fichiers de sauvegarde ou les fichiers à restaurer. Par conséquent, lors d'une restauration de plusieurs fichiers, RDS tente de restaurer chaque fichier dans chaque dossier de ce compartiment S3.

4. Cochez la case Enable Encryption (Activer le chiffrement) pour chiffrer le fichier de sauvegarde. Laissez la case décochée (valeur par défaut) pour que le fichier de sauvegarde ne soit pas chiffré.

   Si vous avez choisi Enable encryption (Activer le chiffrement), choisissez une clé de chiffrement pour AWS KMS key. Pour en savoir plus sur les clés de chiffrement, consultez Mise en route dans le Manuel du développeur AWS Key Management Service.

6. Sélectionnez Ajouter une option.
7. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Cette procédure se base sur les hypothèses suivantes :

- Vous ajoutez l'option SQLSERVER_BACKUP_RESTORE à un groupe d'options qui existe déjà. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
- Vous associez l'option à un rôle IAM qui existe déjà et qui a accès à un compartiment S3 pour stocker les sauvegardes.
- Vous appliquez le groupe d'options à une instance de base de données qui existe déjà. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Pour ajouter l'option de sauvegarde et restauration natives

1. Ajoutez l'option `SQLSERVER_BACKUP_RESTORE` au groupe d'options.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
 --apply-immediately \
 --option-group-name mybackupgroup \
 --options "OptionName=SQLSERVER_BACKUP_RESTORE, \
   OptionSettings=[{Name=IAM_ROLE_ARN,Value=arn:aws:iam::account-id:role/role-
name}]"
```

Dans Windows :

```
aws rds add-option-to-option-group ^
 --option-group-name mybackupgroup ^
 --options "[{\"OptionName\": \"SQLSERVER_BACKUP_RESTORE\", ^
 \"OptionSettings\": [{\"Name\": \"IAM_ROLE_ARN\", ^
 \"Value\": \"arn:aws:iam::account-id:role/role-name"}]}]" ^
 --apply-immediately
```

> **Note**
>
> Lorsque vous utilisez l'invite de commande Windows, vous devez utiliser des guillemets doubles (") d'échappement dans le code JSON en les préfixant d'une barre oblique inverse (\).

2. Appliquez le groupe d'options à l'instance de base de données.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
 --db-instance-identifier mydbinstance \
 --option-group-name mybackupgroup \
 --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
 --db-instance-identifier mydbinstance ^
 --option-group-name mybackupgroup ^
 --apply-immediately
```

Modification des paramètres d'option de sauvegarde et restauration natives

Une fois que vous avez activé l'option de sauvegarde et restauration natives, vous pouvez modifier les paramètres de l'option. Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option.

Suppression de l'option de sauvegarde et restauration natives

Vous pouvez désactiver la fonction de sauvegarde et restauration natives en supprimant l'option de votre instance de base de données. Une fois que vous avez supprimé l'option de sauvegarde et restauration natives, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'option de sauvegarde et restauration natives d'une instance de base de données, effectuez l'une des actions suivantes :

- Supprimez l'option du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.
- Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option de sauvegarde et restauration natives. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Support pour le chiffrement transparent des données sur le SQL serveur

Amazon RDS prend en charge l'utilisation de Transparent Data Encryption (TDE) pour chiffrer les données stockées sur vos instances de base de données exécutant Microsoft SQL Server. TDEchiffre automatiquement les données avant leur écriture dans le stockage et déchiffre automatiquement les données lorsqu'elles sont lues depuis le stockage.

Amazon RDS prend en TDE charge les versions et éditions de SQL serveur suivantes :

- SQLÉditions Server 2022 Standard et Enterprise
- SQLÉditions Standard et Enterprise de Server 2019
- SQLServer 2017 Édition Enterprise
- SQLServer 2016 Édition Enterprise

Transparent Data Encryption for SQL Server permet de gérer les clés de chiffrement en utilisant une architecture de clé à deux niveaux. Un certificat, qui est généré à partir de la clé principale de la base de données, permet de protéger les clés de chiffrement des données. La clé de chiffrement de la base de données exécute le chiffrement et le déchiffrement des données sur la base de données utilisateur. Amazon RDS sauvegarde et gère la clé principale de la base de données et le TDE certificat.

La fonction TDE (Transparent Data Encryption) est utilisée dans les scénarios où vous devez chiffrer des données sensibles. Par exemple, vous pouvez souhaiter fournir des fichiers de données et des sauvegardes à un tiers, ou résoudre des problèmes de conformité réglementaire liés à la sécurité. Vous ne pouvez pas chiffrer les bases de données système pour le SQL serveur, telles que les `master` bases de données `model` or.

Ce guide n'a pas vocation à offrir une présentation détaillée du chiffrement TDE, mais assurez-vous de bien comprendre les points forts et les points faibles de chaque algorithme et de chaque clé. Pour plus d'informations sur le chiffrement transparent des données pour le SQL serveur, voir Transparent Data Encryption (TDE) dans la documentation Microsoft.

Rubriques

- Activation de TDE for RDS for SQL Server
- Chiffrement des données sur le RDS serveur SQL
- Sauvegarde et restauration de certificats TDE sur RDS for SQL Server

- Sauvegarde et restauration des TDE certificats pour les bases de données locales
- Désactiver TDE RDS pour le SQL serveur

Activation de TDE for RDS for SQL Server

Pour activer le chiffrement transparent des données RDS pour une instance de base de données pour SQL serveur, spécifiez l'TDEoption dans un groupe d'RDSoptions associé à cette instance de base de données :

1. Déterminez si votre instance de base de données est déjà associée à un groupe d'options disposant de cette TDE option. Pour afficher le groupe d'options auquel une instance de base de données est associée, utilisez la RDS console, la describe-db-instance AWS CLI commande ou l'APIopération escribeDBInstancesD.
2. Si l'instance de base de données n'est pas associée à un groupe d'options TDE activé, deux choix s'offrent à vous. Vous pouvez créer un groupe d'options et ajouter l'TDEoption, ou vous pouvez modifier le groupe d'options associé pour l'ajouter.

   > **Note**
   >
   > Dans la RDS console, l'option est nomméeTRANSPARENT_DATA_ENCRYPTION. Dans le AWS CLI pays RDSAPI, il est nomméTDE.

   Pour plus d'informations sur la création ou la modification d'un groupe d'options, consultez Utilisation de groupes d'options. Pour de plus amples informations sur l'ajout d'une option à un groupe d'options, veuillez consulter Ajout d'une option à un groupe d'options.
3. Associez l'instance de base de données au groupe d'options qui possède l'TDEoption. Pour plus d'informations sur l'association d'une instance de base de données à un groupe d'options, consultez Modification d'une RDS instance de base de données Amazon.

Considérations relatives au groupe d'options

Il s'agit d'une option persistante. TDE Vous ne pouvez pas la supprimer d'un groupe d'options tant que toutes les instances de base de données et les sauvegardes sont associées au groupe d'options. Une fois que vous avez ajouté l'TDEoption à un groupe d'options, celui-ci ne peut être associé qu'aux instances de base de données qui utilisentTDE. Pour plus d'informations sur les options persistantes dans un groupe d'options, consultez Présentation des groupes d'options.

Comme l'TDEoption est une option persistante, il peut y avoir un conflit entre le groupe d'options et une instance de base de données associée. Un conflit peut se produire dans les cas suivants :

- Le groupe d'options actuel contient l'TDEoption, et vous le remplacez par un groupe d'options qui n'en possède pas. TDE
- Vous effectuez une restauration à partir d'un instantané de base de données vers une nouvelle instance de base de données qui ne possède pas de groupe d'options contenant l'TDEoption. Pour plus d'informations sur ce scénario, consultez Considérations relatives aux groupes d'options.

SQLConsidérations relatives aux performances du serveur

L'utilisation du chiffrement transparent des données peut affecter les performances d'une instance de base de données de SQL serveur.

Les performances des bases de données non chiffrées peuvent aussi être dégradées si les bases de données se trouvent sur une instance de base de données qui possède au moins une base de données chiffrée. En conséquence, il est recommandé de garder les bases de données chiffrées et les bases de données non chiffrées sur des instances de base de données distinctes.

Chiffrement des données sur le RDS serveur SQL

Lorsque l'TDEoption est ajoutée à un groupe d'options, Amazon RDS génère un certificat qui est utilisé dans le processus de chiffrement. Vous pouvez ensuite utiliser le certificat pour exécuter SQL des instructions qui chiffrent les données d'une base de données sur l'instance de base de données.

L'exemple suivant utilise le certificat RDS -created appelé `RDSTDECertificateName` pour chiffrer une base de données appelée. `myDatabase`

```
---------- Turning on TDE -------------

-- Find an RDS TDE certificate to use
USE [master]
GO
SELECT name FROM sys.certificates WHERE name LIKE 'RDSTDECertificate%'
GO

USE [myDatabase]
GO
-- Create a database encryption key (DEK) using one of the certificates from the
 previous step
CREATE DATABASE ENCRYPTION KEY WITH ALGORITHM = AES_256
```

```
ENCRYPTION BY SERVER CERTIFICATE [RDSTDECertificateName]
GO

-- Turn on encryption for the database
ALTER DATABASE [myDatabase] SET ENCRYPTION ON
GO

-- Verify that the database is encrypted
USE [master]
GO
SELECT name FROM sys.databases WHERE is_encrypted = 1
GO
SELECT db_name(database_id) as DatabaseName, * FROM sys.dm_database_encryption_keys
GO
```

Le temps nécessaire pour chiffrer une base de données SQL du serveur TDE dépend de plusieurs facteurs. Il s'agit notamment de la taille de l'instance de base de données, de l'utilisation du IOPS stockage provisionné par l'instance, de la quantité de données et d'autres facteurs.

Sauvegarde et restauration de certificats TDE sur RDS for SQL Server

RDS for SQL Server fournit des procédures stockées pour la sauvegarde, la restauration et la suppression de certificats TDE. RDS for SQL Server fournit également une fonction permettant d'afficher les certificats TDE utilisateur restaurés.

Les certificats TDE utilisateur sont utilisés pour restaurer des bases de données sur site et pour lesquelles TDE est activé sur RDS for SQL Server. Ces certificats ont le préfixe `UserTDECertificate_`. Après avoir restauré les bases de données et avant de les mettre à disposition, RDS modifie les bases de données sur lesquelles TDE est activé pour utiliser les certificats TDE générés par RDS. Ces certificats ont le préfixe `RDSTDECertificate`.

Les certificats TDE utilisateur restent sur l'instance de base de données RDS for SQL Server, sauf si vous les supprimez en utilisant la procédure stockée `rds_drop_tde_certificate`. Pour de plus amples informations, veuillez consulter Suppression de certificats TDE restaurés.

Vous pouvez utiliser un certificat TDE utilisateur pour restaurer d'autres bases de données à partir de l'instance de base de données source. Les bases de données à restaurer doivent utiliser le même certificat TDE et TDE doit être activé sur celles-ci. Il n'est pas nécessaire d'importer (restaurer) à nouveau le même certificat.

Rubriques

- Prérequis
- Limites
- Sauvegarde d'un certificat TDE
- Restauration d'un certificat TDE
- Affichage des certificats TDE restaurés
- Suppression de certificats TDE restaurés

Prérequis

Avant de pouvoir sauvegarder ou restaurer des certificats TDE sur RDS for SQL Server, veillez à effectuer les tâches suivantes. Les trois premières tâches sont décrites dans Configuration pour les sauvegarde et restauration natives.

1. Créez des compartiments Amazon S3 pour y stocker les fichiers à sauvegarder et à restaurer.

   Nous vous recommandons d'utiliser des compartiments distincts pour les sauvegardes de bases de données et pour les sauvegardes de certificats TDE.
2. Créez un rôle IAM pour la sauvegarde et la restauration de fichiers.

   Le rôle IAM doit être à la fois un utilisateur et un administrateur de la AWS KMS key.

   Outre les autorisations requises pour la sauvegarde et la restauration natives SQL Server, le rôle IAM exige également les autorisations suivantes :

   - `s3:GetBucketAcl`, `s3:GetBucketLocation` et `s3:ListBucket` sur la ressource du compartiment S3
   - `s3:ListAllMyBuckets` sur la ressource *
3. Ajoutez l'option `SQLSERVER_BACKUP_RESTORE` à un groupe d'options sur votre instance de base de données.

   Elle vient s'ajouter à l'option `TRANSPARENT_DATA_ENCRYPTION` (`TDE`).
4. Vérifiez que vous disposez d'une clé KMS de chiffrement symétrique. Vous avez les options suivantes :

   - Si vous disposez déjà d'une clé KMS dans votre compte, vous pouvez l'utiliser. Aucune action supplémentaire n'est nécessaire.

- Si votre compte ne contient pas encore de clés de chiffrement KMS symétriques, créez-en une en suivant les instructions de la section Creating keys (Création de clés) du Guide du développeur AWS Key Management Service .

5. Activez l'intégration Amazon S3 pour transférer des fichiers entre l'instance de base de données et Amazon S3.

   Pour plus d'informations sur l'activation de l'intégration d'Amazon S3, consultez Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3.

Limites

L'utilisation de procédures stockées pour sauvegarder et restaurer des certificats TDE présente les limites suivantes :

- Les options `SQLSERVER_BACKUP_RESTORE` et `TRANSPARENT_DATA_ENCRYPTION` (TDE) doivent être ajoutées au groupe d'options que vous avez associé à votre instance de base de données.
- La sauvegarde et la restauration de certificats TDE ne sont pas prises en charge sur les instances de base de données multi-AZ.
- L'annulation des tâches de sauvegarde et de restauration de certificats TDE n'est pas prise en charge.
- Vous ne pouvez pas utiliser de certificat TDE utilisateur pour le chiffrement TDE d'une autre base de données sur votre instance de base de données RDS for SQL Server. Vous pouvez l'utiliser pour restaurer uniquement d'autres bases de données à partir de l'instance de base de données source sur laquelle TDE est activé et qui utilisent le même certificat TDE.
- Vous ne pouvez supprimer que des certificats TDE utilisateur.
- Le nombre maximal de certificats TDE utilisateur pris en charge sur RDS est de 10. Si le nombre dépasse 10, supprimez les certificats TDE inutilisés et réessayez.
- Le nom de certificat ne peut pas être vide ou null.
- Lors de la restauration d'un certificat, le nom du certificat ne peut pas inclure le mot-clé `RDSTDECERTIFICATE` et doit commencer par le préfixe `UserTDECertificate_`.
- Le paramètre `@certificate_name` peut inclure uniquement les caractères suivants : a-z, 0-9, @, $, # et trait de soulignement (_).
- L'extension de fichier de `@certificate_file_s3_arn` doit être .cer (insensible à la casse).
- L'extension de fichier de `@private_key_file_s3_arn` doit être .pvk (insensible à la casse).

- Les métadonnées S3 du fichier de clé privée doivent inclure la balise `x-amz-meta-rds-tde-pwd`. Pour de plus amples informations, veuillez consulter Sauvegarde et restauration des TDE certificats pour les bases de données locales.

Sauvegarde d'un certificat TDE

Pour sauvegarder les certificats TDE, utilisez la procédure stockée `rds_backup_tde_certificate`. Elle possède la syntaxe suivante.

```
EXECUTE msdb.dbo.rds_backup_tde_certificate
    @certificate_name='UserTDECertificate_certificate_name |
 RDSTDECertificatetimestamp',
    @certificate_file_s3_arn='arn:aws:s3:::bucket_name/certificate_file_name.cer',
    @private_key_file_s3_arn='arn:aws:s3:::bucket_name/key_file_name.pvk',
    @kms_password_key_arn='arn:aws:kms:region:account-id:key/key-id',
    [@overwrite_s3_files=0|1];
```

Les paramètres suivants sont obligatoires :

- `@certificate_name` : nom du certificat TDE à sauvegarder.
- `@certificate_file_s3_arn` : Amazon Resource Name (ARN) de destination pour le fichier de sauvegarde de certificat dans Amazon S3.
- `@private_key_file_s3_arn` : ARN S3 de destination du fichier de clé privée qui sécurise le certificat TDE.
- `@kms_password_key_arn` : ARN de la clé KMS symétrique utilisée pour chiffrer le mot de passe de la clé privée.

Le paramètre suivant est facultatif :

- `@overwrite_s3_files` : indique s'il convient de remplacer le certificat existant et les fichiers de clé privée dans S3 :
  - `0` : n'écrase pas les fichiers existants. Cette valeur est celle par défaut.

    Si `@overwrite_s3_files` est défini sur 0, une erreur est renvoyée si un fichier existe déjà.
  - 1 – Écrase le fichier existant qui possède déjà le nom spécifié, même s'il ne s'agit pas d'un fichier de sauvegarde.

Example Exemple de sauvegarde d'un certificat TDE

```
EXECUTE msdb.dbo.rds_backup_tde_certificate
    @certificate_name='RDSTDECertificate20211115T185333',
    @certificate_file_s3_arn='arn:aws:s3:::TDE_certs/mycertfile.cer',
    @private_key_file_s3_arn='arn:aws:s3:::TDE_certs/mykeyfile.pvk',
    @kms_password_key_arn='arn:aws:kms:us-
west-2:123456789012:key/AKIAIOSFODNN7EXAMPLE',
    @overwrite_s3_files=1;
```

Restauration d'un certificat TDE

Vous utilisez la procédure stockée `rds_restore_tde_certificate` pour restaurer (importer) des certificats TDE utilisateur. Elle possède la syntaxe suivante.

```
EXECUTE msdb.dbo.rds_restore_tde_certificate
    @certificate_name='UserTDECertificate_certificate_name',
    @certificate_file_s3_arn='arn:aws:s3:::bucket_name/certificate_file_name.cer',
    @private_key_file_s3_arn='arn:aws:s3:::bucket_name/key_file_name.pvk',
    @kms_password_key_arn='arn:aws:kms:region:account-id:key/key-id';
```

Les paramètres suivants sont obligatoires :

- `@certificate_name` : nom du certificat TDE à restaurer. Le nom doit commencer par le préfixe `UserTDECertificate_`.
- `@certificate_file_s3_arn` : ARN S3 du fichier de sauvegarde utilisé pour restaurer le certificat TDE.
- `@private_key_file_s3_arn` : ARN S3 du fichier de sauvegarde de la clé privée du certificat TDE à restaurer.
- `@kms_password_key_arn` : ARN de la clé KMS symétrique utilisée pour chiffrer le mot de passe de la clé privée.

Example Exemple de restauration d'un certificat TDE

```
EXECUTE msdb.dbo.rds_restore_tde_certificate
    @certificate_name='UserTDECertificate_myTDEcertificate',
    @certificate_file_s3_arn='arn:aws:s3:::TDE_certs/mycertfile.cer',
    @private_key_file_s3_arn='arn:aws:s3:::TDE_certs/mykeyfile.pvk',
```

```
    @kms_password_key_arn='arn:aws:kms:us-
west-2:123456789012:key/AKIAIOSFODNN7EXAMPLE';
```

Affichage des certificats TDE restaurés

Vous utilisez la fonction `rds_fn_list_user_tde_certificates` pour afficher les certificats TDE utilisateur restaurés (importés). Elle possède la syntaxe suivante.

```
SELECT * FROM msdb.dbo.rds_fn_list_user_tde_certificates();
```

La sortie se présente comme suit : Les colonnes ne sont pas toutes affichées ici.

name	certif te_id	princi _id	pvt_ke ncrypt _type_ c	issuer me	cert_s al_num	thumbp t	subjec	start_ e	expiry te	pvt_key_l ast_backu p_date
UserTD rtific _tde_c	343	1	ENCRYP _BY_MA R_KEY	AnyCom y Shippi	79 3e 57 a3 69 fd 1d 9e 47 2c 32 67 1d 9c ca af	0x6BB2 341103 80B FE1BA2 C69509 5B5	AnyCom y Shippi	2022-0 5 19:49: 000000	2023-0 5 19:49: 000000	NULL

Suppression de certificats TDE restaurés

Pour supprimer les certificats TDE utilisateur restaurés (importés) que vous n'utilisez pas, utilisez la procédure stockée `rds_drop_tde_certificate`. Elle possède la syntaxe suivante.

```
EXECUTE msdb.dbo.rds_drop_tde_certificate
 @certificate_name='UserTDECertificate_certificate_name';
```

Les paramètres suivants sont obligatoires :

- `@certificate_name` : nom du certificat TDE à supprimer.

Vous ne pouvez supprimer que les certificats TDE restaurés (importés). Vous ne pouvez pas supprimer les certificats créés par RDS.

Example Exemple de suppression d'un certificat TDE

```
EXECUTE msdb.dbo.rds_drop_tde_certificate
 @certificate_name='UserTDECertificate_myTDEcertificate';
```

Sauvegarde et restauration des TDE certificats pour les bases de données locales

Vous pouvez sauvegarder les TDE certificats des bases de données locales, puis les restaurer RDS ultérieurement sur SQL Server. Vous pouvez également restaurer un TDE certificat RDS for SQL Server sur une instance de base de données sur site.

La procédure suivante sauvegarde un TDE certificat et une clé privée. La clé privée est chiffrée à l'aide d'une clé de données générée à partir de votre KMS clé de chiffrement symétrique.

Pour sauvegarder un certificat sur site TDE

1. Générez la clé de données à l'aide de la AWS CLI generate-data-keycommande.

   ```
   aws kms generate-data-key \
       --key-id my_KMS_key_ID \
       --key-spec AES_256
   ```

   La sortie se présente comme suit :

   ```
   {
   ```

```
"CiphertextBlob": "AQIDAHimL2NEoAlOY6Bn7LJfnxi/OZe9kTQo/
XQXduug1rmerwGiL7g5ux4av9GfZLxYTDATAAAAfjB8BgkqhkiG9w0B
BwagbzBtAgEAMGgGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMyCxLMi7GRZgKqD65AgEQgDtjvZLJo2cQ31Vetng
2RezQy3sAS6ZHrCjfnfn0c65bFdhsXxjSMnudIY7AKw==",
"Plaintext": "U/fpGtmzGCYBi8A2+0/9qcRQRK2zmG/aOn939ZnKi/0=",
"KeyId": "arn:aws:kms:us-west-2:123456789012:key/1234abcd-00ee-99ff-88dd-
aa11bb22cc33"
}
```

Vous utilisez la sortie en texte brut à l'étape suivante comme mot de passe de clé privée.

2. Sauvegardez votre TDE certificat comme indiqué dans l'exemple suivant.

```
BACKUP CERTIFICATE myOnPremTDEcertificate TO FILE = 'D:\tde-cert-backup.cer'
WITH PRIVATE KEY (
FILE = 'C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\DATA\cert-
backup-key.pvk',
ENCRYPTION BY PASSWORD = 'U/fpGtmzGCYBi8A2+0/9qcRQRK2zmG/aOn939ZnKi/0=');
```

3. Enregistrez le fichier de sauvegarde de certificat dans votre compartiment de certificat Amazon S3.
4. Enregistrez le fichier de sauvegarde de clé privée dans votre compartiment de certificat S3, avec la balise suivante dans les métadonnées du fichier :

   - Clé : `x-amz-meta-rds-tde-pwd`
   - Valeur : valeur `CiphertextBlob` issue de la génération de la clé de données, comme dans l'exemple suivant.

     ```
     AQIDAHimL2NEoAlOY6Bn7LJfnxi/OZe9kTQo/
     XQXduug1rmerwGiL7g5ux4av9GfZLxYTDATAAAAfjB8BgkqhkiG9w0B
     BwagbzBtAgEAMGgGCSqGSIb3DQEHATAeBglghkgBZQMEAS4wEQQMyCxLMi7GRZgKqD65AgEQgDtjvZLJo2cQ31Vet
     2RezQy3sAS6ZHrCjfnfn0c65bFdhsXxjSMnudIY7AKw==
     ```

La procédure suivante restaure un TDE certificat RDS for SQL Server sur une instance de base de données sur site. Vous copiez et restaurez le TDE certificat sur votre instance de base de données de destination à l'aide de la sauvegarde du certificat, du fichier de clé privée et de la clé de données correspondants. Le certificat restauré est chiffré par la clé principale de base de données du nouveau serveur.

Pour restaurer un TDE certificat

1. Copiez le fichier de sauvegarde du TDE certificat et le fichier de clé privée depuis Amazon S3 vers l'instance de destination. Pour plus d'informations sur la copie de fichiers depuis Amazon S3, consultez Transfert de fichiers entre RDS for SQL Server et Amazon S3.
2. Utilisez votre KMS clé pour déchiffrer le texte chiffré de sortie afin de récupérer le texte brut de la clé de données. Le texte chiffré se trouve dans les métadonnées S3 du fichier de sauvegarde de la clé privée.

   ```
   aws kms decrypt \
       --key-id my_KMS_key_ID \
       --ciphertext-blob fileb://exampleCiphertextFile | base64 -d \
       --output text \
       --query Plaintext
   ```

   Vous utilisez la sortie en texte brut à l'étape suivante comme mot de passe de clé privée.
3. Utilisez la SQL commande suivante pour restaurer votre TDE certificat.

   ```
   CREATE CERTIFICATE myOnPremTDEcertificate FROM FILE='D:\tde-cert-backup.cer'
   WITH PRIVATE KEY (FILE = N'D:\tde-cert-key.pvk',
   DECRYPTION BY PASSWORD = 'plain_text_output');
   ```

Pour plus d'informations sur le KMS déchiffrement, voir déchiffrer dans la KMS section du manuel de référence des AWS CLI commandes.

Une fois le TDE certificat restauré sur l'instance de base de données de destination, vous pouvez restaurer les bases de données chiffrées avec ce certificat.

> **Note**
>
> Vous pouvez utiliser le même TDE certificat pour chiffrer plusieurs bases de données de SQL serveur sur l'instance de base de données source. Pour migrer plusieurs bases de données vers une instance de destination, copiez le TDE certificat qui leur est associé sur l'instance de destination une seule fois.

Désactiver TDE RDS pour le SQL serveur

TDEPour désactiver une instance de base de données RDS for SQL Server, assurez-vous d'abord qu'il ne reste aucun objet chiffré sur l'instance de base de données. Pour ce faire, déchiffrez les objets ou supprimez-les. Si des objets chiffrés existent sur l'instance de base de données, vous ne pouvez pas désactiver l'instance TDE de base de données. Lorsque vous utilisez la console pour supprimer l'TDEoption d'un groupe d'options, la console indique qu'elle est en cours de traitement. En outre, un événement d'erreur est créé si le groupe d'options est associé à une instance de base de données ou un instantané de bases de données chiffré.

L'exemple suivant supprime le TDE chiffrement d'une base de données appeléecustomerDatabase.

```
------------- Removing TDE ----------------

USE [customerDatabase]
GO

-- Turn off encryption of the database
ALTER DATABASE [customerDatabase]
SET ENCRYPTION OFF
GO

-- Wait until the encryption state of the database becomes 1. The state is 5
 (Decryption in progress) for a while
SELECT db_name(database_id) as DatabaseName, * FROM sys.dm_database_encryption_keys
GO

-- Drop the DEK used for encryption
DROP DATABASE ENCRYPTION KEY
GO

-- Alter to SIMPLE Recovery mode so that your encrypted log gets truncated
USE [master]
GO
ALTER DATABASE [customerDatabase] SET RECOVERY SIMPLE
GO
```

Lorsque tous les objets sont déchiffrés, vous disposez de deux options :

1. Vous pouvez modifier l'instance de base de données à associer à un groupe d'options sans l'TDEoption.

2. Vous pouvez supprimer l'TDEoption du groupe d'options.

SQLAudit du serveur

Dans AmazonRDS, vous pouvez auditer les bases de données Microsoft SQL Server à l'aide du mécanisme d'audit de SQL serveur intégré. Vous pouvez créer des audits et des spécifications d'audit de la même manière que pour des serveurs de base de données sur site.

RDStélécharge les journaux d'audit terminés dans votre compartiment S3, en utilisant le IAM rôle que vous fournissez. Si vous activez la rétention, RDS conserve vos journaux d'audit sur votre instance de base de données pendant la période configurée.

Pour plus d'informations, consultez la section Audit SQL du serveur (moteur de base de données) dans la documentation Microsoft SQL Server.

SQLAudit de serveur avec flux d'activité de base de données

Vous pouvez utiliser les flux d'activité de base de données RDS pour intégrer les événements d'audit SQL du serveur aux outils de surveillance de l'activité des bases de données d'Imperva McAfee, etIBM. Pour plus d'informations sur l'audit avec Database Activity Streams for RDS SQL Server, voir Audit dans Microsoft SQL Server

Rubriques

- Support pour l'audit des SQL serveurs
- Ajout d'SQLun audit de serveur aux options de l'instance de base de données
- Utilisation de l'audit SQL de serveur
- Consultation des journaux d'audit
- Utilisation de SQL Server Audit avec des instances multi-AZ
- Configuration d'un compartiment S3
- Création manuelle d'un rôle IAM pour SQL Server Audit

Support pour l'audit des SQL serveurs

Sur AmazonRDS, depuis SQL Server 2016, toutes les éditions de Server prennent en charge les audits au niveau du SQL serveur, et l'édition Enterprise prend également en charge les audits au niveau de la base de données. À partir de SQL Server 2016 (13.x)SP1, toutes les éditions prennent en charge les audits au niveau du serveur et au niveau de la base de données. Pour plus d'informations, consultez la section Audit SQL du serveur (moteur de base de données) dans la documentation SQL du serveur.

RDSprend en charge la configuration des paramètres d'option suivants pour l'audit SQL du serveur.

Paramètre d'option	Valeurs valides	Description
`IAM_ROLE_ARN`	Un nom de ressource Amazon valide (ARN) au format`arn:aws:iam:: account-id :role/role-name` .	Le IAM rôle qui accorde ARN l'accès au compartiment S3 dans lequel vous souhaitez stocker vos journaux d'audit. Pour plus d'informations, consultez Amazon Resource Names (ARNs) dans le Références générales AWS.
`S3_BUCKET_ARN`	Un valide ARN au format `arn:aws:s3::: amzn-s3-demo-bucket` ou `arn:aws:s3::: amzn-s3-demo-bucket /key-prefix`	ARNPour le compartiment S3 dans lequel vous souhaitez stocker vos journaux d'audit.
`ENABLE_COMPRESSION`	`true` ou `false`	Contrôle la compression des journaux d'audit. Par défaut, la compression est activée (définie sur `true`).
`RETENTION_TIME`	`0` sur `840`	Durée de conservation (en heures) pendant laquelle les enregistrements d'audit SQL du serveur sont conservés sur votre RDS instance. Par défaut, la conservation est désactivée.

Ajout d'SQLun audit de serveur aux options de l'instance de base de données

L'activation de l'audit SQL du serveur nécessite deux étapes : activer l'option sur l'instance de base de données et activer la fonctionnalité dans le SQL serveur. Le processus d'ajout de l'option d'audit SQL du serveur à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajouter et configurer toutes les options requises.
3. Associez le groupe d'options à l'instance de base de données.

Une fois que vous avez ajouté l'option d'audit SQL du serveur, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, vous pouvez créer des audits et stocker les journaux d'audit dans votre compartiment S3.

Pour ajouter et configurer l'audit SQL du serveur sur le groupe d'options d'une instance de base de données

1. Choisissez l'une des méthodes suivantes :
   - Utiliser un groupe d'options existant.
   - Créer un groupe d'options d'instance de base de données personnalisé et utiliser ce groupe d'options. Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'AUDIToption SQLSERVER_ au groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
   - Pour IAMle rôle, si vous possédez déjà un IAM rôle avec les politiques requises, vous pouvez choisir ce rôle. Pour créer un nouveau IAM rôle, choisissez Créer un nouveau rôle. Pour plus d'informations sur les stratégies requises, consultez Création manuelle d'un rôle IAM pour SQL Server Audit.
   - Pour Select S3 destination (Sélectionner une destination S3), si vous avez déjà un compartiment S3 que vous souhaitez utiliser, choisissez-le. Pour créer un compartiment S3, choisissez Create a New S3 Bucket (Créer un nouveau compartiment S3).
   - Pour Enable Compression (Activer la compression), laissez cette option cochée pour compresser les fichiers d'audit. La compression est activée par défaut. Pour désactiver la compression, désélectionnez Enable Compression (Activer la compression).

- Pour Audit log retention (Rétention des journaux d'audit), pour conserver les enregistrement sue l'instance de base de données, choisissez cette option. Spécifiez une durée de rétention en heures. La durée de rétention maximale est de 35 jours.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante. Choisissez l'une des méthodes suivantes :

   - Si vous créez une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance.
   - Sur une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Modification de l'option d'audit SQL du serveur

Après avoir activé l'option Audit SQL du serveur, vous pouvez modifier les paramètres. Pour de plus amples informations sur la modification des paramètres d'option, veuillez consulter Modification d'un paramètre d'option.

Suppression de l'audit SQL du serveur dans les options de l'instance de base de données

Vous pouvez désactiver la fonctionnalité d'audit de SQL serveur en désactivant les audits, puis en supprimant l'option.

Pour supprimer les audits

1. Désactivez tous les paramètres d'audit dans SQL Server. Pour savoir où les audits sont exécutés, interrogez les vues du catalogue de sécurité SQL du serveur. Pour plus d'informations, consultez la section Vues du catalogue de sécurité dans la documentation de Microsoft SQL Server.
2. Supprimez l'option d'audit SQL du serveur de l'instance de base de données. Sélectionnez l'une des méthodes suivantes :

   - Supprimez l'option SQL Server Audit du groupe d'options utilisé par l'instance de base de données. Ce changement affecte toutes les instances de bases de données qui utilisent le même groupe d'options. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.
   - Modifiez l'instance de base de données, puis choisissez un groupe d'options sans l'option SQL Server Audit. Cette modification affecte uniquement l'instance de base de données que vous

modifiez. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

3. Une fois que vous avez supprimé l'option d'audit du SQL serveur de l'instance de base de données, vous n'avez pas besoin de redémarrer l'instance. Supprimez les fichiers d'audit non requis de votre compartiment S3.

Utilisation de l'audit SQL de serveur

Vous pouvez contrôler les audits de serveur, les spécifications d'audit de serveur et les spécifications d'audit de base de données, de la même manière que vous les contrôlez pour vos serveurs de base de données sur site.

Création d'audits

Vous pouvez créer des audits de serveur de la même manière que pour des serveurs de base de données sur site. Pour plus d'informations sur la création d'audits de serveurs, consultez CREATESERVERAUDITla documentation Microsoft SQL Server.

Pour éviter les erreurs, respectez les limitations suivantes :

- Ne dépassez pas le nombre maximal d'audits de serveur pris en charge par instance de 50.
- Demandez au SQL serveur d'écrire des données dans un fichier binaire.
- N'utilisez pas `RDS_` comme préfixe dans le nom de l'audit de serveur.
- Pour `FILEPATH`, spécifiez `D:\rdsdbdata\SQLAudit`.
- Pour `MAXSIZE`, spécifiez une taille comprise entre 2 Mo et 50 Mo.
- Ne configurez pas `MAX_ROLLOVER_FILES` ou `MAX_FILES`.
- Ne configurez pas le SQL serveur pour arrêter l'instance de base de données s'il ne parvient pas à écrire l'enregistrement d'audit.

Création de spécifications d'audit

Vous créez des spécifications d'audit de serveur et des spécifications d'audit de base de données, de la même manière que vous les créez pour vos serveurs de base de données sur site. Pour plus d'informations sur la création de spécifications d'audit, consultez CREATESERVERAUDITSPECIFICATIONet CREATEDATABASEAUDITSPECIFICATIONconsultez la documentation Microsoft SQL Server.

Pour éviter les erreurs, n'utilisez pas `RDS_` comme préfixe dans le nom de la spécification d'audit de base de données ou la spécification d'audit de serveur.

Consultation des journaux d'audit

Vos journaux d'audit sont stockés dans `D:\rdsdbdata\SQLAudit`.

Une fois que le SQL serveur a terminé d'écrire dans un fichier journal d'audit (lorsque le fichier atteint sa limite de taille), Amazon RDS télécharge le fichier dans votre compartiment S3. Si la rétention est activée, Amazon RDS déplace le fichier dans le dossier de rétention :`D:\rdsdbdata\SQLAudit\transmitted`.

Pour plus d'informations sur la rétention, consultez Ajout d'SQLun audit de serveur aux options de l'instance de base de données.

Les enregistrements d'audit sont conservés sur l'instance de base de données jusqu'à ce que le fichier de journal soit chargé. Vous pouvez afficher les enregistrements d'audit en exécutant la commande suivante.

```
SELECT   *
 FROM     msdb.dbo.rds_fn_get_audit_file
              ('D:\rdsdbdata\SQLAudit\*.sqlaudit'
              , default
              , default )
```

Vous pouvez utiliser la même commande pour afficher les enregistrements d'audit de votre dossier de rétention en modifiant le filtre en `D:\rdsdbdata\SQLAudit\transmitted\*.sqlaudit`.

```
SELECT   *
 FROM     msdb.dbo.rds_fn_get_audit_file
              ('D:\rdsdbdata\SQLAudit\transmitted\*.sqlaudit'
              , default
              , default )
```

Utilisation de SQL Server Audit avec des instances multi-AZ

Pour les instances multi-AZ, le processus d'envoi de fichiers de journal d'audit à Amazon S3 est similaire au processus utilisé pour les instances mono-AZ. Cependant, il existe quelques différences importantes :

- Les objets de spécification d'audit de base de données sont répliqués vers tous les nœuds.
- Les audits de serveur et les spécifications d'audit de serveur ne sont pas répliqués vers les nœuds secondaires. Vous devez les créer et les modifier manuellement.

Pour capturer des audits de serveur ou une spécification d'audit de serveur depuis les deux nœuds :

1. Créez un audit de serveur ou une spécification d'audit de serveur sur le nœud primaire.
2. Basculez vers le nœud secondaire et créez un audit de serveur ou une spécification d'audit de serveur portant le même nom et GUID sur le nœud secondaire. Utilisez le `AUDIT_GUID` paramètre pour spécifier leGUID.

Configuration d'un compartiment S3

Les fichiers journaux d'audit sont chargés automatiquement depuis l'instance de base de données vers votre compartiment S3. Les restrictions suivantes s'appliquent au compartiment S3 que vous utilisez comme cible pour vos fichiers d'audit :

- Elle doit se trouver dans la même AWS région que l'instance de base de données.
- Il ne doit pas être ouvert au public.
- Le propriétaire du bucket doit également être le propriétaire du IAM rôle.
- Votre IAM rôle doit disposer d'autorisations pour la KMS clé gérée par le client associée au chiffrement côté serveur du compartiment S3.

La clé cible qui est utilisée pour stocker les données suit ce schéma de dénomination : *`amzn-s3-demo-bucket`*`/key-prefix/instance-name/audit-name/node_file-name.ext`

> **Note**
>
> Vous définissez le nom du compartiment et les valeurs de préfixe de clé avec le paramètre d'option (`S3_BUCKET_ARN`).

Le schéma est composé des éléments suivants :

- ***`amzn-s3-demo-bucket`*** – Nom de votre compartiment S3.
- **`key-prefix`** – Préfixe de clé personnalisé que vous souhaitez utiliser pour les journaux d'audit.

- **`instance-name`**— Le nom de votre RDS instance Amazon.
- **`audit-name`** – Nom de l'audit.
- **`node`** – Identifiant du nœud constituant la source des journaux d'audit (`node1` or `node2`). Il existe un nœud pour une instance mono-AZ et deux nœuds de réplication pour une instance multi-AZ. Il ne s'agit pas des nœuds primaires et secondaires, car les rôles des nœuds primaires et secondaires au fil du temps. L'identificateur de nœud est une simple étiquette.
  - **`node1`** – Premier nœud de la réplication (une instance mono-AZ ne comporte qu'un seul nœud).
  - **`node2`** – Deuxième nœud de la réplication (une instance multi-AZ ne comporte deux nœuds).
- **`file-name`** – Nom du fichier cible. Le nom du fichier est extrait tel quel du SQL serveur.
- **`ext`** – Extension du fichier (`zip` ou `sqlaudit`):
  - **`zip`** – Si la compression est activée (par défaut).
  - **`sqlaudit`** – Si la compression est désactivée.

Création manuelle d'un rôle IAM pour SQL Server Audit

Généralement, lorsque vous créez une nouvelle option, le AWS Management Console rôle IAM et la politique de confiance IAM sont créés pour vous. Cependant, vous pouvez créer manuellement un rôle IAM à utiliser avec les audits SQL Server pour pouvoir le personnaliser avec les exigences supplémentaires que vous pourriez avoir. Pour ce faire, vous créez un rôle IAM et vous déléguez des autorisations pour que le service Amazon RDS puisse utiliser votre compartiment Amazon S3. Lorsque vous créez ce rôle IAM, vous attachez des politiques d'approbation et d'autorisation. La politique d'approbation permet à Amazon RDS d'assumer ce rôle. La politique d'autorisation définit les actions que ce rôle peut exécuter. Pour plus d'informations, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le guide de l'utilisateur d'AWS Identity and Access Management.

Vous pouvez utiliser les exemples de cette section pour créer les relations d'approbation et les politiques d'autorisation dont vous avez besoin.

Voici un exemple de relation d'approbation de rôle pour SQL Server Audit. Elle utilise le principal de service `rds.amazonaws.com` pour autoriser RDS à écrire dans le compartiment S3. Un principal de service est un identifiant utilisé pour accorder des autorisations à un service. Chaque fois que vous autorisez l'accès à `rds.amazonaws.com` de cette manière, vous autorisez RDS à exécuter une action en votre nom. Pour en savoir plus sur les principaux de service, veuillez consulter Éléments de politique JSON d'AWS : Principal.

Example relation d'approbation pour SQL Server Audit

JSON

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
             "Effect": "Allow",
             "Principal": {
                 "Service": "rds.amazonaws.com"
             },
             "Action": "sts:AssumeRole"
         }
     ]
 }
```

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des relations d'approbation basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la relation d'approbation, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'Amazon Resource Name (ARN) complet des ressources qui accèdent au rôle. Pour SQL Server Audit, veillez à inclure à la fois le groupe d'options de base de données et les instances de base de données, comme indiqué dans l'exemple suivant.

Example relation d'approbation avec la clé de contexte de condition globale pour SQL Server Audit

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "rds.amazonaws.com"
            },
            "Action": "sts:AssumeRole",
            "Condition": {
                "StringEquals": {
                    "aws:SourceArn": [

 "arn:aws:rds:Region:my_account_ID:db:db_instance_identifier",
                        "arn:aws:rds:Region:my_account_ID:og:option_group_name"
                    ]
                }
            }
        }
    ]
}
```

Dans l'exemple suivant de politique d'autorisation pour SQL Server Audit, nous spécifions un ARN pour le compartiment Amazon S3. Vous pouvez l'utiliser ARNs pour identifier un compte, un utilisateur ou un rôle spécifique auquel vous souhaitez accorder l'accès. Pour plus d'informations sur l'utilisation ARNs, consultez Amazon resource names (ARNs).

Example politique d'autorisations pour SQL Server Audit

JSON

```
{
     "Version": "2012-10-17",
     "Statement": [
         {
```

```
            "Effect": "Allow",
            "Action": "s3:ListAllMyBuckets",
            "Resource": "*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetBucketACL",
                "s3:GetBucketLocation"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
        },
        {
            "Effect": "Allow",
            "Action": [
                "s3:PutObject",
                "s3:ListMultipartUploadParts",
                "s3:AbortMultipartUpload"
            ],
            "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/key_prefix/*"
        }
    ]
}
```

> **Note**
>
> L'`s3:ListAllMyBuckets`action est requise pour vérifier que le même AWS compte possède à la fois le compartiment S3 et l'instance de base de données SQL Server. L'action répertorie les noms des compartiments du compte.
> Les espaces de noms de compartiment S3 sont globaux. Si vous supprimez accidentellement votre compartiment, un autre utilisateur peut créer un compartiment portant le même nom dans un compte différent. Ensuite, les données d'audit SQL Server sont écrites dans le nouveau compartiment.

Support pour les services d'analyse de SQL serveurs dans Amazon RDS for SQL Server

Microsoft SQL Server Analysis Services (SSAS) fait partie de la suite Microsoft Business Intelligence (MSBI). SSASest un outil de traitement analytique (OLAP) et d'exploration de données en ligne installé dans SQL Server. Vous l'utilisez SSAS pour analyser les données afin de prendre des décisions commerciales. SSASdiffère de la base de données relationnelle SQL Server car elle SSAS est optimisée pour les requêtes et les calculs courants dans un environnement de business intelligence.

Vous pouvez l'activer SSAS pour les instances de base de données existantes ou nouvelles. Il est installé sur la même instance de base de données que votre moteur de base de données. Pour plus d'informationsSSAS, consultez la documentation des services Microsoft Analysis.

Amazon RDS prend en SSAS charge les éditions SQL Server Standard et Enterprise sur les versions suivantes :

- Mode tabulaire :
  - SQLServer 2019, version 15.00.4043.16.v1 et supérieure
  - SQLServer 2017, version 14.00.3223.3.v1 et supérieure
  - SQLServer 2016, version 13.00.5426.0.v1 et supérieure
- Mode multidimensionnel :
  - SQLServer 2019, version 15.00.4153.1.v1 et supérieure
  - SQLServer 2017, version 14.00.3381.3.v1 et supérieure
  - SQLServer 2016, version 13.00.5882.1.v1 et supérieure

Table des matières

- Limites
- Activation d'SSAS
  - Création d'un groupe d'options pour SSAS
  - Ajouter l'SSASoption au groupe d'options
  - Association du groupe d'options à votre instance de base de données
  - Autoriser l'accès entrant à votre groupe VPC de sécurité
  - Activation de l'intégration Amazon S3

- Déploiement SSAS de projets sur Amazon RDS
- Surveillance de l'état d'une tâche de déploiement
- Utilisation SSAS sur Amazon RDS
  - Configuration d'un utilisateur authentifié par Windows pour SSAS
  - Ajout d'un utilisateur de domaine en tant qu'administrateur de base de données
  - Création d'un SSAS proxy
  - Planification du traitement des SSAS bases de données à l'aide SQL de l'agent
  - Révocation de SSAS l'accès depuis le proxy
- Sauvegarde d'une SSAS base de données
- Restauration d'une SSAS base de données
  - Restauration d'une instance de base de données à une date spécifiée
- Changer de SSAS mode
- Désactivation de SSAS
- SSASProblèmes de résolution des problèmes

Limites

Les restrictions suivantes s'appliquent à l'utilisation d'SSASon RDS for SQL Server :

- RDScar le SQL serveur prend en charge SSAS l'exécution en mode tabulaire ou multidimensionnel. Pour de plus amples informations, veuillez consulter Comparaison des solutions tabulaires et multidimensionnelles dans la documentation Microsoft.
- Vous ne pouvez utiliser qu'un seul SSAS mode à la fois. Avant de changer de mode, assurez-vous de supprimer toutes les SSAS bases de données.

  Pour de plus amples informations, veuillez consulter Changer de SSAS mode.
- Les instances multi-AZ ne sont pas prises en charge.
- Les instances doivent utiliser Active Directory autogéré ou AWS Directory Service for Microsoft Active Directory pour SSAS l'authentification. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.
- Les utilisateurs ne disposent pas de l'accès administrateur SSAS du serveur, mais ils peuvent obtenir un accès administrateur au niveau de la base de données.
- Le seul port pris en charge pour l'accès SSAS est le 2383.

- Vous ne pouvez pas déployer de projets directement. Pour ce faire, nous proposons une procédure RDS stockée. Pour de plus amples informations, veuillez consulter Déploiement SSAS de projets sur Amazon RDS.
- Le traitement pendant le déploiement n'est pas pris en charge.
- L'utilisation de fichiers .xmla pour le déploiement n'est pas prise en charge.
- SSASles fichiers d'entrée du projet et les fichiers de sortie de sauvegarde de la base de données ne peuvent se trouver que dans le `D:\S3` dossier de l'instance de base de données.

Activation d'SSAS

Utilisez le processus suivant SSAS pour activer votre instance de base de données :

1. Créez un groupe d'options ou choisissez un groupe d'options existant.
2. Ajoutez l'option `SSAS` au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.
4. Autorisez l'accès entrant au groupe de sécurité du cloud privé virtuel (VPC) pour le port d'SSASécoute.
5. Activez l'intégration Amazon S3.

Création d'un groupe d'options pour SSAS

Utilisez le AWS Management Console ou AWS CLI pour créer un groupe d'options correspondant au moteur de SQL serveur et à la version de l'instance de base de données que vous prévoyez d'utiliser.

> **Note**
>
> Vous pouvez également utiliser un groupe d'options existant s'il correspond au moteur de SQL serveur et à la version appropriés.

Console

La procédure de console suivante crée un groupe d'options pour SQL Server Standard Edition 2017.

Pour créer le groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
   a. Dans Nom, entrez un nom unique au sein de votre AWS compte pour le groupe d'options, tel que**ssas-se-2017**. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
   b. Pour Description, saisissez une brève description du groupe d'options, par exemple **SSAS option group for SQL Server SE 2017**. La description est utilisée à des fins d'affichage.
   c. Pour Moteur, choisissez sqlserver-se.
   d. Pour Version majeure du moteur, choisissez 14.00.
5. Sélectionnez Create (Créer).

CLI

L'CLIexemple suivant crée un groupe d'options pour SQL Server Standard Edition 2017.

Pour créer le groupe d'options

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-option-group \
    --option-group-name ssas-se-2017 \
    --engine-name sqlserver-se \
    --major-engine-version 14.00 \
    --option-group-description "SSAS option group for SQL Server SE 2017"
```

  Dans Windows:

```
aws rds create-option-group ^
```

```
    --option-group-name ssas-se-2017 ^
    --engine-name sqlserver-se ^
    --major-engine-version 14.00 ^
    --option-group-description "SSAS option group for SQL Server SE 2017"
```

Ajouter l'SSASoption au groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'SSASoption au groupe d'options.

Console

Pour ajouter l'SSASoption

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options que vous venez de créer.
4. Sélectionnez Ajouter une option.
5. Sous Détails de l'option, choisissez SSASIe nom de l'option.
6. Sous Option settings (Paramètres d'option), procédez comme suit :

    a. Pour Max memory (Mémoire maximale), saisissez une valeur comprise entre 10 et 80.

    La mémoire maximale indique le seuil supérieur au-dessus duquel SSAS commence à libérer de la mémoire de manière plus agressive pour faire de la place aux demandes en cours d'exécution, ainsi qu'aux nouvelles demandes hautement prioritaires. Le nombre correspond à un pourcentage de la mémoire totale de l'instance de base de données. Les valeurs autorisées sont 10–80, et celle par défaut est 45.

    b. Pour Mode, choisissez le mode SSAS serveur, tabulaire ou multidimensionnel.

    Si le paramètre Mode n'apparaît pas, cela signifie que le mode multidimensionnel n'est pas pris en charge dans votre AWS région. Pour de plus amples informations, veuillez consulter Limites.

    Le mode Tabular (Tabulaire) est le mode par défaut.

    c. Pour les groupes de sécurité, choisissez le groupe de VPC sécurité à associer à l'option.

> **Note**
>
> Le port d'accèsSSAS, 2383, est prérempli.

7. Sous Scheduling (Planification), choisissez si vous souhaitez ajouter l'option immédiatement ou lors du créneau de maintenance suivant.
8. Sélectionnez Ajouter une option.

CLI

Pour ajouter l'SSASoption

1. Créez un JSON fichier, par exemple`ssas-option.json`, avec les paramètres suivants :

   - `OptionGroupName` – Nom du groupe d'options que vous avez créé ou choisi précédemment (`ssas-se-2017` dans l'exemple suivant).
   - `Port`— Le port que vous utilisez pour accéderSSAS. Le seul port pris en charge est 2383.
   - `VpcSecurityGroupMemberships`— Appartenances à des groupes VPC de sécurité pour votre RDS instance de base de données.
   - `MAX_MEMORY`— Le seuil supérieur au-dessus duquel SSAS devrait commencer à libérer de la mémoire de manière plus agressive pour faire de la place aux demandes en cours d'exécution, ainsi qu'aux nouvelles demandes hautement prioritaires. Le nombre correspond à un pourcentage de la mémoire totale de l'instance de base de données. Les valeurs autorisées sont 10–80, et celle par défaut est 45.
   - `MODE`— Le mode SSAS serveur, `Tabular` soit`Multidimensional`. `Tabular`est la valeur par défaut.

     Si vous recevez un message d'erreur indiquant que le paramètre d'`MODE`option n'est pas valide, cela signifie que le mode multidimensionnel n'est pas pris en charge dans votre AWS région. Pour de plus amples informations, veuillez consulter Limites.

   Voici un exemple de JSON fichier avec des paramètres d'SSASoptions.

```
{
"OptionGroupName": "ssas-se-2017",
```

```
"OptionsToInclude": [
 {
 "OptionName": "SSAS",
 "Port": 2383,
 "VpcSecurityGroupMemberships": ["sg-0abcdef123"],
 "OptionSettings": [{"Name":"MAX_MEMORY","Value":"60"},
{"Name":"MODE","Value":"Multidimensional"}]
 }],
"ApplyImmediately": true
}
```

2. Ajoutez l'option SSAS au groupe d'options.

   Example

   Dans Linux, macOS, ou Unix:

   ```
   aws rds add-option-to-option-group \
       --cli-input-json file://ssas-option.json \
       --apply-immediately
   ```

   Dans Windows:

   ```
   aws rds add-option-to-option-group ^
       --cli-input-json file://ssas-option.json ^
       --apply-immediately
   ```

Association du groupe d'options à votre instance de base de données

Vous pouvez utiliser la console ou le CLI pour associer le groupe d'options à votre instance de base de données.

Console

Associez votre groupe d'options à une instance de base de données nouvelle ou existante :

- Pour une nouvelle instance de base de données, associez le groupe d'options à l'instance de base de données lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

- Pour une instance de base de données existante, modifiez l'instance et associez-lui le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

  > **Note**
  >
  > Si vous utilisez une instance existante, elle doit déjà être associée à un domaine Active Directory et à un rôle AWS Identity and Access Management (IAM). Si vous créez une nouvelle instance, spécifiez un domaine et un IAM rôle Active Directory existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.

CLI

Vous pouvez associer votre groupe d'options à une instance de base de données nouvelle ou existante.

> **Note**
>
> Si vous utilisez une instance existante, elle doit déjà être associée à un domaine et à un IAM rôle Active Directory. Si vous créez une nouvelle instance, spécifiez un domaine et un IAM rôle Active Directory existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.

Pour créer une instance de base de données utilisant le groupe d'options

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier myssasinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 14.00.3223.3.v1 \
```

```
    --allocated-storage 100 \
    --manage-master-user-password \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --domain-iam-role-name my-directory-iam-role \
    --domain my-domain-id \
    --option-group-name ssas-se-2017
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier myssasinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 14.00.3223.3.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --domain-iam-role-name my-directory-iam-role ^
    --domain my-domain-id ^
    --option-group-name ssas-se-2017
```

Pour modifier une instance de base de données afin d'y associer le groupe d'options

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier myssasinstance \
    --option-group-name ssas-se-2017 \
    --apply-immediately
```

  Dans Windows:

```
aws rds modify-db-instance ^
```

```
    --db-instance-identifier myssasinstance ^
    --option-group-name ssas-se-2017 ^
    --apply-immediately
```

Autoriser l'accès entrant à votre groupe VPC de sécurité

Créez une règle entrante pour le port d'SSASécouteur spécifié dans le groupe de VPC sécurité associé à votre instance de base de données. Pour de plus amples informations sur la configuration des groupes de sécurité, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Activation de l'intégration Amazon S3

Pour télécharger des fichiers de configuration de modèle sur votre hôte pour le déploiement, utilisez l'intégration Amazon S3. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3.

Déploiement SSAS de projets sur Amazon RDS

ActivéRDS, vous ne pouvez pas déployer de SSAS projets directement à l'aide de SQL Server Management Studio (SSMS). Pour déployer des projets, utilisez une procédure RDS stockée.

> **Note**
>
> L'utilisation de fichiers .xmla pour le déploiement n'est pas prise en charge.

Avant de déployer des projets, vérifiez les points suivants :

- L'intégration Amazon S3 est activée. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3.
- Le paramètre de configuration `Processing Option` est défini sur `Do Not Process`. Ce paramètre signifie qu'aucun traitement n'est effectué après le déploiement.
- Vous disposez des fichiers *`myssasproject`*`.asdatabase` et *`myssasproject`*`.deploymentoptions`. Ils sont générés automatiquement lorsque vous créez le SSAS projet.

Pour déployer un SSAS projet sur RDS

1. Téléchargez le fichier `.asdatabase` (SSASmodèle) de votre compartiment S3 vers votre instance de base de données, comme indiqué dans l'exemple suivant. Pour plus d'informations sur les paramètres de téléchargement, consultez Téléchargement de fichiers depuis un compartiment Amazon S3 vers une instance de base de données de SQL serveur.

```
exec msdb.dbo.rds_download_from_s3
@s3_arn_of_file='arn:aws:s3:::bucket_name/myssasproject.asdatabase',
[@rds_file_path='D:\S3\myssasproject.asdatabase'],
[@overwrite_file=1];
```

2. Téléchargez le fichier `.deploymentoptions` de votre compartiment S3 dans votre instance de base de données.

```
exec msdb.dbo.rds_download_from_s3
@s3_arn_of_file='arn:aws:s3:::bucket_name/myssasproject.deploymentoptions',
[@rds_file_path='D:\S3\myssasproject.deploymentoptions'],
[@overwrite_file=1];
```

3. Déployez le projet.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSAS_DEPLOY_PROJECT',
@file_path='D:\S3\myssasproject.asdatabase';
```

Surveillance de l'état d'une tâche de déploiement

Pour suivre l'état de votre tâche de déploiement (ou de téléchargement), appelez la fonction `rds_fn_task_status`. Deux paramètres sont nécessaires. Le premier paramètre doit toujours être `NULL` parce qu'il ne s'applique pas àSSAS. Le second paramètre accepte l'ID de tâche.

Pour consulter une liste de toutes les tâches, définissez le premier paramètre sur `NULL` et le second sur `0`, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,0);
```

Pour obtenir une tâche spécifique, définissez le premier paramètre sur `NULL` et le second sur l'ID de tâche, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,42);
```

La fonction `rds_fn_task_status` retourne les informations suivantes.

Paramètre de sortie	Description
`task_id`	ID de la tâche
`task_type`	En SSAS effet, les tâches peuvent avoir les types de tâches suivants : • `SSAS_DEPLOY_PROJECT` • `SSAS_ADD_DB_ADMIN_MEMBER` • `SSAS_BACKUP_DB` • `SSAS_RESTORE_DB`
`database_name`	Non applicable aux SSAS tâches.
`% complete`	Progression de la tâche sous forme de pourcentage.
`duration (mins)`	Temps consacré à la tâche, en minutes.
`lifecycle`	État de la tâche. Les statuts possibles sont les suivants : • `CREATED`— Après avoir appelé l'une des procédures SSAS stockées, une tâche est créée et son statut est défini sur`CREATED`. • `IN_PROGRESS` – Après le démarrage d'une tâche, le statut est défini sur `IN_PROGRESS` . Le passage du statut `CREATED` à `IN_PROGRESS` peut prendre jusqu'à cinq minutes. •

Paramètre de sortie	Description
	SUCCESS – Lorsqu'une tâche est terminée, le statut est défini sur SUCCESS. • ERROR – Si une tâche échoue, le statut est défini sur ERROR. Lisez la colonne task_info pour plus d'informations sur l'erreur. • CANCEL_REQUESTED – Après que vous avez appelé rds_cancel_task , le statut de la tâche est défini sur CANCEL_REQUESTED . • CANCELLED – Une fois une tâche annulée avec succès, l'état de la tâche est défini sur CANCELLED .
task_info	Informations supplémentaires sur la tâche. Si une erreur se produit pendant le traitement, cette colonne contient des informations sur l'erreur. Pour de plus amples informations, veuillez consulter SSASProblèmes de résolution des problèmes.
last_updated	Date et heure de la dernière mise à jour de l'état de la tâche.
created_at	Date et heure de création de la tâche.
S3_object_arn	Non applicable aux SSAS tâches.
overwrite_S3_backup_file	Non applicable aux SSAS tâches.

Paramètre de sortie	Description
`KMS_master_key_arn`	Non applicable aux SSAS tâches.
`filepath`	Non applicable aux SSAS tâches.
`overwrite_file`	Non applicable aux SSAS tâches.
`task_metadata`	Métadonnées associées à la SSAS tâche.

Utilisation SSAS sur Amazon RDS

Après avoir déployé le SSAS projet, vous pouvez directement traiter la OLAP base de données surSSMS.

À utiliser SSAS sur RDS

1. DansSSMS, connectez-vous à SSAS l'aide du nom d'utilisateur et du mot de passe du domaine Active Directory.
2. Développez Bases de données. La base de SSAS données récemment déployée apparaît.
3. Localisez la chaîne de connexion et mettez à jour le nom d'utilisateur et le mot de passe pour donner accès à la SQL base de données source. Cela est nécessaire pour le traitement SSAS des objets.

    a. Pour le mode tabulaire, procédez comme suit :

        1. Développez l'onglet Connections (Connexions).
        2. Ouvrez le menu contextuel (clic droit) de l'objet de connexion, puis choisissez Properties (Propriétés).
        3. Mettez à jour le nom d'utilisateur et le mot de passe dans la chaîne de connexion.

    b. Pour le mode multidimensionnel, procédez comme suit :

        1. Développez l'onglet Data Sources (Sources de données).
        2. Ouvrez le menu contextuel (clic droit) de la source de données, puis choisissez Properties (Propriétés).

3. Mettez à jour le nom d'utilisateur et le mot de passe dans la chaîne de connexion.

4. Ouvrez le menu contextuel (clic droit) de la SSAS base de données que vous avez créée et choisissez Process Database.

   Selon la taille des données d'entrée, le traitement peut prendre plusieurs minutes.

Rubriques

- Configuration d'un utilisateur authentifié par Windows pour SSAS
- Ajout d'un utilisateur de domaine en tant qu'administrateur de base de données
- Création d'un SSAS proxy
- Planification du traitement des SSAS bases de données à l'aide SQL de l'agent
- Révocation de SSAS l'accès depuis le proxy

Configuration d'un utilisateur authentifié par Windows pour SSAS

L'utilisateur administrateur principal (parfois appelé utilisateur principal) peut utiliser l'exemple de code suivant pour configurer une connexion authentifiée par Windows et accorder les autorisations de procédure requises. Cela permet à l'utilisateur du domaine d'exécuter les tâches du SSAS client, d'utiliser les procédures de transfert de fichiers S3, de créer des informations d'identification et de travailler avec le proxy de l'agent SQL serveur. Pour plus d'informations, consultez les sections Informations d'identification (moteur de base de données) et Création d'un proxy d'agent SQL serveur dans la documentation Microsoft.

Vous pouvez accorder quelques-unes ou la totalité des autorisations suivantes, selon les besoins, aux utilisateurs authentifiés par Windows.

Example

```
-- Create a server-level domain user login, if it doesn't already exist
USE [master]
GO
CREATE LOGIN [mydomain\user_name] FROM WINDOWS
GO

-- Create domain user, if it doesn't already exist
USE [msdb]
GO
CREATE USER [mydomain\user_name] FOR LOGIN [mydomain\user_name]
```

```
GO

-- Grant necessary privileges to the domain user
USE [master]
GO
GRANT ALTER ANY CREDENTIAL TO [mydomain\user_name]
GO

USE [msdb]
GO
GRANT EXEC ON msdb.dbo.rds_msbi_task TO [mydomain\user_name] with grant option
GRANT SELECT ON msdb.dbo.rds_fn_task_status TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_task_status TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_cancel_task TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_download_from_s3 TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_upload_to_s3 TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_delete_from_filesystem TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.rds_gather_file_details TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_add_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_update_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_grant_login_to_proxy TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_revoke_login_from_proxy TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_delete_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_enum_login_for_proxy to [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_enum_proxy_for_subsystem TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.rds_sqlagent_proxy TO [mydomain\user_name] with grant option
ALTER ROLE [SQLAgentUserRole] ADD MEMBER [mydomain\user_name]
GO
```

Ajout d'un utilisateur de domaine en tant qu'administrateur de base de données

Vous pouvez ajouter un utilisateur de domaine en tant qu'administrateur SSAS de base de données de différentes manières :

- Un administrateur de base de données peut SSMS créer un rôle avec des `admin` privilèges, puis ajouter des utilisateurs à ce rôle.
- Vous pouvez utiliser la procédure stockée suivante.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSAS_ADD_DB_ADMIN_MEMBER',
@database_name='myssasdb',
@ssas_role_name='exampleRole',
@ssas_role_member='domain_name\domain_user_name';
```

Les paramètres suivants sont obligatoires :

- `@task_type`— Le type de MSBI tâche, dans ce cas`SSAS_ADD_DB_ADMIN_MEMBER`.
- `@database_name`— Le nom de la SSAS base de données à laquelle vous accordez des privilèges d'administrateur.
- `@ssas_role_name`— Nom du rôle d'administrateur de SSAS base de données. Si le rôle n'existe pas déjà, il est créé.
- `@ssas_role_member`— L'utilisateur SSAS de base de données que vous ajoutez au rôle d'administrateur.

Création d'un SSAS proxy

Pour pouvoir planifier le traitement de la SSAS base de données à l'aide de l'agent SQL serveur, créez un SSAS identifiant et un SSAS proxy. Exécutez ces procédures en tant qu'utilisateur authentifié par Windows.

Pour créer les SSAS informations d'identification

- Créez les informations d'identification pour le proxy. Pour ce faire, vous pouvez utiliser SSMS l'SQLinstruction suivante.

```
USE [master]
GO
CREATE CREDENTIAL [SSAS_Credential] WITH IDENTITY = N'mydomain\user_name', SECRET =
 N'mysecret'
GO
```

> **Note**
>
> `IDENTITY` doit être une connexion authentifiée par domaine. Remplacez *`mysecret`* par le mot de passe de la connexion authentifiée par le domaine.

Pour créer le SSAS proxy

1. Utilisez l'SQLinstruction suivante pour créer le proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_add_proxy
 @proxy_name=N'SSAS_Proxy',@credential_name=N'SSAS_Credential',@description=N''
GO
```

2. Utilisez l'SQLinstruction suivante pour accorder l'accès au proxy à d'autres utilisateurs.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_grant_login_to_proxy
 @proxy_name=N'SSAS_Proxy',@login_name=N'mydomain\user_name'
GO
```

3. Utilisez l'SQLinstruction suivante pour autoriser le SSAS sous-système à accéder au proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.rds_sqlagent_proxy
 @task_type='GRANT_SUBSYSTEM_ACCESS',@proxy_name='SSAS_Proxy',@proxy_subsystem='SSAS'
GO
```

Pour afficher le proxy et les octrois sur le proxy

1. Utilisez l'SQLinstruction suivante pour voir les bénéficiaires du proxy.

```
USE [msdb]
GO
EXEC sp_help_proxy
GO
```

2. Utilisez l'SQLinstruction suivante pour afficher les autorisations des sous-systèmes.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_enum_proxy_for_subsystem
GO
```

Planification du traitement des SSAS bases de données à l'aide SQL de l'agent

Après avoir créé les informations d'identification et le proxy et accordé l'SSASaccès au proxy, vous pouvez créer une tâche d'agent SQL serveur pour planifier le traitement de la SSAS base de données.

Pour planifier le traitement de la SSAS base de données

- Utilisez SSMS ou T- SQL pour créer la tâche d'agent de SQL serveur. L'exemple suivant utilise T-SQL. Vous pouvez configurer davantage son calendrier de travail via SSMS ou T-SQL.
  - Le `@command` paramètre décrit la commande XML for Analysis (XMLA) à exécuter par le job d'agent de SQL serveur. Cet exemple configure le traitement de base de données SSAS multidimensionnelle.
  - Le `@server` paramètre indique le nom SSAS du serveur cible de la tâche d'agent de SQL serveur.

    Pour appeler le SSAS service au sein de la même RDS instance de base de données où réside la tâche de l'agent SQL serveur, utilisez`localhost:2383`.

    Pour appeler le SSAS service depuis l'extérieur de l'RDSinstance de base de données, utilisez le RDS point de terminaison. Vous pouvez également utiliser le point de terminaison Kerberos Active Directory (AD) (*`your-DB-instance-name`*`.`*`your-AD-domain-name`*) si les RDS instances de base de données sont jointes par le même domaine. Pour les instances de base de données externes, assurez-vous de configurer correctement le groupe VPC de sécurité associé à l'RDSinstance de base de données pour une connexion sécurisée.

  Vous pouvez également modifier la requête pour prendre en charge diverses XMLA opérations. Apportez des modifications soit en modifiant directement la SQL requête T, soit en utilisant l'SSMSinterface utilisateur après la création de la tâche de l'agent SQL serveur.

```
USE [msdb]
GO
DECLARE @jobId BINARY(16)
EXEC msdb.dbo.sp_add_job @job_name=N'SSAS_Job',
    @enabled=1,
    @notify_level_eventlog=0,
    @notify_level_email=0,
    @notify_level_netsend=0,
    @notify_level_page=0,
```

```
    @delete_level=0,
    @category_name=N'[Uncategorized (Local)]',
    @job_id = @jobId OUTPUT
GO
EXEC msdb.dbo.sp_add_jobserver
    @job_name=N'SSAS_Job',
    @server_name = N'(local)'
GO
EXEC msdb.dbo.sp_add_jobstep @job_name=N'SSAS_Job',
 @step_name=N'Process_SSAS_Object',
    @step_id=1,
    @cmdexec_success_code=0,
    @on_success_action=1,
    @on_success_step_id=0,
    @on_fail_action=2,
    @on_fail_step_id=0,
    @retry_attempts=0,
    @retry_interval=0,
    @os_run_priority=0, @subsystem=N'ANALYSISCOMMAND',
    @command=N'<Batch xmlns="http://schemas.microsoft.com/analysisservices/2003/
engine">
        <Parallel>
            <Process xmlns:xsd="http://www.w3.org/2001/XMLSchema"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                xmlns:ddl2="http://schemas.microsoft.com/analysisservices/2003/
engine/2" xmlns:ddl2_2="http://schemas.microsoft.com/analysisservices/2003/
engine/2/2"
                xmlns:ddl100_100="http://schemas.microsoft.com/
analysisservices/2008/engine/100/100" xmlns:ddl200="http://schemas.microsoft.com/
analysisservices/2010/engine/200"
                xmlns:ddl200_200="http://schemas.microsoft.com/
analysisservices/2010/engine/200/200" xmlns:ddl300="http://schemas.microsoft.com/
analysisservices/2011/engine/300"
                xmlns:ddl300_300="http://schemas.microsoft.com/
analysisservices/2011/engine/300/300" xmlns:ddl400="http://schemas.microsoft.com/
analysisservices/2012/engine/400"
                xmlns:ddl400_400="http://schemas.microsoft.com/
analysisservices/2012/engine/400/400" xmlns:ddl500="http://schemas.microsoft.com/
analysisservices/2013/engine/500"
                xmlns:ddl500_500="http://schemas.microsoft.com/
analysisservices/2013/engine/500/500">
                <Object>
                    <DatabaseID>Your_SSAS_Database_ID</DatabaseID>
                </Object>
```

```
                <Type>ProcessFull</Type>
                <WriteBackTableCreation>UseExisting</WriteBackTableCreation>
            </Process>
        </Parallel>
    </Batch>',
    @server=N'localhost:2383',
    @database_name=N'master',
    @flags=0,
    @proxy_name=N'SSAS_Proxy'
GO
```

Révocation de SSAS l'accès depuis le proxy

Vous pouvez révoquer l'accès au SSAS sous-système et supprimer le SSAS proxy à l'aide des procédures stockées suivantes.

Pour révoquer l'accès et supprimer le proxy

1. Révoquez l'accès au sous-système.

```
USE [msdb]
GO
EXEC msdb.dbo.rds_sqlagent_proxy
 @task_type='REVOKE_SUBSYSTEM_ACCESS',@proxy_name='SSAS_Proxy',@proxy_subsystem='SSAS'
GO
```

2. Révoquez les octrois sur le proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_revoke_login_from_proxy
 @proxy_name=N'SSAS_Proxy',@name=N'mydomain\user_name'
GO
```

3. Supprimez le proxy.

```
USE [msdb]
GO
EXEC dbo.sp_delete_proxy @proxy_name = N'SSAS_Proxy'
GO
```

Sauvegarde d'une SSAS base de données

Vous pouvez créer des fichiers SSAS de sauvegarde de base de données uniquement dans le `D:\S3` dossier de l'instance de base de données. Pour déplacer les fichiers de sauvegarde vers votre compartiment S3, utilisez Amazon S3.

Vous pouvez sauvegarder une SSAS base de données comme suit :

- Un utilisateur de domaine ayant le `admin` rôle pour une base de données particulière peut l'utiliser SSMS pour sauvegarder la base de données `D:\S3` dans le dossier.

  Pour de plus amples informations, veuillez consulter Ajout d'un utilisateur de domaine en tant qu'administrateur de base de données.
- Vous pouvez utiliser la procédure stockée suivante. Cette procédure stockée ne prend pas en charge le chiffrement.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSAS_BACKUP_DB',
@database_name='myssasdb',
@file_path='D:\S3\ssas_db_backup.abf',
[@ssas_apply_compression=1],
[@ssas_overwrite_file=1];
```

  Les paramètres suivants sont obligatoires :

  - `@task_type`— Le type de MSBI tâche, dans ce cas`SSAS_BACKUP_DB`.
  - `@database_name`— Le nom de la SSAS base de données que vous sauvegardez.
  - `@file_path`— Le chemin du fichier SSAS de sauvegarde. L'extension `.abf` est requise.

  Les paramètres suivants sont facultatifs :

  - `@ssas_apply_compression`— S'il faut appliquer une compression de SSAS sauvegarde. Les valeurs valides sont 1 (Oui) et 0 (Non).
  - `@ssas_overwrite_file`— S'il faut remplacer le fichier de SSAS sauvegarde. Les valeurs valides sont 1 (Oui) et 0 (Non).

Restauration d'une SSAS base de données

Utilisez la procédure stockée suivante pour restaurer une SSAS base de données à partir d'une sauvegarde.

Vous ne pouvez pas restaurer une base de données s'il existe déjà une SSAS base de données portant le même nom. La procédure stockée pour la restauration ne prend pas en charge les fichiers de sauvegarde chiffrés.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSAS_RESTORE_DB',
@database_name='mynewssasdb',
@file_path='D:\S3\ssas_db_backup.abf';
```

Les paramètres suivants sont obligatoires :

- `@task_type`— Le type de MSBI tâche, dans ce cas`SSAS_RESTORE_DB`.
- `@database_name`— Le nom de la nouvelle SSAS base de données vers laquelle vous effectuez la restauration.
- `@file_path`— Le chemin d'accès au fichier SSAS de sauvegarde.

Restauration d'une instance de base de données à une date spécifiée

Point-in-time recovery (PITR) ne s'applique pas aux SSAS bases de données. Dans ce casPITR, seules les SSAS données du dernier instantané avant l'heure demandée sont disponibles sur l'instance restaurée.

Pour disposer de up-to-date SSAS bases de données sur une instance de base de données restaurée

1. Sauvegardez vos SSAS bases de données `D:\S3` dans le dossier de l'instance source.
2. Transférez les fichiers de sauvegarde dans le compartiment S3.
3. Transférez les fichiers de sauvegarde du compartiment S3 vers le dossier `D:\S3` de l'instance restaurée.
4. Exécutez la procédure stockée pour restaurer les SSAS bases de données sur l'instance restaurée.

   Vous pouvez également retraiter le SSAS projet pour restaurer les bases de données.

Changer de SSAS mode

Vous pouvez modifier le mode d'SSASexécution, tabulaire ou multidimensionnel. Pour changer de mode, utilisez le AWS Management Console ou AWS CLI pour modifier les paramètres des options de l'SSASoption.

> **Important**
>
> Vous ne pouvez utiliser qu'un seul SSAS mode à la fois. Assurez-vous de supprimer toutes les SSAS bases de données avant de changer de mode, sinon vous recevrez un message d'erreur.

Console

La procédure suivante de RDS la console Amazon change le SSAS mode en mode tabulaire et définit le `MAX_MEMORY` paramètre à 70 %.

Pour modifier l'SSASoption

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `SSAS` que vous souhaitez modifier (`ssas-se-2017` dans les exemples précédents).
4. Choisissez Modify option (Modifier l'option).
5. Modifiez les paramètres d'option :
   a. Pour Max memory (Mémoire maximale), saisissez **`70`**.
   b. Pour Mode, choisissez Tabular (Tabulaire).
6. Choisissez Modify option (Modifier l'option).

AWS CLI

L' AWS CLI exemple suivant change le SSAS mode en mode tabulaire et définit le `MAX_MEMORY` paramètre sur 70 %.

Pour que la CLI commande fonctionne, assurez-vous d'inclure tous les paramètres requis, même si vous ne les modifiez pas.

Pour modifier l'SSASoption

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name ssas-se-2017 \
    --options
 "OptionName=SSAS,VpcSecurityGroupMemberships=sg-12345e67,OptionSettings=[{Name=MAX_MEMORY,
{Name=MODE,Value=Tabular}]" \
    --apply-immediately
```

  Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name ssas-se-2017 ^
    --options
 OptionName=SSAS,VpcSecurityGroupMemberships=sg-12345e67,OptionSettings=[{Name=MAX_MEMORY,V
{Name=MODE,Value=Tabular}] ^
    --apply-immediately
```

Désactivation de SSAS

Pour la désactiverSSAS, supprimez l'SSASoption de son groupe d'options.

> **Important**
>
> Avant de supprimer SSAS cette option, supprimez vos SSAS bases de données.
> Nous vous recommandons vivement de sauvegarder vos SSAS bases de données avant de les supprimer et de supprimer l'SSASoption.

Console

Pour supprimer l'SSASoption de son groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `SSAS` que vous souhaitez supprimer (`ssas-se-2017` dans les exemples précédents).
4. Choisissez Supprimer une option.
5. Sous Options de suppression, sélectionnez SSASOptions à supprimer.
6. Sous Appliquer immédiatement, choisissez Oui pour supprimer l'option immédiatement, ou Non pour la supprimer lors du prochain créneau de maintenance.
7. Sélectionnez Delete (Supprimer).

AWS CLI

Pour supprimer l'SSASoption de son groupe d'options

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds remove-option-from-option-group \
    --option-group-name ssas-se-2017 \
    --options SSAS \
    --apply-immediately
```

  Dans Windows:

```
aws rds remove-option-from-option-group ^
    --option-group-name ssas-se-2017 ^
    --options SSAS ^
    --apply-immediately
```

SSASProblèmes de résolution des problèmes

Vous pouvez rencontrer les problèmes suivants lors de l'utilisation deSSAS.

Problème	Type	Suggestions de dépannage
Impossible de configurer l'SSASoption. Le SSAS mode demandé est *new_mode*, mais l'instance de base de données actuelle a *number current_mode* bases de données. Supprimez les bases de données existantes avant de passer à *new_mode* mode. Pour retrouver l'accès à *current_mode* mode de suppression de base de données, mettez à jour le groupe d'options de base de données actuel ou attachez un nouveau groupe d'options avec %s comme valeur de paramètre d'MODEoption pour l'SSASoption.	RDSévénement	Vous ne pouvez pas modifier le SSAS mode si des SSAS bases de données utilisent toujours le mode actuel. Supprimez les SSAS bases de données, puis réessayez.
Impossible de supprimer l'SSASoption car il existe *number* existant *mode* bases de données. L'SSASoption ne peut pas être supprimée tant que toutes les SSAS bases de données ne sont pas supprimées. Ajoutez à nouveau l'SSASoption, supprimez toutes les SSAS bases de données et réessayez.	RDSévénement	Vous ne pouvez pas le désactiver SSAS si vous avez toujours des SSAS bases de données. Supprimez les SSAS bases de données, puis réessayez.
L'SSASoption n'est pas activée ou est en cours d'activation. Réessayez ultérieurement.	RDSprocédure stockée	Vous ne pouvez pas exécuter de procédures SSAS stockées lorsque l'option est désactivée ou lorsqu'elle est activée.

Problème	Type	Suggestions de dépannage
L'SSASoption n'est pas configurée correctement. Assurez-vous que le statut d'adhésion au groupe d'options est « synchronisé » et consultez les journaux d'RDSévénements pour détecter les messages d'erreur SSAS de configuration pertinents. Ensuite, réessayez. Si des erreurs persistent, contactez le AWS Support.	RDSprocédure stockée	Vous ne pouvez pas exécuter de procédures SSAS stockées lorsque le `in-sync` statut de votre appartenance à un groupe d'options n'est pas indiqué. Cela place l'SSASoption dans un état de configuration incorrect. Si le statut de membre de votre groupe d'options change `failed` en raison d'SSASune modification des options, il y a deux raisons possibles : 1. L'SSASoption a été supprimée sans que les SSAS bases de données aient été supprimées. 2. Le SSAS mode a été mis à jour de tabulaire à multidimensionnel, ou de multidimensionnel à tabulaire, sans que les bases de données existantes aient été supprimées. SSAS Reconfigurez l'SSASoption, car elle n'RDSautorise qu'un seul SSAS mode à la fois et ne prend pas en charge la suppression d'SSASoptions en présence de SSAS bases de données. Vérifiez les journaux RDS d'événements pour détecter les erreurs de configuration de votre SSAS instance et résolvez les problèmes en conséquence.

Problème	Type	Suggestions de dépannage
Le déploiement a échoué. La modification ne peut être déployée que sur un serveur exécuté dans *deployment_file_mode* mode. Le mode serveur actuel est *current_mode* .	RDSprocédure stockée	Vous ne pouvez pas déployer une base de données tabulaire sur un serveur multidimensionnel ou une base de données multidimensionnelle sur un serveur tabulaire. Veillez à utiliser des fichiers avec le mode approprié et vérifiez que le paramètre d'option `MODE` est défini sur la valeur appropriée.
La restauration a échoué. Le fichier de sauvegarde ne peut être restauré que sur un serveur exécuté dans *restore_file_mode* mode. Le mode serveur actuel est *current_mode* .	RDSprocédure stockée	Vous ne pouvez pas restaurer une base de données tabulaire sur un serveur multidimensionnel ou une base de données multidimensionnelle sur un serveur tabulaire. Veillez à utiliser des fichiers avec le mode approprié et vérifiez que le paramètre d'option `MODE` est défini sur la valeur appropriée.
La restauration a échoué. Les versions du fichier de sauvegarde et de l'RDSinstance de base de données sont incompatibles.	RDSprocédure stockée	Vous ne pouvez pas restaurer une SSAS base de données dont la version est incompatible avec la version de l'instance SQL du serveur. Pour de plus amples informations, veuillez consulter Niveau de compatibilité pour les modèles tabulaires et Niveau de compatibilité d'une base de données multidimensionnelle (Analysis Services) dans la documentation Microsoft.

Problème	Type	Suggestions de dépannage
La restauration a échoué. Le fichier de sauvegarde spécifié lors de l'opération de restauration est endommagé ou n'est pas un fichier SSAS de sauvegarde. Assurez-vous que @rds_file_path est correctement formaté.	RDSprocédure stockée	Vous ne pouvez pas restaurer une SSAS base de données contenant un fichier endommagé. Assurez-vous que le fichier n'est pas endommagé ou corrompu. Cette erreur peut également être déclenché e lorsque `@rds_file_path` n'est pas correctement formaté (par exemple, il comporte des double barres obliques inverses, comme dans `D:\S3\\incorrect_format.abf`).
La restauration a échoué. Le nom de la base de données restaurée ne peut pas contenir de mots réservés ou de caractères non valides (. , ; ' ` : / \\ * | ? \" & % $! + = ( ) [ ] { } < >) et ne peut pas comporter plus de 100 caractères.	RDSprocédure stockée	Le nom de la base de données restaurée ne peut pas contenir de mots réservés ou de caractères non valides et ne peut pas comporter plus de 100 caractères. Pour les SSAS conventions de dénominat ion des objets, voir Règles de dénomination des objets dans la documentation Microsoft.
Le nom de rôle fourni n'est pas valide. Le nom de rôle ne peut pas contenir de chaînes réservées.	RDSprocédure stockée	Le nom de rôle ne peut pas contenir de chaînes réservées. Pour les SSAS conventions de dénominat ion des objets, voir Règles de dénomination des objets dans la documentation Microsoft.
Le nom de rôle fourni n'est pas valide. Le nom de rôle ne peut pas contenir les caractères réservés suivants : . , ; ' ` : / \\ * | ? \" & % $! + = ( ) [ ] { } < >	RDSprocédure stockée	Le nom de rôle ne peut pas contenir de caractères réservés. Pour les SSAS conventions de dénominat ion des objets, voir Règles de dénomination des objets dans la documentation Microsoft.

Prise en charge de SQL Server Integration Services dans Amazon RDS for SQL Server

Microsoft SQL Server Integration Services (SSIS) est un composant que vous pouvez utiliser pour effectuer un large éventail de tâches de migration de données. SSIS est une plateforme d'intégration de données et d'applications de flux de travail. Elle dispose d'un outil d'entreposage de données utilisé pour l'extraction, la transformation et le chargement des données (ETL). Vous pouvez également utiliser cet outil pour automatiser la maintenance des bases de données SQL Server et les mises à jour des données cube multidimensionnelles.

Les projets SSIS sont organisés en paquets enregistrés en tant que fichiers .dtsx basés sur XML. Les packages peuvent contenir des flux de contrôle et des flux de données. Vous utilisez des flux de données pour représenter les opérations ETL. Après le déploiement, les packages sont stockés dans SQL Server dans la base de données SSISDB. SSISDB est une base de données de traitement des transactions en ligne (OLTP) en mode de récupération complète.

Amazon RDS for SQL Server prend en charge l'exécution de SSIS directement sur une instance de base de données RDS. Vous pouvez activer SSIS sur une instance de base de données existante ou nouvelle. SSIS est installée sur la même instance de base de données que votre moteur de base de données.

RDS prend en charge SSIS pour SQL Server éditions Standard et Enterprise sur les versions suivantes :

- SQL Server 2022, toutes les versions
- SQL Server 2019, version 15.00.4043.16.v1 et ultérieure
- SQL Server 2017, version 14.00.3223.3.v1 et ultérieure
- SQL Server 2016, version 13.00.5426.0.v1 et ultérieure

Table des matières

- Limitations et recommandations
- Activation de SSIS
  - Création du groupe d'options pour SSIS
  - Ajout de l'option SSIS au groupe d'options
  - Création du groupe de paramètres pour SSIS
  - Modification du paramètre pour SSIS

- Association du groupe d'options et du groupe de paramètres à votre instance de base de données
- Activation de l'intégration S3
- Autorisations administratives sur SSISDB
  - Configuration d'un utilisateur authentifié par Windows pour SSIS
- Déploiement d'un projet SSIS
- Surveillance de l'état d'une tâche de déploiement
- En utilisant SSIS
  - Configuration des gestionnaires de connexions à la base de données pour les SSIS projets
  - Création d'un SSIS proxy
  - Planification d'un SSIS package à l'aide de l'agent SQL de serveur
  - Révocation de SSIS l'accès depuis le proxy
- Désactiver et supprimer la SSIS base de données
  - Désactivation SSIS
  - Suppression de la SSISDB base de données

Limitations et recommandations

Les limitations et recommandations suivantes s'appliquent à l'exécution de SSIS sur RDS for SQL Server :

- L'instance de base de données doit avoir un groupe de paramètres associé avec le paramètre `clr enabled` défini sur 1. Pour de plus amples informations, veuillez consulter Modification du paramètre pour SSIS.

  > **Note**
  >
  > Si vous activez le paramètre `clr enabled` sur SQL Server 2017 ou 2019, vous ne pouvez pas utiliser le Common Language Runtime (CLR) sur votre instance de base de données. Pour de plus amples informations, veuillez consulter Fonctions non prises en charge et fonctions avec prise en charge limitée.

- Les tâches de flux de contrôle suivantes sont prises en charge :
  - Analysis Services exécute tâche DDL

  - Tâche de traitement Analysis Services
  - Tâche d'insertion en bloc
  - Tâche de vérification de l'intégrité de la base de données
  - Tâche de flux de données
  - Tâche de requête d'exploration de données
  - Tâche de profilage des données
  - Tâche d'exécution de package
  - Exécuter la tâche de travail de SQL Server Agent
  - Exécuter une tâche SQL
  - Exécuter la tâche d'instruction T-SQL
  - Notifier la tâche de l'opérateur
  - Tâche de reconstruction de l'index
  - Tâche de réorganisation de l'index
  - Tâche de réduction de la base de données
  - Tâche de transfert de la base de données
  - Tâche de transfert des tâches
  - Tâche de transfert des connexions
  - Tâche de transfert d'objets SQL Server
  - Tâche de mise à jour des statistiques
- Seul le déploiement de projet est pris en charge.
- L'exécution de packages SSIS à l'aide de SQL Server Agent est prise en charge.
- Les enregistrements de journaux SSIS peuvent être insérés uniquement dans des bases de données créées par les utilisateurs.
- Utilisez uniquement le dossier `D:\S3` pour travailler avec des fichiers. Les fichiers placés dans un autre répertoire sont supprimés. Soyez conscient de quelques autres détails de l'emplacement des fichiers :
  - Placez les fichiers d'entrée et de sortie du projet SSIS dans le dossier `D:\S3`.
  - Pour la tâche de flux de données, modifiez l'emplacement de `BLOBTempStoragePath` et de `BufferTempStoragePath` vers un fichier à l'intérieur du dossier `D:\S3`. Le chemin d'accès au fichier doit commencer par `D:\S3\`.

  - Assurez-vous que tous les paramètres, variables et expressions utilisés pour les connexions de fichiers pointent vers le dossier `D:\S3`.
  - Sur les instances multi-AZ, les fichiers créés par SSIS dans le dossier `D:\S3` sont supprimés après un basculement. Pour plus d'informations, consultez Limitations Multi-AZ pour l'intégration S3.
  - Téléchargez les fichiers créés par SSIS dans le dossier `D:\S3` dans votre compartiment Amazon S3 pour les rendre durables.
- Les transformations de colonne d'importation et d'exportation, et le composant Script sur la tâche de flux de données ne sont pas prises en charge.
- Vous ne pouvez pas activer le vidage sur l'exécution du package SSIS, et vous ne pouvez pas ajouter des prises de données sur les packages SSIS.
- La fonctionnalité augmentation de la taille des instances SSIS n'est pas prise en charge.
- Vous ne pouvez pas déployer de projets directement. Nous fournissons des procédures stockées RDS pour ce faire. Pour plus d'informations, consultez Déploiement d'un projet SSIS.
- Créez des fichiers de projet SSIS (.ispac) avec le mode de protection `DoNotSavePasswords` pour le déploiement sur RDS.
- SSIS n'est pas pris en charge sur les instances Always On avec des réplicas en lecture.
- Vous ne pouvez pas sauvegarder la base de données SSISDB associée à l'option `SSIS`.
- L'importation et la restauration de la base de données SSISDB à partir d'autres instances de SSIS ne sont pas prises en charge.
- Vous pouvez vous connecter à d'autres instances de base de données SQL Server ou à une source de données Oracle. La connexion à d'autres moteurs de bases de données, tels que MySQL ou PostgreSQL, n'est pas prise en charge pour SSIS sur RDS for SQL Server. Pour plus d'informations sur la connexion à une source de données Oracle, consultez Serveurs liés avec Oracle OLEDB.
- SSIS ne prend pas en charge une instance jointe à un domaine avec une approbation sortante vers un domaine local. Lorsque vous utilisez une approbation sortante, exécutez la tâche SSIS à partir d'un compte du AWS domaine local.

Activation de SSIS

Vous activez SSIS en ajoutant l'option SSIS à votre instance de base de données. Utilisez la procédure suivante :

1. Créez un groupe d'options ou choisissez un groupe d'options existant.
2. Ajoutez l'option `SSIS` au groupe d'options.
3. Créez un nouveau groupe de paramètres ou choisissez un groupe de paramètres existant.
4. Modifiez le groupe de paramètres de manière à définir le paramètre `clr enabled` sur 1.
5. Associez le groupe d'options et le groupe de paramètres à l'instance de base de données.
6. Activez l'intégration Amazon S3

> **Note**
>
> Si une base de données portant le nom SSISDB ou une connexion SSIS réservée existe déjà sur l'instance de base de données, vous ne pouvez pas activer SSIS sur cette dernière.

Création du groupe d'options pour SSIS

Pour utiliser SSIS, créez un groupe d'options ou modifier un groupe d'options correspondant à l'édition et à la version SQL Server de l'instance de base de données que vous prévoyez d'utiliser. Pour ce faire, utilisez le AWS Management Console ou le AWS CLI.

console

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2016.

Pour créer le groupe d'options

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
   a. Dans Nom, entrez un nom unique au sein de votre AWS compte pour le groupe d'options, tel que**`ssis-se-2016`**. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
   b. Pour Description, saisissez une brève description du groupe d'options, par exemple **`SSIS option group for SQL Server SE 2016`**. La description est utilisée à des fins d'affichage.
   c. Pour Moteur, choisissez sqlserver-se.

d. Pour Version majeure du moteur, choisissez 13.00.

5. Choisissez Créer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2016.

Pour créer le groupe d'options

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds create-option-group \
    --option-group-name ssis-se-2016 \
    --engine-name sqlserver-se \
    --major-engine-version 13.00 \
    --option-group-description "SSIS option group for SQL Server SE 2016"
```

  Dans Windows :

```
aws rds create-option-group ^
    --option-group-name ssis-se-2016 ^
    --engine-name sqlserver-se ^
    --major-engine-version 13.00 ^
    --option-group-description "SSIS option group for SQL Server SE 2016"
```

Ajout de l'option SSIS au groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'`SSIS`option à votre groupe d'options.

console

Pour ajouter l'option SSIS

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options que vous venez de créer, ssis-se-2016 dans cet exemple.
4. Sélectionnez Ajouter une option.
5. Sous Détails de l'option, choisissez SSIS pour Nom de l'option.
6. Sous Scheduling (Planification), choisissez si vous souhaitez ajouter l'option immédiatement ou lors du créneau de maintenance suivant.
7. Sélectionnez Ajouter une option.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Pour ajouter l'option SSIS

- Ajoutez l'option `SSIS` au groupe d'options.

  Example

  Pour LinuxmacOS, ou Unix :

  ```
  aws rds add-option-to-option-group \
      --option-group-name ssis-se-2016 \
      --options OptionName=SSIS \
      --apply-immediately
  ```

  Dans Windows :

  ```
  aws rds add-option-to-option-group ^
      --option-group-name ssis-se-2016 ^
      --options OptionName=SSIS ^
      --apply-immediately
  ```

Création du groupe de paramètres pour SSIS

Créez ou modifiez un groupe de paramètres pour le paramètre `clr enabled` qui correspond à l'édition et à la version de SQL Server l'instance de base de données que vous prévoyez d'utiliser pour SSIS.

console

La procédure suivante crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans le volet Créer un groupe de paramètres, faites ce qui suit :
   a. Pour Famille de groupes de paramètres, choisissez sqlserver-se-13.0.
   b. Pour Nom du groupe, saisissez un identifiant pour le groupe de paramètres, tel que **`ssis-sqlserver-se-13`**.
   c. Pour Description, saisissez **`clr enabled parameter group`**.
5. Choisissez Créer.

INTERFACE DE LIGNE DE COMMANDE (CLI)

La procédure suivante crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds create-db-parameter-group \
    --db-parameter-group-name ssis-sqlserver-se-13 \
    --db-parameter-group-family "sqlserver-se-13.0" \
    --description "clr enabled parameter group"
```

  Dans Windows :

```
aws rds create-db-parameter-group ^
    --db-parameter-group-name ssis-sqlserver-se-13 ^
```

```
    --db-parameter-group-family "sqlserver-se-13.0" ^
    --description "clr enabled parameter group"
```

Modification du paramètre pour SSIS

Modifiez le paramètre `clr enabled` dans le groupe de paramètres qui correspond à l'édition et à la version de SQL Server utilisées par votre instance de base de données. Pour SSIS, définissez le paramètre `clr enabled` sur 1.

console

La procédure suivante modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez le groupe de paramètres, par exemple ssis-sqlserver-se-13.
4. Sous Paramètres, filtrez la liste des paramètres pour **`clr`**.
5. Choisissez clr activé.
6. Choisissez Modifier les paramètres.
7. Dans Valeurs, choisissez 1.
8. Sélectionnez Enregistrer les modifications.

INTERFACE DE LIGNE DE COMMANDE (CLI)

La procédure suivante modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name ssis-sqlserver-se-13 \
    --parameters "ParameterName='clr
 enabled',ParameterValue=1,ApplyMethod=immediate"
```

Dans Windows :

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name ssis-sqlserver-se-13 ^
    --parameters "ParameterName='clr
 enabled',ParameterValue=1,ApplyMethod=immediate"
```

Association du groupe d'options et du groupe de paramètres à votre instance de base de données

Pour associer le groupe d'options et le groupe de paramètres SSIS à votre instance de base de données, utilisez le AWS Management Console ou AWS CLI

> **Note**
>
> Si vous utilisez une instance existante, elle doit déjà être associée à un domaine Active Directory et à un rôle AWS Identity and Access Management (IAM). Si vous créez une instance, spécifiez un domaine Active Directory et un rôle IAM existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.

console

Pour terminer l'activation de SSIS, associez votre groupe d'options et votre groupe de paramètres SSIS à une instance de base de données nouvelle ou existante :

- Pour une nouvelle instance de base de données, associez-les lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-les en modifiant l'instance. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

INTERFACE DE LIGNE DE COMMANDE (CLI)

Vous pouvez associer le groupe d'options et le groupe de paramètres SSIS à une instance de base de données nouvelle ou existante.

Pour créer une instance avec le groupe d'options SSIS et le groupe de paramètres

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier myssisinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 13.00.5426.0.v1 \
    --allocated-storage 100 \
    --manage-master-user-password \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --domain-iam-role-name my-directory-iam-role \
    --domain my-domain-id \
    --option-group-name ssis-se-2016 \
    --db-parameter-group-name ssis-sqlserver-se-13
```

  Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier myssisinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 13.00.5426.0.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --domain-iam-role-name my-directory-iam-role ^
```

```
    --domain my-domain-id ^
    --option-group-name ssis-se-2016 ^
    --db-parameter-group-name ssis-sqlserver-se-13
```

Pour modifier une instance et associer le groupe d'options et le groupe de paramètres SSIS

- Exécutez une des commandes suivantes :

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier myssisinstance \
    --option-group-name ssis-se-2016 \
    --db-parameter-group-name ssis-sqlserver-se-13 \
    --apply-immediately
```

  Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier myssisinstance ^
    --option-group-name ssis-se-2016 ^
    --db-parameter-group-name ssis-sqlserver-se-13 ^
    --apply-immediately
```

Activation de l'intégration S3

Pour télécharger les fichiers du projet SSIS (.ispac) sur votre hôte pour le déploiement, utilisez l'intégration de fichiers S3. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS for SQL Server DB avec Amazon S3.

Autorisations administratives sur SSISDB

Lorsque l'instance est créée ou modifiée avec l'SSISoption, le résultat est une SSISDB base de données avec les rôles ssis_admin et ssis_logreader accordés à l'utilisateur principal. L'utilisateur principal dispose des privilèges suivants dans SSISDB :

- modifier le rôle ssis_admin

- modifier le rôle ssis_logreader
- modifier n'importe quel utilisateur

L'utilisateur principal étant un utilisateur SQL authentifié, vous ne pouvez pas utiliser l'utilisateur principal pour exécuter SSIS des packages. L'utilisateur principal peut utiliser ces privilèges pour créer de nouveaux SSISDB utilisateurs et les ajouter aux rôles ssis_admin et ssis_logreader. Cela est utile pour donner accès à votre domaine aux utilisateursSSIS.

Configuration d'un utilisateur authentifié par Windows pour SSIS

L'utilisateur principal peut utiliser l'exemple de code suivant pour configurer une connexion authentifiée par Windows SSISDB et accorder les autorisations de procédure requises. Cela permet à l'utilisateur du domaine de déployer et d'exécuter des SSIS packages, d'utiliser les procédures de transfert de fichiers S3, de créer des informations d'identification et de travailler avec le proxy de l'agent de SQL serveur. Pour plus d'informations, consultez les sections Informations d'identification (moteur de base de données) et Création d'un proxy d'agent SQL serveur dans la documentation Microsoft.

> **Note**
>
> Vous pouvez accorder quelques-unes ou la totalité des autorisations suivantes, selon les besoins, aux utilisateurs authentifiés par Windows.

Example

```
-- Create a server-level SQL login for the domain user, if it doesn't already exist
USE [master]
GO
CREATE LOGIN [mydomain\user_name] FROM WINDOWS
GO

-- Create a database-level account for the domain user, if it doesn't already exist

USE [SSISDB]
GO
CREATE USER [mydomain\user_name] FOR LOGIN [mydomain\user_name]

-- Add SSIS role membership to the domain user
ALTER ROLE [ssis_admin] ADD MEMBER [mydomain\user_name]
```

```
ALTER ROLE [ssis_logreader] ADD MEMBER [mydomain\user_name]
GO

-- Add MSDB role membership to the domain user
USE [msdb]
GO
CREATE USER [mydomain\user_name] FOR LOGIN [mydomain\user_name]

-- Grant MSDB stored procedure privileges to the domain user
GRANT EXEC ON msdb.dbo.rds_msbi_task TO [mydomain\user_name] with grant option
GRANT SELECT ON msdb.dbo.rds_fn_task_status TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_task_status TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_cancel_task TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_download_from_s3 TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_upload_to_s3 TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.rds_delete_from_filesystem TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.rds_gather_file_details TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_add_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_update_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_grant_login_to_proxy TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_revoke_login_from_proxy TO [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_delete_proxy TO [mydomain\user_name] with grant option
GRANT EXEC ON msdb.dbo.sp_enum_login_for_proxy to [mydomain\user_name] with grant
 option
GRANT EXEC ON msdb.dbo.sp_enum_proxy_for_subsystem TO [mydomain\user_name]  with grant
 option
GRANT EXEC ON msdb.dbo.rds_sqlagent_proxy TO [mydomain\user_name] WITH GRANT OPTION


-- Add the SQLAgentUserRole privilege to the domain user
USE [msdb]
GO
ALTER ROLE [SQLAgentUserRole] ADD MEMBER [mydomain\user_name]
GO

-- Grant the ALTER ANY CREDENTIAL privilege to the domain user
USE [master]
GO
GRANT ALTER ANY CREDENTIAL TO [mydomain\user_name]
```

```
GO
```

Déploiement d'un projet SSIS

Sur RDS, vous ne pouvez pas déployer de projets SSAS directement avec SQL Server Management Studio (SSMS) ou des procédures SSIS. Pour télécharger des fichiers de projet à partir de Amazon S3, pour les déployer ensuite, utilisez les procédures stockées RDS.

Pour exécuter les procédures stockées, connectez-vous en tant qu'utilisateur auquel vous avez accordé des autorisations d'exécution pour les procédures stockées. Pour plus d'informations, consultez Configuration d'un utilisateur authentifié par Windows pour SSIS.

Pour déployer le projet SSIS

1. Téléchargez le fichier du projet (.ispac).

   ```
   exec msdb.dbo.rds_download_from_s3
   @s3_arn_of_file='arn:aws:s3:::bucket_name/ssisproject.ispac',
   @rds_file_path='D:\S3\ssisproject.ispac',
   @overwrite_file=1;
   ```

2. Soumettez la tâche de déploiement en vérifiant ce qui suit :

   - Le dossier est présent dans le catalogue SSIS.
   - Le nom du projet correspond au nom du projet que vous avez utilisé lors du développement du projet SSIS.

   ```
   exec msdb.dbo.rds_msbi_task
   @task_type='SSIS_DEPLOY_PROJECT',
   @folder_name='DEMO',
   @project_name='ssisproject',
   @file_path='D:\S3\ssisproject.ispac';
   ```

Surveillance de l'état d'une tâche de déploiement

Pour suivre l'état de votre tâche de déploiement, appelez la fonction `rds_fn_task_status`. Deux paramètres sont nécessaires. Le premier paramètre doit toujours être `NULL` parce qu'il ne s'applique pas àSSIS. Le second paramètre accepte l'ID de tâche.

Pour consulter une liste de toutes les tâches, définissez le premier paramètre sur NULL et le second sur 0, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,0);
```

Pour obtenir une tâche spécifique, définissez le premier paramètre sur NULL et le second sur l'ID de tâche, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,42);
```

La fonction rds_fn_task_status retourne les informations suivantes.

Paramètre de sortie	Description
task_id	ID de la tâche
task_type	SSIS_DEPLOY_PROJECT
database_name	Non applicable aux SSIS tâches.
% complete	Progression de la tâche sous forme de pourcentage.
duration (mins)	Temps consacré à la tâche, en minutes.
lifecycle	État de la tâche. Les statuts possibles sont les suivants : • CREATED – Après avoir appelé la procédure stockée msdb.dbo.rds_msbi_task , une tâche est créée et son état est défini sur CREATED. • IN_PROGRESS – Après le démarrage d'une tâche, le statut est défini sur IN_PROGRESS . Le passage du statut CREATED à IN_PROGRESS peut prendre jusqu'à cinq minutes.

Paramètre de sortie	Description
	• SUCCESS – Lorsqu'une tâche est terminée, le statut est défini sur SUCCESS. • ERROR – Si une tâche échoue, le statut est défini sur ERROR. Lisez la colonne task_info pour plus d'informations sur l'erreur. • CANCEL_REQUESTED – Après que vous avez appelé rds_cancel_task , le statut de la tâche est défini sur CANCEL_REQUESTED . • CANCELLED – Une fois une tâche annulée avec succès, l'état de la tâche est défini sur CANCELLED .
task_info	Informations supplémentaires sur la tâche. Si une erreur se produit pendant le traitement, cette colonne contient des informations sur l'erreur.
last_updated	Date et heure de la dernière mise à jour de l'état de la tâche.
created_at	Date et heure de création de la tâche.
S3_object_arn	Non applicable aux SSIS tâches.
overwrite_S3_backup_file	Non applicable aux SSIS tâches.
KMS_master_key_arn	Non applicable aux SSIS tâches.
filepath	Non applicable aux SSIS tâches.

Paramètre de sortie	Description
`overwrite_file`	Non applicable aux SSIS tâches.
`task_metadata`	Métadonnées associées à la SSIS tâche.

En utilisant SSIS

Après avoir déployé le SSIS projet dans le SSIS catalogue, vous pouvez exécuter des packages directement depuis SSMS ou les planifier à l'aide de l'agent de SQL serveur. Vous devez utiliser un identifiant authentifié par Windows pour exécuter les packages. SSIS Pour de plus amples informations, veuillez consulter Configuration d'un utilisateur authentifié par Windows pour SSIS.

Rubriques

- Configuration des gestionnaires de connexions à la base de données pour les SSIS projets
- Création d'un SSIS proxy
- Planification d'un SSIS package à l'aide de l'agent SQL de serveur
- Révocation de SSIS l'accès depuis le proxy

Configuration des gestionnaires de connexions à la base de données pour les SSIS projets

Lorsque vous utilisez un gestionnaire de connexions, vous pouvez employer les types d'authentification suivants :

- Pour les connexions aux bases de données locales AWS à l'aide de Managed Active Directory, vous pouvez utiliser SQL l'authentification ou l'authentification Windows. Pour l'authentification Windows, utilisez *`DB_instance_name.fully_qualified_domain_name`* comme nom de serveur de la chaîne de connexion.

  `myssisinstance.corp-ad.example.com` en est un exemple, où `myssisinstance` est le nom de l'instance de base de données et `corp-ad.example.com` le nom de domaine entièrement qualifié.
- Pour les connexions à distance, utilisez toujours SQL l'authentification.
- Pour les connexions aux bases de données locales à l'aide d'Active Directory autogéré, vous pouvez utiliser l'SQLauthentification ou l'authentification Windows. Pour l'authentification Windows, utilisez `.` ou *`LocalHost`* comme nom de serveur de la chaîne de connexion.

Création d'un SSIS proxy

Pour pouvoir planifier des SSIS packages à l'aide de l'agent de SQL serveur, créez un SSIS identifiant et un SSIS proxy. Exécutez ces procédures en tant qu'utilisateur authentifié par Windows.

Pour créer les SSIS informations d'identification

- Créez les informations d'identification pour le proxy. Pour ce faire, vous pouvez utiliser SSMS l'SQLinstruction suivante.

```
USE [master]
GO
CREATE CREDENTIAL [SSIS_Credential] WITH IDENTITY = N'mydomain\user_name', SECRET =
 N'mysecret'
GO
```

> **Note**
>
> `IDENTITY` doit être une connexion authentifiée par domaine. Remplacez *`mysecret`* par le mot de passe de la connexion authentifiée par le domaine.
> Chaque fois que l'hôte SSISDB principal est changé, modifiez les informations d'identification du SSIS proxy pour permettre au nouvel hôte d'y accéder.

Pour créer le SSIS proxy

1. Utilisez l'SQLinstruction suivante pour créer le proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_add_proxy
 @proxy_name=N'SSIS_Proxy',@credential_name=N'SSIS_Credential',@description=N''
GO
```

2. Utilisez l'SQLinstruction suivante pour accorder l'accès au proxy à d'autres utilisateurs.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_grant_login_to_proxy
 @proxy_name=N'SSIS_Proxy',@login_name=N'mydomain\user_name'
```

```
GO
```

3. Utilisez l'SQLinstruction suivante pour autoriser le SSIS sous-système à accéder au proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.rds_sqlagent_proxy
 @task_type='GRANT_SUBSYSTEM_ACCESS',@proxy_name='SSIS_Proxy',@proxy_subsystem='SSIS'
GO
```

Pour afficher le proxy et les octrois sur le proxy

1. Utilisez l'SQLinstruction suivante pour voir les bénéficiaires du proxy.

```
USE [msdb]
GO
EXEC sp_help_proxy
GO
```

2. Utilisez l'SQLinstruction suivante pour afficher les autorisations des sous-systèmes.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_enum_proxy_for_subsystem
GO
```

Planification d'un SSIS package à l'aide de l'agent SQL de serveur

Une fois que vous avez créé les informations d'identification et le proxy et que vous avez accordé l'SSISaccès au proxy, vous pouvez créer une tâche d'agent SQL serveur pour planifier le SSIS package.

Pour planifier le SSIS forfait

- Vous pouvez utiliser SSMS ou T- SQL pour créer la tâche d'agent de SQL serveur. L'exemple suivant utilise T-SQL.

```
USE [msdb]
GO
DECLARE @jobId BINARY(16)
```

```
EXEC msdb.dbo.sp_add_job @job_name=N'MYSSISJob',
@enabled=1,
@notify_level_eventlog=0,
@notify_level_email=2,
@notify_level_page=2,
@delete_level=0,
@category_name=N'[Uncategorized (Local)]',
@job_id = @jobId OUTPUT
GO
EXEC msdb.dbo.sp_add_jobserver @job_name=N'MYSSISJob',@server_name=N'(local)'
GO
EXEC msdb.dbo.sp_add_jobstep
 @job_name=N'MYSSISJob',@step_name=N'ExecuteSSISPackage',
@step_id=1,
@cmdexec_success_code=0,
@on_success_action=1,
@on_fail_action=2,
@retry_attempts=0,
@retry_interval=0,
@os_run_priority=0,
@subsystem=N'SSIS',
@command=N'/ISSERVER "\"\SSISDB\MySSISFolder\MySSISProject\MySSISPackage.dtsx\"" /
SERVER "\"my-rds-ssis-instance.corp-ad.company.com/\""
/Par "\"$ServerOption::LOGGING_LEVEL(Int16)\"";1 /Par
 "\"$ServerOption::SYNCHRONIZED(Boolean)\"";True /CALLERINFO SQLAGENT /REPORTING
 E',
@database_name=N'master',
@flags=0,
@proxy_name=N'SSIS_Proxy'
GO
```

Révocation de SSIS l'accès depuis le proxy

Vous pouvez révoquer l'accès au SSIS sous-système et supprimer le SSIS proxy à l'aide des procédures stockées suivantes.

Pour révoquer l'accès et supprimer le proxy

1. Révoquez l'accès au sous-système.

```
USE [msdb]
GO
```

```
EXEC msdb.dbo.rds_sqlagent_proxy
 @task_type='REVOKE_SUBSYSTEM_ACCESS',@proxy_name='SSIS_Proxy',@proxy_subsystem='SSIS'
GO
```

2. Révoquez les octrois sur le proxy.

```
USE [msdb]
GO
EXEC msdb.dbo.sp_revoke_login_from_proxy
 @proxy_name=N'SSIS_Proxy',@name=N'mydomain\user_name'
GO
```

3. Supprimez le proxy.

```
USE [msdb]
GO
EXEC dbo.sp_delete_proxy @proxy_name = N'SSIS_Proxy'
GO
```

Désactiver et supprimer la SSIS base de données

Pour désactiver ou supprimer des SSIS bases de données, procédez comme suit :

Rubriques

- Désactivation SSIS
- Suppression de la SSISDB base de données

Désactivation SSIS

Pour la désactiverSSIS, supprimez l'`SSIS`option de son groupe d'options.

> **Important**
>
> La suppression de l'option ne supprime pas la SSISDB base de données. Vous pouvez donc supprimer l'option en toute sécurité sans perdre les SSIS projets.
> Vous pouvez réactiver l'`SSIS`option après sa suppression pour réutiliser les SSIS projets précédemment déployés dans le SSIS catalogue.

Console

La procédure suivante supprime l'option `SSIS`.

Pour supprimer l'SSISoption de son groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `SSIS` (`ssis-se-2016` dans les exemples précédents).
4. Choisissez Supprimer une option.
5. Sous Options de suppression, sélectionnez SSISOptions à supprimer.
6. Sous Appliquer immédiatement, choisissez Oui pour supprimer l'option immédiatement, ou Non pour la supprimer lors du prochain créneau de maintenance.
7. Sélectionnez Delete.

CLI

La procédure suivante supprime l'option `SSIS`.

Pour supprimer l'SSISoption de son groupe d'options

- Exécutez une des commandes suivantes :

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds remove-option-from-option-group \
    --option-group-name ssis-se-2016 \
    --options SSIS \
    --apply-immediately
```

  Dans Windows:

```
aws rds remove-option-from-option-group ^
    --option-group-name ssis-se-2016 ^
    --options SSIS ^
```

```
    --apply-immediately
```

Suppression de la SSISDB base de données

Une fois l'SSISoption supprimée, la SSISDB base de données n'est pas supprimée. Pour supprimer la SSISDB base de données, utilisez la procédure `rds_drop_ssis_database` stockée après avoir supprimé l'SSISoption.

Pour supprimer la SSIS base de données

- Utilisez la procédure stockée suivante.

```
USE [msdb]
GO
EXEC dbo.rds_drop_ssis_database
GO
```

Après avoir supprimé la SSISDB base de données, si vous réactivez l'SSISoption, vous obtenez un nouveau SSISDB catalogue.

Support pour les services de reporting sur les SQL serveurs dans Amazon RDS for SQL Server

Microsoft SQL Server Reporting Services (SSRS) est une application serveur utilisée pour la génération et la distribution de rapports. Il fait partie d'une suite de services de SQL serveur qui inclut également les services d'analyse de SQL serveur (SSAS) et les services d'intégration de SQL serveurs (SSIS). SSRSest un service basé sur le SQL serveur. Vous pouvez l'utiliser pour collecter des données provenant de diverses sources de données et les présenter de sorte qu'elles soient facilement compréhensibles et prêtes à être analysées.

Amazon RDS for SQL Server prend en charge l'exécution SSRS directe sur les RDS instances de base de données. Vous pouvez l'utiliser SSRS avec des instances de base de données existantes ou nouvelles.

RDSprend en SSRS charge les éditions SQL Server Standard et Enterprise sur les versions suivantes :

- SQLServer 2022, toutes les versions
- SQLServer 2019, version 15.00.4043.16.v1 et supérieure
- SQLServer 2017, version 14.00.3223.3.v1 et supérieure
- SQLServer 2016, version 13.00.5820.21.v1 et supérieure

Table des matières

- Limitations et recommandations
- Activation d'SSRS
  - Création d'un groupe d'options pour SSRS
  - Ajouter l'SSRSoption à votre groupe d'options
  - Association de votre groupe d'options à votre instance de base de données
  - Autoriser l'accès entrant à votre groupe VPC de sécurité
- Bases de données de serveur de rapports
- SSRSfichiers journaux
- Accès au portail SSRS Web
  - En utilisant SSL sur RDS
  - Octroi de l'accès aux utilisateurs du domaine

- Accès au portail Web
- Déploiement de rapports et configuration des sources de données de rapports
  - Déploiement de rapports vers SSRS
  - Configuration de la source de données de rapport
- Utilisation de SSRS Email pour envoyer des rapports
- Révocation des autorisations de niveau système
- Surveillance du statut d'une tâche
- Désactivation et suppression SSRS de bases de données
  - Désactivation de SSRS
  - Supprimer les SSRS bases de données

Limitations et recommandations

Les limitations et recommandations suivantes s'appliquent à l'exécution SSRS sur RDS for SQL Server :

- Vous ne pouvez pas l'utiliser SSRS sur des instances de base de données qui ont des répliques de lecture.
- Les instances doivent utiliser Active Directory autogéré ou AWS Directory Service for Microsoft Active Directory pour l'authentification du portail SSRS Web et du serveur Web. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.
- Vous ne pouvez pas sauvegarder les bases de données du serveur de rapports créées avec SSRS cette option.
- L'importation et la restauration des bases de données du serveur de rapports à partir d'autres instances de SSRS ne sont pas prises en charge. Pour de plus amples informations, veuillez consulter Bases de données de serveur de rapports.
- Vous ne pouvez pas configurer SSRS pour écouter sur le SSL port par défaut (443). Les valeurs autorisées sont comprises entre 1150 et 49511, sauf 1234, 1434, 3260, 3343, 3389 et 47001.
- Les abonnements par partage de fichier Microsoft Windows ne sont pas pris en charge.
- L'utilisation du Gestionnaire de configurations Reporting Services n'est pas prise en charge.
- La création et la modification de rôles n'est pas prise en charge.
- La modification des propriétés du serveur de rapports n'est pas prise en charge.
- Les rôles d'administrateur système et d'utilisateur système ne sont pas accordés.

- Vous ne pouvez pas modifier les affectations de rôle de niveau système via le portail Web.

Activation d'SSRS

Utilisez le processus suivant SSRS pour activer votre instance de base de données :

1. Créez un groupe d'options ou choisissez un groupe d'options existant.
2. Ajoutez l'option SSRS au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.
4. Autorisez l'accès entrant au groupe de sécurité du cloud privé virtuel (VPC) pour le port d'SSRSécoute.

Création d'un groupe d'options pour SSRS

Pour travailler avecSSRS, créez un groupe d'options correspondant au moteur de SQL serveur et à la version de l'instance de base de données que vous prévoyez d'utiliser. Pour ce faire, utilisez le AWS Management Console ou le AWS CLI.

> **Note**
>
> Vous pouvez également utiliser un groupe d'options existant s'il correspond au moteur de SQL serveur et à la version appropriés.

Console

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2017.

Pour créer le groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
    a. Dans Nom, entrez un nom unique au sein de votre groupe d'options Compte AWS, tel que**`ssrs-se-2017`**. Le nom ne peut contenir que des lettres, des chiffres et des tirets.

b. Pour Description, saisissez une brève description du groupe d'options, par exemple **`SSRS option group for SQL Server SE 2017`**. La description est utilisée à des fins d'affichage.

c. Pour Moteur, choisissez sqlserver-se.

d. Pour Version majeure du moteur, choisissez 14.00.

5. Sélectionnez Create (Créer).

CLI

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2017.

Pour créer le groupe d'options

- Exécutez une des commandes suivantes :

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-option-group \
    --option-group-name ssrs-se-2017 \
    --engine-name sqlserver-se \
    --major-engine-version 14.00 \
    --option-group-description "SSRS option group for SQL Server SE 2017"
```

Dans Windows:

```
aws rds create-option-group ^
    --option-group-name ssrs-se-2017 ^
    --engine-name sqlserver-se ^
    --major-engine-version 14.00 ^
    --option-group-description "SSRS option group for SQL Server SE 2017"
```

Ajouter l'SSRSoption à votre groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'SSRSoption à votre groupe d'options.

Console

Pour ajouter l'SSRSoption

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options que vous venez de créer, puis sélectionnez Add option (Ajouter une option).
4. Sous Détails de l'option, choisissez SSRSle nom de l'option.
5. Sous Option settings (Paramètres d'option), procédez comme suit :

   a. Entrez le port du SSRS service à écouter. La valeur par défaut est 8443. Pour obtenir la liste des valeurs autorisées, veuillez consulter Limitations et recommandations.

   b. Entrez une valeur pour Max memory (Volume de mémoire maximal).

      Lamémoire maximale spécifie le seuil supérieur au-dessus duquel aucune nouvelle demande d'allocation de mémoire n'est accordée aux applications serveur de rapports. Le nombre correspond à un pourcentage de la mémoire totale de l'instance de base de données. Les valeurs autorisées sont comprises entre 10 et 80.

   c. Pour les groupes de sécurité, choisissez le groupe de VPC sécurité à associer à l'option. Utilisez le même groupe de sécurité que celui associé à votre instance de base de données.
6. Pour utiliser le SSRS courrier électronique pour envoyer des rapports, cochez la case Configurer les options de livraison par e-mail sous Livraison par e-mail dans les services de reporting, puis procédez comme suit :

   a. Pour Adresse e-mail de l'expéditeur, entrez l'adresse e-mail à utiliser dans le champ From des messages envoyés par SSRS e-mail.

      Spécifiez un compte utilisateur autorisé à envoyer du courrier depuis le SMTP serveur.

   b. Pour le SMTPserveur, spécifiez le SMTP serveur ou la passerelle à utiliser.

      Il peut s'agir d'une adresse IP, du BIOS nom réseau d'un ordinateur sur l'intranet de votre entreprise ou d'un nom de domaine complet.

   c. Pour SMTPport, entrez le port à utiliser pour vous connecter au serveur de messagerie. La valeur par défaut est 25.

   d. Pour utiliser l'authentification :

i. Cochez la case Use authentication (Utiliser l'authentification).

ii. Pour Secret Amazon Resource Name (ARN), entrez les AWS Secrets Manager ARN informations d'identification de l'utilisateur.

Utilisez le format suivant :

arn:aws:secretsmanager:*Region*:*AccountId*:secret:*SecretName*-*6RandomChara*

Par exemple :

arn:aws:secretsmanager:*us-west-2*:*123456789012*:secret:*MySecret-a1b2c3*

Pour obtenir plus d'informations sur la création du secret, consultez Utilisation de SSRS Email pour envoyer des rapports

e. Cochez la case Utiliser le protocole Secure Sockets Layer (SSL) pour chiffrer les messages électroniques à l'aide SSL de.

7. Sous Scheduling (Planification), choisissez si vous souhaitez ajouter l'option immédiatement ou lors du créneau de maintenance suivant.

8. Sélectionnez Ajouter une option.

CLI

Pour ajouter l'SSRSoption

1. Créez un JSON fichier, par exemple`ssrs-option.json`.

a. Définissez les paramètres requis suivants :

- `OptionGroupName` – Nom du groupe d'options que vous avez créé ou choisi précédemment (`ssrs-se-2017` dans l'exemple suivant).
- `Port`— Le port d'écoute du SSRS service. La valeur par défaut est 8443. Pour obtenir la liste des valeurs autorisées, veuillez consulter Limitations et recommandations.
- `VpcSecurityGroupMemberships`— les adhésions à des groupes de VPC sécurité pour votre RDS instance de base de données.
- `MAX_MEMORY` – Seuil supérieur au-dessus duquel aucune nouvelle demande d'attribution de mémoire n'est accordée aux applications de serveur de rapports. Le nombre

correspond à un pourcentage de la mémoire totale de l'instance de base de données. Les valeurs autorisées sont comprises entre 10 et 80.

b. (Facultatif) Définissez les paramètres suivants pour utiliser le SSRS courrier électronique :

- `SMTP_ENABLE_EMAIL`— Paramétré `true` pour utiliser le SSRS courrier électronique. L'argument par défaut est `false`.
- `SMTP_SENDER_EMAIL_ADDRESS`— L'adresse e-mail à utiliser dans le champ From des messages envoyés par SSRS e-mail. Spécifiez un compte utilisateur autorisé à envoyer du courrier depuis le SMTP serveur.
- `SMTP_SERVER`— Le SMTP serveur ou la passerelle à utiliser. Il peut s'agir d'une adresse IP, du BIOS nom réseau d'un ordinateur sur l'intranet de votre entreprise ou d'un nom de domaine complet.
- `SMTP_PORT` : le port à utiliser pour se connecter au serveur de messagerie. La valeur par défaut est 25.
- `SMTP_USE_SSL`— Définissez sur pour `true` crypter les e-mails à l'aide SSL de. L'argument par défaut est `true`.
- `SMTP_EMAIL_CREDENTIALS_SECRET_ARN`— Le Gestionnaire de Secrets ARN qui contient les informations d'identification de l'utilisateur. Utilisez le format suivant :

  `arn:aws:secretsmanager:`*`Region`*`:`*`AccountId`*`:secret:`*`SecretName`*`-6RandomCharact`

  Pour obtenir plus d'informations sur la création du secret, consultez Utilisation de SSRS Email pour envoyer des rapports
- `SMTP_USE_ANONYMOUS_AUTHENTICATION` : définissez ce paramètre sur `true` et n'ajoutez pas `SMTP_EMAIL_CREDENTIALS_SECRET_ARN` si vous ne voulez pas utiliser l'authentification.

  La valeur par défaut est `false` quand `SMTP_ENABLE_EMAIL` est `true`.

L'exemple suivant inclut les paramètres de SSRS courrier électronique, en utilisant le secretARN.

```
{
"OptionGroupName": "ssrs-se-2017",
"OptionsToInclude": [
 {
 "OptionName": "SSRS",
```

```
 "Port": 8443,
 "VpcSecurityGroupMemberships": ["sg-0abcdef123"],
 "OptionSettings": [
           {"Name": "MAX_MEMORY","Value": "60"},
           {"Name": "SMTP_ENABLE_EMAIL","Value": "true"}
           {"Name": "SMTP_SENDER_EMAIL_ADDRESS","Value": "nobody@example.com"},
           {"Name": "SMTP_SERVER","Value": "email-smtp.us-west-2.amazonaws.com"},
           {"Name": "SMTP_PORT","Value": "25"},
           {"Name": "SMTP_USE_SSL","Value": "true"},
           {"Name": "SMTP_EMAIL_CREDENTIALS_SECRET_ARN","Value":
 "arn:aws:secretsmanager:us-west-2:123456789012:secret:MySecret-a1b2c3"}
           ]
 }],
"ApplyImmediately": true
}
```

2. Ajoutez l'option SSRS au groupe d'options.

   Example

   Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --cli-input-json file://ssrs-option.json \
    --apply-immediately
```

   Dans Windows:

```
aws rds add-option-to-option-group ^
    --cli-input-json file://ssrs-option.json ^
    --apply-immediately
```

Association de votre groupe d'options à votre instance de base de données

Utilisez le AWS Management Console ou AWS CLI pour associer votre groupe d'options à votre instance de base de données.

Si vous utilisez une instance de base de données existante, elle doit déjà être associée à un domaine Active Directory et à un rôle AWS Identity and Access Management (IAM). Si vous créez une nouvelle instance, spécifiez un domaine et un IAM rôle Active Directory existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory avec RDS for SQL Server.

Console

Vous pouvez associer votre groupe d'options à une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, associez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, modifiez l'instance et associez le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

CLI

Vous pouvez associer votre groupe d'options à une instance de base de données nouvelle ou existante.

Pour créer une instance de base de données utilisant votre groupe d'options

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier myssrsinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 14.00.3223.3.v1 \
    --allocated-storage 100 \
    --manage-master-user-password  \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --domain-iam-role-name my-directory-iam-role \
    --domain my-domain-id \
    --option-group-name ssrs-se-2017
```

  Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier myssrsinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 14.00.3223.3.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --domain-iam-role-name my-directory-iam-role ^
    --domain my-domain-id ^
    --option-group-name ssrs-se-2017
```

Pour modifier une instance de base de données pour utiliser votre groupe d'options

- Exécutez une des commandes suivantes :

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier myssrsinstance \
    --option-group-name ssrs-se-2017 \
    --apply-immediately
```

  Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier myssrsinstance ^
    --option-group-name ssrs-se-2017 ^
    --apply-immediately
```

Autoriser l'accès entrant à votre groupe VPC de sécurité

Pour autoriser l'accès entrant au groupe de VPC sécurité associé à votre instance de base de données, créez une règle entrante pour le port d'SSRSécouteur spécifié. Pour de plus amples

informations sur la configuration des groupes de sécurité, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Bases de données de serveur de rapports

Lorsque votre instance de base de données est associée à l'SSRSoption, deux nouvelles bases de données sont créées sur votre instance de base de données :

- `rdsadmin_ReportServer`
- `rdsadmin_ReportServerTempDB`

Ces bases de données font office de bases de données ReportServer et de ReportServerTemp bases de données. SSRSstocke ses données dans la ReportServer base de données et les met en cache dans la base de données de base ReportServerTemp de données. Pour en savoir plus, consultez Base de données du serveur de rapports dans la documentation Microsoft.

RDSpossède et gère ces bases de données, de sorte que les opérations de base de données sur celles-ci DROP ne sont pas autorisées. ALTER L'accès à la base de données `rdsadmin_ReportServerTempDB` n'est pas autorisé. Néanmoins, vous pouvez effectuer des opérations de lecture sur la base de données `rdsadmin_ReportServer`.

SSRSfichiers journaux

Vous pouvez répertorier, afficher et télécharger des fichiers SSRS journaux. SSRSles fichiers journaux suivent la convention de dénomination ReportServerService *`_timestamp`*.journal. Ces journaux du serveur de rapports se trouvent dans le répertoire `D:\rdsdbdata\Log\SSRS`. (Le `D:\rdsdbdata\Log` répertoire est également le répertoire parent pour les journaux d'erreurs et les journaux de l'agent de SQL serveur.) Pour de plus amples informations, veuillez consulter Liste et affichage des fichiers journaux de base de données.

Pour les SSRS instances existantes, le redémarrage du SSRS service peut être nécessaire pour accéder aux journaux du serveur de rapports. Vous pouvez redémarrer le service en mettant à jour l'option `SSRS`.

Pour de plus amples informations, veuillez consulter Utilisation des journaux Amazon RDS pour Microsoft SQL Server.

Accès au portail SSRS Web

Pour accéder au portail SSRS Web, procédez comme suit :

1. Activez Secure Sockets Layer (SSL).
2. Accorder l'accès aux utilisateurs du domaine.
3. Accédez au portail Web à l'aide d'un navigateur et des informations d'identification d'un utilisateur du domaine.

En utilisant SSL sur RDS

SSRSutilise le HTTPS SSL protocole pour ses connexions. Pour utiliser ce protocole, importez un SSL certificat dans le système d'exploitation Microsoft Windows de votre ordinateur client.

Pour plus d'informations sur SSL les certificats, consultez. Pour plus d'informations sur l'utilisation SSL avec SQL Server, consultezUtilisation SSL avec une instance de base de données Microsoft SQL Server.

Octroi de l'accès aux utilisateurs du domaine

Lors d'une nouvelle SSRS activation, aucune attribution de rôle n'est prévue dansSSRS. Pour donner à un utilisateur de domaine ou à un groupe d'utilisateurs l'accès au portail Web, RDS fournit une procédure stockée.

Pour accorder l'accès à un utilisateur du domaine sur le portail Web

- Utilisez la procédure stockée suivante.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSRS_GRANT_PORTAL_PERMISSION',
@ssrs_group_or_username=N'AD_domain\user';
```

Le rôle système `RDS_SSRS_ROLE` est accordé à l'utilisateur ou au groupe d'utilisateurs du domaine. Les tâches de niveau système suivantes sont accordées à ce rôle :

- Exécuter des rapports
- Gérer des tâches
- Gérer des calendriers partagés
- Afficher des calendriers partagés

Le rôle de niveau élément `Content Manager` sur le dossier racine est également accordé.

Accès au portail Web

Lorsque la tâche `SSRS_GRANT_PORTAL_PERMISSION` est terminée, vous avez accès au portail à l'aide d'un navigateur Web. Le portail Web URL a le format suivant.

```
https://rds_endpoint:port/Reports
```

Dans ce format, les points suivants s'appliquent :

- *`point de terminaison rds`* — Le point de terminaison de l'RDSinstance de base de données que vous utilisezSSRS.

  Vous trouverez le point de terminaison dans l'onglet Connectivité et sécurité de votre instance de base de données. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données Microsoft SQL Server.
- *`port`*— Le port d'écoute SSRS que vous avez défini dans l'`SSRS`option.

Pour accéder au portail Web

1. Accédez au portail Web URL dans votre navigateur.

   ```
   https://myssrsinstance.cg034itsfake.us-east-1.rds.amazonaws.com:8443/Reports
   ```
2. Connectez-vous avec les informations d'identification d'un utilisateur du domaine auquel vous avez accordé l'accès avec la tâche `SSRS_GRANT_PORTAL_PERMISSION`.

Déploiement de rapports et configuration des sources de données de rapports

Utilisez les procédures suivantes pour déployer des rapports SSRS et configurer les sources de données de rapports :

Rubriques

- Déploiement de rapports vers SSRS
- Configuration de la source de données de rapport

Déploiement de rapports vers SSRS

Une fois que vous avez accès au portail web, vous pouvez y déployer des rapports. Vous pouvez utiliser l'outil de téléchargement du portail Web pour télécharger des rapports ou les déployer directement à partir des outils de données SQL du serveur (SSDT). Lors du déploiement depuisSSDT, assurez-vous de ce qui suit :

- L'utilisateur qui l'SSDTa lancé a accès au portail SSRS Web.
- La `TargetServerURL` valeur dans les propriétés du SSRS projet est définie sur le HTTPS point de terminaison de l'RDSinstance de base de données avec`ReportServer`, par exemple, le suffixe suivant :

```
https://myssrsinstance.cg034itsfake.us-east-1.rds.amazonaws.com:8443/ReportServer
```

Configuration de la source de données de rapport

Après avoir déployé un rapport surSSRS, vous devez configurer la source de données du rapport. Lors de la configuration de la source de données du rapport, assurez-vous de ce qui suit :

- RDSPour les instances de base de données de SQL serveur jointes à AWS Directory Service for Microsoft Active Directory, utilisez le nom de domaine complet (FQDN) comme nom de source de données de la chaîne de connexion. *`myssrsinstance.corp-ad.example.com`* en est un exemple, où *`myssrsinstance`* est le nom de l'instance de base de données et *`corp-ad.example.com`* le nom de domaine entièrement qualifié.
- RDSPour les instances de base de données de SQL serveur jointes à Active Directory autogéré., utilisez ou *`LocalHost`* comme nom de source de données de la chaîne de connexion.

Utilisation de SSRS Email pour envoyer des rapports

SSRS comprend l'extension SSRS Email, que vous pouvez utiliser pour envoyer des rapports aux utilisateurs.

Pour configurer SSRS Email, utilisez les paramètres de l'option SSRS. Pour de plus amples informations, veuillez consulter Ajouter l'SSRSoption à votre groupe d'options.

Après avoir configuré SSRS Email, vous pouvez vous abonner aux rapports sur le serveur de rapports. Pour obtenir plus d'informations, consultez la rubrique Email delivery in Reporting Services (Livraison d'e-mails dans Reporting Services) dans la documentation Microsoft.

L'intégration avec AWS Secrets Manager est requise pour que SSRS Email fonctionne sur RDS. Pour l'intégrer à Secrets Manager, il faut créer un secret.

> **Note**
>
> Si vous modifiez le secret ultérieurement, vous devez également mettre à jour l'option SSRS dans le groupe d'options.

Pour créer un secret pour SSRS Email

1. Suivez les étapes de la section Create a secret (Créer un secret) du Guide de l'utilisateur AWS Secrets Manager .

   a. Pour Select secret type (Sélectionner un type de secret), choisissez Other type of secrets (Autre type de secrets).

   b. Pour Key/value pairs (Paires clé/valeur), entrez ce qui suit :

      - **`SMTP_USERNAME`** : entrez un utilisateur ayant l'autorisation d'envoyer des e-mails à partir du serveur SMTP.
      - **`SMTP_PASSWORD`** : saisissez un mot de passe pour l'utilisateur SMTP.

   c. Pour Encryption key (Clé de chiffrement), n'utilisez pas la valeur AWS KMS key par défaut. Utilisez votre propre clé existante, ou créez-en une.

      La politique de clé KMS doit autoriser l'action `kms:Decrypt`, par exemple :

```
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
    "Principal": {
        "Service": [
            "rds.amazonaws.com"
        ]
    },
    "Action": [
        "kms:Decrypt"
    ],
    "Resource": "*"
}
```

2. Suivez les étapes de la section Attach a permissions policy to a secret (Attacher une politique d'autorisations à un secret) du Guide de l'utilisateur AWS Secrets Manager . La politique d'autorisations transmet l'action `secretsmanager:GetSecretValue` au principal du service `rds.amazonaws.com`.

   Nous vous recommandons d'utiliser les conditions `aws:sourceAccount` et `aws:sourceArn` dans la politique pour éviter le problème de l'adjoint confus. Utilisez votre Compte AWS for `aws:sourceAccount` et l'ARN du groupe d'options pour`aws:sourceArn`. Pour de plus amples informations, veuillez consulter Prévention des problèmes d'adjoint confus entre services.

   Voici un exemple de stratégie d'autorisation.

   JSON

```
{
  "Version" : "2012-10-17",
  "Statement" : [ {
    "Effect" : "Allow",
    "Principal" : {
      "Service" : "rds.amazonaws.com"
    },
    "Action" : "secretsmanager:GetSecretValue",
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "aws:sourceAccount" : "123456789012"
      },
      "ArnLike" : {
        "aws:sourceArn" : "arn:aws:rds:us-west-2:123456789012:og:ssrs-
se-2017"
      }
    }
  } ]
}
```

   Pour plus d'exemples, consultez les exemples de politique d'autorisations pour AWS Secrets Manager dans le guide de AWS Secrets Manager l'utilisateur.

Révocation des autorisations de niveau système

Le rôle système `RDS_SSRS_ROLE` ne dispose pas des autorisations suffisantes pour supprimer les affectations de rôle de niveau système. Pour supprimer un utilisateur ou un groupe d'utilisateurs de `RDS_SSRS_ROLE`, utilisez la même procédure stockée que celle utilisée pour accorder le rôle, mais faites appel au type de tâche `SSRS_REVOKE_PORTAL_PERMISSION`.

Pour révoquer l'accès d'un utilisateur du domaine pour le portail Web

- Utilisez la procédure stockée suivante.

```
exec msdb.dbo.rds_msbi_task
@task_type='SSRS_REVOKE_PORTAL_PERMISSION',
@ssrs_group_or_username=N'AD_domain\user';
```

Cette procédure supprime l'utilisateur du rôle système `RDS_SSRS_ROLE`. Il supprime également l'utilisateur du rôle de niveau élément `Content Manager`, si cet utilisateur en dispose.

Surveillance du statut d'une tâche

Pour suivre le statut de votre tâche d'octroi ou de révocation, appelez la fonction `rds_fn_task_status`. Deux paramètres sont nécessaires. Le premier paramètre doit toujours être `NULL` parce qu'il ne s'applique pas àSSRS. Le second paramètre accepte l'ID de tâche.

Pour consulter une liste de toutes les tâches, définissez le premier paramètre sur `NULL` et le second sur `0`, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,0);
```

Pour obtenir une tâche spécifique, définissez le premier paramètre sur `NULL` et le second sur l'ID de tâche, comme illustré dans l'exemple suivant.

```
SELECT * FROM msdb.dbo.rds_fn_task_status(NULL,42);
```

La fonction `rds_fn_task_status` retourne les informations suivantes.

Paramètre de sortie	Description
`task_id`	ID de la tâche

Paramètre de sortie	Description
`task_type`	En SSRS effet, les tâches peuvent avoir les types de tâches suivants : • `SSRS_GRANT_PORTAL_PERMISSION` • `SSRS_REVOKE_PORTAL_PERMISSION`
`database_name`	Non applicable aux SSRS tâches.
`% complete`	Progression de la tâche sous forme de pourcentage.
`duration (mins)`	Temps consacré à la tâche, en minutes.

Paramètre de sortie	Description
`lifecycle`	État de la tâche. Les statuts possibles sont les suivants : • `CREATED`— Après avoir appelé l'une des procédures SSRS stockées, une tâche est créée et son statut est défini sur`CREATED`. • `IN_PROGRESS` – Après le démarrage d'une tâche, le statut est défini sur `IN_PROGRESS` . Le passage du statut `CREATED` à `IN_PROGRESS` peut prendre jusqu'à cinq minutes. • `SUCCESS` – Lorsqu'une tâche est terminée, le statut est défini sur `SUCCESS`. • `ERROR` – Si une tâche échoue, le statut est défini sur `ERROR`. Lisez la colonne `task_info` pour plus d'informations sur l'erreur. • `CANCEL_REQUESTED` – Après que vous avez appelé la procédure stockée `rds_cancel_task` , le statut de la tâche est défini sur `CANCEL_REQUESTED` . • `CANCELLED` – Une fois une tâche annulée avec succès, l'état de la tâche est défini sur `CANCELLED` .
`task_info`	Informations supplémentaires sur la tâche. Si une erreur se produit pendant le traitement, cette colonne contient des informations sur l'erreur.

Paramètre de sortie	Description
`last_updated`	Date et heure de la dernière mise à jour de l'état de la tâche.
`created_at`	Date et heure de création de la tâche.
`S3_object_arn`	Non applicable aux SSRS tâches.
`overwrite_S3_backup_file`	Non applicable aux SSRS tâches.
`KMS_master_key_arn`	Non applicable aux SSRS tâches.
`filepath`	Non applicable aux SSRS tâches.
`overwrite_file`	Non applicable aux SSRS tâches.
`task_metadata`	Métadonnées associées à la SSRS tâche.

Désactivation et suppression SSRS de bases de données

Pour désactiver SSRS et supprimer des SSRS bases de données, procédez comme suit :

Rubriques

- Désactivation de SSRS
- Supprimer les SSRS bases de données

Désactivation de SSRS

Pour la désactiverSSRS, supprimez l'`SSRS`option de son groupe d'options. La suppression de l'option ne supprime pas les SSRS bases de données. Pour de plus amples informations, veuillez consulter Supprimer les SSRS bases de données.

Vous pouvez SSRS réactiver l'option en réactivant l'`SSRS`option. Si vous avez également supprimé les SSRS bases de données, la lecture de l'option sur la même instance de base de données crée de nouvelles bases de données de serveur de rapports.

Console

Pour supprimer l'SSRSoption de son groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `SSRS` (`ssrs-se-2017` dans les exemples précédents).
4. Choisissez Supprimer une option.
5. Sous Options de suppression, sélectionnez SSRSOptions à supprimer.
6. Sous Appliquer immédiatement, choisissez Oui pour supprimer l'option immédiatement, ou Non pour la supprimer lors du prochain créneau de maintenance.
7. Sélectionnez Delete (Supprimer).

CLI

Pour supprimer l'SSRSoption de son groupe d'options

- Exécutez une des commandes suivantes :

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds remove-option-from-option-group \
    --option-group-name ssrs-se-2017 \
    --options SSRS \
    --apply-immediately
```

  Dans Windows:

```
aws rds remove-option-from-option-group ^
    --option-group-name ssrs-se-2017 ^
    --options SSRS ^
    --apply-immediately
```

Supprimer les SSRS bases de données

La suppression de l'option SSRS ne supprime pas les bases de données du serveur de rapports. Pour les supprimer, utilisez la procédure stockée suivante.

Pour supprimer les bases de données du serveur de rapports, assurez-vous d'abord de supprimer l'option SSRS.

Pour supprimer les SSRS bases de données

- Utilisez la procédure stockée suivante.

```
exec msdb.dbo.rds_drop_ssrs_databases
```

Support pour Microsoft Distributed Transaction Coordinator dans RDS for SQL Server

Une transaction distribuée est une transaction de base de données dans laquelle deux hôtes réseau ou plus sont impliqués. RDSfor SQL Server prend en charge les transactions distribuées entre les hôtes, où un seul hôte peut être l'un des suivants :

- RDSpour l'instance SQL de base de données du serveur
- Hôte de SQL serveur sur site
- EC2Hôte Amazon avec SQL serveur installé
- Tout autre EC2 hôte ou RDS instance de base de données doté d'un moteur de base de données prenant en charge les transactions distribuées

À partir de SQL Server 2012 (version 11.00.5058.0.v1 et versions ultérieures), toutes les éditions de for Server prennent en charge les transactions distribuées. RDS RDS SQL Le support est fourni à l'aide de Microsoft Distributed Transaction Coordinator (MSDTC). Pour des informations détaillées surMSDTC, voir Distributed Transaction Coordinator dans la documentation Microsoft.

Table des matières

- Limites
- Activant MSDTC
  - Création du groupe d'options pour MSDTC
  - Ajouter l'MSDTCoption au groupe d'options
  - Création du groupe de paramètres pour MSDTC
  - Modification du paramètre pour MSDTC
  - Association du groupe d'options et du groupe de paramètres à l'instance de base de données
  - Modifier l'MSDTCoption
- Utilisation des transactions
  - Utilisation des transactions distribuées
  - Utilisation de transactions XA
  - Utilisation du suivi des transactions
- Désactivation MSDTC
- Résolution des problèmes MSDTC pour RDS for SQL Server

Limites

Les restrictions suivantes s'appliquent à l'utilisation d'MSDTCon RDS for SQL Server :

- MSDTCn'est pas pris en charge sur les instances utilisant la mise en miroir de bases de données de SQL serveurs. Pour de plus amples informations, veuillez consulter Transactions - availability groups and database mirroring.
- Le paramètre `in-doubt xact resolution` doit être défini sur 1 ou 2. Pour de plus amples informations, veuillez consulter Modification du paramètre pour MSDTC.
- MSDTCexige que tous les hôtes participant à des transactions distribuées puissent être résolus à l'aide de leur nom d'hôte. RDSgère automatiquement cette fonctionnalité pour les instances jointes à un domaine. Toutefois, pour les instances autonomes, veillez à configurer le DNS serveur manuellement.
- Les transactions Java Database Connectivity (JDBC) XA sont prises en charge pour SQL Server 2017 version 14.00.3223.3 et ultérieure, et Server 2019. SQL
- Les transactions distribuées qui dépendent des bibliothèques de liens dynamiques clientes (DLLs) des RDS instances ne sont pas prises en charge.
- L'utilisation de bibliothèques à liens dynamiques XA personnalisées n'est pas prise en charge.

Activant MSDTC

Utilisez le processus suivant MSDTC pour activer votre instance de base de données :

1. Créez un groupe d'options ou choisissez un groupe d'options existant.
2. Ajoutez l'option `MSDTC` au groupe d'options.
3. Créez un nouveau groupe de paramètres ou choisissez un groupe de paramètres existant.
4. Modifiez le groupe de paramètres de manière à définir le paramètre `in-doubt xact resolution` sur 1 ou 2.
5. Associez le groupe d'options et le groupe de paramètres à l'instance de base de données.

Création du groupe d'options pour MSDTC

Utilisez le AWS Management Console ou AWS CLI pour créer un groupe d'options correspondant au moteur de SQL serveur et à la version de votre instance de base de données.

> **Note**
>
> Vous pouvez également utiliser un groupe d'options existant s'il correspond au moteur de SQL serveur et à la version appropriés.

Console

La procédure suivante crée un groupe d'options pour SQL Server Standard Edition 2016.

Pour créer le groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez Create group.
4. Dans la fenêtre Créer un groupe d'options, procédez comme suit :
    a. Dans Nom, entrez un nom unique au sein de votre AWS compte pour le groupe d'options, tel que**`msdtc-se-2016`**. Le nom ne peut contenir que des lettres, des chiffres et des tirets.
    b. Pour Description, saisissez une brève description du groupe d'options, par exemple **`MSDTC option group for SQL Server SE 2016`**. La description est utilisée à des fins d'affichage.
    c. Pour Moteur, choisissez sqlserver-se.
    d. Pour Version majeure du moteur, choisissez 13.00.
5. Sélectionnez Create (Créer).

CLI

L'exemple suivant crée un groupe d'options pour SQL Server Standard Edition 2016.

Pour créer le groupe d'options

- Utilisez l’une des commandes suivantes.

    Example

    Dans Linux, macOS, ou Unix:

```
aws rds create-option-group \
    --option-group-name msdtc-se-2016 \
    --engine-name sqlserver-se \
    --major-engine-version 13.00 \
    --option-group-description "MSDTC option group for SQL Server SE 2016"
```

Dans Windows:

```
aws rds create-option-group ^
    --option-group-name msdtc-se-2016 ^
    --engine-name sqlserver-se ^
    --major-engine-version 13.00 ^
    --option-group-description "MSDTC option group for SQL Server SE 2016"
```

Ajouter l'MSDTCoption au groupe d'options

Ensuite, utilisez le AWS Management Console ou AWS CLI pour ajouter l'`MSDTCoption` au groupe d'options.

Les paramètres d'option suivants sont requis :

- Port : port que vous utilisez pour accéderMSDTC. Les valeurs autorisées sont comprises entre 1150 et 49151, sauf 1234, 1434, 3260, 3343, 3389 et 47001. La valeur par défaut est 5000.

  Assurez-vous que le port que vous souhaitez utiliser est activé dans vos règles de pare-feu. Assurez-vous également que ce port est activé dans les règles entrantes et sortantes pour le groupe de sécurité associé à votre instance de base de données. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.
- Groupes de sécurité : les appartenances aux groupes de VPC sécurité pour votre RDS instance de base de données.
- Type d'authentification – Mode d'authentification entre les hôtes. Les types d'authentification suivants sont pris en charge :
  - Mutuel : les RDS instances sont authentifiées mutuellement à l'aide de l'authentification intégrée. Si cette option est sélectionnée, toutes les instances associées à ce groupe d'options doivent être jointes au domaine.

  - Aucun(e) – Aucune authentification n'est effectuée entre les hôtes. Nous ne recommandons pas d'utiliser ce mode dans les environnements de production.
- Taille du journal des transactions : taille du journal des MSDTC transactions. Les valeurs autorisées sont comprises entre 4 et 1 024 Mo. La taille par défaut est 4 Mo.

Les paramètres d'option suivants sont facultatifs :

- Activer les connexions entrantes : s'il faut autoriser les MSDTC connexions entrantes aux instances associées à ce groupe d'options.
- Activer les connexions sortantes : s'il faut autoriser les MSDTC connexions sortantes depuis les instances associées à ce groupe d'options.
- Activer XA – Indique si vous souhaitez autoriser les transactions XA. Pour de plus amples informations sur le protocole XA, veuillez consulter XA Specification.
- Activer SNA LU — S'il faut autoriser l'utilisation du protocole SNA LU pour les transactions distribuées. Pour plus d'informations sur la prise en charge du protocole SNA IBMCICSLU, consultez la section Gestion des transactions LU 6.2 dans la documentation Microsoft.

Console

Pour ajouter l'MSDTCoption

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options que vous venez de créer.
4. Sélectionnez Ajouter une option.
5. Sous Détails de l'option, choisissez MSDTCle nom de l'option.
6. Sous Paramètres des options :
    a. Pour Port, entrez le numéro de port d'accèsMSDTC. La valeur par défaut est 5000.
    b. Pour les groupes de sécurité, choisissez le groupe de VPC sécurité à associer à l'option.
    c. Pour Type d'authentification, choisissez Mutuelle ou Aucun(e).
    d. Pour Taille du journal des transactions, entrez une valeur comprise entre 4 et 1 024. La valeur par défaut est 4.

7. Sous Configuration supplémentaire, procédez comme suit :

   a. Pour Connexions, choisissez, selon vos besoins, Activer les connexions entrantes et Activer les connexions sortantes.

   b. Pour les protocoles autorisés, choisissez selon les besoins Enable XA et Enable SNA LU.

8. Sous Scheduling (Planification), choisissez si vous souhaitez ajouter l'option immédiatement ou lors du créneau de maintenance suivant.

9. Sélectionnez Ajouter une option.

   Pour ajouter cette option, aucun redémarrage n'est requis.

CLI

Pour ajouter l'MSDTCoption

1. Créez un JSON fichier, par exemple`msdtc-option.json`, avec les paramètres obligatoires suivants.

```
{
"OptionGroupName":"msdtc-se-2016",
"OptionsToInclude": [
 {
 "OptionName":"MSDTC",
 "Port":5000,
 "VpcSecurityGroupMemberships":["sg-0abcdef123"],
 "OptionSettings":[{"Name":"AUTHENTICATION","Value":"MUTUAL"},
{"Name":"TRANSACTION_LOG_SIZE","Value":"4"}]
 }],
"ApplyImmediately": true
}
```

2. Ajoutez l'option `MSDTC` au groupe d'options.

   Example

   Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --cli-input-json file://msdtc-option.json \
    --apply-immediately
```

Dans Windows:

```
aws rds add-option-to-option-group ^
    --cli-input-json file://msdtc-option.json ^
    --apply-immediately
```

Aucun redémarrage n'est requis.

Création du groupe de paramètres pour MSDTC

Créez ou modifiez un groupe de paramètres pour le `in-doubt xact resolution` paramètre correspondant à l'édition et à la version SQL du serveur de votre instance de base de données.

Console

L'exemple suivant crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez Créer un groupe de paramètres.
4. Dans le volet Créer un groupe de paramètres, faites ce qui suit :
   a. Pour Famille de groupes de paramètres, choisissez sqlserver-se-13.0.
   b. Pour Nom du groupe, saisissez un identifiant pour le groupe de paramètres, tel que **`msdtc-sqlserver-se-13`**.
   c. Pour Description, saisissez **`in-doubt xact resolution`**.
5. Sélectionnez Create (Créer).

CLI

L'exemple suivant crée un groupe de paramètres pour SQL Server Standard Edition 2016.

Pour créer le groupe de paramètres

- Utilisez l'une des commandes suivantes.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-parameter-group \
    --db-parameter-group-name msdtc-sqlserver-se-13 \
    --db-parameter-group-family "sqlserver-se-13.0" \
    --description "in-doubt xact resolution"
```

Dans Windows:

```
aws rds create-db-parameter-group ^
    --db-parameter-group-name msdtc-sqlserver-se-13 ^
    --db-parameter-group-family "sqlserver-se-13.0" ^
    --description "in-doubt xact resolution"
```

Modification du paramètre pour MSDTC

Modifiez le `in-doubt xact resolution` paramètre dans le groupe de paramètres qui correspond à l'édition et à la version SQL du serveur de votre instance de base de données.

PourMSDTC, définissez le `in-doubt xact resolution` paramètre sur l'une des valeurs suivantes :

- 1—`Presume commit`. Toute MSDTC transaction douteuse est présumée avoir été commise.
- 2—`Presume abort`. Toute MSDTC transaction douteuse est présumée avoir été interrompue.

Pour de plus amples informations, veuillez consulter in-doubt xact resolution Server Configuration Option dans la documentation Microsoft.

Console

L'exemple suivant modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Dans le panneau de navigation, choisissez Groupes de paramètres.
3. Choisissez le groupe de paramètres, par exemple msdtc-sqlserver-se-13.
4. Sous Paramètres, filtrez la liste des paramètres pour **`xact`**.
5. Choisissez in-doubt xact resolution.
6. Choisissez Modifier les paramètres.
7. Entrez **1** ou **2**.
8. Sélectionnez Enregistrer les modifications.

CLI

L'exemple suivant modifie le groupe de paramètres que vous avez créé pour SQL Server Standard Edition 2016.

Pour modifier le groupe de paramètres

- Utilisez l'une des commandes suivantes.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name msdtc-sqlserver-se-13 \
    --parameters "ParameterName='in-doubt xact
 resolution',ParameterValue=1,ApplyMethod=immediate"
```

  Dans Windows:

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name msdtc-sqlserver-se-13 ^
    --parameters "ParameterName='in-doubt xact
 resolution',ParameterValue=1,ApplyMethod=immediate"
```

Association du groupe d'options et du groupe de paramètres à l'instance de base de données

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour associer le groupe d'MSDTCoptions et le groupe de paramètres à l'instance de base de données.

Console

Vous pouvez associer le groupe d'MSDTCoptions et le groupe de paramètres à une instance de base de données nouvelle ou existante.

- Pour une nouvelle instance de base de données, associez-les lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, associez-les en modifiant l'instance. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

> **Note**
>
> Si vous utilisez une instance de base de données existante jointe à un domaine, elle doit déjà être associée à un domaine Active Directory et à un rôle AWS Identity and Access Management (IAM). Si vous créez une nouvelle instance jointe à un domaine, spécifiez un domaine et IAM un rôle Active Directory existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory AWS géré avec RDS pour SQL Server.

CLI

Vous pouvez associer le groupe d'MSDTCoptions et le groupe de paramètres à une instance de base de données nouvelle ou existante.

> **Note**
>
> Si vous utilisez une instance de base de données jointe à un domaine existante, elle doit déjà être associée à un domaine Active Directory et à un IAM rôle. Si vous créez une nouvelle instance jointe à un domaine, spécifiez un domaine et IAM un rôle Active Directory existants. Pour de plus amples informations, veuillez consulter Utilisation d'Active Directory AWS géré avec RDS pour SQL Server.

Pour créer une instance de base de données avec le groupe d'MSDTCoptions et le groupe de paramètres

- Spécifiez le type de moteur de base de données et la version majeure utilisés lors de la création du groupe d'options.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --db-instance-class db.m5.2xlarge \
    --engine sqlserver-se \
    --engine-version 13.00.5426.0.v1 \
    --allocated-storage 100 \
    --manage-master-user-password \
    --master-username admin \
    --storage-type gp2 \
    --license-model li \
    --domain-iam-role-name my-directory-iam-role \
    --domain my-domain-id \
    --option-group-name msdtc-se-2016 \
    --db-parameter-group-name msdtc-sqlserver-se-13
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --db-instance-class db.m5.2xlarge ^
    --engine sqlserver-se ^
    --engine-version 13.00.5426.0.v1 ^
    --allocated-storage 100 ^
    --manage-master-user-password ^
    --master-username admin ^
    --storage-type gp2 ^
    --license-model li ^
    --domain-iam-role-name my-directory-iam-role ^
    --domain my-domain-id ^
    --option-group-name msdtc-se-2016 ^
    --db-parameter-group-name msdtc-sqlserver-se-13
```

Pour modifier une instance de base de données et associer le groupe d'MSDTCoptions et le groupe de paramètres

- Utilisez l’une des commandes suivantes.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --option-group-name msdtc-se-2016 \
    --db-parameter-group-name msdtc-sqlserver-se-13 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --option-group-name msdtc-se-2016 ^
    --db-parameter-group-name msdtc-sqlserver-se-13 ^
    --apply-immediately
```

Modifier l'MSDTCoption

Après avoir activé l'option `MSDTC`, vous pouvez modifier ses paramètres. Pour de plus amples informations sur la modification des paramètres d'option, veuillez consulter Modification d'un paramètre d'option.

> **Note**
>
> Certaines modifications apportées aux paramètres des MSDTC options nécessitent le redémarrage du MSDTC service. Cette exigence peut affecter les transactions distribuées en cours d'exécution.

Utilisation des transactions

Utilisation des transactions distribuées

Dans Amazon RDS for SQL Server, vous exécutez des transactions distribuées de la même manière que des transactions distribuées exécutées sur site :

- En utilisant. NETframework de transactions `System.Transactions` promotables, qui optimise les transactions distribuées en reportant leur création jusqu'à ce qu'elles soient nécessaires.

  Dans ce cas, la promotion est automatique et ne nécessite aucune intervention de votre part. S'il n'y a qu'un gestionnaire de ressources dans la transaction, aucune promotion n'est effectuée. Pour de plus amples informations sur les portées des transactions implicites, veuillez consulter Implementing an Implicit Transaction using Transaction Scope dans la documentation Microsoft.

  Les transactions promotionnelles sont prises en charge avec celles-ci. NETimplémentations :

  - En commençant parADO. NET2.0, `System.Data.SqlClient` prend en charge les transactions promotionnelles avec le SQL serveur. Pour plus d'informations, consultez la section Intégration de System.Transactions au SQL serveur dans la documentation Microsoft.
  - ODP. NETsoutient`System.Transactions`. Une transaction locale est créée pour la première connexion ouverte dans la portée `TransactionsScope` à Oracle Database 11g version 1 (version 11.1) et ultérieures. Lorsqu'une deuxième connexion est ouverte, cette transaction est automatiquement promue en tant que transaction distribuée. Pour plus d'informations sur la prise en charge des transactions distribuées dansODP. NET, consultez la section Intégration de Microsoft Distributed Transaction Coordinator dans la documentation Microsoft.
- Utilisation de l'instruction `BEGIN DISTRIBUTED TRANSACTION`. Pour plus d'informations, consultez BEGINDISTRIBUTEDTRANSACTION(Transact-SQL) dans la documentation Microsoft.

Utilisation de transactions XA

À partir de la RDS version 14.00.3223.3 de SQL Server 2017, vous pouvez contrôler les transactions distribuées à l'aide de. JDBC Lorsque vous définissez le paramètre d'`Enable XA`option sur `true` dans l'`MSDTC`option, les JDBC transactions RDS sont automatiquement activées et le `SqlJDBCXAUser` rôle est attribué à l'`guest`utilisateur. Cela permet d'exécuter des transactions distribuées viaJDBC. Pour plus d'informations, y compris pour voir un exemple de code, consultez Comprendre les transactions XA dans la documentation Microsoft.

Utilisation du suivi des transactions

RDSprend en charge le contrôle des traces de MSDTC transactions et leur téléchargement depuis l'RDSinstance de base de données à des fins de dépannage. Vous pouvez contrôler les sessions de suivi des transactions en exécutant la procédure RDS stockée suivante.

```
exec msdb.dbo.rds_msdtc_transaction_tracing 'trace_action',
[@traceall='0|1'],
```

```
[@traceaborted='0|1'],
[@tracelong='0|1'];
```

Les paramètres suivants sont obligatoires :

- `trace_action` – Action de suivi. Il peut être `START`, `STOP` ou `STATUS`.

Les paramètres suivants sont facultatifs :

- `@traceall` – La valeur 1 permet de suivre toutes les transactions distribuées. La valeur par défaut est 0.
- `@traceaborted` – La valeur 1 permet de suivre les transactions distribuées annulées. La valeur par défaut est 0.
- `@tracelong` – La valeur 1 permet de suivre les transactions distribuées de longue durée. La valeur par défaut est 0.

Example de l'action de START traçage

Pour démarrer une nouvelle session de suivi des transactions, exécutez l'exemple d'instruction suivant.

```
exec msdb.dbo.rds_msdtc_transaction_tracing 'START',
@traceall='0',
@traceaborted='1',
@tracelong='1';
```

> **Note**
>
> Une seule session de suivi des transactions peut être active à la fois. Si une nouvelle commande `START` de session de suivi est émise alors qu'une session de suivi est active, une erreur est renvoyée et la session de suivi active reste inchangée.

Example de l'action de STOP traçage

Pour arrêter une session de suivi des transactions, exécutez l'instruction suivante.

```
exec msdb.dbo.rds_msdtc_transaction_tracing 'STOP'
```

Cette instruction arrête la session de suivi des transactions active et enregistre les données de suivi des transactions dans le répertoire des journaux de l'RDSinstance de base de données. La première ligne de la sortie contient le résultat global, et les lignes suivantes indiquent les détails de l'opération.

Vous trouverez ci-après un exemple d'arrêt de session de suivi réussi.

```
OK: Trace session has been successfully stopped.
Setting log file to: D:\rdsdbdata\MSDTC\Trace\dtctrace.log
Examining D:\rdsdbdata\MSDTC\Trace\msdtctr.mof for message formats,  8 found.
Searching for TMF files on path: (null)
Logfile D:\rdsdbdata\MSDTC\Trace\dtctrace.log:
 OS version    10.0.14393  (Currently running on 6.2.9200)
 Start Time    <timestamp>
 End Time      <timestamp>
 Timezone is   @tzres.dll,-932 (Bias is 0mins)
 BufferSize            16384 B
 Maximum File Size     10 MB
 Buffers  Written      Not set (Logger may not have been stopped).
 Logger Mode Settings (11000002) ( circular paged
 ProcessorCount         1
Processing completed   Buffers: 1, Events: 3, EventsLost: 0 :: Format Errors: 0,
 Unknowns: 3
Event traces dumped to d:\rdsdbdata\Log\msdtc_<timestamp>.log
```

Vous pouvez utiliser les informations détaillées pour rechercher le nom du fichier journal généré. Pour plus d'informations sur le téléchargement de fichiers journaux depuis l'RDSinstance de base de données, consultezSurveillance des fichiers journaux RDSAmazon.

Les journaux de session de suivi sont conservés sur l'instance pendant 35 jours. Les journaux de session de suivi plus anciens sont automatiquement supprimés.

Example de l'action de STATUS traçage

Pour suivre le statut d'une session de suivi des transactions, exécutez l'instruction suivante.

```
exec msdb.dbo.rds_msdtc_transaction_tracing 'STATUS'
```

Cette instruction affiche les éléments suivants sous forme de lignes séparées dans l'ensemble de résultats.

```
OK
SessionStatus: <Started|Stopped>
```

```
TraceAll: <True|False>
TraceAborted: <True|False>
TraceLongLived: <True|False>
```

La première ligne indique le résultat global de l'opération : `OK` ou `ERROR` avec des détails, le cas échéant. Les lignes suivantes indiquent des détails sur le statut de la session de suivi :

- `SessionStatus`, les valeurs suivantes sont possibles :
  - `Started` si une session de suivi est en cours d'exécution.
  - `Stopped` si aucune session de suivi n'est en cours d'exécution.
- Les indicateurs de session de suivi peuvent être `True` ou `False` en fonction de la façon dont ils ont été définis dans la commande `START`.

Désactivation MSDTC

Pour la désactiverMSDTC, supprimez l'`MSDTC`option de son groupe d'options.

Console

Pour supprimer l'MSDTCoption de son groupe d'options

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Choisissez le groupe d'options avec l'option `MSDTC` (`msdtc-se-2016` dans les exemples précédents).
4. Choisissez Supprimer une option.
5. Sous Options de suppression, sélectionnez MSDTCOptions à supprimer.
6. Sous Appliquer immédiatement, choisissez Oui pour supprimer l'option immédiatement, ou Non pour la supprimer lors du prochain créneau de maintenance.
7. Sélectionnez Delete (Supprimer).

CLI

Pour supprimer l'MSDTCoption de son groupe d'options

- Utilisez l’une des commandes suivantes.

Example

Dans Linux, macOS, ou Unix:

```
aws rds remove-option-from-option-group \
    --option-group-name msdtc-se-2016 \
    --options MSDTC \
    --apply-immediately
```

Dans Windows:

```
aws rds remove-option-from-option-group ^
    --option-group-name msdtc-se-2016 ^
    --options MSDTC ^
    --apply-immediately
```

Résolution des problèmes MSDTC pour RDS for SQL Server

Dans certains cas, vous pouvez rencontrer des difficultés pour établir une connexion entre l'MSDTCexécution sur un ordinateur client et le MSDTC service exécuté sur une instance de base de données RDS pour le SQL serveur. Dans ce cas, assurez-vous que :

- Les règles entrantes pour le groupe de sécurité associé à l'instance de base de données sont configurées correctement. Pour plus d'informations, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.
- Votre ordinateur client est configuré correctement.
- Les règles de MSDTC pare-feu sur votre ordinateur client sont activées.

Pour configurer l'ordinateur client

1. Ouvrez Component Services (Services de composants).

   Ou, dans Server Manager (Gestionnaire de serveur), choisissez Tools (Outils), puis Component Services (Services de composants).
2. Développez successivement Component Services (Services de composants), Computers (Ordinateurs), My Computer (Mon ordinateur) et Distributed Transaction Coordinator (Coordinateur de transactions distribuées).

3. Ouvrez le menu contextuel (clic droit) pour Local DTC et choisissez Propriétés.
4. Choisissez l'onglet Security (Sécurité).
5. Choisissez toutes les options suivantes :

   - DTCAccès au réseau
   - Allow Inbound (Autoriser le trafic entrant)
   - Allow Outbound (Autoriser le trafic sortant)
6. Assurez-vous que le mode d'authentification correct est choisi :

   - Mutual Authentication Required (Authentification mutuelle requise) – La machine cliente est jointe au même domaine que les autres nœuds participant à une transaction distribuée, ou une relation d'approbation est configurée entre les domaines.
   - No Authentication Required (Aucune authentification requise) – Tous les autres cas.
7. Choisissez OK pour enregistrer vos modifications.
8. Si vous êtes invité à redémarrer le service, choisissez Oui.

Pour activer les règles de MSDTC pare-feu

1. Ouvrez le pare-feu Windows, puis choisissez Paramètres avancés.

   Ou, dans Gestionnaire de serveur, choisissez Outils, puis Pare-feu Windows avec fonctions avancées de sécurité.

   > **Note**
   >
   > En fonction de votre système d'exploitation, le Pare-feu Windows peut s'appeler « Pare-feu Windows Defender ».
2. Choisissez Règles entrantes dans le volet de gauche.
3. Activez, si ce n'est pas déjà le cas, les règles de pare-feu suivantes :

   - Coordinateur des transactions distribuées (RPC)
   - Coordinateur des transactions distribuées (RPC) - EPMAP
   - Coordinateur des transactions distribuées (TCP-In)
4. Fermez le pare-feu Windows.

Tâches DBA courantes pour Amazon RDS pour Microsoft SQL Server

Cette section décrit les implémentations spécifiques à Amazon RDS de certaines tâches d'administration de base de données courantes pour les instances de base de données exécutant le moteur de base de données Microsoft SQL Server. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de bases de données et limite l'accès à certaines tables et procédures système qui requièrent des privilèges avancés.

> **Note**
>
> Lors de l'utilisation d'une instance de base de données SQL Server, vous pouvez exécuter des scripts pour modifier une base de données nouvellement créée, mais vous ne pouvez pas modifier la base de données [model], celle utilisée comme modèle pour les nouvelles bases de données.

Rubriques

- Accès à la base de données tempdb sur les instances de base de données Microsoft SQL Server sur Amazon RDS
- Analyse de la charge de travail de votre base de données sur une instance de base de données Amazon RDS for SQL Server avec Database Engine Tuning Advisor
- Modification du db_ownerrdsa compte pour votre base de données Amazon RDS for SQL Server
- Gestion des classements et des jeux de caractères pour Amazon RDS pour Microsoft SQL Server
- Création d'un utilisateur de base de données pour Amazon RDS for SQL Server
- Déterminer un modèle de restauration pour votre base de données Amazon RDS for SQL Server
- Déterminer l'heure du dernier basculement d'Amazon RDS for Server SQL
- Résolution point-in-time-recovery des défaillances dues à un écart dans le numéro de séquence du journal
- Refuser ou autoriser l'affichage des noms de base de données pour Amazon RDS for SQL Server
- Désactivation des insertions rapides lors du chargement en masse pour Amazon RDS for Server SQL
- Suppression d'une base de données dans une instance de base de données Amazon RDS pour Microsoft SQL Server

- Modification du nom d'une base de données Amazon RDS for Microsoft SQL Server dans le cadre d'un déploiement multi-AZ
- Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL
- Restauration d'instances de base de données dont la licence est terminée pour Amazon for Server RDS SQL
- Transition d'une base de données Amazon RDS for SQL Server de OFFLINE vers ONLINE
- Utilisation de la capture des données de modification pour Amazon RDS for SQL Server
- Utilisation de l'agent SQL Server pour Amazon RDS
- Utilisation des journaux Amazon RDS pour Microsoft SQL Server
- Utilisation de fichiers de trace et de vidage pour Amazon RDS for SQL Server

Accès à la base de données tempdb sur les instances de base de données Microsoft SQL Server sur Amazon RDS

Vous pouvez accéder à la `tempdb` base de données sur vos instances de base de données Microsoft SQL Server sur AmazonRDS. Vous pouvez exécuter du code `tempdb` en utilisant Transact- SQL via Microsoft SQL Server Management Studio (SSMS) ou toute autre application SQL client standard. Pour plus d'informations sur la connexion à votre instance de base de données, consultez Connexion à votre instance de base de données Microsoft SQL Server.

L'utilisateur principal pour votre instance de base de données bénéficie d'un accès `CONTROL` à `tempdb` afin qu'il puisse modifier les options de la base de données `tempdb`. L'utilisateur principal n'est pas le propriétaire de la base de données `tempdb`. Si nécessaire, l'utilisateur principal peut accorder un accès `CONTROL` à d'autres utilisateurs afin qu'ils puissent eux aussi modifier les options de la base de données `tempdb`.

> **Note**
>
> Vous ne pouvez pas exécuter les commandes de console de base de données (DBCC) sur la `tempdb` base de données.

Modification des options de la base de données tempdb

Vous pouvez modifier les options de base de données sur la `tempdb` base de données de vos RDS instances de base de données Amazon. Pour plus d'informations sur les options qui peuvent être modifiées, veuillez consulter Base de données tempdb dans la documentation Microsoft.

Les options de base de données telles que les options de taille maximale des fichiers sont persistantes une fois que vous redémarrez votre instance de base de données. Vous pouvez modifier les options de base de données pour optimiser les performances lors de l'importation des données et pour éviter le manque d'espace de stockage.

Optimisation des performances lors de l'importation de données

Afin d'optimiser les performances lors de l'importation de grandes quantités de données dans votre instance de base de données, définissez les propriétés `SIZE` et `FILEGROWTH` de la base de données tempdb sur des grands chiffres. Pour plus d'informations sur la façon d'optimiser `tempdb`, veuillez consulter Optimisation des performances de la base de données tempdb dans la documentation Microsoft.

L'exemple suivant illustre la définition de la taille sur 100 Go et la croissance des fichiers sur 10 pour cent.

```
alter database[tempdb] modify file (NAME = N'templog', SIZE=100GB, FILEGROWTH = 10%)
```

Prévention des problèmes de stockage

Pour éviter que la base de données `tempdb` utilise tout l'espace disque disponible, définissez la propriété `MAXSIZE`. L'exemple suivant illustre la définition de la propriété sur 2 048 Mo.

```
alter database [tempdb] modify file (NAME = N'templog', MAXSIZE = 2048MB)
```

Réduction de la base de données tempdb

Il existe deux méthodes pour réduire la `tempdb` base de données sur votre RDS instance de base de données Amazon. Vous pouvez utiliser la procédure `rds_shrink_tempdbfile` ou vous pouvez définir la propriété `SIZE`.

Utilisation de la procédure rds_shrink_tempdbfile

Vous pouvez utiliser la RDS procédure Amazon `msdb.dbo.rds_shrink_tempdbfile` pour réduire la `tempdb` base de données. Vous pouvez uniquement appeler `rds_shrink_tempdbfile` si vous disposez de l'accès `CONTROL` à `tempdb`. Lorsque vous appelez `rds_shrink_tempdbfile`, il n'y a aucun temps d'arrêt pour votre instance de base de données.

La procédure `rds_shrink_tempdbfile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`@temp_filename`	SYSNAME	—	obligatoire	Le nom logique du fichier à réduire.
`@target_size`	int	null	facultatif	La nouvelle taille du fichier en mégaoctets.

L'exemple suivant permet d'obtenir les noms des fichiers de la base de données `tempdb`.

```
use tempdb;
```

```
GO

select name, * from sys.sysfiles;
GO
```

L'exemple suivant réduit un fichier de base de données `tempdb` nommé `test_file` et demande une nouvelle taille de `10` mégaoctets :

```
exec msdb.dbo.rds_shrink_tempdbfile @temp_filename = N'test_file', @target_size = 10;
```

Configuration de la SIZE propriété

Vous pouvez également réduire la base de données `tempdb` en configurant la propriété `SIZE` et en redémarrant votre instance de base de données. Pour plus d'informations sur le redémarrage de votre instance de base de données, consultez Redémarrage d'une instance de base de données.

L'exemple suivant illustre la définition de la propriété `SIZE` sur 1 024 Mo.

```
alter database [tempdb] modify file (NAME = N'templog', SIZE = 1024MB)
```

Configuration de TempDB pour les déploiements multi-AZ

Si votre instance de base de données RDS for SQL Server est dans un déploiement multi-AZ utilisant la mise en miroir de base de données (DBM) ou les groupes de disponibilité Always On (AGs), gardez à l'esprit les considérations suivantes concernant l'utilisation de la `tempdb` base de données.

Vous ne pouvez pas répliquer les `tempdb` données de votre instance de base de données principale vers votre instance de base de données secondaire. Lorsque vous basculez vers une instance de base de données secondaire, `tempdb` cette instance de base de données secondaire sera vide.

Vous pouvez synchroniser la configuration des options de `tempdb` base de données, y compris ses paramètres de dimensionnement des fichiers et de croissance automatique, entre votre instance de base de données principale et votre instance de base de données secondaire. La synchronisation de la `tempDB` configuration est prise en charge sur toutes RDS les versions SQL du serveur. Vous pouvez activer la synchronisation automatique de la `tempdb` configuration à l'aide de la procédure stockée suivante :

```
EXECUTE msdb.dbo.rds_set_system_database_sync_objects @object_types = 'TempDbFile';
```

> **Important**
>
> Avant d'utiliser la procédure `rds_set_system_database_sync_objects` stockée, assurez-vous d'avoir défini votre `tempdb` configuration préférée sur votre instance de base de données principale plutôt que sur votre instance de base de données secondaire. Si vous avez modifié la configuration sur votre instance de base de données secondaire, votre `tempdb` configuration préférée peut être supprimée lorsque vous activez la synchronisation automatique.

Vous pouvez utiliser la fonction suivante pour vérifier si la synchronisation automatique de la `tempdb` configuration est activée :

```
SELECT * from msdb.dbo.rds_fn_get_system_database_sync_objects();
```

Lorsque la synchronisation automatique de la `tempdb` configuration est activée, une valeur est renvoyée pour le `object_class` champ. Lorsqu'elle est désactivée, aucune valeur n'est renvoyée.

Vous pouvez utiliser la fonction suivante pour savoir à quel moment les objets ont été synchronisés pour la dernière fois, dans le UTC temps :

```
SELECT * from msdb.dbo.rds_fn_server_object_last_sync_time();
```

Par exemple, si vous avez modifié la `tempdb` configuration à 01h00 puis que vous exécutez la `rds_fn_server_object_last_sync_time` fonction, la valeur renvoyée pour `last_sync_time` doit être postérieure à 01h00, ce qui indique qu'une synchronisation automatique s'est produite.

Si vous utilisez également la réplication des tâches de l'agent SQL serveur, vous pouvez activer la réplication à la fois pour les tâches de l'SQLagent et pour la `tempdb` configuration en les fournissant dans le `@object_type` paramètre :

```
EXECUTE msdb.dbo.rds_set_system_database_sync_objects @object_types =
 'SQLAgentJob,TempDbFile';
```

Pour plus d'informations sur la réplication des tâches de l'agent SQL serveur, consultezActivation de la réplication des tâches de l'agent SQL Server.

Au lieu d'utiliser la procédure `rds_set_system_database_sync_objects` stockée pour garantir la synchronisation automatique des modifications de `tempdb` configuration, vous pouvez utiliser l'une des méthodes manuelles suivantes :

> **Note**
>
> Nous recommandons d'activer la synchronisation automatique de la `tempdb` configuration à l'aide de la procédure `rds_set_system_database_sync_objects` stockée. L'utilisation de la synchronisation automatique évite d'avoir à effectuer ces tâches manuelles chaque fois que vous modifiez votre `tempdb` configuration.

- Tout d'abord, modifiez votre instance de base de données et désactivez le déploiement multi-AZ, puis modifier tempdb, puis enfin réactivez le déploiement multi-AZ. Cette méthode n'entraîne aucun temps d'arrêt.

  Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
- Tout d'abord, modifiez `tempdb` dans l'instance principale d'origine, puis exécutez un basculement manuel et enfin modifiez `tempdb` dans la nouvelle instance principale. Cette méthode implique un temps d'arrêt.

  Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Analyse de la charge de travail de votre base de données sur une instance de base de données Amazon RDS for SQL Server avec Database Engine Tuning Advisor

Database Engine Tuning Advisor est une application cliente fournie par Microsoft qui analyse la charge de travail des bases de données et recommande un ensemble d'index optimal pour vos bases de données Microsoft SQL Server en fonction du type de requêtes que vous exécutez. Comme SQL Server Management Studio, vous exécutez Tuning Advisor à partir d'un ordinateur client qui se connecte à votre RDS instance de base de données Amazon qui exécute SQL Server. L'ordinateur client peut être un ordinateur local que vous exécutez sur site au sein de votre propre réseau ou une instance Amazon EC2 Windows exécutée dans la même région que votre instance Amazon RDS DB.

Cette section montre comment capturer une charge de travail pour que l'Assistant Paramétrage l'analyse. Il s'agit du processus préféré pour capturer une charge de travail, car Amazon RDS restreint l'accès de l'hôte à l'instance SQL du serveur. Pour plus d'informations, consultez Assistant Paramétrage du moteur de base de données dans la documentation Microsoft.

Pour utiliser l'Assistant Paramétrage, vous devez lui fournir ce qu'on appelle une charge de travail. Une charge de travail est un ensemble d'SQLinstructions Transact- qui s'exécutent sur une ou plusieurs bases de données que vous souhaitez ajuster. Database Engine Tuning Advisor utilise des fichiers de trace, des tables de trace, des SQL scripts Transact ou des XML fichiers comme entrée de charge de travail lors du réglage des bases de données. Lorsque vous travaillez avec AmazonRDS, une charge de travail peut être un fichier sur un ordinateur client ou une table de base de données sur une base de données Amazon RDS for SQL Server accessible sur votre ordinateur client. Le fichier ou la table doit contenir des requêtes sur les bases de données que vous voulez régler dans un format adapté à la relecture.

Pour que l'Assistant Paramétrage soit le plus efficace, une charge de travail doit être aussi réaliste que possible. Vous pouvez générer un fichier de charge de travail ou une table en exécutant une trace sur votre instance de base de données. Pendant l'exécution d'une trace, vous pouvez simuler une charge sur votre instance de base de données ou exécuter vos applications avec une charge normale.

Il existe deux types de trace : côté client et côté serveur. Un suivi côté client est plus facile à configurer et vous pouvez suivre les événements de suivi capturés en temps réel dans SQL Server Profiler. Un suivi côté serveur est plus complexe à configurer et nécessite un certain script de transaction. SQL En outre, étant donné que la trace est écrite dans un fichier sur l'RDSinstance de base de données Amazon, l'espace de stockage est consommé par la trace. Il importe de tracer la quantité d'espace de stockage qu'une trace côté serveur utilise, parce que l'instance de base de données peut entrer dans un état de stockage complet et n'être plus disponible si elle se trouve à court d'espace de stockage.

Pour une trace côté client, lorsqu'une quantité suffisante de données de trace a été capturée dans le SQL Server Profiler, vous pouvez ensuite générer le fichier de charge de travail en enregistrant la trace dans un fichier sur votre ordinateur local ou dans une table de base de données sur une instance de base de données disponible sur votre ordinateur client. Le principal désavantage de l'utilisation d'une trace côté client est que la trace peut ne pas capturer toutes les requêtes quand elle est soumise à de lourdes charges. Cela pourrait affaiblir l'efficacité de l'analyse exécutée par l'Assistant Paramétrage du moteur de base de données. Si vous devez exécuter une trace soumise à

des charges massives et que vous voulez vous assurer qu'elle capture chaque requête pendant une session de trace, vous devez utiliser une trace côté serveur.

Pour une trace côté serveur, vous devez obtenir les fichiers de trace de l'instance de base de données en un fichier de charge de travail adapté ou vous pouvez enregistrer la trace sur une table de l'instance de base de données une fois la trace terminée. Vous pouvez utiliser le SQL Server Profiler pour enregistrer la trace dans un fichier sur votre ordinateur local ou demander au Tuning Advisor de lire la table de trace sur l'instance de base de données.

Exécution d'un suivi côté client sur une instance de base de données de SQL serveur

Pour exécuter un suivi côté client sur une instance de base de données de SQL serveur

1. Démarrez SQL Server Profiler. Il est installé dans le dossier Performance Tools du dossier d'instance de votre SQL serveur. Vous devez charger ou définir un modèle de définition de trace pour démarrer une trace côté client.
2. Dans le menu Fichier SQL du profileur de serveur, choisissez New Trace. Dans la boîte de dialogue Connect to Server (Se connecter au serveur), entrez le point de terminaison de l'instance de base de données, le port, l'identifiant principal et le mot de passe de la base de données sur laquelle vous souhaitez exécuter une trace.
3. Dans la boîte de dialogue Propriétés de la trace, entrez un nom de trace et choisissez un modèle de définition de trace. Un modèle par défaut, TSQL _Replay, est fourni avec l'application. Vous pouvez modifier ce modèle pour définir votre trace. Modifiez les événements et les informations relatives aux événements sous l'onglet Sélection des événements de la boîte de dialogue Propriétés de la trace.

   Pour plus d'informations sur les modèles de définition de trace et sur l'utilisation SQL du Server Profiler pour spécifier une trace côté client, consultez Database Engine Tuning Advisor dans la documentation Microsoft.
4. Démarrez le suivi côté client et observez les SQL requêtes en temps réel lorsqu'elles s'exécutent sur votre instance de base de données.
5. Sélectionnez Stop Trace (Arrêter la trace) dans le menu Fichier lorsque vous avez terminé la trace. Enregistrez les résultats comme fichier ou comme table de trace sur votre instance de base de données.

Exécution d'un suivi côté serveur sur une instance de base de données de SQL serveur

L'écriture de scripts pour créer une trace côté serveur peut être complexe et au-delà de la portée de ce document. Cette section contient des scripts que vous pouvez utiliser comme exemples. Comme pour une trace côté client, l'objectif est de créer un fichier de charge de travail ou une table de trace que vous pouvez ouvrir à l'aide de l'Assistant Paramétrage du moteur de base de données.

L'exemple abrégé suivant est un script qui démarre une trace côté serveur et capture les détails dans un fichier de charge de travail. La trace est initialement enregistrée dans le fichier RDSTrace .trc du répertoire D : \ RDSDBDATA \ Log et est reportée tous les 100 Mo, de sorte que les fichiers de trace suivants sont nommés RDSTrace _1.trc, _2.trc, etc. RDSTrace

```
DECLARE @file_name NVARCHAR(245) = 'D:\RDSDBDATA\Log\RDSTrace';
DECLARE @max_file_size BIGINT = 100;
DECLARE @on BIT = 1
DECLARE @rc INT
DECLARE @traceid INT

EXEC @rc = sp_trace_create @traceid OUTPUT, 2, @file_name, @max_file_size
IF (@rc = 0) BEGIN
   EXEC sp_trace_setevent @traceid, 10, 1, @on
   EXEC sp_trace_setevent @traceid, 10, 2, @on
   EXEC sp_trace_setevent @traceid, 10, 3, @on
 . . .
   EXEC sp_trace_setfilter @traceid, 10, 0, 7, N'SQL Profiler'
   EXEC sp_trace_setstatus @traceid, 1
   END
```

L'exemple suivant illustre un script qui arrête une trace. Notez qu'une trace créée par le précédent script continue à s'exécuter jusqu'à ce que vous arrêtiez explicitement la trace ou que le processus ne dispose plus d'espace disque suffisant.

```
DECLARE @traceid INT
SELECT @traceid = traceid FROM ::fn_trace_getinfo(default)
WHERE property = 5 AND value = 1 AND traceid <> 1

IF @traceid IS NOT NULL BEGIN
   EXEC sp_trace_setstatus @traceid, 0
   EXEC sp_trace_setstatus @traceid, 2
END
```

Vous pouvez enregistrer les résultats de la trace côté serveur sur une table de base de données et utiliser celle-ci comme charge de travail pour l'Assistant Paramétrage à l'aide de la fonction fn_trace_gettable. Les commandes suivantes chargent les résultats de tous les fichiers nommés RDSTrace .trc du répertoire D:\rdsdbdata\Log, y compris tous les fichiers de survol tels que RDSTrace _1.trc, dans une table nommée dans la base de données actuelle. RDSTrace

```
SELECT * INTO RDSTrace
FROM fn_trace_gettable('D:\rdsdbdata\Log\RDSTrace.trc', default);
```

Pour enregistrer un fichier de survol spécifique dans une table, par exemple le fichier RDSTrace _1.trc, spécifiez le nom du fichier de survol et remplacez 1 par défaut comme dernier paramètre de fn_trace_gettable.

```
SELECT * INTO RDSTrace_1
FROM fn_trace_gettable('D:\rdsdbdata\Log\RDSTrace_1.trc', 1);
```

Exécution de l'Assistant Paramétrage avec une trace

Une fois que vous créez une trace, comme fichier local ou comme table de base de données, vous pouvez exécuter l'Assistant Paramétrage sur votre instance de base de données. L'utilisation de Tuning Advisor avec Amazon RDS est le même que lorsque vous travaillez avec une instance de SQL serveur autonome et distante. Vous pouvez utiliser l'interface utilisateur de l'Assistant Paramétrage sur votre ordinateur client ou choisir l'utilitaire dta.exe à partir de la ligne de commande. Dans les deux cas, vous devez vous connecter à l'RDSinstance de base de données Amazon à l'aide du point de terminaison de l'instance de base de données et fournir votre nom d'utilisateur principal et votre mot de passe utilisateur principal lorsque vous utilisez Tuning Advisor.

L'exemple de code suivant illustre l'utilisation de l'utilitaire de ligne de commande dta.exe sur une RDS instance de base de données Amazon dont le point de terminaison est**dta.cnazcmklsdei.us-east-1.rds.amazonaws.com**. L'exemple inclut le nom d'utilisateur principal **admin** et le mot de passe de l'utilisateur principal **test**. L'exemple de base de données à régler se nomme ordinateur nommé **C:\RDSTrace.trc**. L'exemple de code de ligne de commande indique également une session de suivi nommée **RDSTrace1** et indique les fichiers de sortie vers la machine locale nommée **RDSTrace.sql** pour le script de SQL sortie, **RDSTrace.txt** pour un fichier de résultats et **RDSTrace.xml** pour un XML fichier d'analyse. Une table d'erreurs est également spécifiée dans la RDSDTA base de données nommée**RDSTraceErrors**.

```
dta -S dta.cnazcmklsdei.us-east-1.rds.amazonaws.com -U admin -P test -D RDSDTA -
if C:\RDSTrace.trc -s RDSTrace1 -of C:\ RDSTrace.sql -or C:\ RDSTrace.txt -ox C:\
 RDSTrace.xml -e RDSDTA.dbo.RDSTraceErrors
```

Voici le même exemple de code de ligne de commande, sauf que la charge de travail en entrée est une table sur l'RDSinstance Amazon distante nommée **`RDSTrace`** qui se trouve dans la **`RDSDTA`** base de données.

```
dta -S dta.cnazcmklsdei.us-east-1.rds.amazonaws.com -U admin -P test -D RDSDTA -it
 RDSDTA.dbo.RDSTrace -s RDSTrace1 -of C:\ RDSTrace.sql -or C:\ RDSTrace.txt -ox C:\
 RDSTrace.xml -e RDSDTA.dbo.RDSTraceErrors
```

Pour obtenir la liste complète des paramètres de ligne de commande de l'utilitaire dta, consultezUtilitaire dta dans la documentation Microsoft.

Modification du **`db_ownerrdsa`** compte pour votre base de données Amazon RDS for SQL Server

Lorsque vous créez ou restaurez une base de données dans une instance de base de données RDS for SQL Server, Amazon RDS définit le propriétaire de la base de données sur`rdsa`. Si vous avez un déploiement multi-AZ utilisant SQL Server Database Mirroring (DBM) ou Always On Availability Groups (AGs), Amazon RDS définit le propriétaire de la base de données sur l'instance de base de données secondaire sur. `NT AUTHORITY\SYSTEM` Le propriétaire de la base de données secondaire ne peut pas être modifié tant que l'instance de base de données secondaire n'est pas promue au rôle principal. Dans la plupart des cas, le fait de définir le propriétaire de la base de données sur `NT AUTHORITY\SYSTEM` ne pose pas de problèmes lors de l'exécution de requêtes, mais cela peut générer des erreurs pendant l'exécution de procédures stockées système telles que `sys.sp_updatestats`, qui ont besoin d'autorisations élevées pour s'exécuter.

Vous pouvez utiliser la requête suivante pour identifier le propriétaire des bases de données détenues par `NT AUTHORITY\SYSTEM` :

```
SELECT name FROM sys.databases WHERE SUSER_SNAME(owner_sid) = 'NT AUTHORITY\SYSTEM';
```

Vous pouvez utiliser la procédure RDS stockée Amazon `rds_changedbowner_to_rdsa` pour remplacer le propriétaire de la base de données par`rdsa`. Les bases de données suivantes ne sont pas autorisées à être utilisées avec `rds_changedbowner_to_rdsa` : `master, model, msdb, rdsadmin, rdsadmin_ReportServer, rdsadmin_ReportServerTempDB, SSISDB`.

Pour changer le propriétaire de la base de données`rdsa`, appelez la procédure `rds_changedbowner_to_rdsa` stockée et indiquez le nom de la base de données.

Example d'utilisation :

```
exec msdb.dbo.rds_changedbowner_to_rdsa 'TestDB1';
```

Les paramètres suivants sont obligatoires :

- `@db_name` – Nom de la base de données dont le propriétaire doit être remplacé par `rdsa`.

> **Important**
>
> Vous ne pouvez pas utiliser `rds_changedbowner_to_rdsa` pour remplacer le propriétaire d'une base de données par un identifiant autre que`rdsa`. Par exemple, vous ne pouvez pas modifier le propriétaire de l'identifiant avec lequel vous avez créé la base de données. Pour rétablir l'adhésion perdue au `db_owner` rôle de votre utilisateur principal alors qu'aucun autre utilisateur de base de données ne peut être utilisé pour accorder l'adhésion, réinitialisez le mot de passe de l'utilisateur principal pour obtenir l'adhésion au `db_owner` rôle. Pour de plus amples informations, veuillez consulter Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL.

Gestion des classements et des jeux de caractères pour Amazon RDS pour Microsoft SQL Server

Cette rubrique fournit des conseils sur la façon de gérer les classements et les jeux de caractères pour Microsoft SQL Server dans Amazon RDS. Il explique comment configurer les classements lors de la création de la base de données et comment les modifier ultérieurement, afin de garantir une gestion appropriée des données textuelles en fonction des exigences linguistiques et locales. En outre, il décrit les meilleures pratiques pour maintenir la compatibilité et les performances dans les environnements SQL Server sur Amazon RDS.

SQL Server prend en charge les classements à différents niveaux. Vous définissez le classement de serveur par défaut lorsque vous créez l'instance de base de données. Vous pouvez remplacer le classement au niveau de la base de données, de la table ou de la colonne.

Rubriques

- Classement de niveau serveur pour Microsoft SQL Server
- Classement au niveau de la base de données pour Microsoft SQL Server

Classement de niveau serveur pour Microsoft SQL Server

Lorsque vous créez une instance de base de données Microsoft SQL Server, vous pouvez définir le classement de serveur que vous souhaitez utiliser. Si vous ne choisissez aucun autre classement, le classement au niveau du serveur prend par défaut la valeur SQL_Latin1_General_ _CI_AS. CP1 Le classement de serveur est appliqué par défaut à toutes les bases de données et à tous les objets de base de données.

> **Note**
>
> Vous ne pouvez pas modifier le classement lorsque vous effectuez une restauration à partir d'un instantané de base de données.

Amazon RDS prend actuellement en charge les classements de serveur suivants :

Classement (Collation)	Description
Arabic_CI_AS	Arabe, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Chinois_PRC_ BIN2	Chinois-PRC, ordre de tri des points de code binaire
Chinese_PRC_CI_AS	Chinois - RPC, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Chinese_Taiwan_Stroke_CI_AS	Chinois de Taiwan, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur

Classement (Collation)	Description
Danish_Norwegian_CI_AS	Danois-Norvégien, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Danish_Norwegian_CI_AS_KS	danois-norvégien, insensible aux majuscule s, aux accents, au type Kana, insensible à la largeur
Danish_Norwegian_CI_AS_KS_WS	danois-norvégien, insensible aux majuscules, aux accents, au type Kana, à la largeur
Danish_Norwegian_CI_AS_WS	danois-norvégien, insensible aux majuscules, aux accents, au type Kana, à la largeur
Danish_Norwegian_CS_AI	danois-norvégien, distinction majuscules/ majuscules, insensible aux accents, insensible au type Kana, insensible à la largeur
Danish_Norwegian_CS_AI_KS	danois-norvégien, distinction majuscules/ majuscules, insensible aux accents, sensible au type kana, insensible à la largeur
Finlandais_Suédois_100_bin	Finno-Suédois-100, tri binaire
Finlandais_Suédois_100_ BIN2	100 finno-suédois, tri par comparaison de points de code binaire
finnois_suédois_100_CI_AI	100 finno-suédois, insensible aux majuscules et minuscules, insensible aux accents, insensible au type Kana, insensible à la largeur
finnois_suédois_100_CI_AS	100 finno-suédois, insensible aux majuscules et minuscules, aux accents, au type Kana, à la largeur

Classement (Collation)	Description
Finnish_Swedish_CI_AS	Finnois, suédois et suédois (Finlande), insensible à la casse, sensible aux accents, sensible aux caractères Kana et insensible à la largeur.
French_CI_AS	French, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Hebrew_BIN	Hebrew, tri binaire
Hebrew_CI_AS	Hebrew, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Japanese_BIN	Japanese, tri binaire
Japanese_CI_AS	Japanese, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Japanese_CS_AS	Japanese, sensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Japanese_XJIS_140_CI_AS	Japonais, insensible à la casse, sensible aux accents, insensible au type de kana, insensibl e à la largeur, caractères supplémentaires, insensible au sélecteur de variante
Japanese_XJIS_140_CI_AS_KS_VSS	Japonais, insensible à la casse, sensible aux accents, sensible au type de kana, insensibl e à la largeur, caractères supplémentaires, sensible au sélecteur de variante

Classement (Collation)	Description
Japanese_XJIS_140_CI_AS_VSS	Japonais, insensible à la casse, sensible aux accents, insensible au type de kana, insensibl e à la largeur, caractères supplémentaires, sensible au sélecteur de variante
Japanese_XJIS_140_CS_AS_KS_WS	Japonais, sensible à la casse, sensible aux accents, sensible au type de kana, sensible à la largeur, caractères supplémentaires, insensible au sélecteur de variante
Korean_Wansung_CI_AS	Korean-Wansung, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Latin1_General_100_BIN	Latin1-General-100, tri binaire
Latin1_Général_100_ BIN2	Latin1-General-100, ordre de tri des points de code binaire
Latin1_Général_100_ _ BIN2 UTF8	Latin1-General-100, ordre de tri des points de code binaire, codé en UTF-8
Latin1_General_100_CI_AS	Latin1-General-100, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Latin1_General_100_CI_AS_SC_ UTF8	Latin1-General-100, insensible à la casse, sensible aux accents, caractères supplémen taires, codé en UTF-8
Latin1_General_BIN	Latin1-General, tri binaire
Latin1_Général_ BIN2	Latin1-General, ordre de tri des points de code binaire

Classement (Collation)	Description
Latin1_General_CI_AI	Latin1-General, insensible à la casse, insensible aux accents, insensible au type de kana, insensible à la largeur
Latin1_General_CI_AS	Latin1-General, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Latin1_General_CI_AS_KS	Latin1-General, insensible à la casse, sensible aux accents, sensible au type de kana, insensible à la largeur
Latin1_General_CS_AS	Latin1-General, sensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Modern_Spanish_CI_AS	Modern-Spanish, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Polish_CI_AS	Polonais, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
SQL_1xCompat_ 0_CI_AS CP85	Latin1-General, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, ordre de tri SQL Server 49 sur la page de codes 850 pour les données non Unicode
SQL_Latin1_Général_ _CI_AI CP1	Latin1-General, insensible à la casse, insensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, l'ordre de tri SQL Server 54 sur la page de codes 1252 pour les données non Unicode

Classement (Collation)	Description
SQL_Latin1_General_ _CI_AS (par défautCP1)	Latin1-General, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, ordre de tri SQL Server 52 sur la page de codes 1252 pour les données non Unicode
SQL_Latin1_General_ _CS_AS CP1	Latin1-General, sensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, l'ordre de tri SQL Server 51 sur la page de codes 1252 pour les données non Unicode
SQL_Latin1_Général_ _CI_AI CP437	Latin1-General, insensible à la casse, insensibl e aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, ordre de tri SQL Server 34 sur la page de codes 437 pour les données non Unicode
SQL_Latin1_General_ 0_BIN CP85	Latin1-General, ordre de tri binaire pour les données Unicode, l'ordre de tri SQL Server 40 sur la page de codes 850 pour les données non Unicode
SQL_Latin1_Général_ 0_ CP85 BIN2	Latin1-General, ordre de tri des points de code binaire pour les données Unicode, l'ordre de tri SQL Server 40 sur la page de codes 850 pour les données non Unicode
SQL_Latin1_General_ 0_CI_AI CP85	Latin1-General, insensible à la casse, insensibl e aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, l'ordre de tri SQL Server 44 sur la page de codes 850 pour les données non Unicode

Classement (Collation)	Description
SQL_Latin1_General_ 0_CI_AS CP85	Latin1-General, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, ordre de tri SQL Server 42 sur la page de codes 850 pour les données non Unicode
SQL_Latin1_General_ _CI_AS CP1256	Latin1-General, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur pour les données Unicode, ordre de tri SQL Server 146 sur la page de codes 1256 pour les données non Unicode
SQL_Latin1_General_ _CS_AS CP1255	Latin1-Général, distinction majuscules/minuscules, sensible aux accents, insensible au type kana, insensible à la largeur pour les données Unicode, ordre de tri 137 de SQL Server sur la page de code 1255 pour les données non Unicode
Thai_CI_AS	Thaï, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur
Turkish_CI_AS	Turc, insensible à la casse, sensible aux accents, insensible au type de kana, insensible à la largeur

Pour choisir la classement :

- Si vous utilisez la console Amazon RDS, lors de la création d'une nouvelle instance de base de données, choisissez Additional configuration (Configuration supplémentaire), puis saisissez le classement dans le menu Collation (Classement). Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

- Si vous utilisez le AWS CLI, utilisez l'`--character-set-name`option avec la `create-db-instance` commande. Pour de plus amples informations, veuillez consulter create-db-instance.
- Si vous utilisez l'API Amazon RDS, utilisez le paramètre `CharacterSetName` avec l'opération `CreateDBInstance`. Pour plus d'informations, consultez la section Créer DBInstance.

Classement au niveau de la base de données pour Microsoft SQL Server

Vous pouvez modifier la classement par défaut au niveau base de données, table ou colonne, en remplaçant le classement de la création d'une nouvelle base de données ou d'un objet de base de données. Par exemple, si le classement de votre serveur par défaut est SQL_Latin1_General_ _CI_AS, vous pouvez le remplacer par CP1 Mohawk_100_CI_AS pour la prise en charge du classement Mohawk. Même les arguments d'une requête peuvent être l'objet d'un cast de type afin d'utiliser un classement différent si nécessaire.

Par exemple, la requête suivante changerait le classement par défaut de la AccountName colonne en Mohawk_100_CI_AS

```
CREATE TABLE [dbo].[Account]
 (
     [AccountID] [nvarchar](10) NOT NULL,
     [AccountName] [nvarchar](100) COLLATE Mohawk_100_CI_AS NOT NULL
 ) ON [PRIMARY];
```

Le moteur de base de données Microsoft SQL Server prend en charge Unicode à l'aide des types de données intégrés NCHAR, NVARCHAR et NTEXT. Par exemple, si vous avez besoin du support CJC, utilisez ces types de données Unicode pour le stockage des caractères et remplacer le classement de serveur par défaut lors de la création de vos bases de données et tables. Voici plusieurs liens depuis Microsoft couvrant le classement et le support Unicode pour SQL Server :

- Working with Collations (Utilisation des collectes)
- Collation and International Terminology (Collecte et terminologie internationale)
- Using SQL Server Collations (Utilisation de collectes SQL Server)
- International Considerations for Databases and Database Engine Applications (Considérations internationales pour les bases de données et les applications de moteur de base de données)

Création d'un utilisateur de base de données pour Amazon RDS for SQL Server

Vous pouvez créer un utilisateur de base de données pour votre instance de base de données Amazon RDS for Microsoft SQL Server en exécutant un script T-SQL, comme dans l'exemple suivant. Utilisez une application telle que SQL Server Management Suite (SSMS). Connectez-vous à l'instance de base de données en tant que l'utilisateur principal créé lors de la création de l'instance de base de données.

```
--Initially set context to master database
USE [master];
GO
--Create a server-level login named theirname with password theirpassword
CREATE LOGIN [theirname] WITH PASSWORD = 'theirpassword';
GO
--Set context to msdb database
USE [msdb];
GO
--Create a database user named theirname and link it to server-level login theirname
CREATE USER [theirname] FOR LOGIN [theirname];
GO
```

Pour consulter un exemple d'ajout d'utilisateur de base de données à un rôle, consultez Ajouter un utilisateur au SQLAgentUser rôle.

> **Note**
>
> Si vous obtenez des erreurs d'autorisation lors de l'ajout d'un utilisateur, vous pouvez restaurer les privilèges en modifiant le mot de passe de l'utilisateur principal de l'instance de base de données. Pour de plus amples informations, veuillez consulter Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL. Il n'est pas recommandé de cloner les autorisations des utilisateurs principaux dans vos applications. Pour plus d'informations, consultez Comment cloner les autorisations des utilisateurs principaux dans Amazon RDS for SQL Server.

Déterminer un modèle de restauration pour votre base de données Amazon RDS for SQL Server

Dans AmazonRDS, le modèle de restauration, la période de rétention et l'état de la base de données sont liés.

Il est important de comprendre les conséquences avant de modifier l'un de ces paramètres. Chaque paramètre peut affecter les autres. Par exemple :

- Si vous modifiez le modèle de restauration d'une base de données en SIMPLE ou BULK _ LOGGED alors que la conservation des sauvegardes est activée, Amazon RDS réinitialise le modèle de restauration FULL dans les cinq minutes. Cela entraîne également la RDS prise d'un instantané de l'instance de base de données.
- Si vous définissez la durée de conservation des sauvegardes sur plusieurs `0` jours, RDS définit le mode de restauration surSIMPLE.
- Si vous remplacez le modèle de restauration d'une base de SIMPLE données par une autre option alors que la durée de conservation des sauvegardes est définie sur `0` plusieurs jours, RDS réinitialise le modèle de SIMPLE restauration sur.

> **Important**
>
> Ne modifiez jamais le modèle de restauration sur les instances multi-AZ, même s'il semble possible de le faire, par exemple en utilisant. ALTER DATABASE La conservation des sauvegardes, et donc le mode FULL restauration, sont nécessaires pour le Multi-AZ. Si vous modifiez le modèle de restauration, revenez-le RDS immédiatement àFULL.
> Cette réinitialisation automatique oblige RDS à reconstruire complètement le rétroviseur. Au cours de cette reconstruction, la disponibilité de la base de données est dégradée pendant environ 30 à 90 minutes jusqu'à ce que le miroir soit prêt pour le basculement. L'instance de base de données connaît également une dégradation de performances de la même façon que lors d'une conversion du mode mono-AZ au mode multi-AZ. La durée pendant laquelle les performances sont dégradées dépend de la taille de stockage de base de données (plus la base de données stockée est grande, plus la dégradation est longue).

Pour plus d'informations sur les modèles de restauration SQL du serveur, consultez la section Modèles de restauration (SQLserveur) dans la documentation Microsoft.

Déterminer l'heure du dernier basculement d'Amazon RDS for Server SQL

Pour déterminer l'heure du dernier basculement, utilisez la procédure stockée suivante :

```
execute msdb.dbo.rds_failover_time;
```

Cette procédure renvoie les informations suivantes.

Paramètre de sortie	Description
errorlog_available_from	Affiche l'heure à partir de laquelle les journaux d'erreurs sont disponibles dans le répertoire des journaux.
recent_failover_time	Affiche l'heure du dernier basculement si elle est disponible à partir des journaux d'erreurs. Sinon, la valeur affichée est `null`.

> **Note**
>
> La procédure stockée recherche tous les journaux d'erreurs SQL du serveur disponibles dans le répertoire des journaux pour récupérer l'heure de basculement la plus récente. Si les messages de basculement ont été remplacés par le SQL serveur, la procédure ne permet pas de récupérer l'heure de basculement.

Example d'Aucun basculement récent

Cet exemple illustre la sortie lorsqu'il n'y a pas de basculement récent dans les journaux d'erreurs. Aucun basculement ne s'est produit depuis le 29/04/2020 à 23:59:00.01.

errorlog_available_from	recent_failover_time
2020-04-29 23:59:00.0100000	null

Example de Basculement récent

Cet exemple illustre la sortie lorsqu'un basculement récent est détecté dans les journaux d'erreurs. Le basculement le plus récent a eu lieu le 05/05/2020 à 18:57:51.89.

errorlog_available_from	recent_failover_time
2020-04-29 23:59:00.0100000	2020-05-05 18:57:51.8900000

Résolution point-in-time-recovery des défaillances dues à un écart dans le numéro de séquence du journal

Lors d'une tentative point-in-time-recovery (PITR) dans RDS pour SQL Server, vous risquez de rencontrer des échecs en raison de lacunes dans les numéros de séquence des journaux ()LSNs. Ces lacunes empêchent RDS de restaurer votre base de données à l'heure demandée et RDS met en `incompatible-restore` état votre instance de restauration.

Les causes courantes de ce problème sont les suivantes :

- Modifications manuelles du modèle de restauration de base de données.
- Le modèle de restauration automatique est modifié par RDS en raison de ressources insuffisantes pour effectuer les sauvegardes du journal des transactions.

Pour identifier les lacunes LSN dans votre base de données, exécutez cette requête :

```
SELECT * FROM msdb.dbo.rds_fn_list_tlog_backup_metadata(database_name)
ORDER BY backup_file_time_utc desc;
```

Si vous découvrez une lacune dans le LSN, vous pouvez :

- Choisissez un point de restauration avant l'écart LSN.
- Attendez et restaurez jusqu'à un point une fois la sauvegarde de l'instance suivante terminée.

Pour éviter ce problème, nous vous recommandons de ne pas modifier manuellement le modèle de restauration de vos bases de données RDS for SQL Server, car cela interrompt la durabilité de

l'instance. Nous vous recommandons également de choisir un type d'instance doté de ressources suffisantes pour votre charge de travail afin de garantir des sauvegardes régulières du journal des transactions.

Pour plus d'informations sur la gestion du journal des transactions, consultez le guide d'architecture et de gestion du journal des transactions SQL Server dans la documentation Microsoft SQL Server.

Refuser ou autoriser l'affichage des noms de base de données pour Amazon RDS for SQL Server

L'utilisateur principal ne peut pas définir `DENY VIEW ANY DATABASE TO` *`LOGIN`* pour masquer les bases de données à un utilisateur.
Pour modifier cette autorisation, utilisez plutôt la procédure stockée suivante :

- Refuser l'accès à la vue de la base *`LOGIN`*:

```
EXEC msdb.dbo.rds_manage_view_db_permission @permission='DENY',
 @server_principal='LOGIN'

go
```

- Autoriser l'accès à la vue de base de données *`LOGIN`*:

```
EXEC msdb.dbo.rds_manage_view_db_permission @permission='GRANT',
 @server_principal='LOGIN'

go
```

Tenez compte des points suivants lorsque vous utilisez cette procédure stockée :

- Les noms des bases de données sont masqués en interne SSMS et en interne DMV (vues de gestion dynamique). Cependant, les noms des bases de données sont toujours visibles dans les tables d'audit, les journaux et les tables de métadonnées. Il s'agit d'autorisations `VIEW ANY DATABASE` de serveur sécurisées. Pour plus d'informations, voir DENYAutorisations du serveur.
- Une fois l'autorisation rétablie `GRANT` (autorisée), *`LOGIN`* peut afficher toutes les bases de données.
- Si vous supprimez et recréez *`LOGIN`*, l'autorisation d'affichage associée au LOGIN est réinitialisée à`ALLOW`.

- Pour les instances Multi-AZ, définissez l'GRANTautorisation DENY ou uniquement pour *LOGIN* sur l'hôte principal. Les modifications sont automatiquement répercutées sur l'hôte secondaire.
- Cette autorisation change uniquement si un identifiant peut afficher les noms de base de données. Toutefois, l'accès aux bases de données et aux objets qu'elles contiennent est géré séparément.

Désactivation des insertions rapides lors du chargement en masse pour Amazon RDS for Server SQL

À partir de SQL Server 2016, les insertions rapides sont activées par défaut. Les insertions rapides tirent parti d'une journalisation minimale quand la base de données est en mode de récupération simple ou utilisant les journaux de transactions pour optimiser les performances d'insertion. Avec les insertions rapides, chaque lot de chargement en masse acquiert de nouvelles extensions en ignorant la recherche d'allocation des extensions existantes avec l'espace libre disponible pour optimiser les performances d'insertion.

Avec les insertions rapides, les chargements en masse de lots de petite taille peuvent aboutir à une plus grande quantité d'espace inutilisé consommée par les objets. S'il n'est pas possible d'augmenter la taille de lot, l'activation de cet indicateur de trace 692 peut contribuer à réduire l'espace réservé inutilisé, mais au détriment des performances. L'activation de cet indicateur de trace désactive les insertions rapides lors du chargement en masse de données dans un segment de mémoire ou un index cluster.

Vous activez l'indicateur de trace 692 en tant que paramètre de démarrage à l'aide de groupes de paramètres de base de données. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).

L'indicateur de trace 692 est pris en charge pour Amazon RDS on SQL Server 2016 et versions ultérieures. Pour plus d'informations sur les indicateurs de trace, voir DBCCTRACEON- indicateurs de trace dans la documentation Microsoft.

Suppression d'une base de données dans une instance de base de données Amazon RDS pour Microsoft SQL Server

Vous pouvez supprimer la base de données d'une instance de base de données Amazon RDS exécutant Microsoft SQL Server dans un déploiement Multi-AZ ou à zone de disponibilité unique. Pour supprimer la base de données, utilisez la commande suivante :

```
--replace your-database-name with the name of the database you want to drop
```

```
EXECUTE msdb.dbo.rds_drop_database  N'your-database-name'
```

> **Note**
>
> Utilisez des apostrophes droites dans la commande. Tout autre type de guillemet entraînera une erreur.

Après cette procédure de suppression de la base de données, Amazon RDS supprime toutes les connexions existantes à cette dernière, ainsi que son historique de sauvegarde.

Pour accorder une autorisation de sauvegarde et de restauration à d'autres utilisateurs, procédez comme suit :

```
USE master
GO
CREATE LOGIN user1 WITH PASSWORD=N'changeThis', DEFAULT_DATABASE=master,
 CHECK_EXPIRATION=OFF, CHECK_POLICY=OFF
GO
USE msdb
GO
CREATE USER user1 FOR LOGIN user1
GO
use msdb
GO
GRANT EXECUTE ON msdb.dbo.rds_backup_database TO user1
GO
GRANT EXECUTE ON msdb.dbo.rds_restore_database TO user1
GO
```

Modification du nom d'une base de données Amazon RDS for Microsoft SQL Server dans le cadre d'un déploiement multi-AZ

Pour renommer une instance de base de données Microsoft SQL Server qui utilise un déploiement Multi-AZ, procédez comme suit :

1. Commencez par désactiver Multi-AZ pour l'instance de base de données.
2. Renommez la base de données en exécutant `rdsadmin.dbo.rds_modify_db_name`.
3. Ensuite, activez la mise en miroir multi-AZ ou l'option Groupes de disponibilité AlwaysOn pour l'instance de base de données, afin de rétablir son état d'origine.

Pour plus d'informations, consultez Ajout de Multi-AZ à une instance de base de données Microsoft SQL Server.

> **Note**
>
> Si votre instance n'utilise pas le mode multi-AZ, vous n'avez pas besoin de modifier d'autres paramètres avant ou après l'exécution de `rdsadmin.dbo.rds_modify_db_name`.
> Vous ne pouvez pas renommer une base de données sur une instance source de réplication en lecture.

Exemple : dans l'exemple suivant, la procédure stockée `rdsadmin.dbo.rds_modify_db_name` change le nom d'une base de données de **`MOO`** à **`ZAR`**. Cela revient à exécuter l'instruction DDL `ALTER DATABASE [MOO] MODIFY NAME = [ZAR]`.

```
EXEC rdsadmin.dbo.rds_modify_db_name N'MOO', N'ZAR'
GO
```

Réinitialisation de l'appartenance au rôle db_owner pour l'utilisateur principal d'Amazon for Server RDS SQL

Si vous empêchez votre utilisateur principal d'appartenir au `db_owner` rôle dans votre base de données RDS for SQL Server et qu'aucun autre utilisateur de base de données ne peut accorder cette adhésion, vous pouvez rétablir l'adhésion perdue en modifiant le mot de passe de l'utilisateur principal de l'instance de base de données.

En modifiant le mot de passe utilisateur principal de l'instance de base de données, vous RDS `db_owner` accordez l'adhésion aux bases de données de l'instance de base de données susceptibles d'avoir été révoquées accidentellement. Vous pouvez modifier le mot de passe de l'instance de base de données à l'aide de la RDS console Amazon modify-db-instance, de la AWS CLI commande ou de l'odifyDBInstanceAPIopération M. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Restauration d'instances de base de données dont la licence est terminée pour Amazon for Server RDS SQL

Microsoft a demandé à certains RDS clients Amazon qui n'ont pas communiqué leurs informations Microsoft License Mobility de résilier leur instance de base de données. Amazon RDS prend des instantanés de ces instances de base de données, et vous pouvez les restaurer à partir de cet instantané vers une nouvelle instance de base de données dotée du modèle License Included.

Vous pouvez effectuer la restauration à partir d'un instantané de Standard Edition vers Standard Edition ou Enterprise Edition.

Vous pouvez effectuer la restauration à partir d'un instantané d'Enterprise Edition vers Standard Edition ou Enterprise Edition.

Pour effectuer une restauration à partir d'un instantané SQL du serveur après RDS qu'Amazon a créé un instantané final de votre instance

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez le snapshot de votre instance de base de données de SQL serveur. Amazon RDS crée un instantané final de votre instance de base de données. Le nom de l'instantané de l'instance terminée est au format *`instance_name`*`-final-snapshot`. Par exemple, si le nom de votre instance de base de données est**`mytest.cdxgahslksma.us-east-1.rds.com`**, le cliché final est appelé **`mytest-final-snapshot`** et se trouve dans la même AWS région que l'instance de base de données d'origine.
4. Pour Actions, choisissez Restore Snapshot (Restaurer l'instantané).

   La fenêtre Restore DB Instance (Restituer l'instance de base de données) s'affiche.
5. Pour Modèle de licence, sélectionnez licence incluse.
6. Choisissez le moteur de base de données du SQL serveur que vous souhaitez utiliser.
7. Pour DB Instance Identifier (Identifiant d'instance DB), entrez le nom de l'instance de base de données restaurée.
8. Choisissez Restore DB Instance.

Pour plus d'informations sur la restauration à partir d'un instantané, consultez Restauration vers une instance de base de données.

Transition d'une base de données Amazon RDS for SQL Server de OFFLINE vers ONLINE

Vous pouvez faire passer votre base de données Microsoft SQL Server sur une RDS instance de base de données Amazon de `OFFLINE` à`ONLINE`.

SQLMéthode du serveur	RDSMéthode Amazon
ALTER DATABASE *db_name* SET ONLINE;	EXECrdsadmin.dbo.rds_set_database_online *db_name*

Utilisation de la capture des données de modification pour Amazon RDS for SQL Server

Amazon RDS prend en charge la capture de données modifiées (CDC) pour vos instances de base de données s'exécutant sur Microsoft SQL Server. CDC capture les modifications apportées aux données de vos tables. CDC stocke les métadonnées relatives à chaque modification et vous pouvez y accéder ultérieurement. Pour plus d'informations sur le fonctionnement de CDC, consultez Capture de données modifiées dans la documentation Microsoft. Pour utiliser la fonction CDC avec vos instances de base de données Amazon RDS, vous devez tout d'abord l'activer au niveau de la base de données en exécutant `msdb.dbo.rds_cdc_enable_db`. Une fois la fonction CDC activée, tout utilisateur `db_owner` de cette base de données peut activer ou désactiver CDC sur les tables de cette base de données.

> **Important**
>
> Pendant les restaurations, la fonction CDC est désactivée. L'ensemble des métadonnées associées est automatiquement supprimé de la base de données. Cela s'applique aux restaurations de snapshots, aux point-in-time restaurations et aux restaurations natives de SQL Server à partir de S3. Après avoir exécuté l'un de ces types de restaurations, vous pouvez réactiver CDC et respécifier les tables à suivre.

Pour activer CDC pour une instance de base de données, exécutez la procédure stockée `msdb.dbo.rds_cdc_enable_db`.

```
exec msdb.dbo.rds_cdc_enable_db 'database_name'
```

Pour désactiver le CDC pour une instance de base de données, exécutez la procédure stockée `msdb.dbo.rds_cdc_disable_db`.

```
exec msdb.dbo.rds_cdc_disable_db 'database_name'
```

Pour accorder des autorisations CDC à un utilisateur, procédez comme suit :

```
go
  GRANT EXECUTE ON msdb.dbo.rds_cdc_enable_db TO User1
  GRANT EXECUTE ON msdb.dbo.rds_cdc_disable_db TO User1
```

Rubriques

- Suivi des tables avec CDC
- Tâches CDC
- Capture de données modifiées (CDC) pour les instances multi-AZ

Suivi des tables avec CDC

Une fois que CDC est activé sur la base de données, vous pouvez démarrer le suivi de tables spécifiques. Vous pouvez choisir les tableaux à suivre en exécutant sys.sp_cdc_enable_table.

```
--Begin tracking a table
exec sys.sp_cdc_enable_table
   @source_schema           = N'source_schema'
,  @source_name             = N'source_name'
,  @role_name               = N'role_name'

--The following parameters are optional:

--, @capture_instance       = 'capture_instance'
--, @supports_net_changes   = supports_net_changes
--, @index_name             = 'index_name'
--, @captured_column_list   = 'captured_column_list'
```

```
--, @filegroup_name         = 'filegroup_name'
--, @allow_partition_switch = 'allow_partition_switch'
;
```

Pour afficher la configuration CDC de vos tableaux, exécutez sys.sp_cdc_help_change_data_capture.

```
--View CDC configuration
exec sys.sp_cdc_help_change_data_capture

--The following parameters are optional and must be used together.
--  'schema_name', 'table_name'
;
```

Pour plus d'informations sur les tables, fonctions et procédures stockées CDC dans la documentation SQL Server, consultez les rubriques suivantes :

- Procédures stockées CDC (Transact-SQL)
- Fonctions CDC (Transact-SQL)
- Tableaux CDC (Transact-SQL)

Tâches CDC

Quand vous activez CDC, SQL Server crée les tâches CDC. Les propriétaires de base de données (`db_owner`) peuvent afficher, créer, modifier et supprimer les tâches CDC. Cependant, le compte système RDS en est propriétaire. Par conséquent, les tâches ne sont pas visibles des vues natives, des procédures ou de SQL Server Management Studio.

Pour contrôler le comportement de CDC dans une base de données, utilisez les procédures SQL Server natives telles que sp_cdc_enable_table et sp_cdc_start_job . Pour modifier les paramètres des tâches CDC, comme `maxtrans` et `maxscans`, vous pouvez utiliser sp_cdc_change_job..

Pour obtenir plus d'informations sur les tâches CDC, vous pouvez interroger les vues de gestion dynamiques suivantes :

- sys.dm_cdc_errors
- sys.dm_cdc_log_scan_sessions
- sysjobs
- sysjobhistory

Capture de données modifiées (CDC) pour les instances multi-AZ

Si vous utilisez CDC sur une instance multi-AZ, assurez-vous que la configuration de la tâche CDC du miroir correspond à celle du mandataire. Les tâches CDC sont mappées au `database_id`. Si la IDs base de données secondaire est différente de la base de données principale, les tâches ne seront pas associées à la base de données appropriée. Pour éviter toute erreur après le basculement, RDS supprimer et recrée les tâches sur le nouveau mandataire. Les tâches recréées utilisent les paramètres que le mandataire a enregistrés avant le basculement.

Même si ce processus s'exécute rapidement, il est toujours possible que les tâches CDC puissent s'exécuter avant que RDS puisse les corriger. Voici trois moyens de contraindre les paramètres à être cohérents entre les réplicas principaux et secondaires :

- Utilisez les mêmes paramètres de tâche pour toutes les bases de données pour lesquelles CDC est activé.
- Avant de modifier la configuration des tâches CDC, convertissez l'instance multi-AZ en mono-AZ.
- Transférez manuellement les paramètres chaque fois que vous les modifiez sur le principal.

Pour afficher et définir les paramètres CDC utilisés pour recréer les tâches CDC après un basculement, utilisez `rds_show_configuration` et `rds_set_configuration`.

L'exemple suivant renvoie la valeur définie pour `cdc_capture_maxtrans`. Pour tout paramètre défini sur `RDS_DEFAULT`, RDS configure automatiquement la valeur.

```
-- Show configuration for each parameter on either primary and secondary replicas.
exec rdsadmin.dbo.rds_show_configuration 'cdc_capture_maxtrans';
```

Pour définir la configuration sur le réplica secondaire, exécutez `rdsadmin.dbo.rds_set_configuration`. Cette procédure définit les valeurs de paramètre pour toutes les bases de données du serveur secondaire. Ces paramètres ne sont utilisés qu'après un basculement. L'exemple suivant définit `maxtrans` pour toutes les tâches de capture CDC la valeur suivante *`1000`* :

```
--To set values on secondary. These are used after failover.
exec rdsadmin.dbo.rds_set_configuration 'cdc_capture_maxtrans', 1000;
```

Pour définir les paramètres de tâche CDC sur le principal, utilisez plutôt sys.sp_cdc_change_job.

Utilisation de l'agent SQL Server pour Amazon RDS

Avec Amazon RDS, vous pouvez utiliser SQL Server Agent sur une instance de base de données exécutant Microsoft SQL Server Enterprise Edition, Standard Edition ou Web Edition. SQL Server Agent est un service Microsoft Windows qui exécute des tâches administratives planifiées, appelées travaux. Vous pouvez utiliser SQL Server Agent pour exécuter les travaux T-SQL jobs afin de reconstruire les index, d'exécuter les contrôles de corruption et de regrouper les données dans une instance de base de données SQL Server.

Lorsque vous créez une instance de base de données SQL Server, l'utilisateur principal est inscrit dans le rôle `SQLAgentUserRole`.

SQL Server Agent peut exécuter un travail en fonction d'une planification, en réponse à un événement spécifique, ou à la demande. Pour plus d'informations, consultez SQL Server Agent dans la documentation Microsoft.

> **Note**
>
> Évitez de planifier l'exécution de travaux pendant les fenêtres de maintenance et de sauvegarde de votre instance de base de données. Les processus de maintenance et de sauvegarde lancés par AWS peuvent interrompre une tâche ou entraîner son annulation. Dans les déploiements Multi-AZ, les tâches de l'agent SQL Server sont répliquées de l'hôte principal vers l'hôte secondaire lorsque la fonction de réplication des tâches est activée. Pour de plus amples informations, veuillez consulter Activation de la réplication des tâches de l'agent SQL Server.
>
> Les déploiements multi-AZ sont limités à 10 000 tâches SQL Server Agent. Si vous avez besoin d'une limite plus élevée, demandez une augmentation en contactant Support. Ouvrez la page du Centre AWS Support, connectez-vous si nécessaire, puis choisissez Create case (Créer une demande de support). Sélectionnez Service Limit increase (Augmentation des limites de service). Remplissez et envoyez le formulaire.

Pour afficher l'historique d'un travail SQL Server Agent dans SQL Server Management Studio (SSMS), ouvrez l'Explorateur d'objet, cliquez avec le bouton droit sur le travail, puis cliquez sur View History (Afficher l'historique).

Étant donné que SQL Server Agent est exécuté sur un hôte géré dans une instance de base de données, certaines actions ne sont pas prises en charge :

- L'exécution de tâches de réplication et l'exécution de scripts de ligne de commande à l'aide d'ActiveX, de l'interface de commande Windows ou de Windows PowerShell ne sont pas prises en charge.
- Vous ne pouvez pas démarrer, arrêter ou redémarrer manuellement SQL Server Agent.
- Les notifications par e-mail via SQL Server Agent ne sont pas disponibles à partir d'une instance de base de données.
- Les alertes et les opérateurs SQL Server Agent ne sont pas pris en charge.
- L'utilisation de SQL Server Agent pour créer des sauvegardes n'est pas prise en charge. Utilisez Amazon RDS for sauvegarder votre instance de base de données.
- Actuellement, RDS pour SQL Server ne prend pas en charge l'utilisation de jetons SQL Server Agent.

Activation de la réplication des tâches de l'agent SQL Server

Vous pouvez activer la réplication des tâches de l'agent SQL Server à l'aide de la procédure stockée suivante :

```
EXECUTE msdb.dbo.rds_set_system_database_sync_objects @object_types = 'SQLAgentJob';
```

Vous pouvez exécuter la procédure stockée sur toutes les versions de SQL Server prises en charge par Amazon RDS for SQL Server. Les tâches des catégories suivantes sont répliquées :

- [Uncategorized (Local)] ([Non classé (local)])
- [Uncategorized (Multi-Server)] ([Non classé (multi-serveurs)])
- [Uncategorized] ([Non classé])
- Data Collector (Collecteur de données)
- Database Engine Tuning Advisor (Assistant Paramétrage du moteur de base de données)
- Database Maintenance (Maintenance de base de données)
- Full-Text (Texte intégral)

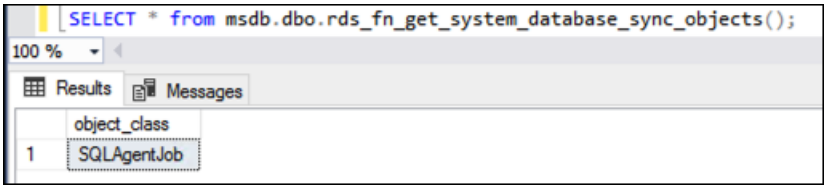
Seules les tâches qui utilisent des étapes de travail T-SQL sont répliquées. Les tâches comportant des types d'étapes tels que SQL Server Integration Services (SSIS), SQL Server Reporting Services (SSRS), Replication, PowerShell ne sont pas répliquées. Les tâches qui utilisent Database Mail (Messagerie de base de données) et les objets au niveau du serveur ne sont pas répliquées.

> **Important**
>
> L'hôte principal est la source de vérité pour la réplication. Avant d'activer la réplication des tâches, assurez-vous que vos tâches SQL Server Agent se trouvent sur l'hôte principal. Si vous ne le faites pas, cela pourrait entraîner la suppression de vos tâches SQL Server Agent si vous activez la fonctionnalité lorsque des tâches plus récentes se trouvent sur l'hôte secondaire.

Vous pouvez utiliser la fonction suivante pour confirmer si la réplication est activée.

```
SELECT * from msdb.dbo.rds_fn_get_system_database_sync_objects();
```

La requête T-SQL renvoie le résultat suivant si les tâches de l'agent SQL Server sont répliquées. Si elles ne se répliquent pas, la requête ne renvoie rien pour `object_class`.

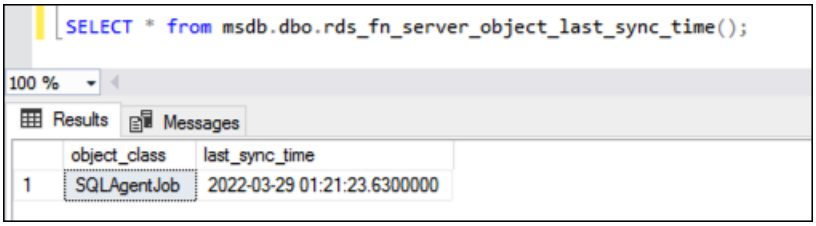


Vous pouvez utiliser la fonction suivante pour trouver la dernière synchronisation des objets selon le fuseau UTC.

```
SELECT * from msdb.dbo.rds_fn_server_object_last_sync_time();
```

Par exemple, supposons que vous modifiez une tâche de l'agent du serveur SQL à 01:00. Vous vous attendez à ce que l'heure de synchronisation la plus récente soit postérieure à 01:00, indiquant que la synchronisation a eu lieu.

Après la synchronisation, les valeurs renvoyées pour `date_created` et `date_modified` sur le nœud secondaire doivent correspondre.



Si vous utilisez également la `tempdb` réplication, vous pouvez activer la réplication à la fois pour les tâches SQL Agent et pour la `tempdb` configuration en les fournissant dans le `@object_type` paramètre :

```
EXECUTE msdb.dbo.rds_set_system_database_sync_objects @object_types =
 'SQLAgentJob,TempDbFile';
```

Pour plus d'informations sur `tempdb` la réplication, consultezConfiguration de TempDB pour les déploiements multi-AZ.

Rôles de l'agent SQL Server

RDS pour SQL Server prend en charge les rôles d'agent SQL Server suivants avec différents niveaux d'autorisation pour la gestion des tâches :

- SQLAgentUserRole

  Autorisations

  - Créez et gérez leurs propres tâches, horaires et opérateurs
  - Afficher les propriétés de leurs propres tâches et horaires
  - Impossible d'afficher ou de gérer les tâches créées par d'autres utilisateurs

  Ce rôle convient aux utilisateurs qui doivent créer et gérer leurs propres tâches, mais qui n'ont pas besoin d'accéder aux tâches créées par d'autres utilisateurs.
- SQLAgentReaderRole

  Autorisations

  - Toutes les autorisations de SQLAgent UserRole
  - Afficher la liste de toutes les tâches et de tous les plannings, y compris ceux créés par d'autres
  - Afficher les propriétés de tous les emplois
  - Consulter l'historique des emplois

  Ce rôle convient aux utilisateurs qui ont besoin de surveiller l'état de toutes les tâches mais qui n'ont pas besoin de les gérer.
- SQLAgentOperatorRole

  Autorisations

  - Toutes les autorisations de SQLAgent UserRole et SQLAgent ReaderRole

- Exécuter, arrêter ou démarrer des tâches
- Gérer l'historique des emplois
- Activer/désactiver les tâches et les plannings
- Afficher les opérateurs et les proxys

Ce rôle fournit les autorisations les plus complètes et convient aux utilisateurs qui ont besoin d'un contrôle total sur toutes les tâches.

Utilisez la commande suivante pour attribuer les rôles à votre identifiant SQL Server :

```
USE msdb;
EXEC sp_addrolemember 'SQLAgentOperatorRole', 'username';
```

Gestion SQLAgent OperatorRole dans RDS pour SQL Server

Pour afficher les tâches en cours, vous devez les ajouter SQLAgent OperatorRole à votre identifiant SQL Server et les supprimer avant de vous déconnecter de votre base de données.

Pour visualiser l'arborescence de l'agent SQL Server dans SQL Server Management Studio, suivez les instructions suivantes :

Afficher l'agent SQL Server sur SQL Server Management Studio (SSMS)

1. À l'aide des informations d'identification principales RDS, connectez-vous à l'instance RDS SQL Server et accordez à l'utilisateur souhaité le. SQLAgent UserRole

   ```
   USE msdb
   GO
   IF NOT EXISTS(SELECT name FROM sys.database_principals WHERE name = 'UserName')
   BEGIN
   CREATE USER UserName FROM LOGIN UserName
   END
   GO
   ALTER ROLE SQLAgentUserRole ADD MEMBER UserName
   GO
   GRANT ALTER ON ROLE::[SQLAgentOperatorRole] to UserName
   GO
   ```

Ces commandes créent l'utilisateur dans la `msdb` base de données, au cas où elle n'existerait pas. Il ajoute également l'utilisateur sur le SQLAgentUserRole, de sorte que l'arborescence de l'agent SQL Server sur SSMS soit visible. Enfin, il accorde des autorisations de modification SQLAgent OperatorRole à l'utilisateur. Cela permet à l'utilisateur de add/remove s'émanciper de ce rôle.

2. Pour vous ajouter au rôle mentionné ci-dessus, connectez-vous à l'instance RDS SQL Server, avec l'utilisateur qui a besoin de voir les tâches, et exécutez le script suivant.

```
use msdb
go
ALTER ROLE SQLAgentOperatorRole ADD MEMBER UserName
GO
```

Ensuite, cliquez avec le bouton droit sur le dossier Jobs, puis choisissez Actualiser.

3. Lorsque vous effectuez cette action, l'onglet Tâches affiche un bouton + (plus). Cliquez pour développer la liste des tâches de l'agent SQL Server.

4. **Important**

   Avant de vous déconnecter de l'instance RDS SQL Server, vous devez vous retirer du SQLAgentOperatorRole.

   Pour supprimer votre identifiant du SQLAgentOperatorRole, exécutez la requête suivante avant de déconnecter ou de fermer le Management Studio :

```
USE msdb
GO
ALTER ROLE SQLAgentOperatorRole DROP MEMBER UserName
GO
```

Pour plus d'informations, consultez la section Exploitation SQLAgent OperatorRole dans RDS SQL Server.

Ajouter un utilisateur au SQLAgentUser rôle

Pour autoriser un identifiant ou un utilisateur supplémentaire à utiliser l'agent SQL serveur, connectez-vous en tant qu'utilisateur principal et procédez comme suit :

1. Créez une autre connexion de niveau serveur à l'aide de la commande `CREATE LOGIN`.
2. Créez un utilisateur dans `msdb` avec la commande `CREATE USER` puis liez cet utilisateur à la connexion que vous avez créée à l'étape précédente.
3. Ajoutez l'utilisateur à la procédure `SQLAgentUserRole` à l'aide de la procédure stockée système `sp_addrolemember`.

Supposons, par exemple, que votre nom d'utilisateur principal soit **`admin`** et que vous souhaitiez donner accès à l'agent SQL serveur à un utilisateur doté d'un mot de passe**`theirpassword`**. **`theirname`** Dans ce cas, vous pouvez utiliser la procédure suivante.

Pour ajouter un utilisateur au SQLAgentUser rôle

1. Connectez-vous en tant qu'utilisateur principal.
2. Exécutez les commandes suivantes :

```
--Initially set context to master database
USE [master];
GO
--Create a server-level login named theirname with password theirpassword
CREATE LOGIN [theirname] WITH PASSWORD = 'theirpassword';
GO
--Set context to msdb database
USE [msdb];
GO
--Create a database user named theirname and link it to server-level login
 theirname
CREATE USER [theirname] FOR LOGIN [theirname];
GO
--Added database user theirname in msdb to SQLAgentUserRole in msdb
EXEC sp_addrolemember [SQLAgentUserRole], [theirname];
```

Suppression d'une tâche SQL Server Agent

Vous utilisez la procédure stockée `sp_delete_job` pour supprimer les travaux de l'agent SQL Server sur Amazon RDS pour Microsoft SQL Server.

Vous ne pouvez pas utiliser SSMS pour supprimer des travaux de l'agent SQL Server Agent. Si vous le faites, vous obtenez un message d'erreur similaire au suivant :

```
The EXECUTE permission was denied on the object 'xp_regread', database
 'mssqlsystemresource', schema 'sys'.
```

Cette erreur survient parce que, en tant que service géré, RDS est empêché d'exécuter les procédures qui accèdent au registre Windows. Lorsque vous utilisez SSMS, celui-ci tente d'exécuter un processus (`xp_regread`) pour lequel RDS n'est pas autorisé.

> **Note**
>
> Sur RDS for SQL Server, seuls les membres du rôle d'administrateur système sont autorisés à mettre à jour ou à supprimer des tâches appartenant à un identifiant de connexion différent. Pour plus d'informations, consultez Tirer parti SQLAgent OperatorRole de RDS SQL Server.

Pour supprimer un travail SQL Server Agent

- Exécutez l'instruction T-SQL suivante :

```
EXEC msdb..sp_delete_job @job_name = 'job_name';
```

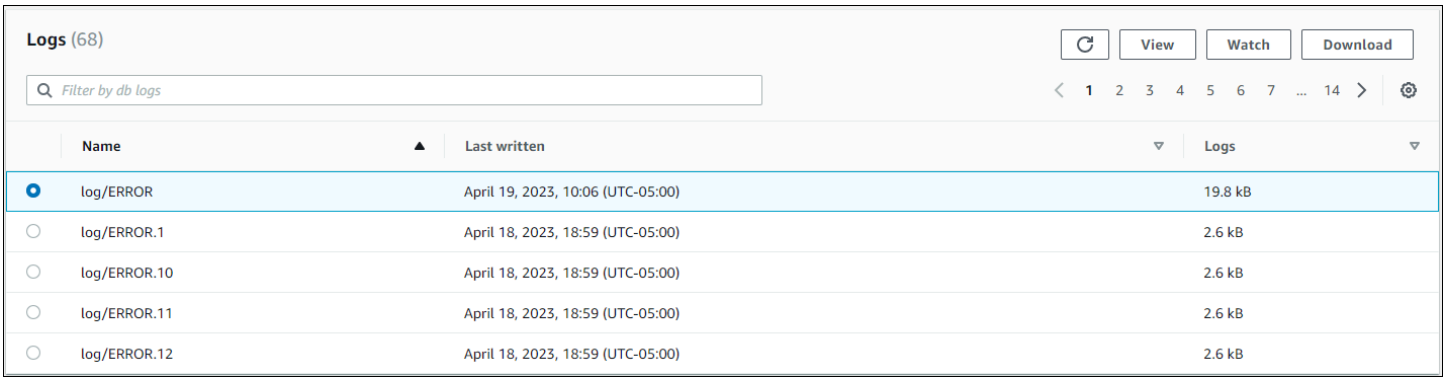
Utilisation des journaux Amazon RDS pour Microsoft SQL Server

Vous pouvez utiliser la console Amazon RDS for afficher, consulter et télécharger les journaux SQL Server Agent, les journaux d'erreurs Microsoft SQL Server et les journaux SQL Server Reporting Services (SSRS).

Consultation des fichiers journaux

Si vous affichez un journal dans la console Amazon RDS, vous pouvez voir son contenu tel qu'il est à ce moment-là. L'observation d'un journal dans la console l'ouvre dans un état dynamique de telle sorte que vous puissiez voir ses mises à jour pratiquement en temps réel.

Seul le dernier journal est actif pour pouvoir être observé. Par exemple, supposons que les journaux affichent les informations suivantes :



Seul log/ERROR, comme le journal le plus récent est en mise à jour active. Vous pouvez choisir d'en observer d'autres, mais ils sont statiques et ne sont pas mis à jour.

Archivage des fichiers journaux

La console Amazon RDS affiche les journaux de la semaine écoulée jusqu'au même. Vous pouvez télécharger et archiver les journaux pour les garder comme référence au-delà de cette date. Une solution pour archiver les journaux consiste à les charger dans un compartiment Amazon S3. Pour savoir comment configurer un compartiment Amazon S3 et comment charger un fichier, consultez Bases Amazon S3 dans le Guide de démarrage Amazon Simple Storage Service, puis cliquez sur Mise en route.

Affichage des journaux des erreurs et des agents

Pour consulter les journaux des erreurs et de l'agent Microsoft SQL Server, utilisez la procédure stockée Amazon RDS `rds_read_error_log` avec les paramètres suivants :

- **`@index`** – version du journal à récupérer. La valeur par défaut est 0, qui permet de récupérer le journal des erreurs actuel. Spécifiez 1 pour récupérer le journal précédent, spécifiez 2 pour récupérer celui d'avant, et ainsi de suite.
- **`@type`** – type de journal à récupérer. Spécifiez 1 pour récupérer un journal des erreurs. Spécifiez 2 pour récupérer un journal de l'agent.

Example

L'exemple suivant demande le journal des erreurs actuel.

```
EXEC rdsadmin.dbo.rds_read_error_log @index = 0, @type = 1;
```

Pour plus d'informations sur les erreurs SQL Server, consultez la section Erreurs du moteur de base de données dans la documentation Microsoft.

Utilisation de fichiers de trace et de vidage pour Amazon RDS for SQL Server

Cette section décrit l'utilisation des fichiers de trace et des fichiers de vidage pour vos instances Amazon RDS DB exécutant Microsoft SQL Server.

Génération d'une SQL requête de suivi

```
declare @rc int
declare @TraceID int
declare @maxfilesize bigint

set @maxfilesize = 5

exec @rc = sp_trace_create @TraceID output,  0, N'D:\rdsdbdata\log\rdstest',
 @maxfilesize, NULL
```

Affichage d'une trace ouverte

```
select * from ::fn_trace_getinfo(default)
```

Affichage du contenu d'une trace

```
select * from ::fn_trace_gettable('D:\rdsdbdata\log\rdstest.trc', default)
```

Configuration de la période de rétention pour les fichiers de trace et de vidage

Les fichiers de trace et de vidage peuvent s'accumuler et consommer de l'espace sur le disque. Par défaut, Amazon RDS purge les fichiers de trace et de vidage datant de plus de sept jours.

Pour consulter la période actuelle de rétention des fichiers de trace et de vidage, utilisez la procédure `rds_show_configuration`, comme illustré dans l'exemple suivant.

```
exec rdsadmin..rds_show_configuration;
```

Pour modifier la période de rétention des fichiers de trace, utilisez la procédure `rds_set_configuration` et définissez `tracefile retention` en minutes. L'exemple ci-dessous définit la période de rétention des fichiers de trace à 24 heures.

```
exec rdsadmin..rds_set_configuration 'tracefile retention', 1440;
```

Pour modifier la période de rétention des fichiers de vidage, utilisez la procédure `rds_set_configuration` et définissez `dumpfile retention` en minutes. L'exemple ci-dessous définit la période de rétention des fichiers de vidage à 3 jours.

```
exec rdsadmin..rds_set_configuration 'dumpfile retention', 4320;
```

Pour des raisons de sécurité, vous ne pouvez pas supprimer un fichier de trace ou de vidage spécifique sur une instance de base de données de SQL serveur. Pour supprimer tous les fichiers de trace ou de vidage inutilisés, définissez la période de rétention des fichiers à 0.

Amazon RDS pour moi SQL

Amazon RDS prend en charge plusieurs versions des instances My SQL for DB. Pour obtenir des informations complètes sur les versions prises en charge, consultezVersions de MySQL sur Amazon RDS.

Pour créer une SQL instance de base de données Amazon RDS for My, utilisez les outils ou interfaces de RDS gestion Amazon. Vous pouvez alors effectuer ce qui suit :

- Redimensionner votre instance de base de données
- Autorisation des connexions à votre instance de base de données
- Créer et restaurer à partir de sauvegardes ou d'instantanés
- Créer des secondaires Multi-AZ
- Créer des réplicas en lecture
- Surveiller les performances de votre instance de base de données

Pour stocker et accéder aux données de votre instance de base de données, vous utilisez les SQL utilitaires et applications My standard.

Amazon RDS for My SQL est conforme à de nombreuses normes du secteur. Par exemple, vous pouvez utiliser RDS for My SQL databases pour créer des applications HIPAA conformes. Vous pouvez utiliser Mes SQL bases de données RDS pour stocker des informations relatives à la santé, y compris des informations de santé protégées (PHI) dans le cadre d'un accord de partenariat commercial (BAA) avec AWS. Amazon RDS for My répond SQL également aux exigences de sécurité du Federal Risk and Authorization Management Program (FedRAMP). En outre, Amazon RDS for My SQL a reçu une autorisation provisoire du Conseil d'autorisation RAMP conjoint de la Fed (JAB) pour opérer (P-ATO) sur le site de RAMP HIGH référence de la Fed dans les AWS GovCloud (US) régions. Pour plus d'informations sur les normes de conformité prises en charge, veuillez consulter Conformité du Cloud AWS.

Pour plus d'informations sur les fonctionnalités de chaque version de MySQL, consultez la section Les principales fonctionnalités de My SQL dans la SQL documentation My.

Avant de créer une instance de base de données, effectuez les étapes de la section Configuration de votre environnement Amazon RDS. Lorsque vous créez une instance de base de données, l'utilisateur RDS principal obtient DBA des privilèges, avec certaines limites. Utilisez ce compte pour des tâches administratives telles que la création de comptes de base de données supplémentaires.

Vous pouvez créer ce qui suit :

- Instances DB
- Instantanés de base de données
- P oint-in-time restaure
- Sauvegardes automatiques
- Sauvegardes manuelles

Vous pouvez utiliser des instances de base de données exécutant My SQL dans un cloud privé virtuel (VPC) basé sur AmazonVPC. Vous pouvez également ajouter des fonctionnalités à votre instance My SQL DB en activant diverses options. Amazon RDS prend en charge les déploiements multi-AZ pour My SQL en tant que solution de basculement à haute disponibilité.

> **Important**
>
> Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui nécessitent des privilèges avancés. Vous pouvez accéder à votre base de données à l'aide de SQL clients standard tels que le client MySQL. Cependant, vous ne pouvez pas accéder directement à l'hôte à l'aide de Telnet ou de Secure Shell (SSH).

Rubriques

- Fonctionnalités MySQL prises en charge sur Amazon RDS
- Versions de MySQL sur Amazon RDS
- Connexion à votre instance de base de données MySQL
- Sécurisation des connexions d'instance de base de données MySQL
- Améliorer les performances des requêtes RDS pour My SQL grâce à Amazon RDS Optimized Reads
- Améliorer les performances d'écriture grâce à RDS Optimized Writes for My SQL
- Mises à niveau du moteur de base de données RDS pour MySQL
- Mise à niveau d'une version du moteur de snapshots de base de données MySQL
- Importation de données dans une instance de base de données Amazon RDS for MySQL
- Utilisation de la réplication MySQL dans Amazon RDS

- Configuration de clusters actifs-actifs pour for My RDS SQL
- Exportation de données depuis une instance My SQL DB à l'aide de la réplication
- Options pour les instances My SQL DB
- Paramètres pour My SQL
- DBATâches courantes pour les instances My SQL DB
- Fuseau horaire local pour les instances My SQL DB
- Limites et problèmes connus pour Amazon RDS for MySQL
- RDSpour Ma référence de procédure SQL stockée

Fonctionnalités MySQL prises en charge sur Amazon RDS

RDS for MySQL prend en charge la plupart des fonctionnalités et des capacités de MySQL. Certaines fonctions peuvent avoir une prise en charge limitée ou des privilèges restreints.

Vous pouvez filtrer les nouvelles fonctions de Amazon RDS sur la page Nouveautés en matière de base de données. Pour Produits, choisissez Amazon RDS. Ensuite, effectuez une recherche à l'aide de mots clés tels que **`MySQL 2022`**.

> **Note**
>
> Les listes suivantes ne sont pas exhaustives.

Rubriques

- Support des fonctionnalités MySQL sur les versions majeures d'Amazon RDS for MySQL
- Moteurs de stockage pris en charge pour RDS for MySQL
- Utilisation de memcached et d'autres options avec MySQL sur Amazon RDS
- Préparation du cache InnoDB pour MySQL sur Amazon RDS
- Changements linguistiques inclusifs pour RDS for MySQL 8.4
- Fonctions MySQL non prises en charge par Amazon RDS

Support des fonctionnalités MySQL sur les versions majeures d'Amazon RDS for MySQL

Dans les sections suivantes, vous trouverez des informations sur la prise en charge des fonctionnalités MySQL dans les versions majeures d'Amazon RDS for MySQL :

Rubriques

- Support de MySQL 8.4 sur Amazon RDS

Pour plus d'informations sur les versions mineures prises en charge d'Amazon RDS for MySQL, Versions de MySQL mineures prises en charge sur Amazon RDS consultez.

Support de MySQL 8.4 sur Amazon RDS

Amazon RDS prend en charge les nouvelles fonctionnalités suivantes pour vos instances de base de données exécutant MySQL version 8.4 ou supérieure.

- Bibliothèque cryptographique — RDS pour MySQL a remplacé OpenSSL par AWS Libcrypto (AWS-LC), qui est certifié FIPS 140-3. Pour plus d'informations, consultez le AWS-LC GitHub référentiel à l'adressehttps://github.com/aws/aws-lc.
- Modifications apportées au protocole TLS — RDS pour MySQL ne prend en charge que les protocoles TLS 1.2 et TLS 1.3. Pour de plus amples informations, veuillez consulter the section called “SSL/TLSsupport avec My SQL”.
- support memcached — L'interface memcached n'est plus disponible sur MySQL 8.4. Pour de plus amples informations, veuillez consulter Prise en charge memcached MySQL.
- Plugin d'authentification par défaut — Le plugin d'authentification par défaut est`caching_sha2_password`. Pour de plus amples informations, veuillez consulter the section called “Plugin d'authentification par défaut MySQL”.
- **`mysqlpump`**utilitaire client — L'utilitaire `mysqlpump` client n'est plus disponible dans MySQL 8.4. Pour plus d'informations, consultez mysqldump Modèle de privilèges basé sur les rôles pour for My RDS SQL et mysqlpump dans le guide prescriptif.AWS
- Procédures stockées de réplication gérée : lorsque vous utilisez des procédures stockées pour gérer la réplication avec un utilisateur de réplication configuré avec`caching_sha2_password`, vous devez configurer le protocole TLS en `SOURCE_SSL=1` spécifiant. `caching_sha2_password`est le plugin d'authentification par défaut pour RDS for MySQL 8.4.
- Modifications du comportement des paramètres — Les paramètres suivants ont changé pour MySQL 8.4.
  - `innodb_dedicated_server`— Ce paramètre est désormais activé par défaut. Pour de plus amples informations, veuillez consulter Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL.
  - `innodb_buffer_pool`— Le moteur de base de données calcule désormais ce paramètre, mais vous pouvez le remplacer. Pour de plus amples informations, veuillez consulter Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL.
  - `innodb_redo_log_capacity`— Ce paramètre contrôle désormais la taille des fichiers de journalisation. Le moteur de base de données calcule désormais ce paramètre, mais vous

pouvez le remplacer. Pour de plus amples informations, veuillez consulter Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL.

- Paramètres obsolètes ou supprimés — RDS pour MySQL a supprimé les paramètres suivants des groupes de paramètres pour les instances de base de données MySQL 8.4. Le `innodb_redo_log_capacity` paramètre contrôle désormais la taille des fichiers de journalisation.
  - `innodb_log_file_size`
  - `innodb_log_files_in_group`
- Nouvelles valeurs par défaut pour les paramètres — Les paramètres suivants ont de nouvelles valeurs par défaut pour les instances de base de données MySQL 8.4 :
  - Divers paramètres de la communauté MySQL liés aux performances ont changé. Pour plus d'informations, consultez Nouveautés de MySQL 8.4 depuis MySQL 8.0.

    Nous vous recommandons de tester les performances de votre application sur RDS for MySQL 8.4 avant de migrer une instance de production.
  - `innodb_purge_threads`— La valeur par défaut est définie sur la formule pour `LEAST({DBInstanceVCPU/2},4)` éviter que la longueur de la liste d'historique d'InnoDB ne devienne trop longue.
  - `group_replication_exit_state_action`— La valeur par défaut est`OFFLINE_MODE`, ce qui correspond à la valeur par défaut dans MySQL Community. Pour de plus amples informations, veuillez consulter Considérations et bonnes pratiques relatives à RDS for My SQL active-active clusters.
  - `binlog_format`— La valeur par défaut est`ROW`, ce qui correspond à la valeur par défaut dans MySQL Community. Vous pouvez modifier le paramètre pour les instances de base de données mono-AZ ou les instances de base de données multi-AZ, mais pas pour les clusters de base de données multi-AZ. Les clusters de bases de données multi-AZ utilisent la réplication semi-synchrone, et lorsqu'ils `binlog_format` sont définis sur ou, la réplication échoue. `MIXED STATEMENT`
- Changements linguistiques inclusifs — RDS pour MySQL 8.4 inclut les modifications par rapport à RDS pour MySQL 8.0 relatives aux mots clés et aux schémas système pour un langage inclusif. Pour de plus amples informations, veuillez consulter Changements linguistiques inclusifs pour RDS for MySQL 8.4.

Pour une liste de toutes les fonctionnalités et modifications apportées à MySQL 8.4, consultez la section Nouveautés de MySQL 8.4 depuis MySQL 8.0 dans la documentation MySQL.

Pour obtenir la liste des fonctions non prises en charge, consultez the section called “Fonctions non prises en charge”.

Moteurs de stockage pris en charge pour RDS for MySQL

Même si MySQL prend en charge plusieurs moteurs de stockage aux capacités diverses, ils ne sont pas tous optimisés pour la récupération et la durabilité des données. Amazon RDS prend entièrement en charge le moteur de stockage InnoDB pour les instances de base de données MySQL. Les fonctionnalités Amazon RDS telles que la Point-In-Time restauration et la restauration de snapshots nécessitent un moteur de stockage récupérable et sont prises en charge uniquement par le moteur de stockage InnoDB. Pour de plus amples informations, veuillez consulter Prise en charge memcached MySQL.

Le Federated Storage Engine n'est pour l'instant pas pris en charge par Amazon RDS for MySQL.

Pour les schémas créés par l'utilisateur, le moteur de stockage MyISAM ne permet pas une restauration fiable et peut entraîner la perte ou la corruption de données lorsque MySQL est redémarré après une restauration, empêchant ainsi la Point-In-Time restauration ou la restauration par capture instantanée de fonctionner comme prévu. Néanmoins, si vous choisissez tout de même d'utiliser MyISAM avec Amazon RDS, les instantanés peuvent être utiles dans certaines conditions.

> **Note**
>
> Les tables système du schéma `mysql` peuvent être dans le stockage MyISAM.

Si vous souhaitez convertir des tables MyISAM existantes en tables InnoDB, vous pouvez utiliser la commande `ALTER TABLE` (par exemple, `alter table TABLE_NAME engine=innodb;`). N'oubliez pas que MyISAM et InnoDB ont des forces et des faiblesses différentes, vous devriez donc commencer par évaluer de façon exhaustive l'impact de ce basculement sur vos applications.

Les versions MySQL 5.1, 5.5 et 5.6 ne sont plus prises en charge dans Amazon RDS. Cependant, vous pouvez restaurer des instantanés MySQL 5.1, 5.5 et 5.6 existants. Lorsque vous restaurez un instantané MySQL 5.1, 5.5 ou 5.6, l'instance de base de données est automatiquement mise à niveau vers MySQL 5.7.

Utilisation de memcached et d'autres options avec MySQL sur Amazon RDS

La plupart des moteurs de base de données Amazon RDS prennent en charge des groupes d'options qui vous permettent de sélectionner des fonctions supplémentaires pour votre instance de base de données. Les instances de bases de données RDS for MySQL prennent en charge l'option `memcached`, un cache simple basée sur les clés. Pour plus d'informations sur `memcached` et d'autres options, consultez Options pour les instances My SQL DB. Pour plus d'informations sur l'utilisation de groupes d'options, consultez Utilisation de groupes d'options.

Préparation du cache InnoDB pour MySQL sur Amazon RDS

La préparation du cache InnoDB peut fournir des gains de performances pour votre instance de base de données MySQL en enregistrant l'état actuel du pool de mémoires tampons lorsque l'instance de base de données est arrêtée, puis en rechargeant le pool de mémoires tampons à partir des informations enregistrées au démarrage de l'instance de base de données. Cette approche contourne la nécessité de « préparer » le pool de mémoires tampons à partir d'une utilisation normale de la base de données et précharge à la place le pool de mémoires tampons avec les pages des requêtes courantes connues. Le fichier qui stocke les informations du pool de tampons enregistré stocke uniquement les métadonnées pour les pages qui sont dans le pool de mémoires tampons et pas les pages elles-mêmes. Par conséquent, le fichier ne nécessite pas un important espace de stockage. La taille du fichier représente environ 0,2 pour cent de la taille du cache. Par exemple, pour un cache 64 Gio, la taille du fichier de préparation de cache est de 128 Mio. Pour de plus amples informations sur la préparation du cache InnoDB, veuillez consulter Saving and Restoring the Buffer Pool State dans la documentation MySQL.

Les instances de bases de données RDS for MySQL prennent en charge la préparation du cache InnoDB. Pour activer la préparation du cache InnoDB, définissez les paramètres `innodb_buffer_pool_dump_at_shutdown` et `innodb_buffer_pool_load_at_startup` avec la valeur 1 dans le groupe de paramètres de votre instance de base de données. La modification de ces valeurs dans un groupe de paramètres affecte toutes les instances de bases de données MySQL qui utilisent ce groupe de paramètres. Pour activer la préparation du cache InnoDB pour des instances de bases de données MySQL spécifiques, vous devrez peut-être créer un groupe de paramètres pour ces instances. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

La préparation du cache InnoDB fournit principalement une amélioration des performances pour les instances de bases de données qui utilisent le stockage standard. Si vous utilisez le stockage PIOPS, vous ne constatez généralement pas d'amélioration significative des performances.

Important

Si votre instance de base de données MySQL ne se ferme pas normalement, comme lors d'un basculement, l'état du pool de mémoires tampons n'est pas enregistré sur le disque. Dans ce cas, MySQL charge n'importe quel fichier du pool de mémoires tampons disponible au redémarrage de l'instance de base de données. Il n'en résulte aucun dommage, mais le pool de tampons restauré peut ne pas refléter l'état le plus récent du pool de tampons avant le redémarrage. Pour vous assurer d'avoir un état récent du pool de mémoires tampons disponible afin de préparer le cache InnoDB au démarrage, il est recommandé que vous vidiez régulièrement le pool de mémoires tampons « à la demande ».
Vous pouvez créer un événement pour vider le pool de mémoires tampons automatiquement et à intervalles réguliers. Par exemple, l'instruction suivante crée un événement nommé `periodic_buffer_pool_dump` qui vide le pool de mémoires tampons toutes les heures.

```
CREATE EVENT periodic_buffer_pool_dump
ON SCHEDULE EVERY 1 HOUR
DO CALL mysql.rds_innodb_buffer_pool_dump_now();
```

Pour de plus amples informations sur les événements MySQL, veuillez consulter Syntaxe d'événement dans la documentation MySQL.

Vidage et chargement du pool de tampons à la demande

Vous pouvez enregistrer et charger le cache InnoDB « à la demande ».

- Pour vider l'état actuel du pool de mémoires tampons sur le disque, appelez la procédure stockée mysql.rds_innodb_buffer_pool_dump_now.
- Pour charger l'état enregistré du pool de mémoires tampons à partir du disque, appelez la procédure stockée mysql.rds_innodb_buffer_pool_load_now.
- Pour annuler une opération de chargement en cours, appelez la procédure stockée mysql.rds_innodb_buffer_pool_load_abort.

Changements linguistiques inclusifs pour RDS for MySQL 8.4

RDS for MySQL 8.4 inclut des modifications par rapport à l'édition communautaire de MySQL 8.4 relatives aux mots clés et aux schémas système pour un langage inclusif. Par exemple, la `SHOW REPLICA STATUS` commande remplace`SHOW SLAVE STATUS`.

Rubriques

- Modifications du nom des paramètres de configuration
- Changements de nom de procédure stockée

Modifications du nom des paramètres de configuration

Les paramètres de configuration suivants ont de nouveaux noms dans RDS for MySQL 8.4.

Pour des raisons de compatibilité, vous pouvez vérifier les noms des paramètres dans le `mysql` client en utilisant l'un ou l'autre des noms. Vous ne pouvez utiliser les nouveaux noms que lorsque vous modifiez des valeurs dans un groupe de paramètres MySQL 8.4 personnalisé. Pour de plus amples informations, veuillez consulter Groupes de paramètres par défaut et personnalisés.

Nom à supprimer	Nom nouveau ou préféré
`init_slave`	`init_replica`
`log_slave_updates`	`log_replica_updates`
`log_slow_slave_statements`	`log_slow_replica_statements`
`rpl_stop_slave_timeout`	`rpl_stop_replica_timeout`
`skip_slave_start`	`skip_replica_start`
`slave_checkpoint_group`	`replica_checkpoint_group`
`slave_checkpoint_period`	`replica_checkpoint_period`

Nom à supprimer	Nom nouveau ou préféré
slave_compressed_protocol	replica_compressed_protocol
slave_exec_mode	replica_exec_mode
slave_load_tmpdir	replica_load_tmpdir
slave_max_allowed_packet	replica_max_allowed_packet
slave_net_timeout	replica_net_timeout
slave_parallel_type	replica_parallel_type
slave_parallel_workers	replica_parallel_workers
slave_pending_jobs_size_max	replica_pending_jobs_size_max
slave_preserve_commit_order	replica_preserve_commit_order
slave_skip_errors	replica_skip_errors
slave_sql_verify_checksum	replica_sql_verify_checksum
slave_transaction_retries	replica_transaction_retries
slave_type_conversions	replica_type_conversions
sql_slave_skip_counter	sql_replica_skip_counter

> **Note**
>
> Le paramètre `replica_allow_batching` n'est pas disponible car Amazon RDS ne prend pas en charge les clusters NDB.

Changements de nom de procédure stockée

Les procédures stockées suivantes ont de nouveaux noms dans RDS for MySQL 8.4.

Pour des raisons de compatibilité, vous pouvez utiliser l'un ou l'autre nom dans la version 8.4 initiale de RDS pour MySQL. Les anciens noms de procédures seront supprimés dans une version ultérieure. Pour de plus amples informations, veuillez consulter Configuration, démarrage et arrêt de la réplication du journal binaire (binlog).

Nom à supprimer	Nom nouveau ou préféré
`mysql.rds_next_master_log`	`mysql.rds_next_source_log`
`mysql.rds_reset_external_master`	`mysql.rds_reset_external_source`
`mysql.rds_set_external_master`	`mysql.rds_set_external_source`
`mysql.rds_set_external_master_with_auto_position`	`mysql.rds_set_external_source_with_auto_position`
`mysql.rds_set_external_master_with_delay`	`mysql.rds_set_external_source_with_delay`
`mysql.rds_set_master_auto_position`	`mysql.rds_set_source_auto_position`

Fonctions MySQL non prises en charge par Amazon RDS

Amazon RDS ne prend pas en charge actuellement les fonctions MySQL suivantes :

- Plug-in d'authentification
- Erreur de journalisation dans le journal système
- Chiffrement d'espace de tables InnoDB
- Clusters NDB
- Plug-in de niveau de sécurité du mot de passe
- Variables système persistantes
- Plugin de réécriture de requêtes Rewriter
- Réplication semi-synchrone, sauf pour les clusters de bases de données multi-AZ
- Espace de table transportable
- Plug-in X

Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui requièrent des privilèges avancés. Amazon RDS prend en charge l'accès aux bases de données sur une instance de base de données en utilisant toute application cliente SQL standard. Amazon RDS ne permet pas d'accès d'hôte direct à une instance de base de données via Telnet, Secure Shell (SSH) ou une connexion Bureau à distance Windows. Lorsque vous créez une instance de base de données, vous êtes assigné en tant que db_owner pour toutes les bases de données de cette instance, et vous disposez de toutes les autorisations au niveau de la base de données, à l'exception de celles utilisées pour les sauvegardes. Amazon RDS gère les sauvegardes pour vous.

Versions de MySQL sur Amazon RDS

Dans MySQL, les numéros de version sont organisés en versions X.Y.Z. Dans la terminologie Amazon RDS, X.Y indique la version majeure et Z le numéro de la version mineure. Pour les implémentations Amazon RDS, un changement de version sera considéré majeur si le numéro de version majeure change—par exemple, en passant de la version 5.7 à 8.0. Un changement de version est considéré comme mineur si seul le numéro de version mineure change, par exemple, si vous passez de la version 8.0.32 à la version 8.0.34.

Rubriques

- Versions de MySQL mineures prises en charge sur Amazon RDS
- Versions de MySQL majeures prises en charge sur Amazon RDS
- Versions de support étendu d'Amazon RDS pour RDS pour MySQL
- Utilisation de l'environnement de prévisualisation de base de données
- MySQL version 9.3 dans l'environnement Database Preview
- MySQL version 9.2 dans l'environnement de prévisualisation de la base de données
- MySQL version 9.1 dans l'environnement de prévisualisation de la base de données
- Versions rendues obsolètes pour Amazon RDS for MySQL

Versions de MySQL mineures prises en charge sur Amazon RDS

Amazon RDS prend actuellement en charge les versions mineures suivantes de MySQL.

> **Note**
>
> Les dates qui comportent uniquement un mois et une année sont approximatives et seront mises à jour lorsqu'une date exacte sera connue.
> Amazon RDS Extended Support n'est pas disponible pour les versions mineures.

Le tableau suivant indique les versions mineures de MySQL 8.4 actuellement prises en charge par Amazon RDS.

Version du moteur MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
8.4.6	22 juillet 2025	1er août 2025	1er août 2026
8.4.5	15 avril 2025	29 avril 2025	septembre 2026
8.4.4	21 janvier 2025	19 février 2025	mars 2026
8.4.3	15 octobre 2024	21 novembre 2024	mars 2026

Le tableau suivant indique les versions mineures de MySQL 8.0 actuellement prises en charge par Amazon RDS.

> **Note**
>
> Les versions mineures peuvent atteindre la fin du support standard avant les versions majeures. Par exemple, la version mineure 8.0.28 a atteint sa date de fin de support standard le 28 mars 2024, tandis que la version majeure 8.0 atteint cette date le 31 juillet 2026. RDS prendra en charge les versions mineures 8.0.* supplémentaires publiées par la communauté MySQL entre ces dates. Nous vous recommandons de passer à la dernière version mineure disponible aussi souvent que possible pour toutes les versions majeures.

Version du moteur MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
8,0,43	22 juillet 2025	1er août 2025	1er août 2026
8,0,42	15 avril 2025	29 avril 2025	Juillet 2026
8,0,41	21 janvier 2025	19 février 2025	mars 2026
8,0,40	15 octobre 2024	13 novembre 2024	mars 2026
8,0,39	23 juillet 2024	12 août 2024	octobre 2025

Version du moteur MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin de la prise en charge standard de RDS
8,0,37	30 avril 2024	18 juin 2024	octobre 2025

Le tableau suivant présente les versions mineures de MySQL 5.7 disponibles sous Amazon RDS Extended Support.

> **Note**
>
> Les versions mineures peuvent atteindre la fin du Support étendu avant les versions majeures. Par exemple, la version mineure 5.7.44-RDS.20240529 atteint la date de fin du Support étendu en septembre 2025, tandis que la version majeure 5.7 atteint cette date le 28 février 2027. RDS générera et publiera des versions mineures 5.7.44-RDS.xxyyzz supplémentaires entre ces dates. Nous vous recommandons de passer à la dernière version mineure disponible aussi souvent que possible pour toutes les versions majeures.

Version du moteur MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin du support étendu RDS
5.7.44-RDS.20250508*	Ne s’applique pas	20 mai 2025	septembre 2026
5.7.44-RDS.20250213*	Ne s’applique pas	12 mars 2025	septembre 2026
5.7.44-RDS.20250103*	Ne s’applique pas	13 février 2025	mars 2026
5.7.44-RDS.20240808*	Ne s’applique pas	28 août 2024	octobre 2025
5.7.44-RDS.20240529*	Ne s’applique pas	25 juin 2024	octobre 2025

Version du moteur MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin du support étendu RDS
5.7.44-RDS.20240408*	Ne s'applique pas	17 mai 2024	octobre 2025

* MySQL Community a retiré la version majeure 5.7 et ne publiera pas de nouvelles versions mineures. Il s'agit d'une version mineure publiée par Amazon RDS avec des correctifs de sécurité critiques et des corrections de bogues pour les bases de données MySQL 5.7 couvertes par le support étendu RDS. Pour plus d'informations sur ces versions mineures, consultezthe section called "Versions de support étendu RDS". Pour plus d'informations sur le Support étendu RDS, consultez.

Vous pouvez spécifier n'importe quelle version MySQL actuellement prise en charge lorsque vous créez une instance de base de données. Vous pouvez spécifier la version majeure (par exemple MySQL 5.7), puis toute version mineure prise en charge pour la version majeure spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version prise en charge, généralement la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée. Pour voir la liste des versions prises en charge, ainsi que les valeurs par défaut pour les instances de base de données nouvellement créées, exécutez la `describe-db-engine-versions` AWS CLI commande.

Par exemple, pour répertorier les versions de moteur prises en charge pour RDS for MySQL, exécutez la commande CLI suivante :

```
aws rds describe-db-engine-versions --engine mysql --query "*[].
{Engine:Engine,EngineVersion:EngineVersion}" --output text
```

La version par défaut de MySQL peut varier selon la Région AWS. Pour créer une instance de base de données avec une version mineure spécifique, spécifiez la version mineure lors de la création de l'instance de base de données. Vous pouvez déterminer la version mineure par défaut pour un Région AWS en exécutant la AWS CLI commande suivante :

```
aws rds describe-db-engine-versions --default-only --engine mysql
 --engine-version major_engine_version --region region --query "*[].
{Engine:Engine,EngineVersion:EngineVersion}" --output text
```

*major_engine_version*Remplacez-le par la version principale du moteur, puis remplacez *region* par le Région AWS. Par exemple, la AWS CLI commande suivante renvoie la version mineure du moteur MySQL par défaut pour la version majeure 5.7 et pour l'ouest des États-Unis (Oregon) Région AWS (us-west-2) :

```
aws rds describe-db-engine-versions --default-only --engine mysql --engine-version 5.7
 --region us-west-2 --query "*[].{Engine:Engine,EngineVersion:EngineVersion}" --output
 text
```

Avec Amazon RDS, vous contrôlez à quel moment vous mettez à niveau votre instance MySQL vers une nouvelle version majeure prise en charge par Amazon RDS. Vous pouvez maintenir la compatibilité avec des versions MySQL spécifiques, tester de nouvelles versions avec votre application avant le déploiement en production et effectuer des mises à niveau de versions majeures aux moments qui correspondent le mieux à votre calendrier.

Lorsque la mise à niveau automatique de versions mineures est activée, votre instance de base de données sera automatiquement mise à niveau vers de nouvelles versions mineures MySQL, celles-ci étant prises en charge par Amazon RDS. Ces correctifs sont appliqués pendant le créneau de maintenance planifié. Vous pouvez modifier une instance de base de données pour activer ou désactiver les mises à niveau automatiques des versions mineures.

Si vous refusez les mises à niveau automatiques planifiées, vous pouvez procéder manuellement à une mise à niveau vers une version mineure prise en charge en suivant la même procédure que pour une mise à jour de la version majeure. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Amazon RDS prend actuellement en charge les mises à niveau suivantes pour les versions majeures du moteur de base de données MySQL :

- MySQL 5.7 vers MySQL 8.0
- MySQL 8.0 vers MySQL 8.4

Étant donné que les mises à niveau de version majeures impliquent quelques risques de compatibilité, elles ne sont pas appliquées automatiquement et vous devez donc faire une demande de modification de l'instance de base de données. Vous devez tester soigneusement toute mise à niveau avant de procéder à la mise à niveau de vos instances de production. Pour plus d'informations sur la mise à niveau d'une instance de base de données MySQL, consultez Mises à niveau du moteur de base de données RDS pour MySQL.

Vous pouvez tester une instance de base de données par rapport à une nouvelle version avant la mise à niveau. Pour ce faire, créez un instantané de base de données de votre instance de base de données existante, restaurez à partir de l'instantané de base de données pour créer une instance de base de données et lancez une mise à niveau de version pour la nouvelle instance de base de données. Vous pouvez ensuite procéder en toute sécurité à une expérimentation sur le clone mis à niveau de votre instance de base de données avant de décider de mettre à niveau ou pas votre instance de base de données d'origine.

Versions mineures de MySQL sur Amazon RDS

Pour les modifications apportées par la communauté MySQL aux versions mineures, consultez les mises à jour critiques, les alertes de sécurité et les bulletins sur le site Web d'Oracle. Sous Critical Patch Update, choisissez le mois où Oracle a publié la version mineure. Ensuite, choisissez la version mineure de MySQL sous Produits et versions concernés.

Versions mineures

- MySQL version 8.4.6
- MySQL version 8.4.5
- MySQL version 8.4.4
- MySQL version 8.0.43
- MySQL version 8.0.42
- MySQL version 8.0.41
- MySQL version 8.0.40
- MySQL version 8.0.39
- MySQL version 8.0.37

MySQL version 8.4.6

MySQL version 8.4.6 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Ajout d'un plugin `rds_security_audit` de base de données, un plugin RDS pour MySQL interne qui collecte les métriques liées à la sécurité. Aucune action du client n'est requise.

MySQL version 8.4.5

MySQL version 8.4.5 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

MySQL version 8.4.4

MySQL version 8.4.4 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025a`.
- Correction d'un bogue qui provoquait une erreur de classement lors de l'exécution des procédures stockées Amazon RDS `mysql.rds_set_configuration` et `mysql.rds_kill`.

MySQL version 8.0.43

MySQL version 8.0.43 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

MySQL version 8.0.42

MySQL version 8.0.42 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025b`.

MySQL version 8.0.41

MySQL version 8.0.41 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Les informations de fuseau horaire ont été mises à jour pour les baser sur celles-ci `tzdata2025a`.
- Correction d'un bogue qui provoquait une erreur de classement lors de l'exécution des procédures stockées Amazon RDS `mysql.rds_set_configuration` et `mysql.rds_kill`.

MySQL version 8.0.40

MySQL version 8.0.40 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Correction d'un bug qui provoquait des incohérences entre les jeux de caractères lors des mises à niveau de la base de données.

MySQL version 8.0.39

MySQL version 8.0.39 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Correction d'un bogue qui `sql_log_off` empêchait le bon fonctionnement avec `SESSION_VARIABLES_ADMIN` ce privilège.
- Correction d'un bogue qui empêchait l'utilisateur principal d'accorder le `SESSION_VARIABLE_ADMIN` privilège aux autres utilisateurs de la base de données.
- Correction d'un bogue qui provoquait un mélange illégal de classements lors de l'exécution des procédures stockées fournies par RDS.

MySQL version 8.0.37

MySQL version 8.0.37 est désormais disponible sur Amazon RDS. Cette version contient des correctifs et des améliorations ajoutés par la communauté MySQL et Amazon RDS.

Nouvelles fonctionnalités et améliorations

- Correction d'un bogue lié à l'exécution d'une instruction DDL instantanée suivie d'une mise à jour qui entraînait un échec d'assertion.

Versions de MySQL majeures prises en charge sur Amazon RDS

Les versions majeures de RDS for MySQL sont disponibles sous le support standard au moins jusqu'à la fin de vie de la version correspondante de la communauté. Vous pouvez continuer à exécuter une version majeure après la date de fin du support standard RDS moyennant des frais. Pour plus d'informations, consultez et Tarification d'Amazon RDS for MySQL.

Vous pouvez utiliser les dates suivantes pour planifier vos cycles de test et de mise à niveau.

> **Note**
>
> Les dates avec seulement un mois et une année sont approximatives et sont mises à jour avec une date exacte quand elles sont connues.
> Vous pouvez également consulter les informations relatives aux dates de support des principales versions du moteur à l'aide de l'API AWS CLI ou de l'API RDS. Pour de plus amples informations, veuillez consulter Affichage des dates de support pour les versions du moteur dans Amazon RDS Extended Support.

Version majeure de MySQL	Date de parution communau aire	Date de parution de RDS	Date de fin de vie de la communau é	Date de fin de la prise en charge standard de RDS	Date de début de la première année de tarificat ion du support étendu RDS	Date de début de la troisième année de tarificat ion du support étendu RDS	Date de fin du support étendu RDS
MySQL 8.4	30 avril 2024	21 novembre 2024	avril 2029	31 juillet 20	1 août 2029	1er août 2031	31 juillet 2032
MySQL 8.0	19 avril 201	23 octobre 2018	Avril 2026	31 juillet 20	1er août 20	1er août 20	31 juillet 2029

Version majeure de MySQL	Date de parution communautaire	Date de parution de RDS	Date de fin de vie de la communauté	Date de fin de la prise en charge standard de RDS	Date de début de la première année de tarification du support étendu RDS	Date de début de la troisième année de tarification du support étendu RDS	Date de fin du support étendu RDS
MySQL 5.7*	21 octobre 2015	22 février 2	Octobre 20	29 février 2	1er mars 2(	1er mars 2(	28 février 2027

* MySQL 5.7 n'est désormais disponible que dans le cadre du support étendu RDS. Pour de plus amples informations, veuillez consulter .

Versions de support étendu d'Amazon RDS pour RDS pour MySQL

Le contenu suivant répertorie toutes les versions de RDS Extended Support pour les versions de RDS for MySQL.

Versions

- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250508
- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250213
- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250103
- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240808
- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240529
- Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240408

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250508

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250508 est disponible.

Bugs corrigés :

- Indice virtuel fixe instable après le rollback lorsqu'il `index_id` est supérieur au maximum. `uint32`
- Les tests corrigés échouent en raison d'un problème de mémoire.
- Fixed `<COMMAND_CLASS>` est vide pour`<NAME>Execute</NAME>`.
- Correction : compilez MySQL avec GCC 14 [noclose 5.7].

CVEs fixe :

- CVE-2025-30682
- CVE-2025-30687
- CVE-2025-30688
- CVE-2025-21581
- CVE-2025-21585
- CVE-2025-30689
- CVE-2025-30722
- CVE-2025-21577
- CVE-2025-30693
- CVE-2025-30695
- CVE-2025-30703

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250213

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250213 est disponible.

Bugs corrigés :

- Correction de l'échec de l'assertion d'InnoDB. `result != FTS_INVALID`
- Correction du plantage et de la corruption généralisée des index spatiaux après l'`ALTER TABLE`opération de reconstruction de la table InnoDB à l'aide de l'algorithme. `INPLACE`
- Corrigé `ON DELETE CASCADE` par le crash de la colonne générée. `innobase_get_computed_value`
- Correction d'un échec d'assertion dans`row_MySQL_pad_col`.
- Correction d'un problème où le DDL en ligne provoquait l'erreur suivante :`ERROR 1712 (HY000): Index PRIMARY is corrupted`.

- Correction de crashs à. `Item_rollup_sum_switcher::current_arg`
- Correction d'un problème en raison duquel le cache de la base de données n'était pas vidé. `DROP USER`
- Correction d'un dépassement de la mémoire tampon dans. `my_print_help`
- Correction d'un problème d'InnoDB qui limitait `FULLTEXT` l'index `FTS_DOC_ID` à une valeur maximale de 32 bits non signée.

CVEs fixe :

- CVE-2025-21497
- CVE-2025-21555
- CVE-2025-21559
- CVE-2025-21490
- CVE-2025-21491
- CVE-2025-21500
- CVE-2025-21501
- CVE-2025-21540
- CVE-2025-21543
- CVE-2025-21520

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250103

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20250103 est disponible.

Bugs corrigés :

- Correction d'un problème de nettoyage FTS lors de la suppression et de l'ajout d'un `FULLTEXT` index dans la même transaction.
- Optimisation du temps d'allocation de mémoire dans le client MySQL pour éviter toute fuite potentielle.
- Correction de la troncature des résultats à 34 octets lors de l'utilisation de l'`UNION`opérateur.
- out-of-boundsAccès potentiel fixe `ulong bitmask` en raison du code d'autorisation.

CVEs fixe :

- CVE-2024-21230
- CVE-2024-21201
- CVE-2024-21241
- CVE-2024-21203

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240808

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240808 est disponible.

Bugs corrigés :

- Correction d'un échec d'assertion lié à l'index des colonnes du dictionnaire.
- Correction d'un problème lié à la `is_binlog_cache_empty()` fonction.
- Correction `heap-use-after-free` d'erreurs dans `sql/item.cc` les fichiers.
- Correction de plusieurs problèmes d'index spatial en les désactivant pour les `index-only` lectures.
- Correction d'un problème d'instrumentation avec le `LOCK_ORDER: CONNECTION_CONTROL` plugin.
- Correction de threads bloquant avec le `CONNECTION_CONTROL` plugin.
- Correction de la `PSI_THREAD_INFO` non-mise à jour pour`PREPARED STATEMENTS`.
- Correction du double traitement des mots d'index FTS avec`innodb_optimize_fulltext_only`.

CVEs fixe :

- CVE-2024-21177

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240529

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240529 est disponible.

Bugs corrigés :

- Correction d'un échec d'`field.cc`assertion par implémentation`fix_after_pullout`.
- Correction d'une défaillance du pointeur nul lors du renvoi des métadonnées au client pour certaines requêtes SQL. Ces requêtes contenaient des paramètres dynamiques et des sous-requêtes dans des `SELECT` clauses.

- Correction de résultats incorrects lors de l'utilisation `GROUP BY` pour des scans d'index lâches ou des scans de plages non contiguës d'un index.
- Correction de la perte d'informations GTID en cas de crash de MySQL pendant la persistance.
- Correction d'une condition de course qui pouvait entraîner le blocage indéfiniment d'une transaction InnoDB.
- Correction d'une condition de course lors du nettoyage des informations de certification par Group Replication.
- Correction d'un problème de numérisation de l'index rétrograde avec des opérations de page simultanées.
- Correction d'un problème d'état de recherche en texte intégral (FTS) incohérent dans des scénarios concurrents.
- Correction d'un problème d'assertion lié à la mémoire tampon de modification lors de la suppression de tables.
- Comportement unifié pour `deinit` la fonction d'appel pour tous les types de plugins.

CVEs fixe :

- CVE-2024-20963
- CVE-2024-20993
- CVE-2024-20998
- CVE-2024-21009
- CVE-2024-21054
- CVE-2024-21055
- CVE-2024-21057
- CVE-2024-21062
- CVE-2024-21008
- CVE-2024-21013
- CVE-2024-21047
- CVE-2024-21087
- CVE-2024-21096

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240408

Support étendu RDS pour RDS pour MySQL version 5.7.44-RDS.20240408 est disponible.

Cette version contient des correctifs pour les éléments suivants CVEs :

- CVE-2024-20963

Utilisation de l'environnement de prévisualisation de base de données

En juillet 2023, Oracle a annoncé un nouveau modèle de version pour MySQL. Ce modèle inclut deux types de versions : les versions Innovation et les versions LTS. Amazon RDS met à disposition les versions Innovation de MySQL dans l'environnement de prévisualisation RDS. Pour en savoir plus sur les versions Innovation de MySQL, consultez le blog Introducing MySQL Innovation and Long-Term Support (LTS) versions.

Les instances de base de données RDS for MySQL dans l'environnement de prévisualisation de base de données sont similaires sur le plan fonctionnel à d'autres instances de base de données RDS for MySQL. Cependant, vous ne pouvez pas utiliser l'environnement de prévisualisation de base de données pour les charges de travail de production.

Les environnements de prévisualisation présentent les limitations suivantes :

- Amazon RDS supprime toutes les instances de base de données 60 jours après leur création, en même temps que leurs sauvegardes et leurs instantanés.
- Vous ne pouvez utiliser que les stockages SSD à usage général et les stockages SSD IOPS provisionnés.
- Vous ne pouvez pas obtenir d'aide Support avec les instances de base de données. Vous pouvez plutôt publier vos questions sur la communauté de AWS questions-réponses gérée,AWS Re:post.
- Vous ne pouvez pas copier un instantané d'instance de base de données dans un environnement de production.

Les options suivantes sont prises en charge par la prévisualisation.

- Vous pouvez créer des instances de base de données à l'aide des classes d'instance de base de données db.m6i, db.r6i, db.m6g, db.m5, db.t3, db.r6g et db.r5. Pour plus d'informations sur les classes d'instances RDS, consultez Classes d'instances de base de données .

- Vous pouvez utiliser à la fois des déploiements mono-AZ et multi-AZ.
- Vous pouvez utiliser les fonctions de vidage et de chargement MySQL standard pour exporter des bases de données depuis l'environnement de prévisualisation de la base de données ou pour importer des bases de données dans cet environnement.

Fonctions non prises en charge dans l'environnement de prévisualisation de base de données

Les fonctions suivantes ne sont pas disponibles dans l'environnement de prévisualisation de base de données :

- Copie d'instantanés entre Régions
- Réplicas en lecture entre Régions
- RDS Proxy (Proxy RDS)

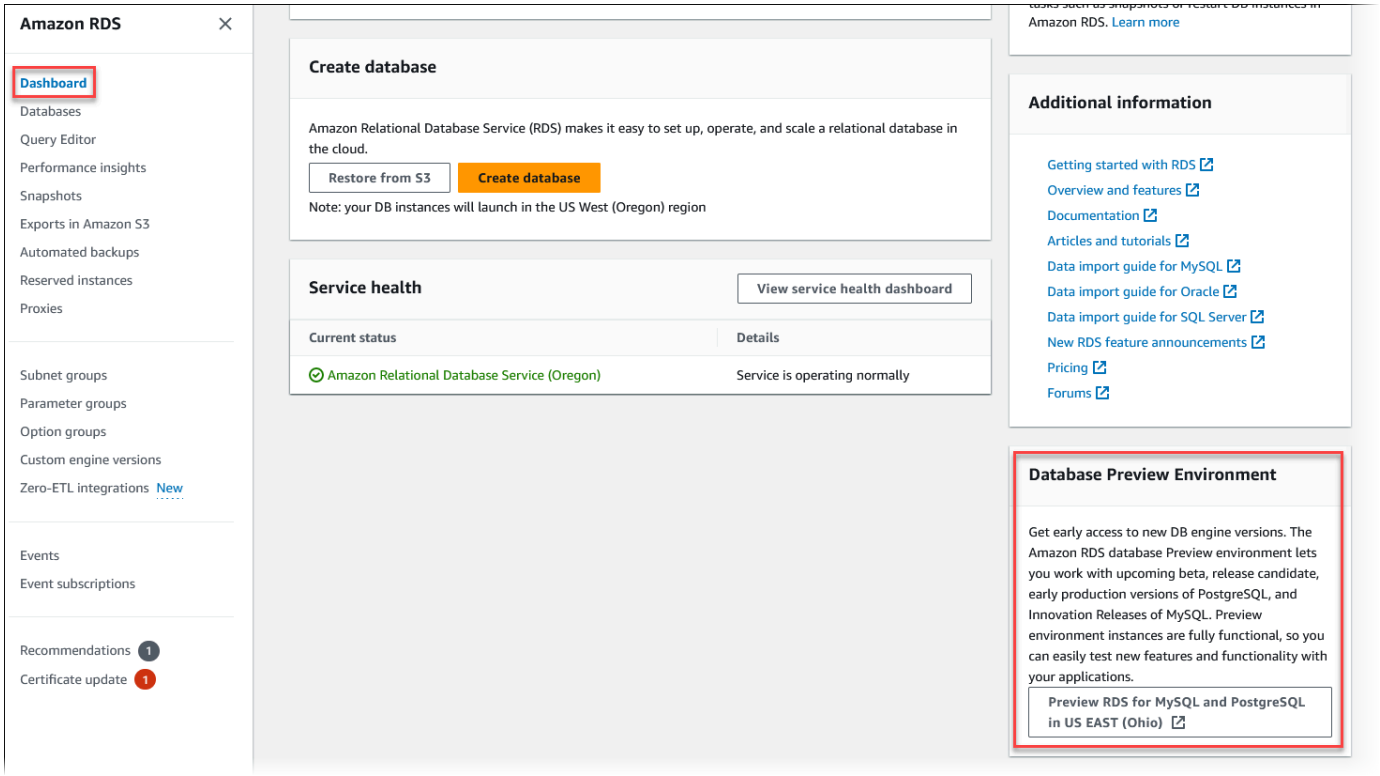
Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données

Vous pouvez créer une instance de base de données dans l'environnement Database Preview à l'aide de l'API AWS Management Console AWS CLI, ou RDS.

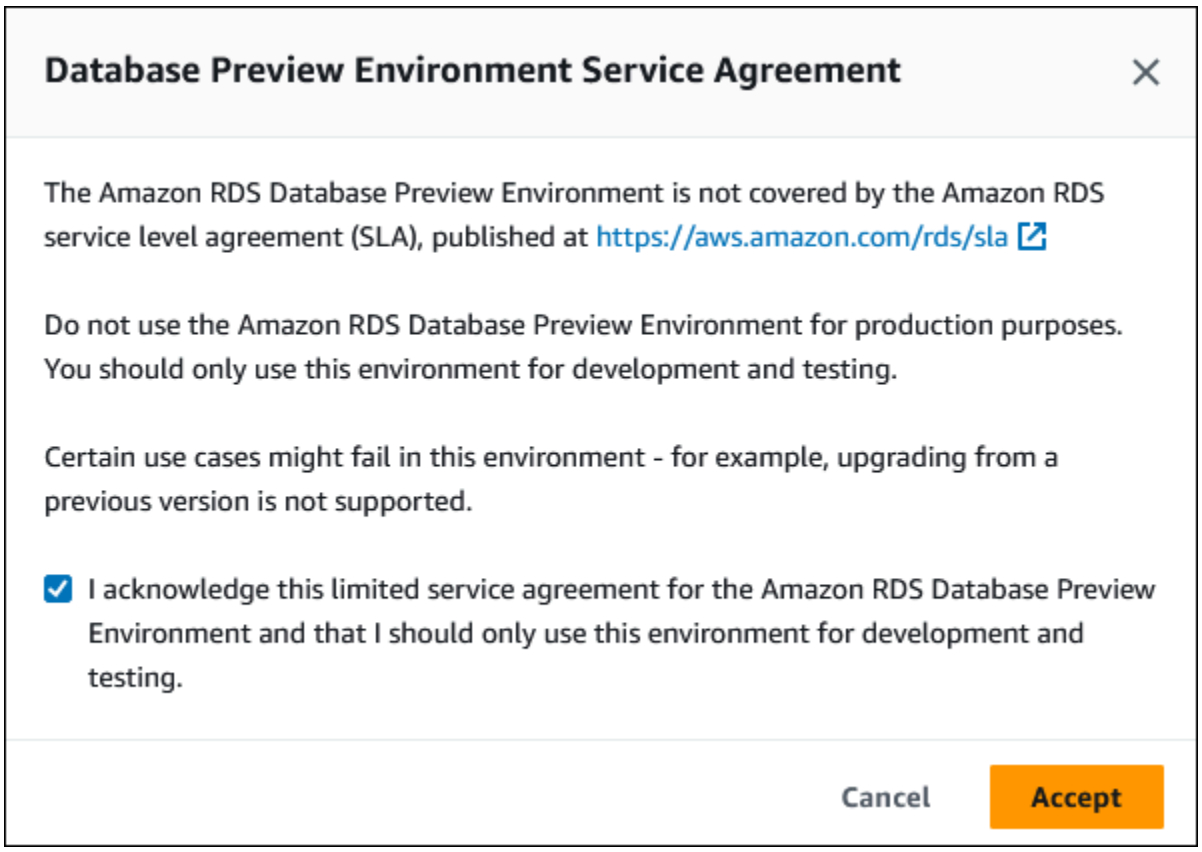
console

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Dashboard (Tableau de bord) dans le panneau de navigation.
3. Sur la page Tableau de bord, recherchez la section Environnement de prévisualisation de base de données, comme illustré dans l'image suivante.



Vous pouvez accéder directement à l'environnement de prévisualisation de base de données. Avant de poursuivre, vous devez reconnaître et accepter les limites.



4. Pour créer l'instance de base de données RDS for MySQL, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure console dans Création d'une instance de base de données.

AWS CLI

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données à l'aide de l' AWS CLI, utilisez le point de terminaison suivant.

```
rds-preview.us-east-2.amazonaws.com
```

Pour créer l'instance de base de données RDS for MySQL, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure AWS CLI dans Création d'une instance de base de données.

API RDS

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données à l'aide de l'API RDS, utilisez le point de terminaison suivant.

```
rds-preview.us-east-2.amazonaws.com
```

Pour créer l'instance de base de données RDS for MySQL, suivez le même processus que pour créer n'importe quelle instance de base de données Amazon RDS. Pour plus d'informations, consultez la procédure API RDS dans Création d'une instance de base de données.

MySQL version 9.3 dans l'environnement Database Preview

MySQL version 9.3 est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 9.3 de MySQL contient plusieurs améliorations décrites dans Modifications apportées à MySQL 9.3.0.

Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement d'aperçu de base de données”. Pour accéder à l'environnement de prévisualisation depuis la console, sélectionnez https://console.aws.amazon.com/rds-preview/.

MySQL version 9.2 dans l'environnement de prévisualisation de la base de données

MySQL version 9.2 est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 9.2 de MySQL contient plusieurs améliorations décrites dans Modifications apportées à MySQL 9.2.0.

Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement d'aperçu de base de données”. Pour accéder à l'environnement de prévisualisation depuis la console, sélectionnez https://console.aws.amazon.com/rds-preview/.

MySQL version 9.1 dans l'environnement de prévisualisation de la base de données

MySQL version 9.1 est désormais disponible dans l'environnement Amazon RDS Database Preview. La version 9.1 de MySQL contient plusieurs améliorations décrites dans Modifications apportées à MySQL 9.1.0.

Pour plus d'informations sur l'environnement de prévisualisation de base de données, consultez the section called “Environnement d'aperçu de base de données”. Pour accéder à l'environnement de prévisualisation depuis la console, sélectionnez https://console.aws.amazon.com/rds-preview/.

Versions rendues obsolètes pour Amazon RDS for MySQL

Les versions Amazon RDS for MySQL 5.1, 5.5 et 5.6 sont rendues obsolètes.

Pour plus d'informations sur la politique d'obsolescence d'Amazon RDS pour MySQL, consultez Amazon RDS. FAQs

Connexion à votre instance de base de données MySQL

Avant de vous connecter à une instance de base de données qui exécute le moteur de base de données MySQL, vous devez créer une instance de base de données. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS. Après qu'Amazon RDS a provisionné votre instance de base de données, vous pouvez utiliser n'importe quelle application cliente MySQL standard pour vous connecter à l'instance. Dans la chaîne de connexion, vous spécifiez l'adresse DNS du point de terminaison de l'instance de base de données comme paramètre de l'hôte, et le numéro de port du point de terminaison de l'instance de base de données comme paramètre du port.

Pour vous authentifier auprès de votre instance de base de données RDS, vous pouvez utiliser l'une des méthodes d'authentification pour MySQL et l'authentification de base de données AWS Identity and Access Management (IAM) :

- Pour en savoir plus sur l'authentification sur MySQL à l'aide de l'une des méthodes d'authentification pour MySQL, consultez Méthode d'authentification dans la documentation MySQL.
- Pour en savoir plus sur l'authentification sur MySQL à l'aide de l'authentification de base de données IAM, consultez Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

Vous pouvez vous connecter à une instance de base de données MySQL en utilisant des outils tels que le client de ligne de commande MySQL. Pour plus d'informations sur l'utilisation du client de ligne de commande MySQL, consultez mysql - Le client de ligne de commande MySQL dans la documentation MySQL. MySQL Workbench est une application basée sur l'interface utilisateur graphique que vous pouvez utiliser pour la connexion. Pour plus d'informations, consultez la page Download MySQL Workbench. Pour plus d'informations sur l'installation de MySQL (y compris le client de ligne de commande MySQL), consultez Installation et mise à niveau de MySQL.

Pour se connecter à une instance de base de données hors de son Amazon VPC, l'instance de base de données doit être accessible au public, l'accès doit être accordé en utilisant les règles entrantes du groupe de sécurité de l'instance de base de données et d'autres exigences doivent être respectées. Pour de plus amples informations, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

Vous pouvez utiliser le chiffrement Secure Sockets Layer (SSL) ou Transport Layer Security (TLS) pour les connexions à une instance de base de données MySQL. Pour plus d'informations, veuillez

consulter SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS. Si vous utilisez l'authentification de base de données AWS Identity and Access Management (IAM), assurez-vous d'utiliser une connexion SSL/TLS. Pour plus d'informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

Vous pouvez également vous connecter à une instance de base de données à partir d'un serveur Web. Pour plus d'informations, consultez Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS.

> **Note**
>
> Pour de plus amples informations sur la connexion à une instance de base de données MariaDB, veuillez consulter Connexion à votre instance de base de données MariaDB.

Pour rechercher et vous connecter à une instance de base de données RDS pour MySQL, consultez les rubriques suivantes.

Rubriques

- Trouver les informations de connexion pour une instance RDS for My SQL DB
- Installation du client de SQL ligne de commande My
- Connexion depuis le client de SQL ligne de commande My (non chiffré)
- Connexion depuis My SQL Workbench
- Connexion à RDS for My SQL avec le AWS JDBC pilote, le pilote AWS Python et AWS ODBC le pilote pour mon SQL
- Résolution des problèmes de connexion à votre instance My SQL DB

Trouver les informations de connexion pour une instance RDS for My SQL DB

Les informations de connexion d'une instance de base de données incluent son point de terminaison, son port et un utilisateur de base de données valide, tel que l'utilisateur principal. Par exemple, supposons qu'une valeur de point de terminaison soit `mydb.123456789012.us-east-1.rds.amazonaws.com`. Dans ce cas, la valeur du port est `3306`, et l'utilisateur de base de données est `admin`. Compte tenu de ces informations, vous spécifiez les valeurs suivantes dans une chaîne de connexion :

- Pour l'hôte ou le nom ou le DNS nom d'hôte, spécifiez`mydb.123456789012.us-east-1.rds.amazonaws.com`.
- Pour un port, spécifiez `3306`.
- Pour l'utilisateur, spécifiez `admin`.

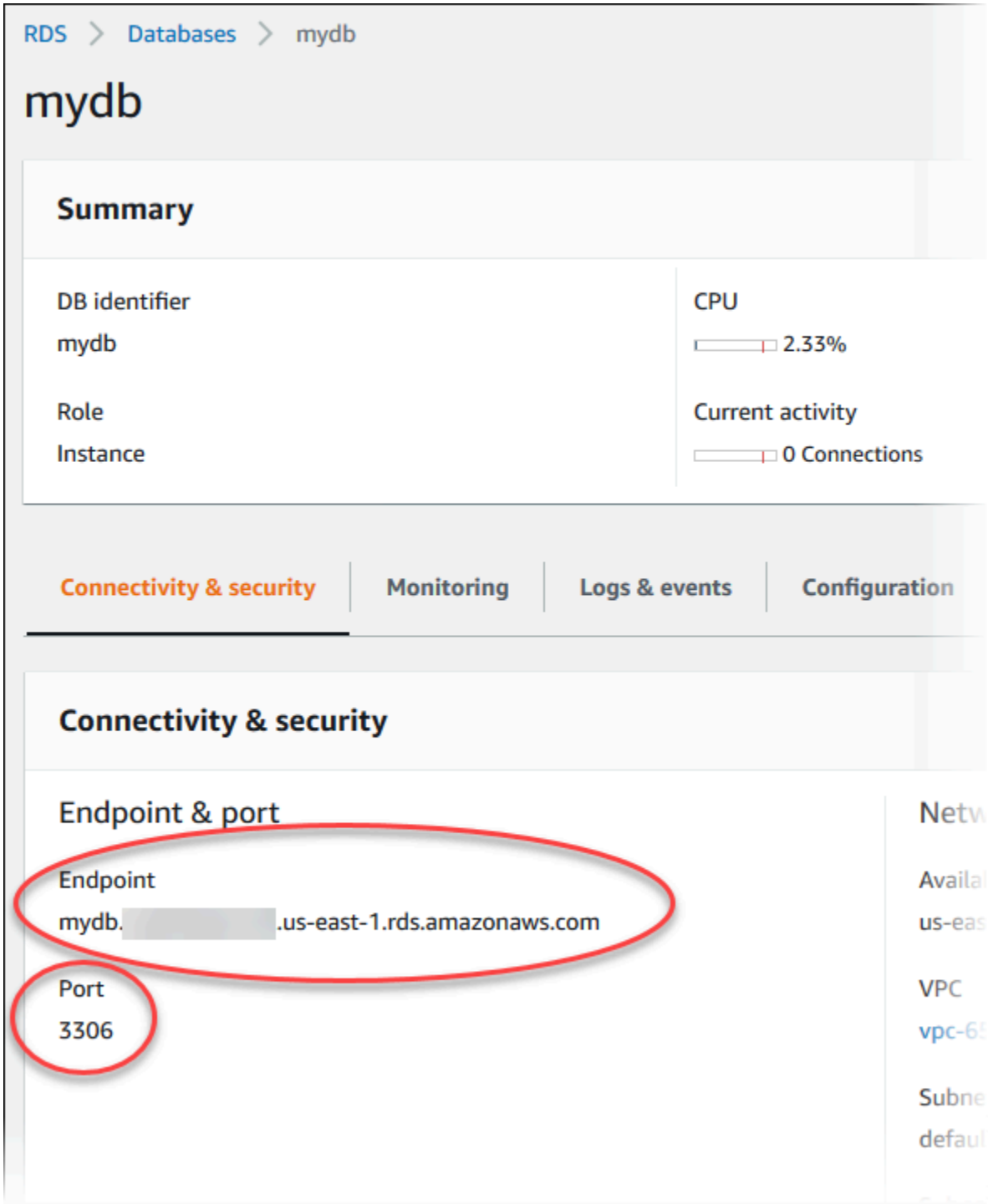
Pour vous connecter à une instance de base de données, utilisez n'importe quel client pour le moteur My SQL DB. Par exemple, vous pouvez utiliser le client de SQL ligne de commande My ou My SQL Workbench.

Pour trouver les informations de connexion d'une instance de base de données AWS Management Console, vous pouvez utiliser la AWS CLI describe-db-instancescommande ou l'escribeDBInstancesopération Amazon RDS API D pour répertorier ses détails.

Console

Pour trouver les informations de connexion d'une instance de base de données dans AWS Management Console

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez Databases (Bases de données) pour afficher la liste de vos instances de base de données.
3. Choisissez le nom de l'instance My SQL DB pour afficher ses détails.
4. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



5. Si vous devez rechercher le nom d'utilisateur principal, choisissez l'onglet Configuration et affichez la valeur Master username (Identifiant principal.

AWS CLI

Pour rechercher les informations de connexion d'une instance My SQL DB à l'aide de AWS CLI, exécutez la describe-db-instancescommande. Dans l'appel, recherchez l'ID d'instance de base de données, le point de terminaison, le port et l'identifiant principal.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-instances \
  --filters "Name=engine,Values=mysql" \
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Dans Windows:

```
aws rds describe-db-instances ^
  --filters "Name=engine,Values=mysql" ^
  --query "*[].[DBInstanceIdentifier,Endpoint.Address,Endpoint.Port,MasterUsername]"
```

Votre sortie doit ressembler à ce qui suit.

```
[
    [
        "mydb1",
        "mydb1.123456789012.us-east-1.rds.amazonaws.com",
        3306,
        "admin"
    ],
    [
        "mydb2",
        "mydb2.123456789012.us-east-1.rds.amazonaws.com",
        3306,
        "admin"
    ]
]
```

RDS API

Pour trouver les informations de connexion d'une instance de base de données à l'aide d'Amazon RDSAPI, appelez l'escribeDBInstancesopération D. Dans la sortie, recherchez les valeurs de l'adresse du point de terminaison, du port du point de terminaison et du nom d'utilisateur principal.

Installation du client de SQL ligne de commande My

La plupart des distributions Linux incluent le client MariaDB au lieu du client Oracle My. SQL Pour installer le client de SQL ligne de commande My sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install mariadb105
```

Pour installer le client de SQL ligne de commande My sur Amazon Linux 2, exécutez la commande suivante :

```
sudo yum install mariadb
```

Pour installer le client de SQL ligne de commande My sur la plupart des DEB distributions Linux, exécutez la commande suivante :

```
apt-get install mariadb-client
```

Pour vérifier la version de votre client de SQL ligne de commande My, exécutez la commande suivante :

```
mysql --version
```

Pour lire la SQL documentation relative à la version actuelle de votre client, exécutez la commande suivante :

```
man mysql
```

Connexion depuis le client de SQL ligne de commande My (non chiffré)

> **Important**
>
> N'utilisez une SQL connexion My non cryptée que lorsque le client et le serveur sont connectés VPC et que le réseau est fiable. Pour plus d'informations sur l'utilisation de connexions chiffrées, consultez Connexion à votre instance My SQL DB sur Amazon RDS SSL avec/ TLS depuis le client de SQL ligne de commande My (crypté).

Pour vous connecter à une instance de base de données à l'aide du client de SQL ligne de commande My, entrez la commande suivante à l'invite de commande. Pour le paramètre -h, remplacez le DNS nom (point de terminaison) de votre instance de base de données. Pour le paramètre -P, remplacez le port pour votre instance de base de données. Pour le paramètre -u, remplacez le nom d'utilisateur d'un utilisateur de base de données valide, par exemple l'utilisateur principal. Entrez le mot de passe de l'utilisateur principal quand vous y êtes invité.

```
mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com -P 3306 -
u mymasteruser -p
```

Après avoir entré le mot de passe pour l'utilisateur, le résultat suivant devrait normalement s'afficher.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9738
Server version: 8.0.28 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

Connexion depuis My SQL Workbench

Pour vous connecter depuis My SQL Workbench

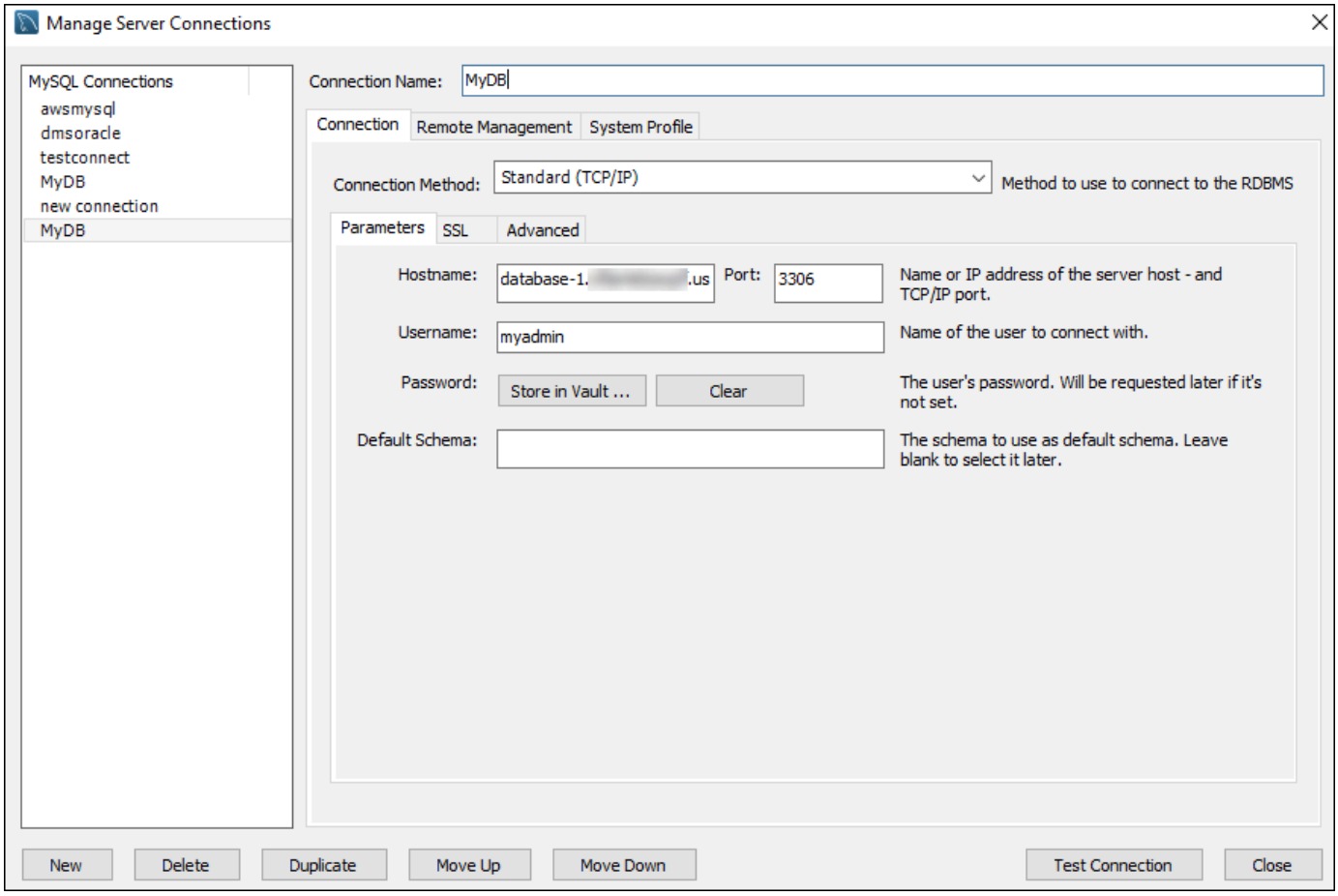
1. Téléchargez et installez My SQL Workbench sur Download My SQL Workbench.
2. Ouvrez My SQL Workbench.



3. Dans Base de données, choisissez Gérer les connexions.
4. Dans la fenêtre Gérer les connexions du serveur, choisissez Nouveau.
5. Dans la fenêtre Se connecter à la base de données, entrez les informations suivantes :

   - Connexion stockée – Entrez un nom pour la connexion, par exemple **MyDB**.
   - Nom d'hôte – Entrez le point de terminaison de l'instance de base de données.
   - Port – Entrez le port utilisé par l'instance de base de données.
   - Nom d'utilisateur – Entrez le nom d'utilisateur d'un utilisateur de base de données valide, par exemple l'utilisateur principal.
   - Mot de passe – Facultatif. Choisissez Stocker dans le coffre, puis entrez et enregistrez le mot de passe de l'utilisateur.

   La fenêtre ressemble à ce qui suit :



   Vous pouvez utiliser les fonctionnalités de My SQL Workbench pour personnaliser les connexions. Par exemple, vous pouvez utiliser l'SSLonglet pour configurer les TLS

connexionsSSL//. Pour plus d'informations sur l'utilisation SQL de My Workbench, consultez la documentation My SQL Workbench. Chiffrer les connexions client aux instances My SQL DB avecSSL/TLS, voirChiffrement des connexions client SSL avec/ TLS vers les instances My SQL DB sur Amazon RDS.

6. Facultatif. Vous pouvez choisir Tester la connexion pour vous assurer que la connexion à l'instance de base de données est réussie.
7. Choisissez Close (Fermer).
8. Dans Base de données, choisissez Se connecter à la base de données.
9. Dans Connexion stockée, choisissez votre connexion.
10. Choisissez OK.

Connexion à RDS for My SQL avec le AWS JDBC pilote, le pilote AWS Python et AWS ODBC le pilote pour mon SQL

Connectez-vous aux instances de RDS for My SQL DB avec le AWS JDBC pilote, le pilote AWS Python et le AWS ODBC pilote pour MySQL. Pour plus d'informations, consultez les rubriques suivantes.

Rubriques

- Connexion à RDS for My SQL avec le JDBC pilote Amazon Web Services (AWS)
- Connexion à RDS for My SQL avec le pilote Python Amazon Web Services (AWS)
- Connexion à RDS for My SQL avec le ODBC pilote Amazon Web Services (AWS) pour My SQL

Connexion à RDS for My SQL avec le JDBC pilote Amazon Web Services (AWS)

Le JDBC pilote Amazon Web Services (AWS) est conçu comme un JDBC wrapper avancé. Ce wrapper complète et étend les fonctionnalités d'un JDBC pilote existant. Le pilote est compatible directement avec le pilote communautaire My SQL Connector/J et le pilote communautaire MariaDB Connector/J.

Pour installer le AWS JDBC pilote, ajoutez le fichier .jar du AWS JDBC pilote (situé dans l'application`CLASSPATH`) et conservez les références au pilote communautaire correspondant. Mettez à jour le URL préfixe de connexion correspondant comme suit :

- `jdbc:mysql://` sur `jdbc:aws-wrapper:mysql://`

- `jdbc:mariadb://` sur `jdbc:aws-wrapper:mariadb://`

Pour plus d'informations sur le AWS JDBC pilote et des instructions complètes pour son utilisation, consultez le GitHub référentiel de JDBC pilotes Amazon Web Services (AWS).

Connexion à RDS for My SQL avec le pilote Python Amazon Web Services (AWS)

Le pilote Python Amazon Web Services (AWS) est conçu comme un wrapper Python avancé. Ce wrapper complète et étend les fonctionnalités du pilote open source Psycopg. Le pilote AWS Python prend en charge les versions 3.8 et supérieures de Python. Vous pouvez installer le `aws-advanced-python-wrapper` package à l'aide de la `pip` commande, en même temps que les packages `psycopg` open source.

Pour plus d'informations sur le pilote AWS Python et des instructions complètes pour son utilisation, consultez le GitHub référentiel de pilotes Python Amazon Web Services (AWS).

Connexion à RDS for My SQL avec le ODBC pilote Amazon Web Services (AWS) pour My SQL

The AWS ODBC Driver for My SQL est un pilote client conçu pour assurer la haute disponibilité de RDS for MySQL. Le pilote peut coexister avec le pilote My SQL Connector/le ODBC pilote et est compatible avec les mêmes flux de travail.

Pour plus d'informations sur le AWS ODBC Driver for My SQL et des instructions complètes pour son installation et son utilisation, consultez le ODBCpilote Amazon Web Services (AWS) pour My SQL GitHub repository.

Résolution des problèmes de connexion à votre instance My SQL DB

Les deux causes les plus courantes d'échec de connexion à une nouvelle instance de base de données sont :

- L'instance de base de données a été créée à l'aide d'un groupe de sécurité qui n'autorise pas les connexions depuis l'appareil ou l'EC2instance Amazon sur laquelle l'SQLapplication ou l'utilitaire My est exécuté. L'instance de base de données doit disposer d'un groupe de VPC sécurité qui autorise les connexions. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS.

Vous pouvez ajouter ou modifier une règle entrante dans le groupe de sécurité. Pour Source, choisissez Mon IP. Cela autorise à accéder à l'instance de base de données à partir de l'adresse IP détectée dans votre navigateur.

- L'instance de base de données a été créée à l'aide du port par défaut 3306, et votre entreprise dispose de règles de pare-feu bloquant les connexions à ce port depuis les appareils de votre réseau d'entreprise. Pour corriger le problème, recréez l'instance avec un port différent.

Pour de plus amples informations sur les problèmes de connexion, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

Sécurisation des connexions d'instance de base de données MySQL

Vous pouvez mettre en œuvre des mesures de sécurité robustes pour protéger les instances de base de données MySQL contre les accès non autorisés et les menaces potentielles. Les groupes de sécurité, le chiffrement SSL/TLS et l'authentification de base de données IAM fonctionnent ensemble pour créer plusieurs niveaux de sécurité de connexion pour vos instances de base de données MySQL. Ces contrôles de sécurité vous aident à respecter les exigences de conformité, à prévenir les violations de données et à maintenir des canaux de communication sécurisés entre les applications et les bases de données. Vous pouvez sécuriser vos instances de base de données MySQL en chiffrant les données en transit, en limitant l'accès à des plages d'adresses IP spécifiques et en gérant l'authentification des utilisateurs via des rôles IAM plutôt que des mots de passe de base de données.

La sécurité des instances de bases de données MySQL est gérée à trois niveaux :

- AWS Identity and Access Management contrôle qui peut effectuer des actions de gestion Amazon RDS sur les instances de base de données. Lorsque vous vous connectez à AWS l'aide d'informations d'identification IAM, votre compte IAM doit disposer de politiques IAM qui accordent les autorisations requises pour effectuer les opérations de gestion Amazon RDS. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.
- Lorsque vous créez une instance de base de données, vous utilisez un groupe de sécurité VPC pour contrôler quels appareils et EC2 instances Amazon peuvent ouvrir des connexions au point de terminaison et au port de l'instance de base de données. Ces connexions peuvent être établies en utilisant le protocole SSL (Secure Sockets Layer) et le protocole TLS (Transport Layer Security). En outre, les règles de pare-feu de votre entreprise peuvent contrôler si les appareils en cours d'exécution dans votre entreprise peuvent ouvrir des connexions à l'instance de base de données.
- Pour authentifier la connexion et les autorisations pour une instance de base de données MySQL, vous pouvez adopter l'une des approches suivantes ou une combinaison des deux :
  - Vous pouvez adopter la même approche qu'avec une instance autonome de MySQL. Les commandes telles que `CREATE USER`, `RENAME USER`, `GRANT`, `REVOKE` et `SET PASSWORD` fonctionnent de la même façon que dans les bases de données sur site, comme le fait la modification directe des tables du schéma de base de données. Cependant, la modification directe des tables du schéma de base de données n'est pas une bonne pratique, et à partir de la version 8.0.36 de RDS pour MySQL, elle n'est plus prise en charge. Pour de plus amples

informations, veuillez consulter Access Control and Account Management dans la documentation MySQL.

- Vous pouvez également utiliser l'authentification de base de données IAM. L'authentification de base de données IAM vous permet de vous authentifier sur votre instance de base de données à l'aide d'un utilisateur IAM ou d'un rôle IAM et d'un jeton d'authentification. Un jeton d'authentification est une valeur unique qui est générée à l'aide du processus de signature Signature Version 4. L'authentification de base de données IAM vous permet d'utiliser les mêmes informations d'identification pour contrôler l'accès à vos ressources AWS et à vos bases de données. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.
- Vous pouvez également utiliser l'authentification Kerberos pour RDS for MySQL. L'instance de base de données fonctionne avec AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD) pour activer l'authentification Kerberos. Lorsque les utilisateurs s'authentifient avec une instance de base de données MySQL jointe au domaine d'approbation, les demandes d'authentification sont transférées. Les demandes transférées sont redirigées vers le répertoire de domaines que vous avez créé AWS Directory Service. Pour de plus amples informations, veuillez consulter Utilisation de Kerberos l'authentification pour Amazon RDS for MySQL.

Lorsque vous créez une instance de base de données Amazon RDS, l'utilisateur principal a les privilèges par défaut suivants :

Version de moteur	Privilège système	Rôle de base de données
RDS pour MySQL version 8.4.3 et supérieure	`GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, RELOAD, PROCESS, REFERENCES ,INDEX, ALTER, SHOW DATABASES , CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE , REPLICATION CLIENT , CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE ROLE, DROP ROLE, APPLICATION_PASSWORD_ADMIN , FLUSH_OPTIMIZER_COSTS , FLUSH_PRIVILEGES , FLUSH_STATUS ,`	`rds_superuser_role` Pour plus d'informations sur `rds_superuser_role` , consultez Modèle de privilèges basé sur les rôles pour for My RDS SQL.

Version de moteur	Privilège système	Rôle de base de données
	`FLUSH_TABLES` , `FLUSH_USER_RESOURCES` , `ROLE_ADMIN` , `SENSITIVE_VARIABLES_OBSERVER` , `SESSION_VARIABLES_ADMIN` , `SET_ANY_DEFINER` , `SHOW_ROUTINE` , `XA_RECOVER_ADMIN`	
RDS pour MySQL version 8.0.36 et supérieure	`SELECT`, `INSERT`, `UPDATE`, `DELETE`, `CREATE`, `DROP`, `RELOAD`, `PROCESS`, `REFERENCES` , `INDEX`, `ALTER`, `SHOW DATABASES` , `CREATE TEMPORARY TABLES`, `LOCK TABLES`, `EXECUTE`, `REPLICATION SLAVE`, `REPLICATION CLIENT` , `CREATE VIEW`, `SHOW VIEW`, `CREATE ROUTINE`, `ALTER ROUTINE`, `CREATE USER`, `EVENT`, `TRIGGER`, `CREATE ROLE`, `DROP ROLE`, `APPLICATION_PASSWORD_ADMIN` , `ROLE_ADMIN` , `SET_USER_ID` , `XA_RECOVER_ADMIN`	`rds_superuser_role` Pour plus d'informations sur `rds_superuser_role` , consultez Modèle de privilèges basé sur les rôles pour for My RDS SQL.
RDS pour les versions de MySQL inférieures à 8.0.36	`SELECT`, `INSERT`, `UPDATE`, `DELETE`, `CREATE`, `DROP`, `RELOAD`, `PROCESS`, `REFERENCES` , `INDEX`, `ALTER`, `SHOW DATABASES` , `CREATE TEMPORARY TABLES`, `LOCK TABLES`, `EXECUTE`, `REPLICATION CLIENT` , `CREATE VIEW`, `SHOW VIEW`, `CREATE ROUTINE`, `ALTER ROUTINE`, `CREATE USER`, `EVENT`, `TRIGGER`, `REPLICATION SLAVE`	Aucun

> **Note**
>
> Bien qu'il soit possible de supprimer l'utilisateur maître sur l'instance de base de données, il n'est pas recommandé de le faire. Pour recréer l'utilisateur principal, utilisez l'opération Modify DBInstance RDS API ou exécutez la modify-db-instance AWS CLI commande et

spécifiez un nouveau mot de passe utilisateur principal avec le paramètre approprié. Si l'utilisateur maître n'existe pas dans l'instance, il est créé avec le mot de passe spécifié.

Pour fournir des services de gestion à chaque instance de base de données, l'utilisateur `rdsadmin` est créé lors de la création de l'instance de base de données. Les tentatives de supprimer, renommer et modifier le mot de passe du compte `rdsadmin`, ou d'en modifier les privilèges, génèrent une erreur.

Pour autoriser la gestion de l'instance de base de données, les commandes standard `kill` et `kill_query` ont fait l'objet de restrictions. Les commandes Amazon RDS `rds_kill` et `rds_kill_query` sont fournies pour vous permettre de mettre fin aux requêtes ou aux sessions utilisateur sur les instances de base de données.

Validation du mot de passe RDS pour for My SQL

My SQL fournit le `validate_password` plugin pour améliorer la sécurité. Le plugin applique les politiques de mot de passe à l'aide des paramètres du groupe de paramètres de base de données de votre instance My SQL DB. Le plugin est pris en charge pour les instances de base de données exécutant My SQL versions 5.7, 8.0 et 8.4. Pour plus d'informations sur le `validate_password` plugin, consultez le plugin de validation des mots de passe dans la section Ma SQL documentation.

Pour activer le **`validate_password`** plugin pour une instance My SQL DB

1. Connectez-vous à votre instance My SQL DB et exécutez la commande suivante.

   ```
   INSTALL PLUGIN validate_password SONAME 'validate_password.so';
   ```

2. Configurez les paramètres du plug-in dans le groupe de paramètres de base de données utilisé par l'instance de base de données.

   Pour plus d'informations sur les paramètres, consultez la section Options et variables du plugin de validation des mots de passe dans Ma SQL documentation.

   Pour plus d'informations sur la modification des paramètres d'instance de base de données, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

3. Redémarrez l'instance de base de données.

Après avoir activé le `validate_password` plugin, réinitialisez les mots de passe existants pour les mettre en conformité avec vos nouvelles politiques de validation.

Amazon RDS ne valide pas les mots de passe. L'instance My SQL DB effectue la validation du mot de passe. Si vous définissez un mot de passe utilisateur à l' AWS Management Console aide de la `modify-db-instance` AWS CLI commande ou de l'`ModifyDBInstance`RDSAPIopération, la modification peut réussir même si le nouveau mot de passe ne répond pas à vos politiques en matière de mots de passe. Toutefois, un nouveau mot de passe n'est défini dans l'instance My SQL DB que s'il est conforme aux politiques de mot de passe. Dans ce cas, Amazon RDS enregistre l'événement suivant.

```
   "RDS-EVENT-0067" - An attempt to reset the master password for the DB instance has
failed.
```

Pour plus d'informations sur les RDS événements Amazon, consultezUtilisation des notifications d'RDSÉvénements Amazon.

Chiffrement des connexions client SSL avec/ TLS vers les instances My SQL DB sur Amazon RDS

Secure Sockets Layer (SSL) est un protocole standard pour sécuriser les connexions réseau entre le client et le serveur. Après SSL la version 3.0, le nom a été changé en Transport Layer Security (TLS). Amazon RDS prend en charge TLS le SSL chiffrement/chiffrement pour les instances My SQL DB. À l'aide deSSL/TLS, vous pouvez chiffrer une connexion entre votre client d'application et votre instance My SQL DB. SSL/TLSle support est disponible dans tous les Régions AWS domaines pour MySQL.

Avec AmazonRDS, vous pouvez sécuriser les données en transit en chiffrant les connexions client aux instances My SQL DB avecSSL/TLS, en exigeantSSL/TLSpour toutes les connexions à une instance My SQL DB et en vous connectant depuis le client de SQL ligne de commande My avecSSL/TLS(crypté). Les sections suivantes fournissent des conseils sur la configuration et l'utilisation deSSL/TLSencryption pour les instances My SQL DB sur AmazonRDS.

Rubriques

- SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS

- ExigerSSL/TLSpour des comptes utilisateur spécifiques sur une instance My SQL DB sur Amazon RDS
- ExigerSSL/TLSpour toutes les connexions à une instance My SQL DB sur Amazon RDS
- Connexion à votre instance My SQL DB sur Amazon RDS SSL avec/ TLS depuis le client de SQL ligne de commande My (crypté)

SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS

Amazon RDS crée un SSL/TLS certificate and installs the certificate on the DB instance when Amazon RDS provisions the instance. These certificates are signed by a certificate authority. The SSL/TLS certificate includes the DB instance endpoint as the Common Name (CN) for the SSL/TLS certificat pour se prémunir contre les attaques par usurpation d'identité.

Un TLS certificatSSL/créé par Amazon RDS est l'entité racine fiable et devrait fonctionner dans la plupart des cas, mais il peut échouer si votre application n'accepte pas les chaînes de certificats. Si votre application n'accepte pas les chaînes de certificats, essayez d'utiliser un certificat intermédiaire pour vous connecter à votre Région AWS. Par exemple, vous devez utiliser un certificat intermédiaire pour vous connecter aux AWS GovCloud (US) régions avecSSL/TLS.

Pour plus d'informations sur le téléchargement de certificats, veuillez consulter . Pour plus d'informations sur l'utilisation deSSL/TLSavec MySQL, consultezMise à jour des applications pour se connecter aux instances My SQL DB à l'aide de nouveaux TLS certificatsSSL/.

Pour les SQL versions 8.0 et antérieures de My, Amazon RDS for My SQL utilise Open SSL pour sécuriser les connexions. Pour My SQL version 8.4 et supérieure, Amazon RDS for My est SQL utilisé AWS-LC. TLSle support dépend de la SQL version My. Le tableau suivant indique la TLS prise en charge de Mes SQL versions.

Ma SQL version	TLS 1.0	TLS1.1	TLS1.2	TLS1.3
Mon SQL 8,4	Non pris en charge	Non pris en charge	Pris en charge	Pris en charge
Mon SQL 8.0	Non pris en charge	Non pris en charge	Pris en charge	Pris en charge
Mon SQL 5.7	Pris en charge	Pris en charge	Pris en charge	Non pris en charge

ExigerSSL/TLSpour des comptes utilisateur spécifiques sur une instance My SQL DB sur Amazon RDS

Vous pouvez exiger le TLS chiffrementSSL/pour les connexions de compte utilisateur spécifiées à vos instances My SQL DB sur AmazonRDS. La protection des informations sensibles contre tout accès non autorisé ou toute interception est essentielle pour appliquer les politiques de sécurité lorsque la confidentialité des données est une préoccupation.

Pour exiger des TLS connexionsSSL/pour des comptes utilisateurs spécifiques, utilisez l'une des instructions suivantes, en fonction de votre SQL version My version, pour exiger des TLS connexionsSSL/sur le compte utilisateur`encrypted_user`.

Pour cela, utilisez l'instruction suivante.

```
ALTER USER 'encrypted_user'@'%' REQUIRE SSL;
```

Pour plus d'informations sur TLS les connexionsSSL//avec MySQL, consultez la section Utilisation de connexions cryptées dans la SQL documentation My.

ExigerSSL/TLSpour toutes les connexions à une instance My SQL DB sur Amazon RDS

Utilisez le `require_secure_transport` paramètre pour exiger que toutes les connexions utilisateur à votre instance My SQL DB utilisentSSL/TLS. Par défaut, le paramètre `require_secure_transport` est défini sur `OFF`. Vous pouvez définir le `require_secure_transport` paramètre sur pour `ON` requérirSSL/TLSpour les connexions à votre instance de base de données.

Vous pouvez définir la valeur du paramètre `require_secure_transport` en mettant à jour le groupe de paramètres de base de données pour votre instance de base de données. Vous n'avez pas besoin de redémarrer votre instance de base de données pour que la modification prenne effet.

Lorsque le paramètre `require_secure_transport` est défini sur `ON` pour une instance de base de données, un client de base de données peut s'y connecter s'il peut établir une connexion chiffrée. Sinon, un message d'erreur similaire au suivant est renvoyé au client :

```
MySQL Error 3159 (HY000): Connections using insecure transport are prohibited while --
require_secure_transport=ON.
```

Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Pour plus d'informations sur le `require_secure_transport` paramètre, consultez la section Ma SQL documentation.

Connexion à votre instance My SQL DB sur Amazon RDS SSL avec/ TLS depuis le client de SQL ligne de commande My (crypté)

Les paramètres du programme `mysql` client sont légèrement différents selon la version de My SQL ou MariaDB que vous utilisez.

Pour savoir quelle version vous avez, exécutez la commande `mysql` avec l'option `--version`. Dans l'exemple suivant, la sortie indique que le programme client provient de MariaDB.

```
$ mysql --version
mysql  Ver 15.1 Distrib 10.5.15-MariaDB, for osx10.15 (x86_64) using readline 5.1
```

La plupart des distributions Linux, telles qu'Amazon Linux, CentOS et DebianSUSE, ont remplacé My par SQL MariaDB, et la version qu'elles contiennent provient de MariaDB. `mysql`

Pour vous connecter à votre instance de base de données à l'aide deSSL/TLS, procédez comme suit :

Pour vous connecter à une instance de base de données SSL avec/ à TLS l'aide du client de ligne de SQL commande My

1. Téléchargez un certificat racine qui fonctionne pour tous Régions AWS.

   Pour plus d'informations sur le téléchargement de certificats, veuillez consulter .
2. Utilisez un client de SQL ligne de commande My pour vous connecter à une instance de base de données avec le chiffrementSSL/TLS. Pour le `-h` paramètre, remplacez le DNS nom (point de terminaison) de votre instance de base de données. Remplacez le `--ssl-ca` paramètre par le nom du fichierSSL/TLScertificate. Pour le paramètre `-P`, remplacez le port pour votre instance de base de données. Pour le paramètre `-u`, remplacez le nom d'utilisateur d'un utilisateur de base de données valide, par exemple l'utilisateur principal. Entrez le mot de passe de l'utilisateur principal quand vous y êtes invité.

   L'exemple suivant montre comment lancer le client à l'aide du `--ssl-ca` paramètre à l'aide du client My SQL 5.7 ou version ultérieure :

```
mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl-mode=REQUIRED -P 3306 -u myadmin -p
```

Pour exiger le SSL/TLS connection verifies the DB instance endpoint against the endpoint in the SSL/TLS certificat, entrez la commande suivante :

```
mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl-mode=VERIFY_IDENTITY -P 3306 -u myadmin -p
```

L'exemple suivant montre comment lancer le client à l'aide du paramètre `--ssl-ca` en utilisant le client MariaDB.

```
mysql -h mysql-instance1.123456789012.us-east-1.rds.amazonaws.com --ssl-ca=global-bundle.pem --ssl -P 3306 -u myadmin -p
```

3. Entrez le mot de passe de l'utilisateur principal quand vous y êtes invité.

Vous verrez des résultats similaires à ce qui suit.

```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9738
Server version: 8.0.28 Source distribution

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

Mise à jour des applications pour se connecter aux instances My SQL DB à l'aide de nouveaux TLS certificatsSSL/

Depuis le 13 janvier 2023, Amazon RDS a publié de nouveaux certificats d'autorité de certification (CA) pour la connexion à vos RDS instances de base de données à l'aide de Secure Socket Layer ou de Transport Layer Security (SSL/TLS). Vous trouverez ci-après des informations sur la mise à jour de vos applications afin d'utiliser les nouveaux certificats.

Cette rubrique peut vous aider à déterminer si des applications clientes utilisentSSL/TLSpour se connecter à vos instances de base de données. Si tel est le cas, il vous est alors possible de vérifier si ces applications nécessitent une vérification du certificat pour se connecter.

> **Note**
>
> Certaines applications sont configurées pour se connecter aux instances My SQL DB uniquement si elles parviennent à vérifier le certificat sur le serveur. Pour ces applications, vous devez mettre à jour les magasins d'approbations des applications clientes afin d'inclure les nouveaux certificats de l'autorité de certification.
> Vous pouvez spécifier les SSL modes suivants : `disabledpreferred`, et`required`. Lorsque vous utilisez le `preferred` SSL mode et que le certificat CA n'existe pas ou n'est pas à jour, la connexion revient à ne pas être utilisée SSL et se connecte sans chiffrement. Nous recommandons d'éviter le mode `preferred`. En mode `preferred`, si la connexion rencontre un certificat non valide, elle cesse d'utiliser le chiffrement et continue sans chiffrement.

Une fois que vous avez mis à jour les certificats de l'autorité de certification dans les magasins d'approbations des applications clientes, vous pouvez soumettre les certificats de vos instances de bases de données à une rotation. Nous vous recommandons vivement de tester ces procédures dans un environnement de développement ou intermédiaire avant de les implémenter dans vos environnements de production.

Pour de plus amples informations sur la rotation de certificats, veuillez consulter Rotation de votre certificat SSL/TLS. Pour en savoir plus sur le téléchargement de certificats, consultez . Pour plus d'informations sur l'utilisation deSSL/TLSavec les instances My SQL DB, consultezSSL/TLSsupport pour les instances My SQL DB sur Amazon RDS.

Rubriques

- Déterminer si des applications se connectent à votre instance My SQL DB à l'aide de SSL
- Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter
- Mise à jour du magasin d'approbations de votre application
- Exemple de code Java pour établir des SSL connexions

Déterminer si des applications se connectent à votre instance My SQL DB à l'aide de SSL

Si vous utilisez Amazon RDS for My SQL version 5.7, 8.0 ou 8.4 et que le schéma de performance est activé, exécutez la requête suivante pour vérifier si les connexions utilisentSSL/TLS. Pour plus

d'informations sur l'activation du schéma de performance, consultez la section Démarrage rapide du schéma de performance dans ma SQL documentation.

```
mysql> SELECT id, user, host, connection_type
       FROM performance_schema.threads pst
       INNER JOIN information_schema.processlist isp
       ON pst.processlist_id = isp.id;
```

Dans cet exemple de sortie, vous pouvez voir à la fois votre propre session (`admin`) et une application connectée en cours `webapp1` d'utilisationSSL.

```
+----+-----------------+------------------+-----------------+
| id | user            | host             | connection_type |
+----+-----------------+------------------+-----------------+
|  8 | admin           | 10.0.4.249:42590 | SSL/TLS         |
|  4 | event_scheduler | localhost        | NULL            |
| 10 | webapp1         | 159.28.1.1:42189 | SSL/TLS         |
+----+-----------------+------------------+-----------------+
3 rows in set (0.00 sec)
```

Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter

Vous pouvez vérifier si les JDBC clients et Mes SQL clients ont besoin d'une vérification de certificat pour se connecter.

JDBC

L'exemple suivant avec My SQL Connector/J 8.0 montre un moyen de vérifier les propriétés de JDBC connexion d'une application afin de déterminer si les connexions réussies nécessitent un certificat valide. Pour plus d'informations sur toutes les options de JDBC connexion pour MySQL, consultez la section Propriétés de configuration dans la SQL documentation My.

Lorsque vous utilisez My SQL Connector/J 8.0, une SSL connexion doit être vérifiée par rapport au certificat du serveur de base de données si vos propriétés de connexion sont `sslMode` définies sur `VERIFY_CA` ou`VERIFY_IDENTITY`, comme dans l'exemple suivant.

```
Properties properties = new Properties();
properties.setProperty("sslMode", "VERIFY_IDENTITY");
properties.put("user", DB_USER);
properties.put("password", DB_PASSWORD);
```

> **Note**
>
> Si vous utilisez le My SQL Java Connector v5.1.38 ou version ultérieure, ou le My SQL Java Connector v8.0.9 ou version ultérieure pour vous connecter à vos bases de données, même si vous n'avez pas explicitement configuré vos applications pour l'utiliserSSL/TLS when connecting to your databases, these client drivers default to using SSL/TLS. In addition, when using SSL/TLS, elles effectuent une vérification partielle du certificat et ne parviennent pas à se connecter si le certificat du serveur de base de données est expiré.

Mon SQL

Les exemples suivants avec My SQL Client montrent deux manières de vérifier Ma SQL connexion d'un script afin de déterminer si les connexions réussies nécessitent un certificat valide. Pour plus d'informations sur toutes les options de connexion avec My SQL Client, consultez la section Configuration côté client pour les connexions chiffrées dans la documentation MySQL.

Lorsque vous utilisez la version 5.7 ou supérieure de My SQL Client, une SSL connexion doit être vérifiée par rapport au certificat CA du serveur, s'il s'agit de l'`--ssl-mode`option que vous spécifiez `VERIFY_CA` ou`VERIFY_IDENTITY`, comme dans l'exemple suivant.

```
mysql -h mysql-database.rds.amazonaws.com -uadmin -ppassword --ssl-ca=/tmp/ssl-cert.pem
 --ssl-mode=VERIFY_CA
```

Mise à jour du magasin d'approbations de votre application

Pour plus d'informations sur la mise à jour du trust store pour Mes SQL applications, consultez la section Installation de SSL certificats dans la SQL documentation Ma.

Pour plus d'informations sur le téléchargement du certificat racine, consultez .

Pour obtenir des exemples de scripts qui importent des certificats, consultez Exemple de script pour importer les certificats dans votre magasin d'approbations.

> **Note**
>
> Lors de la mise à jour du magasin d'approbations, vous pouvez conserver les certificats plus anciens en complément de l'ajout des nouveaux certificats.

Si vous utilisez le JDBC pilote MySQL dans une application, définissez les propriétés suivantes dans l'application.

```
System.setProperty("javax.net.ssl.trustStore", certs);
System.setProperty("javax.net.ssl.trustStorePassword", "password");
```

Lorsque vous démarrez l'application, définissez les propriétés suivantes.

```
java -Djavax.net.ssl.trustStore=/path_to_trust_store/MyTruststore.jks -
Djavax.net.ssl.trustStorePassword=my_trust_store_password com.companyName.MyApplication
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Exemple de code Java pour établir des SSL connexions

L'exemple de code suivant montre comment configurer la SSL connexion qui valide le certificat de serveur à l'aide JDBC de.

```
public class MySQLSSLTest {

        private static final String DB_USER = "username";
        private static final String DB_PASSWORD = "password";
```

```
        // This trust store has only the prod root ca.
        private static final String TRUST_STORE_FILE_PATH = "file-path-to-trust-store";
        private static final String TRUST_STORE_PASS = "trust-store-password";

        public static void test(String[] args) throws Exception {
            Class.forName("com.mysql.jdbc.Driver");

            System.setProperty("javax.net.ssl.trustStore", TRUST_STORE_FILE_PATH);
            System.setProperty("javax.net.ssl.trustStorePassword", TRUST_STORE_PASS);

            Properties properties = new Properties();
            properties.setProperty("sslMode", "VERIFY_IDENTITY");
            properties.put("user", DB_USER);
            properties.put("password", DB_PASSWORD);


            Connection connection = null;
            Statement stmt = null;
            ResultSet rs = null;
            try {
                connection =
 DriverManager.getConnection("jdbc:mysql://mydatabase.123456789012.us-
east-1.rds.amazonaws.com:3306",properties);
                stmt = connection.createStatement();
                rs=stmt.executeQuery("SELECT 1 from dual");
            } finally {
                if (rs != null) {
                    try {
                        rs.close();
                    } catch (SQLException e) {
                    }
                }
                if (stmt != null) {
                   try {
                        stmt.close();
                    } catch (SQLException e) {
                   }
                }
                if (connection != null) {
                    try {
                        connection.close();
                    } catch (SQLException e) {
                        e.printStackTrace();
                    }
```

```
                }
            }
            return;
        }
    }
```

> **Important**
>
> Une fois que vous avez déterminé que vos connexions à la base de données utilisentSSL/TLSet que vous avez mis à jour le magasin de confiance des applications, vous pouvez mettre à jour votre base de données pour utiliser les certificats rds-ca-rsa 2048-g1. Pour obtenir des instructions, veuillez consulter l'étape 3 dans Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données.
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Utilisation de Kerberos l'authentification pour Amazon RDS for MySQL

Vous pouvez utiliser l'authentification Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de base de données MySQL. L'instance de base de données fonctionne avec AWS Directory Service for Microsoft Active Directory (AWS Managed Microsoft AD) pour activer l'authentification Kerberos. Lorsque les utilisateurs s'authentifient avec une instance de base de données MySQL jointe au domaine d'approbation, les demandes d'authentification sont transférées. Les demandes transférées sont redirigées vers le répertoire de domaines que vous avez créé AWS Directory Service.

Vous pouvez gagner du temps et de l'argent en conservant toutes les informations d'identification dans le même annuaire. Cette approche vous permet d'avoir un endroit centralisé de stockage et de gestion des informations d'identification pour plusieurs instances de base de données. L'utilisation d'un annuaire peut également améliorer votre profil de sécurité global.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions d'Amazon RDS avec authentification Kerberos, consultez Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS.

Vue d'ensemble de la configuration de l'authentification Kerberos pour les instances de bases de données MySQL

Pour configurer l'authentification Kerberos pour une instance de base de données MySQL, effectuez les étapes générales suivantes, décrites plus en détails par la suite :

1. AWS Managed Microsoft AD À utiliser pour créer un AWS Managed Microsoft AD répertoire. Vous pouvez utiliser le AWS Management Console AWS CLI, le ou le AWS Directory Service pour créer le répertoire. Pour plus de détails à ce sujet, consultez la section Créer votre AWS Managed Microsoft AD répertoire dans le Guide d'AWS Directory Service administration.
2. Créez un rôle AWS Identity and Access Management (IAM) qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess` Le rôle autorise Amazon RDS à effectuer des appels vers votre annuaire.

   Pour que le rôle autorise l'accès, le point de terminaison AWS Security Token Service (AWS STS) doit être activé dans le Région AWS AWS compte. AWS STS les points de terminaison sont actifs par défaut dans tous les cas Régions AWS, et vous pouvez les utiliser sans autre action. Pour plus d'informations, consultez la section Activation et désactivation AWS STS dans et Région AWS dans le guide de l'utilisateur IAM.
3. Créez et configurez des utilisateurs dans l' AWS Managed Microsoft AD annuaire à l'aide des outils Microsoft Active Directory. Pour plus d'informations sur la création d'utilisateurs dans votre Active Directory, voir Gérer les utilisateurs et les groupes dans Microsoft AD AWS géré dans le Guide d'AWS Directory Service administration.
4. Créez ou modifiez une instance de base de données MySQL. Si vous utilisez CLI ou l'API RDS dans la demande de création, spécifiez un identificateur de domaine avec le paramètre `Domain`. Utilisez l'identificateur `d-*` généré lors de la création de votre annuaire et le nom du rôle que vous avez créé.

   Si vous modifiez une instance de base de données MySQL existante pour utiliser l'authentification Kerberos, définissez les paramètres de domaine et de rôle IAM pour l'instance de base de données. Recherchez l'instance de base de données dans le même VPC que l'annuaire de domaine.
5. Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données MySQL. Créez l'utilisateur dans MySQL en utilisant la clause `CREATE USER IDENTIFIED WITH 'auth_pam'`. Les utilisateurs que vous créez de cette façon peuvent se connecter à l'instance de base de données MySQL en utilisant l'authentification Kerberos.

Configuration de l'authentification Kerberos pour les instances de base de données MySQL

Vous l'utilisez AWS Managed Microsoft AD pour configurer l'authentification Kerberos pour une instance de base de données MySQL. Pour configurer l'authentification Kerberos, procédez comme suit :

Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD

AWS Directory Service crée un Active Directory entièrement géré dans le AWS cloud. Lorsque vous créez un AWS Managed Microsoft AD annuaire, il AWS Directory Service crée deux contrôleurs de domaine et des serveurs DNS (Domain Name System) en votre nom. Les serveurs de répertoire sont créés dans des sous-réseaux différents d'un VPC. Cette redondance permet de s'assurer que votre annuaire reste accessible, y compris en cas de défaillance.

Lorsque vous créez un AWS Managed Microsoft AD répertoire, il AWS Directory Service exécute les tâches suivantes en votre nom :

- Configuration d'un annuaire Active Directory dans le VPC.
- Création d'un compte d'administrateur d'annuaire avec le nom d'utilisateur Admin et le mot de passe spécifié. Ce compte est utilisé pour gérer votre annuaire.

  > **Note**
  >
  > N'oubliez pas d'enregistrer ce mot de passe. AWS Directory Service ne le stocke pas. Vous pouvez le réinitialiser, mais vous ne pouvez pas le récupérer.

- Création d'un groupe de sécurité pour les contrôleurs de l'annuaire.

Lorsque vous lancez un AWS Managed Microsoft AD, AWS crée une unité organisationnelle (UO) qui contient tous les objets de votre répertoire. Cette unité d'organisation, qui porte le nom NetBIOS que vous avez saisi lorsque vous avez créé votre annuaire, est située dans la racine du domaine. La racine du domaine est détenue et gérée par AWS.

Le compte administrateur créé avec votre AWS Managed Microsoft AD annuaire dispose d'autorisations pour les activités administratives les plus courantes de votre unité d'organisation :

- Création, mise à jour et suppression des utilisateurs

- Ajouter des ressources à votre domaine, comme des serveurs de fichiers ou d'impression, puis attribuer des autorisations pour ces ressources aux utilisateurs dans votre unité d'organisation
- Créez des conteneurs OUs et des conteneurs supplémentaires
- Déléguer des autorités
- Restaurer des objets supprimés de la corbeille Active Directory
- Exécuter les PowerShell modules Windows AD et DNS sur le service Web Active Directory

Le compte Admin dispose également de droits pour exécuter les activités suivantes au niveau du domaine :

- Gérer les configurations DNS (ajouter, supprimer ou mettre à jour des enregistrements, des zones et des redirecteurs)
- Afficher les journaux d'évènements DNS
- Afficher les journaux d'évènements de sécurité

Pour créer un répertoire avec AWS Managed Microsoft AD

1. Connectez-vous à la AWS Directory Service console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/directoryservicev2/.
2. Dans le panneau de navigation, choisissez Directories (Répertoires), puis Set up Directory (Configurer un répertoire).
3. Choisissez AWS Managed Microsoft AD. AWS Managed Microsoft AD est la seule option que vous pouvez actuellement utiliser avec Amazon RDS.
4. Entrez les informations suivantes :

   Nom de DNS de l'annuaire

   Nom complet de l'annuaire, par exemple **`corp.example.com`**.

   Nom NetBIOS de l'annuaire

   Nom court de l'annuaire, par exemple **`CORP`**.

   Description de l'annuaire

   (Facultatif) Une description de l'annuaire.

Mot de passe administrateur

Mot de passe de l'administrateur de l'annuaire. Le processus de création d'un annuaire crée un compte d'administrateur avec le nom d'utilisateur Admin et ce mot de passe.

Le mot de passe de l'administrateur de l'annuaire ne peut pas contenir le terme « admin ». Le mot de passe est sensible à la casse et doit comporter entre 8 et 64 caractères. Il doit également contenir au moins un caractère de trois des quatre catégories suivantes :

- Lettres minuscules (a–z)
- Lettres majuscules (A–Z)
- Chiffres (0–9)
- Caractères non alphanumériques (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)

Confirmer le mot de passe

Saisissez à nouveau le mot de passe de l'administrateur.

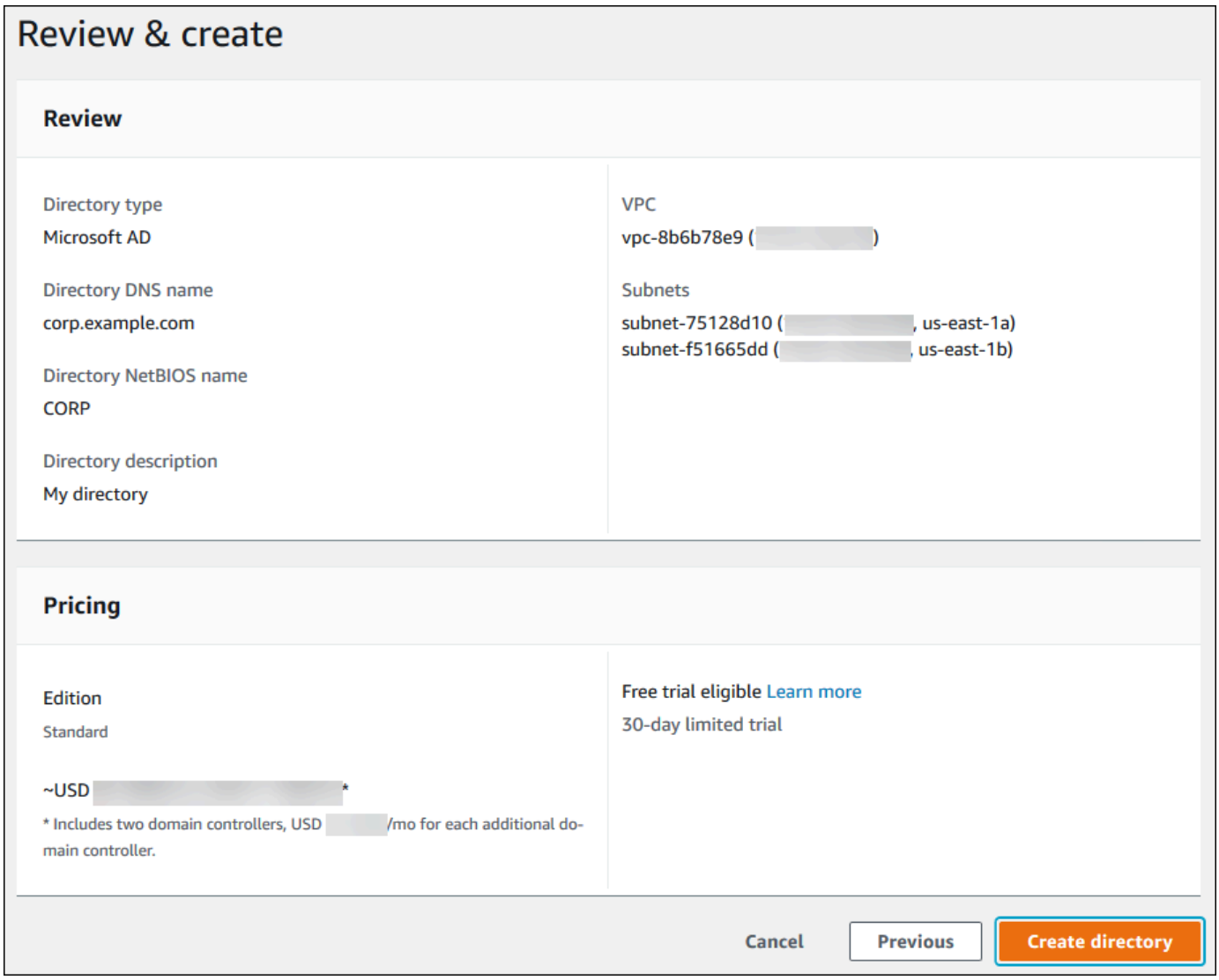
5. Choisissez Suivant.
6. Entrez les informations suivantes dans la section Networking (Réseaux), puis choisissez Suivant (Next) :

   VPC

   VPC de l'annuaire. Créez l'instance de base de données MySQL dans ce même VPC.

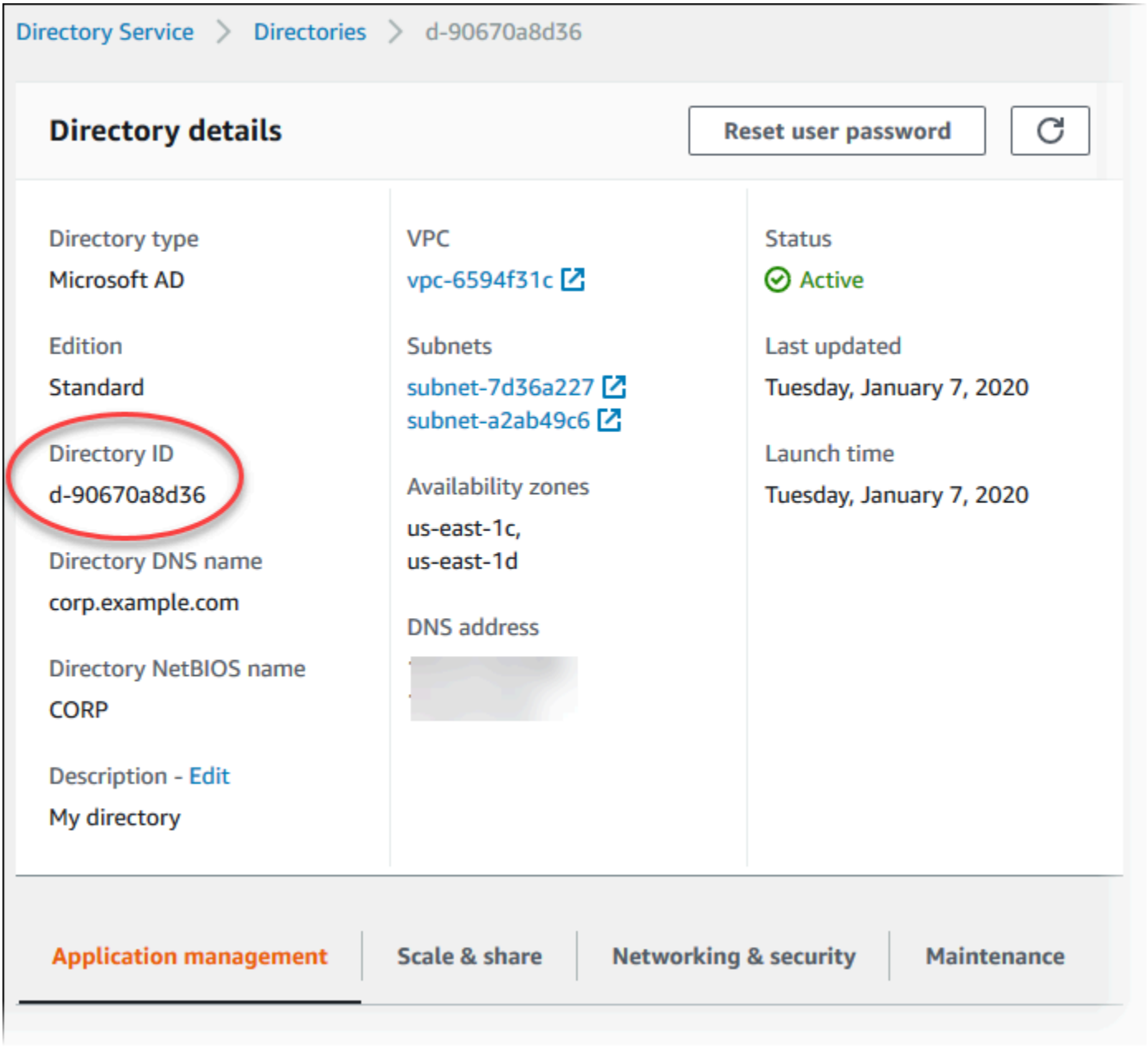
   Sous-réseaux

   Sous-réseaux pour les serveurs d'annuaires. Les deux sous-réseaux doivent être dans des zones de disponibilité différentes.
7. Vérifiez les informations concernant l'annuaire et effectuez les modifications nécessaires. Lorsque les informations sont correctes, choisissez Create directory (Créer le répertoire).



La création de l'annuaire prend plusieurs minutes. Lorsqu'il est créé, la valeur du champ Status (Statut) devient Active (Actif).

Pour consulter les informations relatives à votre annuaire, choisissez le nom de l'annuaire dans la liste. Notez la valeur ID de l'annuaire, car vous avez besoin de cette valeur lorsque vous créez ou modifiez votre instance de base de données MySQL.



Étape 2 : Créer le rôle IAM qui sera utilisé par Amazon RDS

Pour qu'Amazon RDS puisse vous appeler AWS Directory Service , un rôle IAM utilisant la politique `AmazonRDSDirectoryServiceAccess` IAM gérée est requis. Ce rôle permet à Amazon RDS d'appeler l' AWS Directory Service.

Lorsqu'une instance de base de données est créée à l'aide de AWS Management Console et que l'utilisateur de la console dispose de l'`iam:CreateRole`autorisation, la console crée automatiquement ce rôle. Dans ce cas, le nom du rôle est `rds-directoryservice-kerberos-access-role`. Sinon, vous devez créer le rôle IAM manuellement. Lorsque

vous créez ce rôle IAM`Directory Service`, choisissez et associez la politique AWS gérée `AmazonRDSDirectoryServiceAccess` à celui-ci.

Pour plus d'informations sur la création de rôles IAM pour un service, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'utilisateur IAM.

> **Note**
>
> Le rôle IAM utilisé pour l'authentification Windows pour RDS for SQL Server ne peut pas être utilisé pour RDS for MySQL.

Vous pouvez également créer des politiques avec les autorisations obligatoires au lieu d'utiliser la politique gérée IAM `AmazonRDSDirectoryServiceAccess`. Dans ce cas, le rôle IAM doit avoir la politique d'approbation IAM suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.amazonaws.com",
          "rds.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Le rôle doit également avoir la politique de rôle IAM suivante.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ds:DescribeDirectories",
        "ds:AuthorizeApplication",
        "ds:UnauthorizeApplication",
        "ds:GetAuthorizedApplicationDetails"
      ],
    "Effect": "Allow",
    "Resource": "*"
    }
  ]
}
```

Étape 3 : Créer et configurer des utilisateurs

Vous pouvez créer des utilisateurs avec l'outil Active Directory Users and Computers (Utilisateurs et ordinateurs Active Directory). Cet outil fait partie des outils Active Directory Domain Services et Active Directory Lightweight Directory Services (Services de domaine Active Directory et Services d'annuaire légers Active Directory). Les utilisateurs représentent des individus ou des entités individuelles qui ont accès à votre annuaire.

Pour créer des utilisateurs dans un AWS Directory Service annuaire, vous devez être connecté à une EC2 instance Amazon basée sur Microsoft Windows. Cette instance doit être membre de l' AWS Directory Service annuaire et être connectée en tant qu'utilisateur autorisé à créer des utilisateurs. Pour de plus amples informations, veuillez consulter Gérer des utilisateurs et des groupes dans AWS Managed Microsoft AD dans le Guide d'administration d'AWS Directory Service.

Étape 4 : Créer ou modifier une instance de base de données MySQL

Créez ou modifiez une instance de base de données MySQL à utiliser avec votre annuaire. Vous pouvez utiliser la console, CLI ou l'API RDS pour associer une instance de base de données à un annuaire. Vous pouvez effectuer cette opération de différentes manières :

- Créez une nouvelle instance de base de données MySQL à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.

- Modifiez une instance de base de données MySQL existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstanceModify RDS API.

  Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.

- Restaurez une instance de base de données MySQL à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération DBInstance Restore DBSnapshot From RDS API.

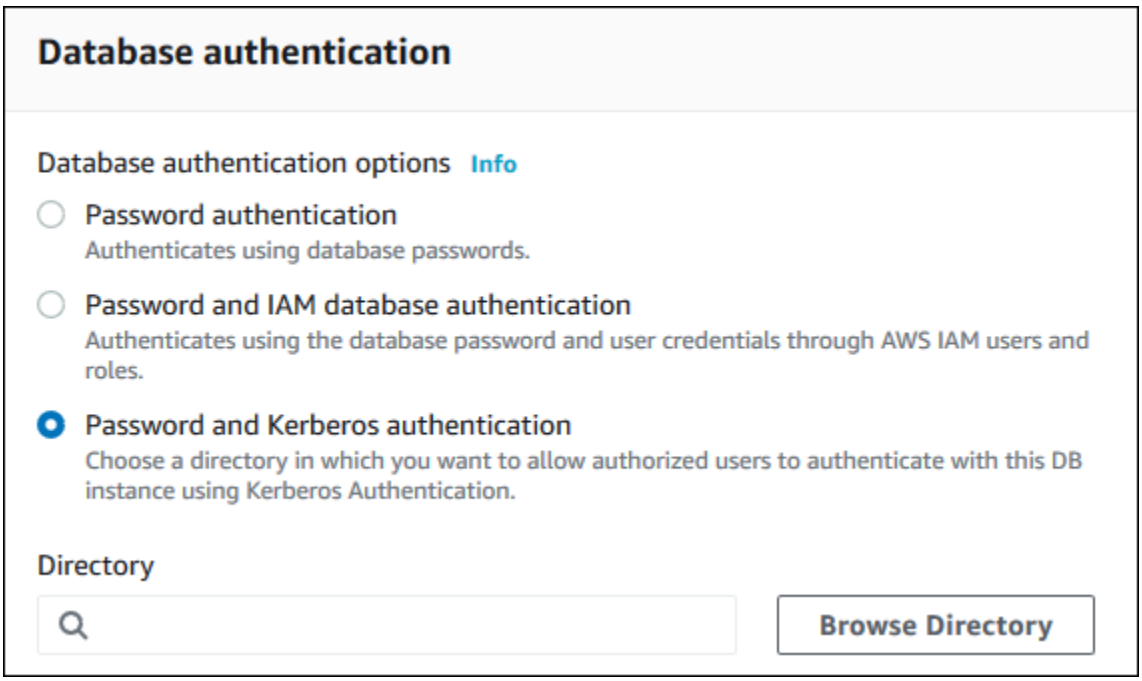
  Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.

- Restaurez une instance de base de données MySQL à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération Restore DBInstance ToPointInTime RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

L'authentification Kerberos est uniquement prise en charge pour les instances de base de données MySQL dans un VPC. L'instance de base de données peut être dans le même VPC que l'annuaire ou dans un VPC différent. L'instance de base de données doit utiliser un groupe de sécurité qui accepte les sorties du VPC, afin que l'instance de base de données puisse communiquer avec l'annuaire.

Lorsque vous utilisez la console pour créer, modifier ou restaurer une instance de bases de données, choisissez Password and Kerberos authentication (Mot de passe et authentification Kerberos) dans la section Database authentication (Authentification de base de données). Choisissez Browse Directory (Parcourir les répertoires), puis sélectionnez le répertoire, ou choisissez Create a new directory (Créer un nouveau répertoire).



Lorsque vous utilisez l'API AWS CLI ou RDS, associez une instance de base de données à un répertoire. Les paramètres suivants sont nécessaires pour que l'instance de base de données utilise l'annuaire du domaine que vous avez créé :

- Pour le paramètre `--domain`, vous devez indiquer l'identifiant du domaine (identifiant « d-* ») généré lors de la création de l'annuaire.
- Pour le paramètre `--domain-iam-role-name`, utilisez le rôle que vous avez créé qui utilise la politique IAM gérée `AmazonRDSDirectoryServiceAccess`.

Par exemple, la commande de CLI suivante modifie une instance de base de données de façon à utiliser un annuaire.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --domain d-ID \
    --domain-iam-role-name role-name
```

Dans Windows :

```
aws rds modify-db-instance ^
```

```
    --db-instance-identifier mydbinstance ^
    --domain d-ID ^
    --domain-iam-role-name role-name
```

> **Important**
>
> Si vous modifiez une instance de base de données de façon à activer l'authentification Kerberos, redémarrez l'instance de base de données après avoir effectué la modification.

Étape 5 : Créer les connexions MySQL d'authentification Kerberos

Utilisez les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données MySQL comme vous le faites pour n'importe quelle instance de base de données. L'instance de base de données est jointe au AWS Managed Microsoft AD domaine. Vous pouvez ainsi allouer les connexions et utilisateurs MySQL depuis les utilisateurs Active Directory de votre domaine. Les autorisations de base de données sont gérées via des autorisations MySQL standard qui sont accordées et révoquées à partir de ces connexions.

Vous pouvez autoriser un utilisateur Active Directory à s'authentifier avec MySQL. Pour ce faire, utilisez d'abord les informations d'identification de l'utilisateur principal Amazon RDS pour vous connecter à l'instance de base de données MySQL comme avec n'importe quelle autre instance de base de données. Une fois connecté, créez un utilisateur authentifié en externe avec PAM (Pluggable Authentication Modules) dans MySQL en exécutant la commande suivante. Remplacez *testuser* par le nom de l'utilisateur.

```
CREATE USER 'testuser'@'%' IDENTIFIED WITH 'auth_pam';
```

Les utilisateurs (personnes et applications) de votre domaine peuvent désormais se connecter à l'instance de base de données à partir d'un ordinateur client joint au domaine à l'aide de l'authentification Kerberos.

> **Important**
>
> Nous recommandons vivement aux clients d'utiliser des SSL/TLS connexions lorsqu'ils utilisent l'authentification PAM. S'ils n'utilisent pas de SSL/TLS connexion, le mot de passe peut être envoyé en texte clair dans certains cas. Pour exiger une connexion SSL/TLS

cryptée pour votre utilisateur AD, exécutez la commande suivante et remplacez-la *testuser* par le nom d'utilisateur :

```
ALTER USER 'testuser'@'%' REQUIRE SSL;
```

Pour de plus amples informations, veuillez consulter SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS.

Gestion d'une instance de base de données dans un domaine

Vous pouvez utiliser CLI ou l'API RDS pour gérer votre instance de base de données et sa relation avec votre annuaire géré Active Directory. Par exemple, vous pouvez associer un annuaire Active Directory pour l'authentification Kerberos et dissocier un annuaire Active Directory pour désactiver l'authentification Kerberos. Vous pouvez également transférer une instance de base de données vers une autre afin qu'elle soit authentifiée en externe par un annuaire Active Directory.

Par exemple, l'API Amazon RDS vous permet d'effectuer les actions suivantes :

- Pour retenter l'activation de l'authentification Kerberos en cas d'échec d'appartenance, utilisez l'opération d'API `ModifyDBInstance` et spécifiez l'ID d'annuaire d'appartenance actuelle.
- Pour mettre à jour le nom du rôle IAM de l'appartenance, utilisez l'opération d'API `ModifyDBInstance` et spécifiez l'ID d'annuaire de l'appartenance actuelle et le nouveau rôle IAM.
- Pour désactiver l'authentification Kerberos sur une instance de base de données, utilisez l'opération d'API `ModifyDBInstance` et spécifiez none comme paramètre de domaine.
- Pour déplacer une instance de base de données d'un domaine à un autre, utilisez l'opération d'API `ModifyDBInstance` et spécifiez l'identifiant du nouveau domaine en tant que paramètre de domaine.
- Pour répertorier l'appartenance pour chaque instance de base de données, utilisez l'opération d'API `DescribeDBInstances`.

Présentation de l'appartenance au domaine

Après la création ou la modification de votre instance de base de données, elle devient un membre du domaine. Vous pouvez consulter l'état de l'appartenance au domaine de l'instance de base de données en exécutant la commande describe-db-instancesCLI. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- `kerberos-enabled` – L'instance de base de données a l'authentification Kerberos activée.
- `enabling-kerberos`— AWS est en train d'activer l'authentification Kerberos sur cette instance de base de données.
- `pending-enable-kerberos` – L'activation de l'authentification Kerberos est en attente sur cette instance de base de données.
- `pending-maintenance-enable-kerberos`— AWS tentera d'activer l'authentification Kerberos sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `pending-disable-kerberos` – La désactivation de l'authentification Kerberos est en attente sur cette instance de base de données.
- `pending-maintenance-disable-kerberos`— AWS tentera de désactiver l'authentification Kerberos sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `enable-kerberos-failed` – Un problème de configuration a empêché AWS d'activer l'authentification Kerberos sur l'instance de base de données. Vérifiez et corrigez votre configuration avant d'émettre à nouveau la commande de modification de l'instance de base de données.
- `disabling-kerberos`— AWS est en train de désactiver l'authentification Kerberos sur cette instance de base de données.

Une demande d'activation de l'authentification Kerberos peut échouer à cause d'un problème de connectivité réseau ou d'un rôle IAM incorrect. Par exemple, supposons que vous créez une instance de base de données ou modifiez une instance de base de données et que la tentative d'activation de l'authentification Kerberos échoue. Si cela se produit, réémettez la commande modify ou modifiez l'instance de base de données nouvellement créée pour joindre le domaine.

Connexion à MySQL avec l'authentification Kerberos

Pour vous connecter à MySQL avec l'authentification Kerberos, vous devez vous connecter à l'aide du type d'authentification Kerberos.

Pour créer un utilisateur de base de données auquel vous pouvez vous connecter à l'aide de l'authentification Kerberos, utilisez une clause `IDENTIFIED WITH` avec l'instruction `CREATE USER`. Pour obtenir des instructions, consultez Étape 5 : Créer les connexions MySQL d'authentification Kerberos.

Pour éviter les erreurs, utilisez le client `mysql` MariaDB. Vous pouvez télécharger le logiciel MariaDB à l'adresse https://downloads.mariadb.org/.

À partir d'une invite de commande, connectez-vous à un des points de terminaison associés à votre instance de base de données MySQL. Suivez les procédures générales décrites dans Connexion à votre instance de base de données MySQL. Lorsque vous êtes invité à entrer le mot de passe, entrez le mot de passe Kerberos associé à ce nom d'utilisateur.

Restauration d'une instance de base de données MySQL et ajout de cette instance à un domaine

Vous pouvez restaurer un instantané de base de données ou terminer la point-in-time restauration d'une instance de base de données MySQL, puis l'ajouter à un domaine. Une fois que l'instance de base de données est restaurée, modifiez l'instance de base de données à l'aide du processus expliqué dans Étape 4 : Créer ou modifier une instance de base de données MySQL afin d'ajouter l'instance de base de données à un domaine.

Limitations MySQL de l'authentification Kerberos

Les limitations suivantes s'appliquent à l'authentification Kerberos pour MySQL :

- Seul un AWS Managed Microsoft AD est pris en charge. Toutefois, vous pouvez joindre des instances de base de données RDS for MySQL à des domaines Microsoft AD gérés partagés qui appartiennent à différents comptes dans la même Région AWS.
- Vous devez redémarrer l'instance de base de données après avoir activé la fonctionnalité.
- La longueur du nom de domaine ne peut pas dépasser 61 caractères.
- Vous ne pouvez pas activer l'authentification Kerberos et l'authentification IAM en même temps. Choisissez l'une ou l'autre des méthodes d'authentification pour votre instance de base de données MySQL.
- Ne modifiez pas le port d'instance de base de données après avoir activé la fonctionnalité.
- N'utilisez pas l'authentification Kerberos avec les réplicas en lecture.
- Si la mise à niveau automatique des versions mineures est activée pour une instance de base de données MySQL qui utilise l'authentification Kerberos, vous devez désactiver cette authentification, puis la réactiver après une mise à niveau automatique. Pour plus d'informations sur les mises à niveau automatiques des versions mineures, consultez Mises à niveau automatiques des versions mineures pour RDS for My SQL.
- Pour supprimer une instance de base de données pour laquelle cette fonctionnalité est activée, désactivez d'abord la fonctionnalité. Pour ce faire, exécutez la commande `modify-db-instance` CLI pour l'instance de base de données et spécifiez `none` le `--domain` paramètre.

Si vous utilisez CLI ou l'API RDS pour supprimer une instance de base de données pour laquelle cette fonctionnalité est activée, laissez s'écouler un délai.

- RDS pour MySQL ne prend pas en charge l'authentification Kerberos via une confiance forestière entre votre AD sur site ou auto-hébergé et le. AWS Managed Microsoft AD

Améliorer les performances des requêtes RDS pour My SQL grâce à Amazon RDS Optimized Reads

Vous pouvez accélérer le traitement des requêtes RDS pour My SQL grâce à Amazon RDS Optimized Reads. Une instance RDS for My SQL DB ou un cluster de base de données multi-AZ qui utilise des lectures RDS optimisées peut permettre un traitement des requêtes jusqu'à deux fois plus rapide qu'une instance ou un cluster de base de données qui ne l'utilise pas.

Rubriques

- Vue d'ensemble des lectures RDS optimisées
- Cas d'utilisation pour des lectures RDS optimisées
- Bonnes pratiques pour des lectures RDS optimisées
- Utilisation de lectures RDS optimisées
- Surveillance des instances de base de données qui utilisent des lectures RDS optimisées
- Limites relatives aux lectures RDS optimisées

Vue d'ensemble des lectures RDS optimisées

Lorsque vous utilisez une instance RDS for My SQL DB ou un cluster de base de données multi-AZ sur lequel les lectures RDS optimisées sont activées, les performances de requête sont plus rapides grâce à l'utilisation d'un magasin d'instances. Un stockage d'instances fournit un stockage temporaire de niveau bloc pour votre instance de base de données ou votre cluster de bases de données multi-AZ. Le stockage est situé sur des disques SSD Non-Volatile Memory Express (NVMeSSDs) connectés physiquement au serveur hôte. Ce stockage est optimisé pour une faible latence, de hautes performances d'E/S aléatoires et un haut débit de lecture séquentielle.

RDSLes lectures optimisées sont activées par défaut lorsqu'une instance de base de données ou un cluster de base de données multi-AZ utilise une classe d'instance de base de données avec un magasin d'instance, tel que db.m5d ou db.m6gd. Avec les lectures RDS optimisées, certains objets temporaires sont stockés dans le magasin d'instance. Ces objets temporaires incluent des fichiers temporaires internes, des tables temporaires internes sur disque, des fichiers de mappage de mémoire et des fichiers de cache de journal binaire (binlog). Pour plus d'informations sur le magasin d'instances, consultez le magasin d'EC2instances Amazon dans le guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux.

Les charges de travail qui génèrent des objets temporaires dans My SQL pour le traitement des requêtes peuvent tirer parti du magasin d'instances pour accélérer le traitement des requêtes. Ce type de charge de travail inclut les requêtes impliquant des tris, des agrégations de hachage, des jointures à charge élevée, des expressions de table communes (CTEs) et des requêtes sur des colonnes non indexées. Ces volumes de stockage d'instance offrent des performances IOPS et des performances supérieures, quelles que soient les configurations de stockage utilisées pour le EBS stockage Amazon persistant. Dans la RDS mesure où Optimized Reads transfère les opérations sur les objets temporaires vers le magasin d'instance, les opérations d'entrée/sortie par seconde (IOPS) ou le débit du stockage persistant (AmazonEBS) peuvent désormais être utilisés pour des opérations sur des objets persistants. Ces opérations incluent des lectures et des écritures régulières de fichiers de données, ainsi que des opérations de moteur en arrière-plan, telles que le vidage et la fusion de mémoires tampon par insertion.

> **Note**
>
> Les RDS instantanés manuels et automatisés contiennent uniquement des fichiers de moteur pour les objets persistants. Les objets temporaires créés dans le magasin d'instances ne sont pas inclus dans les RDS instantanés.

Cas d'utilisation pour des lectures RDS optimisées

Si vos charges de travail dépendent fortement d'objets temporaires, tels que des tables ou des fichiers internes, pour l'exécution des requêtes, vous pouvez tirer parti de l'activation des lectures RDS optimisées. Les cas d'utilisation suivants sont éligibles aux lectures RDS optimisées :

- Applications qui exécutent des requêtes analytiques avec des expressions de table communes complexes (CTEs), des tables dérivées et des opérations de regroupement
- Réplicas en lecture qui traitent un trafic de lecture important avec des requêtes non optimisées
- Applications exécutant des requêtes de création de rapport à la demande ou dynamiques impliquant des opérations complexes, telles que des requêtes avec des clauses `GROUP BY` et `ORDER BY`
- Charges de travail utilisant des tables temporaires internes pour le traitement des requêtes

  Vous pouvez surveiller la variable de statut du moteur `created_tmp_disk_tables` pour déterminer le nombre de tables temporaires sur disque créées sur votre instance de base de données.

- Applications qui créent de grandes tables temporaires, directement ou dans le cadre de procédures, pour stocker des résultats intermédiaires
- Requêtes de base de données qui regroupent ou trient des colonnes non indexées

Bonnes pratiques pour des lectures RDS optimisées

Utilisez les meilleures pratiques suivantes pour RDS optimiser les lectures :

- Ajoutez une logique de nouvelle tentative pour les requêtes en lecture seule au cas où elles échoueraient en raison d'un stockage d'instances complet pendant l'exécution.
- Surveillez l'espace de stockage disponible sur le magasin d'instances à l'aide de la CloudWatch métrique`FreeLocalStorage`. Si le stockage d'instances atteint sa limite en raison de la charge de travail sur l'instance de base de données, modifiez l'instance de base de données pour utiliser une classe d'instances de base de données plus grande.
- Lorsque votre instance de base de données ou votre cluster de bases de données multi-AZ dispose de suffisamment de mémoire mais atteint toujours la limite de stockage sur le stockage d'instances, augmentez la valeur `binlog_cache_size` pour conserver en mémoire les entrées binlog spécifiques à la session. Cette configuration empêche l'écriture des entrées binlog dans les fichiers de cache binlog temporaires sur le disque.

  Le paramètre `binlog_cache_size` est spécifique à la session. Vous pouvez modifier cette valeur pour chaque nouvelle session. Le réglage de ce paramètre peut augmenter l'utilisation de la mémoire sur l'instance de base de données pendant les pics de charge de travail. Par conséquent, envisagez d'augmenter la valeur du paramètre en fonction du modèle de charge de travail de votre application et de la mémoire disponible sur l'instance de base de données.
- Pour les versions My SQL 8.0 et antérieures, utilisez la valeur par défaut de `MIXED` pour le `binlog_format` paramètre. En fonction de la taille des transactions, le réglage de `binlog_format` sur `ROW` peut entraîner la création de fichiers de cache binlog volumineux sur le stockage d'instances. Pour My SQL 8.4 et versions ultérieures, utilisez la valeur par défaut de `ROW` pour le `binlog_format` paramètre.
- Définissez le paramètre internal_tmp_mem_storage_engine sur `TempTable` et définissez le paramètre temptable_max_mmap pour qu'il corresponde à la taille du stockage disponible sur le stockage d'instances.
- Évitez d'effectuer des modifications en bloc dans une transaction unique. Ces types de transactions peuvent générer de gros fichiers de cache binlog sur le stockage d'instances et peuvent provoquer des problèmes lorsque le stockage d'instances est plein. Envisagez de diviser

les écritures en plusieurs petites transactions afin de réduire au maximum l'utilisation de l'espace de stockage pour les fichiers de cache binlog.

- Utilisez la valeur par défaut de `ABORT_SERVER` pour le paramètre `binlog_error_action`. Cela évite les problèmes liés à la journalisation binaire sur les instances de base de données lorsque les sauvegardes sont activées.

Utilisation de lectures RDS optimisées

Lorsque vous provisionnez une instance RDS for My SQL DB avec l'une des classes d'instance DB suivantes dans le cadre d'un déploiement d'instance de base de données mono-AZ, d'un déploiement d'instance de base de données multi-AZ ou d'un déploiement de cluster de base de données multi-AZ, l'instance de base de données utilise automatiquement des RDS lectures optimisées.

Pour activer les lectures RDS optimisées, effectuez l'une des opérations suivantes :

- Créez une instance RDS for My SQL DB ou un cluster de base de données Multi-AZ à l'aide de l'une de ces classes d'instances de base de données. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données existante RDS pour My SQL DB ou un cluster de base de données Multi-AZ pour utiliser l'une de ces classes d'instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

RDSLes lectures optimisées sont disponibles partout Régions AWS RDS où une ou plusieurs classes d'instance de base de données avec NVMe SSD stockage local sont prises en charge. Pour plus d'informations sur les classes d'instances de base de données, consultez the section called “Classes d'instances de base de données”.

La disponibilité des classes d'instances de base de données diffère pour Régions AWS. Pour déterminer si une classe d'instance de base de données est prise en charge dans une classe spécifique Région AWS, consultezthe section called “Déterminer le support des classes d'instance de base de données dans Régions AWS”.

Si vous ne souhaitez pas utiliser les lectures RDS optimisées, modifiez votre instance de base de données ou votre cluster de base de données multi-AZ afin qu'il n'utilise pas de classe d'instance de base de données prenant en charge cette fonctionnalité.

Surveillance des instances de base de données qui utilisent des lectures RDS optimisées

Vous pouvez surveiller les instances de base de données qui utilisent des lectures RDS optimisées à l'aide CloudWatch des métriques suivantes :

- `FreeLocalStorage`
- `ReadIOPSLocalStorage`
- `ReadLatencyLocalStorage`
- `ReadThroughputLocalStorage`
- `WriteIOPSLocalStorage`
- `WriteLatencyLocalStorage`
- `WriteThroughputLocalStorage`

Ces métriques fournissent des données sur le stockage d'instanceIOPS, le stockage et le débit disponibles. Pour plus d'informations sur ces métriques, consultez Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS.

Limites relatives aux lectures RDS optimisées

Les limites suivantes s'appliquent aux lectures RDS optimisées :

- RDSLes lectures optimisées sont prises en charge pour les versions suivantes :
  - RDSpour My SQL version 8.0.28 et versions supérieures majeures et mineures

  Pour plus d'informations sur RDS les versions pour Mes SQL versions, consultezVersions de MySQL sur Amazon RDS.
- Vous ne pouvez pas modifier l'emplacement des objets temporaires vers le stockage persistant (AmazonEBS) sur les classes d'instance de base de données qui prennent en charge les lectures RDS optimisées.
- Lorsque la journalisation binaire est activée sur une instance de base de données, la taille maximale des transactions est limitée par la taille du stockage d'instances. Dans MySQL, toute session qui nécessite plus de stockage que la valeur de `binlog_cache_size` écrit les modifications de transaction dans les fichiers de cache binlog temporaires, qui sont créés sur le magasin d'instance.

- Les transactions peuvent échouer lorsque le stockage d'instances est plein.

Améliorer les performances d'écriture grâce à RDS Optimized Writes for My SQL

Vous pouvez améliorer les performances des transactions d'écriture avec RDS Optimized Writes for MySQL. Lorsque votre SQL base de données RDS for My utilise des écritures RDS optimisées, elle peut atteindre un débit de transactions d'écriture jusqu'à deux fois supérieur.

Rubriques

- Vue d'ensemble des écritures RDS optimisées
- Utilisation d'écritures RDS optimisées
- Activation des écritures RDS optimisées sur une base de données existante
- Limites relatives aux écritures RDS optimisées

Vue d'ensemble des écritures RDS optimisées

Lorsque vous activez les écritures RDS optimisées, vos SQL bases de données RDS for My n'écrivent qu'une seule fois lors du transfert des données vers un stockage durable, sans avoir besoin de la mémoire tampon à double écriture. Les bases de données continuent de fournir des protections de ACID propriété pour des transactions de base de données fiables, ainsi que des performances améliorées.

Les bases de données relationnelles, telles que MySQL, fournissent les ACIDpropriétés d'atomicité, de cohérence, d'isolation et de durabilité nécessaires à des transactions de base de données fiables. Pour fournir ces propriétés, My SQL utilise une zone de stockage de données appelée tampon doublewrite qui empêche les erreurs d'écriture de page partielles. Ces erreurs se produisent en cas de panne matérielle alors que la base de données met à jour une page, par exemple en cas de panne de courant. Une SQL base de données My peut détecter des écritures de page partielles et effectuer une restauration avec une copie de la page dans le tampon d'écriture double. Cette technique offre une protection, mais elle entraîne également des opérations d'écriture supplémentaires. Pour plus d'informations sur le tampon My SQL doublewrite, voir Doublewrite Buffer dans la documentation My. SQL

Lorsque les écritures RDS optimisées sont activées, RDS pour Mes SQL bases de données, n'écrivez qu'une seule fois lors du transfert des données vers un stockage durable sans utiliser la mémoire tampon à double écriture. RDS Les écritures optimisées sont utiles si vous exécutez des charges de travail intensives en écriture sur vos bases de données RDS for My. SQL Parmi les

bases de données soumises à de lourdes charges de travail en écriture, citons celles qui prennent en charge les paiements numériques, les transactions financières et les applications de jeu.

Ces bases de données s'exécutent sur des classes d'instances de base de données qui utilisent le système AWS Nitro. En raison de la configuration matérielle dans ces systèmes, la base de données peut écrire des pages de 16 Kio directement dans des fichiers de données de manière fiable et durable, en une seule étape. Le système AWS Nitro permet d'RDSoptimiser les écritures.

Vous pouvez définir le nouveau paramètre de base de données `rds.optimized_writes` pour contrôler la fonctionnalité d'écritures RDS optimisées RDS pour mes SQL bases de données. Accédez à ce paramètre dans les groupes de paramètres de base de données RDS pour My SQL version 8.0 et RDS pour My SQL version 8.4. Définissez ce paramètre sur l'une des valeurs suivantes :

- `AUTO`— Activez les écritures RDS optimisées si la base de données le permet. Désactivez les écritures RDS optimisées si la base de données ne les prend pas en charge. Il s'agit de la valeur par défaut.
- `OFF`— Désactive les écritures RDS optimisées même si la base de données les prend en charge.

Si vous avez une base de données existante avec une version de moteur, une classe d'instance de base de données, un and/or file system format that doesn't support RDS Optimized Writes, you can enable the feature by creating a blue/green déploiement. Pour de plus amples informations, veuillez consulter the section called “Activation sur une base de données existante”.

Si vous migrez une SQL base de données RDS for My configurée pour utiliser les écritures RDS optimisées vers une classe d'instance de base de données qui ne prend pas en charge cette fonctionnalité, désactive RDS automatiquement les écritures RDS optimisées pour la base de données.

Lorsque les écritures RDS optimisées sont désactivées, la base de données utilise le tampon My SQL doublewrite.

Pour déterminer si une SQL base de données RDS for My utilise des écritures RDS optimisées, consultez la valeur actuelle du `innodb_doublewrite` paramètre de la base de données. Si la base de données utilise des écritures RDS optimisées, ce paramètre est défini sur `FALSE` (`0`).

Utilisation d'écritures RDS optimisées

Vous pouvez activer les écritures RDS optimisées lorsque vous créez une base de données RDS pour Ma SQL base de données à l'aide de la RDS console AWS CLI, du ou du RDSAPI. RDS Les écritures optimisées sont activées automatiquement lorsque les deux conditions suivantes s'appliquent lors de la création de la base de données :

- Vous spécifiez une version du moteur de base de données et une classe d'instance de base de données qui prennent en charge les écritures RDS optimisées.
  - RDSLes écritures optimisées sont prises en charge RDS pour My SQL version 8.0.30 et supérieures. Pour plus d'informations sur RDS les versions pour Mes SQL versions, consultezVersions de MySQL sur Amazon RDS.
  - RDSOptimized Writes est pris en charge RDS pour les SQL bases de données For My qui utilisent les classes d'instances de base de données suivantes :
    - db.m7i
    - db.m7g
    - db.m6g
    - db.m6gd
    - db.m6i
    - db.m5
    - db.m5d
    - db.r7i
    - db.r7g
    - db.r6g
    - db.r6gd
    - db.r6i
    - db.r5
    - db.r5b
    - db.r5d
    - db.x2idn
    - db.x2iedn

Pour plus d'informations sur les classes d'instances de base de données, consultez the section called “Classes d'instances de base de données”.

La disponibilité des classes d'instances de base de données diffère pour Régions AWS. Pour déterminer si une classe d'instance de base de données est prise en charge dans une classe spécifique Région AWS, consultezthe section called “Déterminer le support des classes d'instance de base de données dans Régions AWS”.

Pour mettre à niveau votre base de données vers une classe d'instance de base de données prenant en charge les écritures RDS optimisées, vous pouvez créer un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter the section called “Activation sur une base de données existante”.

- Dans le groupe de paramètres associé à la base de données, le paramètre `rds.optimized_writes` est défini sur `AUTO`. Dans les groupes de paramètres par défaut, ce paramètre est toujours défini sur `AUTO`.

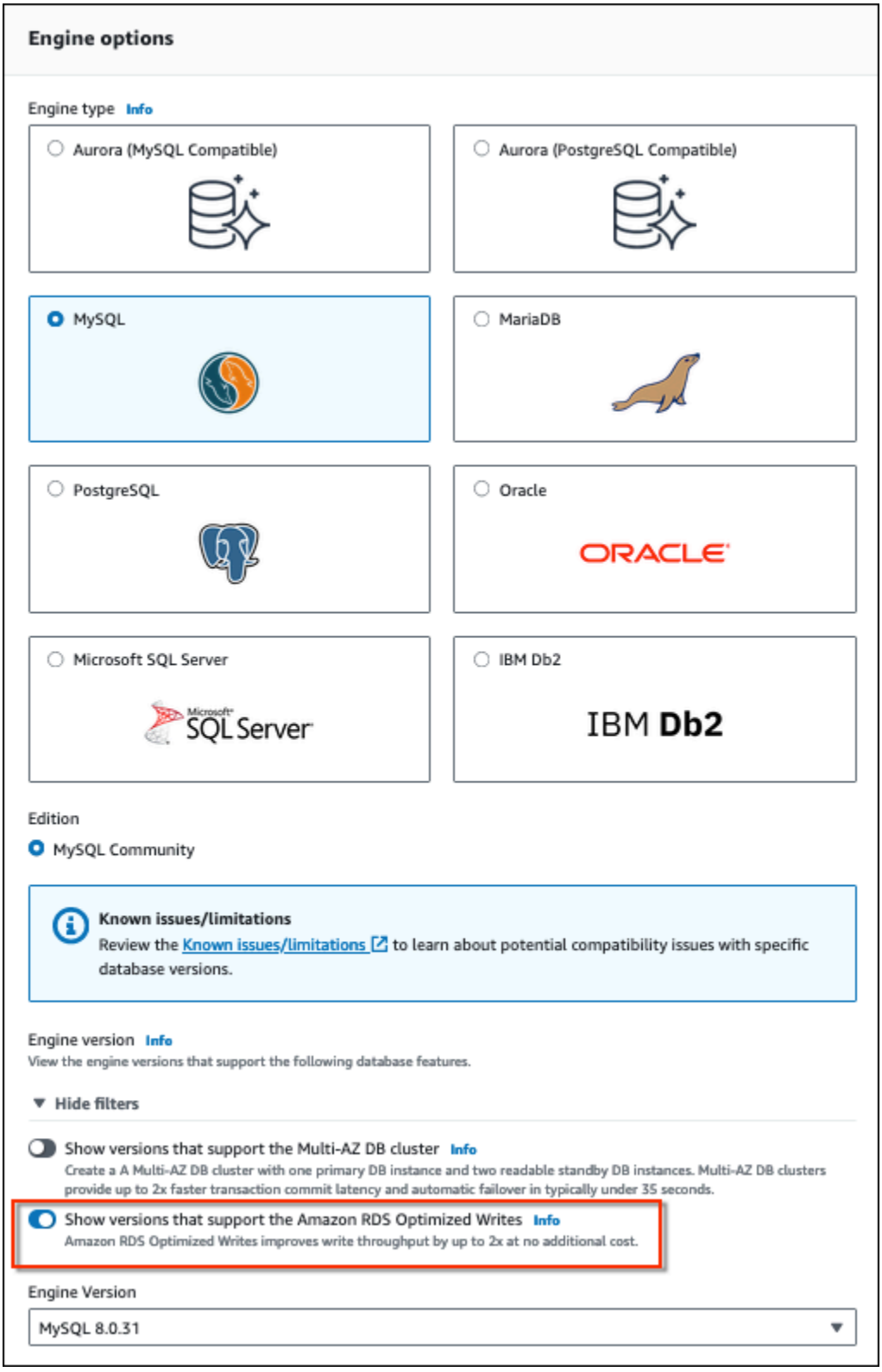
Si vous souhaitez utiliser une version du moteur de base de données et une classe d'instance de base de données qui prennent en charge les écritures RDS optimisées, mais que vous ne souhaitez pas utiliser cette fonctionnalité, spécifiez un groupe de paramètres personnalisé lors de la création de la base de données. Dans le groupe de paramètres, définissez le paramètre `rds.optimized_writes` sur `OFF`. Si vous souhaitez que la base de données utilise les écritures RDS optimisées ultérieurement, vous pouvez définir le paramètre sur `AUTO` pour l'activer. Pour obtenir des informations sur la création des groupes de paramètres personnalisés et sur la définition des paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.

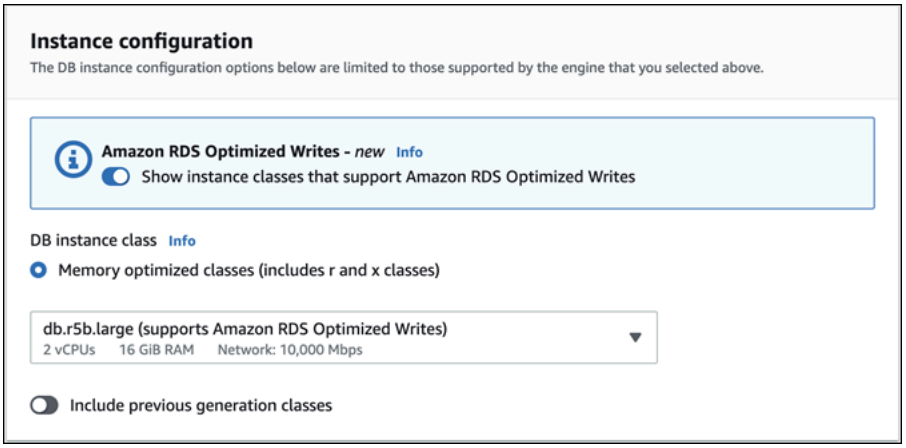
Console

Lorsque vous utilisez la RDS console pour créer une base de données RDS for My, vous pouvez filtrer les versions du moteur de SQL base de données et les classes d'instances de base de données qui prennent en charge les écritures RDS optimisées. Après avoir activé les filtres, vous pouvez choisir parmi les versions du moteur de base de données et les classes d'instances de base de données disponibles.

Pour choisir une version du moteur de base de données qui prend en charge les écritures RDS optimisées, filtrez les versions du moteur My SQL DB qui la prennent en charge dans la version du moteur, puis choisissez une version. RDS



Dans la section Configuration de l'instance, filtrez les classes d'instance de base de données qui prennent en charge les écritures RDS optimisées, puis choisissez une classe d'instance de base de données.



Après avoir effectué ces sélections, vous pouvez choisir d'autres paramètres qui répondent à vos besoins et terminer de créer la base de données RDS pour Ma SQL base de données à l'aide de la console.

AWS CLI

Pour créer une instance de base de données à l'aide de AWS CLI, exécutez la create-db-instancecommande. Assurez-vous que les `--db-instance-class` valeurs `--engine-version` et prennent en charge les écritures RDS optimisées. De plus, veillez à ce que le paramètre `rds.optimized_writes` du groupe de paramètres associé à l'instance de base de données soit défini sur AUT0. Cet exemple associe le groupe de paramètres par défaut à l'instance de base de données.

Example Création d'une instance de base de données utilisant des écritures RDS optimisées

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --engine mysql \
    --engine-version 8.0.30 \
    --db-instance-class db.r5b.large \
    --manage-master-user-password \
    --master-username admin \
    --allocated-storage 200
```

Dans Windows:

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --engine mysql ^
    --engine-version 8.0.30 ^
    --db-instance-class db.r5b.large ^
    --manage-master-user-password ^
    --master-username admin ^
    --allocated-storage 200
```

RDS API

Vous pouvez créer une instance de base de données à l'aide de reateDBInstance l'opération C. Lorsque vous utilisez cette opération, assurez-vous que les `DBInstanceClass` valeurs `EngineVersion` et prennent en charge les écritures RDS optimisées. De plus, veillez à ce que le paramètre `rds.optimized_writes` du groupe de paramètres associé à l'instance de base de données soit défini sur `AUTO`.

Activation des écritures RDS optimisées sur une base de données existante

Pour modifier une base de données existante RDS pour que Ma SQL base de données active les écritures RDS optimisées, la base de données doit avoir été créée avec une version de moteur de base de données et une classe d'instance de base de données prises en charge. En outre, la base de données doit avoir été créée après la publication d'RDSOptimized Writes le 27 novembre 2022, car la configuration requise du système de fichiers sous-jacent est incompatible avec celle des bases de données créées avant sa publication. Si ces conditions sont remplies, vous pouvez activer les écritures RDS optimisées en définissant le `rds.optimized_writes` paramètre sur`AUTO`.

Si votre base de données n'a pas été créée avec une version de moteur, une classe d'instance ou une configuration de système de fichiers prise en charge, vous pouvez utiliser le RDS Blue/Green Deployments to migrate to a supported configuration. While creating the blue/green déploiement. Procédez comme suit :

- Sélectionnez Activer les écritures optimisées sur la base de données verte, puis spécifiez une version du moteur et une classe d'instance de base de données qui prennent en charge les écritures RDS optimisées. Pour obtenir la liste des versions de moteur et des classes d'instance prises en charge, consultez Utilisation d'écritures RDS optimisées.
- Sous Stockage, choisissez Mettre à niveau la configuration du système de fichiers de stockage. Cette option met à niveau la base de données vers une configuration de système de fichiers sous-jacent compatible.

Lorsque vous créez le déploiement bleu/vert, si le `rds.optimized_writes` paramètre est défini surAUT0, les écritures RDS optimisées seront automatiquement activées dans l'environnement vert. Vous pouvez ensuite basculer le déploiement bleu/vert, qui favorise l'environnement vert comme nouvel environnement de production.

Pour de plus amples informations, veuillez consulter the section called “Création d'un déploiement bleu/vert”.

Limites relatives aux écritures RDS optimisées

Lorsque vous restaurez une SQL base de données RDS for My à partir d'un instantané, vous ne pouvez activer les écritures RDS optimisées pour la base de données que si toutes les conditions suivantes sont réunies :

- L'instantané a été créé à partir d'une base de données qui prend en charge les écritures RDS optimisées.
- L'instantané a été créé à partir d'une base de données créée après la publication d'RDSOptimized Writes.
- L'instantané est restauré dans une base de données qui prend en charge les écritures RDS optimisées.
- La base de données restaurée est associée à un groupe de paramètres où le paramètre `rds.optimized_writes` est défini sur AUT0.

Mises à niveau du moteur de base de données RDS pour MySQL

Lorsque Amazon RDS prend en charge une nouvelle version d'un moteur de base de données, vous pouvez mettre à niveau vos instances de base de données vers cette nouvelle version. Il existe deux types de mises à niveau pour les bases de données MySQL : les mises à niveau des versions majeures et les mises à niveau des versions mineures.

Mises à niveau de version majeure.

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. En conséquence, vous devez effectuer manuellement les mises à niveau de version majeure de vos instances de base de données. Vous pouvez lancer une mise à niveau de version majeure en modifiant votre instance de base de données. Avant d'effectuer une mise à niveau de version majeure, nous vous recommandons de suivre les instructions figurant dansMises à niveau des versions majeures de RDS pour MySQL.

Pour les mises à niveau de versions majeures des déploiements d'instances de base de données multi-AZ, Amazon RDS met à niveau simultanément les répliques principales et de secours. Votre instance de base de données ne sera pas disponible tant que la mise à niveau ne sera pas terminée. Pour les mises à niveau de versions majeures des déploiements de clusters de bases de données multi-AZ, Amazon RDS met à niveau les instances membres du cluster une par une.

> **Tip**
>
> Vous pouvez minimiser le temps d'arrêt requis pour une mise à niveau de version majeure en utilisant un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Mises à niveau de version mineure.

Les mises à niveau de versions mineures incluent uniquement les modifications rétrocompatibles avec les applications existantes. Vous pouvez lancer manuellement une mise à niveau de version mineure en modifiant votre instance de base de données. Vous pouvez également activer l'option de mise à niveau automatique des versions mineures lors de la création ou de la modification d'une instance de base de données. Cela signifie qu'Amazon RDS met automatiquement à niveau

votre instance de base de données après avoir testé et approuvé la nouvelle version. Pour de plus amples informations sur la mise à niveau, veuillez consulter Mise à niveau de la version du moteur d'une instance de base de données.

Lorsque vous effectuez une mise à niveau de version mineure d'un cluster de base de données multi-AZ, Amazon RDS met à niveau les instances de base de données du lecteur une par une. Ensuite, l'une des instances de base de données du lecteur devient la nouvelle instance de base de données du rédacteur. Amazon RDS met ensuite à niveau l'ancienne instance d'écriture (qui est désormais une instance de lecteur).

> **Note**
>
> Le temps d'arrêt lié à une mise à niveau de version mineure d'un déploiement d'instance de base de données multi-AZ peut durer plusieurs minutes. Les clusters de bases de données multi-AZ réduisent généralement le temps d'arrêt des mises à niveau de versions mineures à environ 35 secondes. Lorsqu'il est utilisé avec le proxy RDS, vous pouvez réduire davantage les temps d'arrêt à une seconde ou moins. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS. Vous pouvez également utiliser un proxy de base de données open source tel que ProxySQL ou le pilote AWS Advanced JDBC Wrapper. PgBouncer

Si votre instance de base de données MySQL utilise des répliques en lecture, vous devez mettre à niveau toutes les répliques en lecture avant de mettre à niveau l'instance source.

Rubriques

- Considérations relatives aux mises à niveau de MySQL
- Trouver des cibles de mise à niveau valides
- Numéros de version de MySQL
- Numéros de version RDS dans RDS pour MySQL
- Mises à niveau des versions majeures de RDS pour MySQL
- Tester et effectuer ma RDS mise SQL à niveau
- Mise à niveau d'une instance de base de données MySQL
- Mises à niveau automatiques des versions mineures pour RDS for My SQL
- Utilisation d'une réplique en lecture pour réduire les temps d'arrêt lors de la mise à niveau d'une base de données RDS for MySQL

Considérations relatives aux mises à niveau de MySQL

Amazon RDS prend deux instantanés de base de données ou plus au cours du processus de mise à niveau. Amazon RDS prend jusqu'à deux instantanés de l'instance de base de données avant d'apporter des modifications à la mise à niveau. Si la mise à niveau ne fonctionne pas pour vos bases de données, vous pouvez restaurer l'un de ces instantanés pour créer une instance de base de données exécutant l'ancienne version. Amazon RDS prend un autre instantané de l'instance de base de données une fois la mise à niveau terminée. Amazon RDS prend ces instantanés, qu'il AWS Backup gère ou non les sauvegardes de l'instance de base de données.

> **Note**
>
> Amazon RDS ne prend des instantanés de base de données que si vous avez défini la période de rétention des sauvegardes de votre instance de base de données sur un nombre supérieur à 0. Pour modifier la période de rétention des sauvegardes, consultez Modification d'une RDS instance de base de données Amazon.

Une fois la mise à niveau terminée, vous ne pouvez pas rétablir la version précédente du moteur de base de données. Si vous souhaitez revenir à la version précédente, restaurez le premier instantané de base de données pris pour créer une nouvelle instance de base de données.

Vous contrôlez à quel moment vous mettez à niveau votre instance de base de données vers une nouvelle version prise en charge par Amazon RDS. Ce niveau de contrôle vous aide à maintenir la compatibilité avec des versions de base de données spécifiques et à tester les nouvelles versions avec votre application avant un déploiement en production. Lorsque vous êtes prêt, vous pouvez effectuer des mises à niveau de version aux moments qui conviennent le mieux à votre planning.

Si votre instance de base de données utilise la réplication en lecture, vous devez mettre à niveau toutes les répliques en lecture avant de mettre à niveau l'instance source.

Trouver des cibles de mise à niveau valides

Lorsque vous utilisez le AWS Management Console pour mettre à niveau une instance de base de données, il affiche les cibles de mise à niveau valides pour l'instance de base de données. Vous pouvez également exécuter la AWS CLI commande suivante pour identifier les cibles de mise à niveau valides pour une instance de base de données :

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine mysql \
  --engine-version version_number \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine mysql ^
  --engine-version version_number ^
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Par exemple, pour identifier les cibles de mise à niveau valides pour une instance de base de données MySQL version 8.0.28, exécutez la commande suivante : AWS CLI

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine mysql \
  --engine-version 8.0.28 \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine mysql ^
  --engine-version 8.0.28 ^
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Numéros de version de MySQL

La séquence de numérotation des versions du moteur de base de données RDS pour MySQL se présente sous la forme major.minor.patch.YYYYMMDD ou major.minor.patch, par exemple 8.0.33.R2.20231201 ou 5.7.44. Le format utilisé dépend de la version du moteur MySQL. Pour plus d'informations sur la numérotation des versions de RDS Extended Support, consultez. Dénomination de la version d'Amazon RDS Extended Support

majeur

Le numéro de version principal est à la fois le nombre entier et la première partie fractionnaire du numéro de version, par exemple 8.0. Une mise à niveau majeure augmente la partie majeure du numéro de version. Par exemple, une mise à niveau de 5.7 .44 vers 8.0.33 est une mise à niveau de version majeure, où 5.7 et 8.0 sont les numéros de version principaux.

mineur

Le numéro de version secondaire est la troisième partie du numéro de version, par exemple, le 33 dans la version 8.0.33.

patch

Le correctif est la quatrième partie du numéro de version, par exemple le R2 dans 8.0.33.R2. Une version de correctif RDS inclut des corrections de bogues importantes apportées à une version mineure après sa publication.

YYYYMMDD

La date est la cinquième partie du numéro de version, par exemple, le 20231201 dans 8.0.33.R2.20231201. Une version RDS date est un correctif de sécurité qui inclut des correctifs de sécurité importants ajoutés à une version mineure après sa publication. Il n'inclut aucun correctif susceptible de modifier le comportement d'un moteur.

Le tableau suivant explique le schéma de dénomination de RDS pour MySQL version 8.4.

8.4 version mineure	Schéma de dénomination
≥ 3	Les nouvelles instances de base de données utilisent Major.Minor.Patch.YYMMDD, par exemple 8.4.3.R2.20241201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple 8.4.3.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.

Le tableau suivant explique le schéma de dénomination de RDS pour MySQL version 8.0.

Version mineure 8.0	Schéma de dénomination
≥ 33	Les nouvelles instances de base de données utilisent Major.Minor.Patch.YYMMDD, par exemple 8.0.33.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple 8.0.33.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.
< 33	Les instances de base de données existantes utilisent major.minor.patch, par exemple 8.0.32.R2.

Le tableau suivant explique le schéma de dénomination de RDS pour MySQL version 5.7.

5.7 version mineure	Schéma de dénomination
≥ 42	Les nouvelles instances de base de données utilisent Major.Minor.Patch.YYMMDD, par exemple 5.7.42.R2.20231201. Les instances de base de données existantes peuvent utiliser major.minor.patch, par exemple 5.7.42.R2, jusqu'à votre prochaine mise à niveau de version majeure ou mineure.

Numéros de version RDS dans RDS pour MySQL

Les numéros de version RDS utilisent le schéma de dénomination *major*.*minor*.*patch* ou le schéma de *major*.*minor*.*patch*.*YYYYMMDD* dénomination. Les versions d'Amazon RDS Extended Support utilisent le schéma de dénomination *minor*-*RDS*.*YYYYMMDD* des versions secondaires.

Une version de correctif RDS inclut des corrections de bogues importantes apportées à une version mineure après sa publication. Une version de date RDS (*YYYYMMDD*) est un correctif de sécurité. Un correctif de sécurité n'inclut aucun correctif susceptible de modifier le comportement du moteur. Pour plus d'informations sur la numérotation des versions de RDS Extended Support, consultez. Dénomination de la version d'Amazon RDS Extended Support

Vous pouvez connaître le numéro de version RDS de votre base de données RDS pour MySQL à l'aide de la requête SQL suivante :

```
mysql> select mysql.rds_version();
```

Par exemple, l'interrogation d'une base de données RDS for MySQL 8.0.34 renvoie le résultat suivant :

```
+---------------------+
| mysql.rds_version() |
+---------------------+
| 8.0.34.R2.20231201  |
+---------------------+
1 row in set (0.01 sec)
```

Mises à niveau des versions majeures de RDS pour MySQL

Amazon RDS prend en charge les mises à niveau sur place des versions majeures suivantes du moteur de base de données MySQL :

- MySQL 5.7 vers MySQL 8.0
- MySQL 8.0 vers MySQL 8.4

> **Note**
>
> Vous ne pouvez créer des instances de base de données MySQL versions 5.7, 8.0 et 8.4 qu'avec des classes d'instance de base de données de dernière génération et de génération actuelle.
> Dans certains cas, vous souhaitez mettre à niveau une instance de base de données exécutée sur une classe d'instance de base de données de génération précédente vers une instance de base de données dotée d'une version supérieure du moteur MySQL. Dans ce cas, commencez par modifier l'instance de base de données afin d'utiliser une classe d'instance de base de données de génération actuelle ou de dernière génération. Ensuite, vous pouvez modifier l'instance de base de données pour utiliser la version supérieure du moteur de base de données MySQL. Pour de plus amples informations sur les classes d'instance de base de données Amazon RDS, veuillez consulter Classes d'instances de base de données .

Rubriques

- Présentation des mises à niveau de version majeure MySQL
- Vérifications préalables pour les mises à niveau
- Annulation après échec de la mise à niveau

Présentation des mises à niveau de version majeure MySQL

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. En conséquence, Amazon RDS n'applique pas automatiquement les mises à niveau de version majeure. Vous devez modifier manuellement votre instance de base de données. Nous vous recommandons de tester soigneusement toute mise à niveau avant de l'appliquer à vos instances de production.

Pour effectuer une mise à niveau de version majeure, effectuez d'abord toutes les mises à jour du système d'exploitation disponibles. Une fois les mises à jour du système d'exploitation terminées, effectuez une mise à niveau vers chaque version majeure, par exemple de 5.7 à 8.0, puis de 8.0 à 8.4. Pour plus d'informations sur la mise à niveau d'un cluster de bases de données multi-AZ RDS pour MySQL, consultez. Mise à niveau de la version du moteur d'un cluster de base de données multi-AZ pour Amazon RDS Les instances de base de données MySQL créées avant le 24 avril 2014 affichent une mise à jour de système d'exploitation disponible jusqu'à ce que celle-ci soit appliquée. Pour de plus amples informations sur les mises à jour du système d'exploitation, veuillez consulter Appliquer des mises à jour à un d'instances de base de données.

Au cours d'une mise à niveau de version majeure de MySQL, Amazon RDS exécute le fichier binaire MySQL `mysql_upgrade` pour mettre à niveau les tables, si nécessaire. Amazon RDS vide également les tables `slow_log` et `general_log` pendant une mise à niveau de version majeure. Pour conserver les informations de journal, enregistrez le contenu du journal avant la mise à niveau de version majeure.

Les mises à niveau des versions majeures de MySQL durent généralement environ 10 minutes. Certaines mises à niveau peuvent durer plus longtemps en raison de la taille de la classe d'instance de base de données ou parce que l'instance ne suit pas certaines directives opérationnelles indiquées dans Bonnes pratiques relatives à Amazon RDS.. Si vous effectuez une mise à niveau d'une instance de base de données à partir de la console Amazon RDS, le statut de l'instance de base de données indique quand la mise niveau est terminée. Si vous effectuez la mise à niveau à l'aide de AWS Command Line Interface (AWS CLI), utilisez la describe-db-instancescommande et vérifiez la `Status` valeur.

Vérifications préalables pour les mises à niveau

Amazon RDS effectue des prévérifications avant la mise à niveau afin de vérifier les incompatibilités. Ces incompatibilités varient en fonction de la version de MySQL vers laquelle la mise à niveau est effectuée.

Certaines vérifications préalables sont incluses avec MySQL et d'autres ont été spécifiquement créées par l'équipe Amazon RDS. Pour de plus amples informations sur les vérifications préalables fournies par MySQL, veuillez consulter Upgrade Checker Utility.

Les vérifications préalables s'exécutent avant que l'instance de base de données soit arrêtée pour la mise à niveau, ce qui signifie que leur exécution n'entraîne aucun temps d'arrêt. Si les vérifications préalables trouvent une incompatibilité, Amazon RDS annule automatiquement la mise à niveau avant que l'instance de base de données soit arrêtée. Amazon RDS génère également un événement pour l'incompatibilité. Pour de plus amples informations sur les événements Amazon RDS, veuillez consulte Utilisation des notifications d'RDSévénements Amazon.

Amazon RDS enregistre des informations détaillées sur chaque incompatibilité dans le fichier journal `PrePatchCompatibility.log`. Dans la plupart des cas, l'entrée de journal inclut un lien vers la documentation MySQL permettant de corriger l'incompatibilité. Pour de plus amples informations sur l'affichage des fichiers journaux, veuillez consulter Liste et affichage des fichiers journaux de base de données.

En raison de la nature des prévérifications, elles analysent les objets de votre base de données. L'analyse entraîne la consommation de ressources et augmente le temps nécessaire pour la mise à niveau.

Rubriques

- Prévérifications pour les mises à niveau de MySQL 8.0 à 8.4
- Vérifications préalables aux mises à jour de MySQL 5.7 vers 8.0

Prévérifications pour les mises à niveau de MySQL 8.0 à 8.4

MySQL 8.4 inclut un certain nombre d'incompatibilités avec MySQL 8.0. Ces incompatibilités peuvent provoquer des problèmes lors d'une mise à niveau de MySQL 8.0 vers MySQL 8.4. Une certaine préparation est donc nécessaire sur votre base de données pour garantir la réussite de la mise à niveau. La liste suivante est une liste généralisée de ces incompatibilités :

- Les tables ne doivent pas utiliser de fonctions ou de types de données obsolètes.

- Les déclencheurs ne doivent pas avoir de definer manquant ou vide, ni de contexte de création invalide.
- Il ne doit y avoir aucune violation de mot clé ou de mot réservé. Certains mots clés peuvent être réservés dans MySQL 8.4 alors qu'ils n'étaient pas réservés auparavant.

  Pour en savoir plus, consultez Mots clés et mots réservés dans la documentation MySQL.
- Aucune table de la base de données `mysql` système MySQL 8.0 ne doit porter le même nom qu'une table utilisée par le dictionnaire de données MySQL 8.4.
- Aucun mode SQL obsolète ne doit être défini dans la configuration variable de votre système `sql_mode`.
- Aucune table ni procédure stockée avec des éléments de colonne `ENUM` ou `SET` individuels ne doit dépasser 255 caractères ou 1 020 octets de longueur.
- Votre installation de MySQL 8.0 ne doit pas utiliser de fonctionnalités qui ne sont pas prises en charge dans MySQL 8.4.

  Pour plus d'informations, consultez la section Fonctionnalités supprimées dans MySQL 8.4 dans la documentation MySQL.
- Aucun nom de contrainte de clé étrangère ne doit dépasser 64 caractères.
- Pour une meilleure prise en charge de l'Unicode, consultez les informations suivantes :
  - Envisagez de convertir les objets qui utilisent le jeu de `utf8mb3` caractères pour utiliser le jeu de `utf8mb4` caractères. Le jeu de caractères `utf8mb3` est obsolète.
  - Envisagez d'utiliser `utf8mb4` pour les références au jeu de caractères au lieu de`utf8`, car il `utf8` s'agit actuellement d'un alias pour le `utf8mb3` jeu de caractères. Si possible, passez d'`utf8mb4`abord `utf8` à la base de données, puis mettez à niveau votre base de données.
  - Comme les clients plus anciens peuvent recevoir une erreur de jeu de caractères inconnue pour`utf8mb3`, mettez à niveau vos clients de base de données avant de mettre à niveau votre base de données.

  Pour en savoir plus, consultez Le jeu de caractères utf8mb3 (encodage Unicode 3 octets en UTF-8) dans la documentation MySQL.

  Pour modifier les jeux de caractères, vous pouvez effectuer manuellement une sauvegarde, une restauration et une réplication de votre base de données. Vous pouvez également utiliser Amazon RDS Blue/Green Deployments. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Lorsque vous lancez une mise à niveau de MySQL 8.0 vers la version 8.4, Amazon RDS exécute automatiquement des prévérifications pour détecter ces incompatibilités. Pour plus d'informations sur la mise à niveau vers MySQL 8.4, consultez la section Mise à niveau de MySQL dans la documentation MySQL.

Ces vérifications préalables sont obligatoires. Vous ne pouvez pas choisir de les ignorer. Elles offrent les avantages suivants :

- Elles vous permettent d'éviter les temps d'arrêts non planifiés pendant la mise à niveau.
- Si vous avez des incompatibilités, Amazon RDS empêche la mise à niveau et vous fournit un journal pour que vous en sachiez plus à leur sujet. Vous pouvez ensuite utiliser le journal pour préparer votre base de données pour la mise à niveau vers MySQL 8.4 en réduisant les incompatibilités. Pour des informations détaillées sur la suppression des incompatibilités, consultez Préparation de votre installation pour la mise à niveau dans la documentation MySQL.

Vérifications préalables aux mises à jour de MySQL 5.7 vers 8.0

MySQL 8.0 inclut plusieurs incompatibilités avec MySQL 5.7. Ces incompatibilités peuvent provoquer des problèmes lors d'une mise à niveau de MySQL 5.7 vers MySQL 8.0. Une certaine préparation est donc nécessaire sur votre base de données pour garantir la réussite de la mise à niveau. La liste suivante est une liste généralisée de ces incompatibilités :

- Les tables ne doivent pas utiliser de fonctions ou de types de données obsolètes.
- Il ne doit y avoir aucun fichier *.frm orphelin.
- Les déclencheurs ne doivent pas avoir de definer manquant ou vide, ni de contexte de création invalide.
- Aucune table partitionnée ne doit utiliser de moteur de stockage dépourvu de prise en charge native du partitionnement.
- Il ne doit y avoir aucune violation de mot clé ou de mot réservé. Certains mots clés doivent être réservés dans MySQL 8.0 alors qu'ils ne l'étaient pas par le passé.

  Pour en savoir plus, consultez Mots clés et mots réservés dans la documentation MySQL.
- Aucune table de la base de données du système `mysql`dans MySQL 5.7 ne doit avoir le même nom que la table utilisée dans le dictionnaire de données MySQL 8.0.
- Aucun mode SQL obsolète ne doit être défini dans la configuration variable de votre système `sql_mode`.

- Aucune table ni procédure stockée avec des éléments de colonne `ENUM` ou `SET` individuels ne doit dépasser 255 caractères ou 1 020 octets de longueur.
- Avant de mettre à niveau vers la version MySQL 8.0.13 ou une version ultérieure, aucune partition de table ne doit se trouver dans les espaces de stockage InnoDB partagés.
- Aucune requête ni définition de programme de la version MySQL 8.0.12 ou d'une version antérieure ne doit utiliser de qualificateur `ASC` ou `DESC` pour les clauses `GROUP BY`.
- Votre installation MySQL ne doit pas utiliser de fonctionnalités incompatibles avec MySQL 8.0.

  Pour en savoir plus, consultez Fonctionnalités supprimées dans MySQL 8.0 dans la documentation MySQL.
- Aucun nom de contrainte de clé étrangère ne doit dépasser 64 caractères.
- Pour une meilleure prise en charge de l'Unicode, consultez les informations suivantes :
  - Envisagez de convertir les objets qui utilisent le jeu de `utf8mb3` caractères pour utiliser le jeu de `utf8mb4` caractères. Le jeu de caractères `utf8mb3` est obsolète.
  - Envisagez d'utiliser `utf8mb4` pour les références au jeu de caractères au lieu de`utf8`, car il `utf8` s'agit actuellement d'un alias pour le `utf8mb3` jeu de caractères. Si possible, passez d'`utf8mb4`abord `utf8` à la base de données, puis mettez à niveau votre base de données.
  - Comme les clients plus anciens peuvent recevoir une erreur de jeu de caractères inconnue pour`utf8mb3`, mettez à niveau vos clients de base de données avant de mettre à niveau votre base de données.

  Pour en savoir plus, consultez Le jeu de caractères utf8mb3 (encodage Unicode 3 octets en UTF-8) dans la documentation MySQL.

  Pour modifier les jeux de caractères, vous pouvez effectuer manuellement une sauvegarde, une restauration et une réplication de votre base de données. Vous pouvez également utiliser Amazon RDS Blue/Green Deployments. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Lorsque vous démarrez une mise à niveau de MySQL 5.7 vers 8.0, Amazon RDS exécute automatiquement des vérifications préalables pour détecter ces incompatibilités. Pour plus d'informations sur la mise à niveau vers MySQL 8.0, consultez Mise à niveau de MySQL dans la documentation MySQL.

Ces vérifications préalables sont obligatoires. Vous ne pouvez pas choisir de les ignorer. Elles offrent les avantages suivants :

- Elles vous permettent d'éviter les temps d'arrêts non planifiés pendant la mise à niveau.
- Si vous avez des incompatibilités, Amazon RDS empêche la mise à niveau et vous fournit un journal pour que vous en sachiez plus à leur sujet. Vous pouvez ensuite utiliser le journal pour préparer votre base de données pour la mise à niveau vers la version 8.0 de MySQL en réduisant ces incompatibilités. Pour obtenir des informations détaillées sur la suppression des incompatibilités, consultez Préparation de votre installation pour la mise à niveau (langue française non garantie) dans la documentation MySQL et Mise à niveau vers MySQL 8.0 ? Ce que vous devez savoir… (langue française non garantie) sur le blog MySQL Server.

Annulation après échec de la mise à niveau

Lorsque vous mettez à niveau une instance de base de données de MySQL 5.7 vers MySQL version 8.0 ou de MySQL version 8.0 vers 8.4, la mise à niveau peut échouer. Elle peut échouer en particulier si le dictionnaire de données contient des incompatibilités qui n'ont pas été détectées alors des vérifications préalables. Dans ce cas, la base de données ne démarre pas correctement dans la nouvelle version MySQL 8.0 ou 8.4. À ce stade, Amazon RDS annule les modifications effectuées pour la mise à niveau. Après le rollback, l'instance de base de données MySQL exécute la version d'origine :

- MySQL version 8.0 (pour une restauration depuis MySQL 8.4)
- MySQL version 5.7 (pour une restauration depuis MySQL 8.0)

Lorsqu'une mise à niveau échoue et qu'elle est annulée, Amazon RDS génère un événement avec l'ID RDS-EVENT-0188.

Généralement, une mise à niveau échoue car il existe des incompatibilités dans les métadonnées entre les bases de données de votre instance de base de données et la version MySQL cible. En cas d'échec d'une mise à niveau, vous pouvez afficher les détails de ces incompatibilités dans le fichier `upgradeFailure.log`. Vous devez résoudre les incompatibilités avant de répéter la tentative de mise à niveau.

Lors d'une tentative de mise à niveau infructueuse et d'une restauration, votre instance de base de données est redémarrée. Tous les changements de paramètres en attente sont appliqués lors du redémarrage et sont maintenus après la restauration.

Pour plus d'informations sur la mise à niveau vers MySQL 8.0, consultez les rubriques suivantes de la documentation MySQL :

- Préparation de votre installation pour la mise à niveau
- Mettre à niveau vers MySQL 8.0 ? Voici ce que vous devez savoir…

Pour plus d'informations sur la mise à niveau vers MySQL 8.4, consultez Préparation de votre installation pour la mise à niveau dans la documentation MySQL.

Tester et effectuer ma RDS mise SQL à niveau

Avant d'effectuer une mise à niveau de version majeure sur votre instance de base de données, testez soigneusement la compatibilité de votre base de données avec la nouvelle version. En outre, vous devez tester soigneusement la compatibilité de toutes les applications qui accèdent à la base de données avec la nouvelle version. Nous vous recommandons d'utiliser la procédure suivante.

Pour tester une mise à niveau de version majeure

1. Passez en revue la documentation de la mise à niveau pour la nouvelle version du moteur de base de données afin de voir si des problèmes de compatibilité peuvent affecter votre base de données ou vos applications :
   - Changements dans My SQL 5.7
   - Changements dans My SQL 8.0
   - Changements apportés à My SQL 8.4
2. Si votre instance de base de données est membre d'un groupe de paramètres de base de données personnalisé, créez un nouveau groupe de paramètres de base de données avec vos paramètres existants, qui soit compatible avec la nouvelle version majeure. Spécifiez le nouveau groupe de paramètres de base de données lorsque vous mettez à niveau votre instance de test afin que les tests de mise à niveau puissent vérifier son bon fonctionnement. Pour de plus amples informations sur la création d'un groupe de paramètres de base de données, veuillez consulter Groupes de paramètres pour (Amazon RDS).
3. Créez un instantané de base de données de l'instance de base de données à mettre à niveau. Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
4. Restaurez l'instantané de base de données pour créer une nouvelle instance de base de données de test. Pour plus d'informations, consultez Restauration vers une instance de base de données.

5. Modifiez cette nouvelle instance de base de données de test pour la mettre à niveau vers la nouvelle version, en utilisant l'une des méthodes détaillées plus loin. Si vous avez créé un groupe de paramètres à l'étape 2, spécifiez ce groupe de paramètres.
6. Évaluez le stockage utilisé par l'instance mise à niveau pour déterminer si la mise à niveau requiert un stockage supplémentaire.
7. Exécutez sur l'instance de base de données mise à niveau autant de tests d'assurance qualité que nécessaire pour garantir que votre base de données et votre application fonctionnent correctement avec la nouvelle version. Implémentez tous les nouveaux tests requis pour évaluer l'impact des éventuels problèmes de compatibilité que vous avez identifiés à l'étape 1. Testez toutes les fonctions et procédures stockées. Dirigez les versions de test de vos applications vers l'instance de base de données mise à niveau.
8. En cas de succès de tous les tests, effectuez la mise à niveau sur votre instance de base de données de production. Nous vous recommandons de ne pas autoriser les opérations d'écriture sur l'instance de base de données tant que vous n'avez pas confirmé que tout fonctionne correctement.

Mise à niveau d'une instance de base de données MySQL

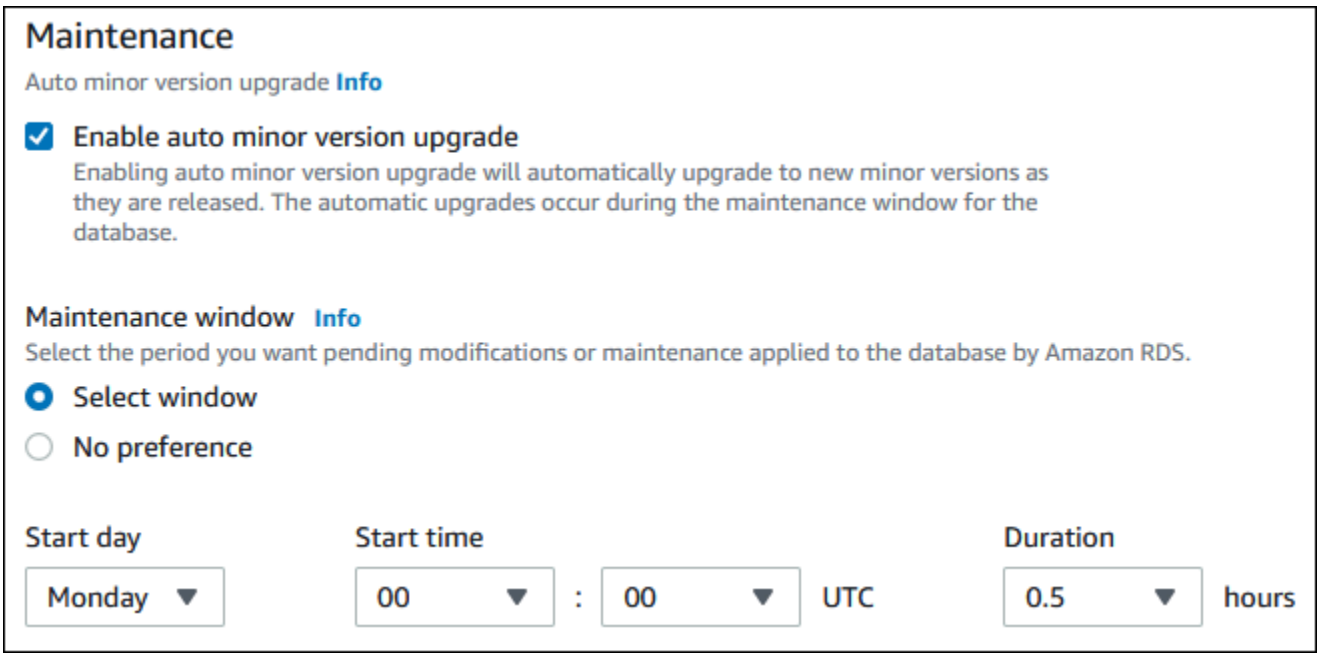
Pour de plus amples informations sur la mise à niveau manuelle ou automatique d'une instance de base de données MySQL, veuillez consulter Mise à niveau de la version du moteur d'une instance de base de données.

Mises à niveau automatiques des versions mineures pour RDS for My SQL

Si vous spécifiez les paramètres suivants lors de la création ou de la modification d'une instance de base de données, celle-ci peut être mise à niveau automatiquement.

- Le paramètre Mise à niveau automatique des versions mineures est activé.
- Le paramètre Période de conservation des sauvegardes est supérieur à 0.

Dans le AWS Management Console, ces paramètres se trouvent sous Configuration supplémentaire. L'image suivante illustre le réglage Auto minor version upgrade (Mise à niveau automatique de versions mineures).



Pour plus d'informations sur ces paramètres, consultez la page Paramètres des instances de base de données.

Dans certains cas, RDS pour Mes versions SQL majeures, dans d'autres Régions AWS, une version mineure est désignée RDS comme version de mise à niveau automatique. Une fois qu'une version mineure a été testée et approuvée par AmazonRDS, la mise à niveau de la version mineure s'effectue automatiquement pendant votre période de maintenance. RDSne définit pas automatiquement les nouvelles versions mineures publiées comme version de mise à niveau automatique. Avant qu'RDS ne désigne une version de mise à niveau automatique plus récente, plusieurs critères sont pris en compte, à savoir :

- Problèmes de sécurité connus
- Bugs dans la version Ma SQL communauté
- Stabilité globale du parc depuis la publication de la version mineure

Vous pouvez exécuter la AWS CLI commande suivante pour déterminer la version cible de mise à niveau mineure automatique actuelle pour une version My SQL minor spécifiée dans une version spécifique Région AWS.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine mysql \
```

```
--engine-version minor_version \
--region region \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine mysql ^
--engine-version minor_version ^
--region region ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output text
```

Par exemple, la AWS CLI commande suivante détermine la cible de mise à niveau mineure automatique pour My SQL minor version 8.0.11 dans l'est des États-Unis (Ohio) (Région AWS us-east-2).

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine mysql \
--engine-version 8.0.11 \
--region us-east-2 \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output table
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine mysql ^
--engine-version 8.0.11 ^
--region us-east-2 ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output table
```

Votre sortie est similaire à ce qui suit.

```
---------------------------------
|    DescribeDBEngineVersions    |
+--------------+-----------------+
|  AutoUpgrade |  EngineVersion  |
+--------------+-----------------+
|  False       |  8.0.15         |
|  False       |  8.0.16         |
|  False       |  8.0.17         |
|  False       |  8.0.19         |
|  False       |  8.0.20         |
|  False       |  8.0.21         |
|  True        |  8.0.23         |
|  False       |  8.0.25         |
+--------------+-----------------+
```

Dans cet exemple, la `AutoUpgrade` valeur correspond `True` à My SQL version 8.0.23. La cible de mise à niveau mineure automatique est donc Ma SQL version 8.0.23, qui est surlignée dans le résultat.

Une instance My SQL DB est automatiquement mise à niveau pendant votre période de maintenance si les critères suivants sont remplis :

- Le paramètre Mise à niveau automatique des versions mineures est activé.
- Le paramètre Période de conservation des sauvegardes est supérieur à 0.
- L'instance de base de données exécute une version mineure du moteur de base de données qui est inférieure à la version mineure de la mise à niveau automatique actuelle.

Pour plus d'informations, consultez Mise à niveau automatique de la version mineure du moteur.

Utilisation d'une réplique en lecture pour réduire les temps d'arrêt lors de la mise à niveau d'une base de données RDS for MySQL

Dans la plupart des cas, un déploiement bleu/vert est la meilleure option pour réduire les temps d'arrêt lors de la mise à niveau d'une instance de base de données MySQL. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.

Si vous ne pouvez pas utiliser un déploiement bleu/vert et que votre instance de base de données MySQL est en cours d'utilisation avec une application de production, vous pouvez utiliser la

procédure suivante pour mettre à niveau la version de la base de données pour votre instance de base de données. Cette procédure peut réduire les temps d'arrêt de votre application.

En utilisant un réplica en lecture, vous pouvez effectuer la plupart des étapes de maintenance à l'avance et ainsi réduire les modifications nécessaires lors d'une panne réelle. Cette technique vous permet de tester et de préparer la nouvelle instance de base de données sans apporter de modifications à votre instance de base de données existante.

La procédure suivante illustre un exemple de mise à niveau de MySQL version 5.7 vers MySQL version 8.0. Vous pouvez utiliser les mêmes étapes générales pour des mises à niveau vers d'autres versions majeures. Vous pouvez utiliser les mêmes étapes générales pour des mises à niveau vers d'autres versions majeures.

> **Note**
>
> Lorsque vous effectuez une mise à niveau de MySQL 5.7 vers MySQL version 8.0, ou de MySQL version 8.0 vers MySQL version 8.4, effectuez les prévérifications avant d'effectuer la mise à niveau. Pour plus d'informations, consultez Vérifications préalables aux mises à jour de MySQL 5.7 vers 8.0 et Prévérifications pour les mises à niveau de MySQL 8.0 à 8.4.

Pour mettre à niveau une base de données MySQL alors qu'une instance de base de données est en cours d'utilisation

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Créez un réplica en lecture de votre instance de base de données MySQL 5.7. Ce processus crée une copie pouvant être mise à niveau de votre base de données. D'autres réplicas en lecture de l'instance de base de données peuvent également exister.
    a. Sur la console, choisissez Bases de données, puis sélectionnez l'instance de base de données que vous souhaitez mettre à niveau.
    b. Sous Actions, choisissez Créer des réplicas en lecture.
    c. Spécifiez une valeur dans le champ Identifiant de l'instance de base de données de votre réplica en lecture et assurez-vous que la Classe d'instance de base de données et les autres paramètres correspondent à votre instance de base de données MySQL 5.7.
    d. Choisissez Créer un réplica en lecture.

3. (Facultatif) Lorsque le réplica en lecture a été créé et que le champ État indique Disponible, convertissez le réplica en lecture en déploiement multi-AZ et activez les sauvegardes.

   Par défaut, une réplique en lecture est créée avec les sauvegardes désactivées. Comme la réplique en lecture devient finalement l'instance de base de données de production, il est recommandé de configurer un déploiement multi-AZ et d'activer les sauvegardes.

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de créer.
   b. Sélectionnez Modify.
   c. Dans le champ Déploiement multi-AZ, choisissez Créer une instance de secours.
   d. Dans le champ Backup Retention Period (Période de rétention des sauvegardes), choisissez une valeur positive différente de zéro (par exemple, 3 jours), puis sélectionnez Continue (Continuer).
   e. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.
   f. Choisissez Modifier l'instance DB.

4. Lorsque le champ État du réplica en lecture indique Disponible, procédez à sa mise à niveau vers MySQL 8.0 :

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de créer.
   b. Sélectionnez Modify.
   c. Dans le champ Version du moteur de base de données, choisissez la version de MySQL 8.0 vers laquelle vous souhaitez effectuer la mise à niveau, puis sélectionnez Continuer.
   d. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.
   e. Choisissez Modifier l'instance de base de données pour démarrer la mise à niveau.

5. Lorsque la mise à niveau est terminée et que le statut indique Disponible, vérifiez que la réplique de lecture mise à niveau correspond up-to-date à l'instance de base de données MySQL 5.7 source. Pour vérifier, connectez-vous au réplica en lecture et exécutez la commande `SHOW REPLICA STATUS`. Si le `Seconds_Behind_Master` champ l'est`0`, la réplication l'est up-to-date.

> **Note**
>
> Les versions précédentes de MySQL utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version MySQL antérieure à la version 8.0.23, utilisez alors `SHOW SLAVE STATUS`.

6. (Facultatif) Créez un réplica en lecture de votre réplica en lecture.

   Si vous souhaitez que l'instance de base de données dispose d'un réplica en lecture une fois celle-ci promue en tant qu'instance de base de données autonome, vous pouvez créer le réplica en lecture dès maintenant.

   a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de mettre à niveau.

   b. Sous Actions, choisissez Créer des réplicas en lecture.

   c. Spécifiez une valeur dans le champ Identifiant de l'instance de base de données de votre réplica en lecture et assurez-vous que la Classe d'instance de base de données et les autres paramètres correspondent à votre instance de base de données MySQL 5.7.

   d. Choisissez Créer un réplica en lecture.

7. (Facultatif) Configurez un groupe de paramètres de base de données personnalisé pour le réplica en lecture.

   Si vous souhaitez que l'instance de base de données utilise un groupe de paramètres personnalisé une fois celle-ci promue en tant qu'instance de base de données autonome, vous pouvez créer le groupe de paramètres de base de données dès maintenant et l'associer au réplica en lecture.

   a. Créez un groupe de paramètres de base de données personnalisé pour MySQL 8.0. Pour obtenir des instructions, consultez Création d'un groupe de paramètres de base de données dans RDSAmazon.

   b. Modifiez les paramètres que vous souhaitez modifier dans le groupe de paramètres de base de données fraîchement créé. Pour obtenir des instructions, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

   c. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture.

   d. Sélectionnez Modify.

e. Dans le champ Groupe de paramètres de base de données, choisissez le groupe de paramètres de base de données MySQL 8.0 que vous venez de créer, puis sélectionnez Continuer.

f. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.

g. Choisissez Modifier l'instance de base de données pour démarrer la mise à niveau.

8. Faites de votre réplica en lecture MySQL 8.0 une instance de base de données autonome.

> **Important**
>
> Une fois promu en tant qu'instance de base de données autonome, votre réplica en lecture MySQL 8.0 cesse d'être un réplica de votre instance de base de données MySQL 5.7. Nous vous conseillons d'effectuer la promotion de votre réplica en lecture MySQL 8.0 au cours d'un créneau de maintenance, lorsque votre instance de base de données MySQL 5.7 source est en mode lecture seule et que toutes les opérations d'écriture sont suspendues. Au terme de l'opération de promotion, vous pouvez diriger vos opérations d'écriture vers l'instance de base de données MySQL 8.0 mise à niveau pour veiller à ce qu'aucune opération d'écriture ne se perde.
> En outre, avant la promotion de votre réplica en lecture MySQL 8.0, nous vous conseillons d'effectuer toutes les opérations DDL (Data Definition Language) nécessaires sur votre réplica en lecture MySQL 8.0. Par exemple, la création d'index. Cette approche permet d'éviter tout effet négatif sur les performances du réplica en lecture MySQL 8.0 après sa promotion. Pour promouvoir un réplica en lecture.

a. Sur la console, choisissez Bases de données, puis sélectionnez le réplica en lecture que vous venez de mettre à niveau.

b. Pour Actions, choisissez Promote (Promouvoir).

c. Choisissez Oui pour activer les sauvegardes automatiques pour l'instance du réplica en lecture. Pour plus d'informations, consultez Présentation des sauvegardes.

d. Choisissez Continuer.

e. Choisissez Promouvoir le réplica en lecture.

9. Vous disposez à présent d'une version mise à niveau de votre base de données MySQL. À ce stade, vous pouvez diriger vos applications vers la nouvelle instance de base de données MySQL 8.0.

Mise à niveau d'une version du moteur de snapshots de base de données MySQL

Amazon RDS vous permet de créer un instantané de base de données de volume de stockage de votre instance de base de données MySQL. Lorsque vous créez un instantané de base de données, celui-ci est basé sur la version du moteur utilisée par votre instance de base de données. Vous pouvez mettre à niveau la version du moteur pour vos instantanés de base de données.

Pour RDS for MySQL, vous pouvez mettre à niveau un instantané de la version 5.7 vers la version 8.0, ou un instantané de la version 8.0 vers la version 8.4. Vous pouvez mettre à niveau des instantanés de base de données chiffrés ou non chiffrés.

Pour afficher les versions de moteur disponibles pour votre instantané de base de données RDS pour MySQL, utilisez l' AWS CLI exemple suivant.

```
aws rds describe-db-engine-versions --engine mysql --include-all --engine-
version example-engine-version --query "DBEngineVersions[*].ValidUpgradeTarget[*].
{EngineVersion:EngineVersion}" --output text
```

Si vous ne voyez aucun résultat pour votre instantané, la version de votre moteur est peut-être obsolète. Si la version de votre moteur est obsolète, nous vous recommandons de passer à la dernière cible de mise à niveau de la version majeure ou à l'une des autres cibles de mise à niveau disponibles pour cette version. Pour de plus amples informations, veuillez consulter Options de mise à niveau pour les instantanés de base de données avec des versions de moteur non prises en charge pour RDS for MySQL.

Après avoir restauré un instantané de base de données mis à niveau vers une nouvelle version de moteur, veillez à vérifier que la mise à jour est réussie. Pour de plus amples informations sur une mise à niveau des versions majeures, veuillez consulter the section called “Mises à niveau du moteur de base de données MySQL”. Pour savoir comment restaurer un instantané de base de données, consultez the section called “Restauration vers une instance de base de données”.

> **Note**
>
> Vous ne pouvez pas mettre à niveau les instantanés de base de données automatisés créés au cours du processus de sauvegarde automatique.

Vous pouvez mettre à niveau un instantané de base de données à l'aide de l'API AWS Management Console AWS CLI, ou RDS.

Console

Pour mettre à niveau une version du moteur de snapshots de base de données à l'aide de AWS Management Console, procédez comme suit.

Pour mettre à niveau un instantané de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané que vous souhaitez mettre à niveau.
4. Pour Actions, choisissez Upgrade Snapshot (Mettre à niveau l'instantané). La page Upgrade Snapshot (Mettre à niveau l'instantané) s'affiche.
5. Choisissez la New engine version (Version du nouveau moteur) vers laquelle mettre à niveau.
6. Choisissez Save changes (Enregistrer les changements) pour mettre à niveau l'instantané.

   Pendant le processus de mise à niveau, toutes les actions d'instantané sont désactivées pour l'instantané de base de données. En outre, le statut du snapshot de base de données passe de Disponible à Mise à niveau, puis passe à Actif une fois terminé. Si le snapshot de base de données ne peut pas être mis à niveau en raison de problèmes de corruption du snapshot, le statut passe à Indisponible. Vous ne pouvez pas récupérer l'instantané lorsqu'il a ce statut.

   > **Note**
   >
   > Si la mise à niveau de l'instantané de base de données échoue, l'instantané revient à l'état d'origine avec la version originale.

AWS CLI

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, exécutez la AWS CLI modify-db-snapshotcommande.

Options

- `--db-snapshot-identifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être unique pour un Amazon Resource Name (ARN). Pour plus d'informations, consultez Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `--engine-version` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot \

    --db-snapshot-identifier my_db_snapshot \
    --engine-version new_version
```

Dans Windows:

```
aws rds modify-db-snapshot ^
    --db-snapshot-identifier my_db_snapshot ^
    --engine-version new_version
```

Amazon RDS API

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, appelez l'DBSnapshotopération RDS API Modify.

Paramètres

- `DBSnapshotIdentifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être unique pour un Amazon Resource Name (ARN). Pour plus d'informations, consultez Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `EngineVersion` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Options de mise à niveau pour les instantanés de base de données avec des versions de moteur non prises en charge pour RDS for MySQL

Le tableau suivant indique les versions de moteur vers lesquelles vous pouvez effectuer une mise à niveau à partir d'une version de moteur non prise en charge pour les instantanés de base de données RDS for MySQL.

> **Note**
>
> Il se peut que vous deviez mettre à niveau votre instantané de base de données plusieurs fois pour passer à la version du moteur que vous avez choisie.

Version du moteur de capture d'écran DB	Versions du moteur disponibles pour la mise à niveau
5.5.8	5,5,62, 5,6,51
5,5,12	5,5,62, 5,6,51
5,5,20	5,5,62, 5,6,51
5,5,23	5,5,62, 5,6,51
5,5,25a	5,5,62, 5,6,51
5,5,27	5,5,62, 5,6,51
5,5,31	5,5,62, 5,6,51
5,5,33	5,5,62, 5,6,51
5,5,37	5,5,62, 5,6,51
5,5,38	5,5,62, 5,6,51
5,5,40	5,5,62, 5,6,51
5,5,40a	5,5,62, 5,6,51

Version du moteur de capture d'écran DB	Versions du moteur disponibles pour la mise à niveau
5,5,40b	5,5,62, 5,6,51
5,5,41	5,5,62, 5,6,51
5,5,42	5,5,62, 5,6,51
5.5.59	5,5,62, 5,6,51
5,6,12	5,6,51, 5,7,44
5,6,13	5,6,51, 5,7,44
5,6,17	5,6,51, 5,7,44
5,6,19	5,6,51, 5,7,44
5,6.19a	5,6,51, 5,7,44
5,6,19b	5,6,51, 5,7,44
5,6,21	5,6,51, 5,7,44
5,6,21b	5,6,51, 5,7,44
5,6,22	5,6,51, 5,7,44
5,6,23	5,6,51, 5,7,44
5,6,27	5,6,51, 5,7,44
5,6.27a	5,6,51, 5,7,44
5,7,10	5.7.44, 5.7.44-rds.20240408, 5.7.44-rds.20240529, 5.7.44-rds.20250103, 5.7.44-rds.20250213, 8.0.32, 8,0.33, 8,0.34, 8,0.35, 8,0.36, 8,0.37, 8,0.39, 8,0.40, 8,0.41

Version du moteur de capture d'écran DB	Versions du moteur disponibles pour la mise à niveau
5.7,11	5.7.44, 5.7.44-rds.20240408, 5.7.44-rds.20240529, 5.7.44-rds.20250103, 5.7.44-rds.20250213, 8.0.32, 8,0.33, 8,0.34, 8,0.35, 8,0.36, 8,0.37, 8,0.39, 8,0.40, 8,0.41
5,7,12	5.7.44, 5.7.44-rds.20240408, 5.7.44-rds.20240529, 5.7.44-rds.20250103, 5.7.44-rds.20250213, 8.0.32, 8,0.33, 8,0.34, 8,0.35, 8,0.36, 8,0.37, 8,0.39, 8,0.40, 8,0.41

Importation de données dans une instance de base de données Amazon RDS for MySQL

Vous pouvez utiliser plusieurs techniques pour importer des données dans une instance de base de données RDS for MySQL. La meilleure approche dépend d'un certain nombre de facteurs :

- Source des données
- Quantité de données
- Importation ponctuelle ou continue
- Durée du temps d'arrêt

Si vous migrez également une application avec les données, il est important de prendre en compte le temps d'arrêt.

Le tableau suivant répertorie les techniques d'importation de données dans une instance de base de données RDS pour MySQL :

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Base de données MySQL existante sur site ou sur Amazon EC2	N'importe quel compte	Une seule fois	Momentanée	Créez une sauvegarde de votre base de données sur site, stockez-la sur Amazon S3, puis restituez le fichier de sauvegarde sur une nouvelle instance de base de données Amazon RDS exécutant MySQL.	Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Base de données MySQL existante sur site ou sur Amazon EC2	N'importe quel compte	En continu	Minimale	Configurez la réplication avec une base de données MySQL existante comme source de réplication.	Configuration d'une réplication de position de fichier journal binaire avec une instance source externe Importation de données vers une base de données Amazon RDS for MySQL avec réduction des

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
					temps d'arrêt

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Toute base de données existante	N'importe quel compte	Une seule fois ou en continu	Minimale	AWS Database Migration Service À utiliser pour migrer la base de données avec un temps d'arrêt minimal et, pour de nombreux moteurs de base de données, poursuivre la réplication continue.	Présentation d'AWS Database Migration Service et Utilisation d'une base de données compatible MySQL comme cible pour AWS DMS dans le Guide de l'utilisateur AWS Database Migration Service

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Instance de base de données MySQL existante	N'importe quel compte	Une seule fois ou en continu	Minimale	Créez un réplica en lecture pour la réplication continue. Promouvez le réplica en lecture pour la création unique d'une nouvelle instance de base de données.	Utilisation des réplicas en lecture d'instance de base de données
Base de données MySQL existante	Petite	Une seule fois	Momentanée	Copiez les données directement dans votre instance de base de données MySQL à l'aide d'un utilitaire de ligne de commande.	Importation de données depuis une base de données MySQL externe vers une instance de base de données Amazon RDS for MySQL

Source	Quantité de données	Une seule fois ou en continu	Interruption de l'application	Technique	En savoir plus
Données non stockées dans une base de données existante	Medium	Une seule fois	Momentanée	Créez des fichiers plats et importez-les à l'aide `LOAD DATA LOCAL INFILE` des instructions MySQL.	Importation de données depuis n'importe quelle source vers une instance de base de données Amazon RDS for MySQL

> **Note**
>
> La base de données système `mysql` contient les informations d'authentification et d'autorisation requises pour se connecter à l'instance de base de données et accéder aux données. La suppression, la modification, le renommage ou la troncation de tables, de données ou d'autres contenus de la base de données `mysql` de votre instance de base de données peut produire une erreur et rendre inaccessibles l'instance de base de données et vos données. Dans ce cas, vous pouvez restaurer l'instance de base de données à partir d'un instantané à l'aide du AWS CLI restore-db-instance-from-db-snapshotcommande. Vous pouvez récupérer l'instance de base de données à l'aide du AWS CLI restore-db-instance-to-point-in-timecommande.

Considérations relatives à l'importation de données pour MySQL

Le contenu suivant contient des informations techniques relatives au chargement de données dans MySQL. Ce contenu s'adresse aux utilisateurs familiarisés avec l'architecture du serveur MySQL.

Journalisation binaire

L'activation de la journalisation binaire réduit les performances de chargement des données et nécessite jusqu'à quatre fois plus d'espace disque que la journalisation désactivée. La taille de transaction utilisée pour charger les données influe directement sur les performances du système et les besoins en espace disque. Les transactions plus importantes nécessitent davantage de ressources.

Taille de la transaction

La taille des transactions influence les aspects suivants des chargements de données MySQL :

- Consommation de ressources
- Utilisation de l'espace disque
- Processus de CV
- Il est temps de récupérer
- Format d'entrée (fichiers plats ou SQL)

Cette section décrit comment la taille de la transaction influe sur la journalisation binaire. En outre, elle plaide en faveur de la désactivation de cette même journalisation lors de chargements de données volumineux. Vous pouvez activer et désactiver la journalisation binaire en définissant la période de conservation automatique des sauvegardes Amazon RDS. Les valeurs différentes de zéro activent la journalisation binaire, tandis que la valeur zéro la désactive. Pour de plus amples informations, veuillez consulter Période de rétention des sauvegardes.

Cette section décrit également l'impact des transactions importantes sur InnoDB et explique pourquoi il est important de réduire la taille des transactions.

Petites transactions

Pour les petites transactions, la journalisation binaire multiplie par deux le nombre d'écritures sur disque requises pour charger les données. Elle peut ainsi affecter gravement les performances

des autres sessions de base de données et accroître le temps requis pour charger les données. La dégradation subie dépend en partie des facteurs suivants :

- Taux de téléchargement
- Autres activités de base de données ayant lieu pendant le chargement
- Capacité de votre instance de base de données Amazon RDS

Les journaux binaires consomment également un espace disque à peu près égal à la quantité de données chargées jusqu'à ce que les journaux soient sauvegardés et supprimés. Amazon RDS minimise ce problème en sauvegardant et en supprimant fréquemment les journaux binaires.

Transactions volumineuses

Pour les transactions importantes, la journalisation binaire triple les IOPS et l'utilisation du disque pour les raisons suivantes :

- Le cache du journal binaire stocke temporairement les données de transaction sur le disque.
- Ce cache augmente avec la taille de la transaction, ce qui consomme de l'espace disque.
- Lorsque la transaction (validation ou annulation) est terminée, le système copie le cache dans le journal binaire.

Ce processus crée trois copies des données :

- Les données d'origine
- Le cache sur le disque
- La dernière entrée du journal binaire

Chaque opération d'écriture entraîne des E/S supplémentaires, ce qui a un impact supplémentaire sur les performances.

De ce fait, la journalisation binaire nécessite trois fois plus d'espace disque que la journalisation désactivée. Par exemple, le chargement de 10 GiB de données en une seule transaction crée trois copies :

- 10 GiB pour les données du tableau
- 10 GiB pour le cache du journal binaire

- 10 GiB pour le fichier journal binaire

L'espace disque temporaire total requis est de 30 GiB.

Considérations importantes relatives à l'espace disque :

- Le fichier de cache est conservé jusqu'à la fin de la session ou jusqu'à ce qu'une nouvelle transaction crée un autre cache.
- Le journal binaire est conservé jusqu'à ce qu'il soit sauvegardé et peut contenir 20 GiB (cache et journal) pendant une période prolongée.

Si vous avez l'`LOAD DATA LOCAL INFILE`habitude de charger les données, Data Recovery crée une quatrième copie au cas où la base de données doive être restaurée à partir d'une sauvegarde effectuée avant le chargement. Pendant le récupération, MySQL extrait les données du journal binaire dans un fichier plat. MySQL s'exécute ensuite`LOAD DATA LOCAL INFILE`. Sur la base de l'exemple précédent, cette restauration nécessite un espace disque temporaire total de 40 Go, soit 10 Go chacun pour la table, le cache, le journal et le fichier local. Sans au moins 40 GiB d'espace disque disponible, la restauration échoue.

Optimisation des charges de données volumineuses

Pour les chargements de données importants, désactivez la journalisation binaire afin de réduire les frais généraux et les besoins en espace disque. Vous pouvez désactiver la journalisation binaire en définissant la période de conservation des sauvegardes sur 0. Une fois le chargement terminé, restaurez la période de conservation des sauvegardes à la valeur appropriée différente de zéro. Pour plus d'informations, reportez-vous Modification d'une RDS instance de base de données Amazon à la section « Durée de conservation des sauvegardes » dans le tableau des paramètres.

> **Note**
>
> Si l'instance de base de données est une instance de base de données source pour les répliques de lecture, vous ne pouvez pas définir la période de rétention des sauvegardes sur 0.

Avant de charger les données, nous vous recommandons de créer un instantané de base de données. Pour de plus amples informations, veuillez consulter Gestion des sauvegardes manuelles.

InnoDB

Les informations suivantes sur les options d'annulation de journalisation et de restauration permettent de limiter les transactions InnoDB afin d'optimiser les performances de la base de données.

Comprendre la journalisation des annulations par InnoDB

L'annulation est un mécanisme de journalisation qui permet l'annulation des transactions et prend en charge le contrôle de simultanéité multiversion (MVCC).

Pour MySQL 5.7 et versions antérieures, les journaux d'annulation sont stockés dans le tablespace du système InnoDB (généralement ibdata1) et sont conservés jusqu'à ce que le thread de purge les supprime. Par conséquent, des transactions de chargement de données importantes peuvent entraîner une augmentation importante de l'espace disque logique du système et consommer de l'espace disque que vous ne pouvez récupérer que si vous recréez la base de données.

Pour toutes les versions de MySQL, le thread de purge doit attendre que la transaction active la plus ancienne soit validée ou annulée pour supprimer tout journal d'annulation. Si la base de données traite d'autres transactions pendant le chargement, leurs journaux d'annulation s'accumulent également et ne peuvent pas être supprimés, même si les transactions sont validées et qu'aucune autre transaction n'a besoin des journaux d'annulation pour MVCC. Dans ce cas, toutes les transactions, y compris les transactions en lecture seule, ralentissent. Ce ralentissement se produit parce que toutes les transactions accèdent à toutes les lignes modifiées par toutes les transactions, et pas seulement par les transactions de chargement. En effet, les transactions doivent parcourir les journaux d'annulation que les transactions de chargement de longue durée ont empêchées d'être purgées lors du nettoyage du journal d'annulation. Cela affecte les performances de toute opération accédant aux lignes modifiées.

Options de récupération des transactions InnoDB

Bien qu'InnoDB optimise les opérations de validation, les annulations de transactions importantes sont lentes. Pour accélérer la restauration, effectuez une point-in-time restauration ou restaurez un instantané de base de données. Pour plus d’informations, consultez Point-in-time rétablissement et Restauration vers une instance de base de données.

Formats d'importation de données

MySQL prend en charge deux formats d'importation de données : les fichiers plats et le SQL. Passez en revue les informations relatives à chaque format afin de déterminer l'option la mieux adaptée à vos besoins.

Fichiers plats

Pour les petites transactions, chargez des fichiers plats avec`LOAD DATA LOCAL INFILE`. Ce format d'importation de données peut offrir les avantages suivants par rapport à l'utilisation de SQL :

- Moins de trafic réseau
- Réduction des coûts de transmission de données
- Diminution des frais de traitement des bases
- Traitement plus rapide

`LOAD DATA LOCAL INFILE`charge l'intégralité du fichier plat en une seule transaction. Réduisez la taille de chaque fichier pour bénéficier des avantages suivants :

- Fonctionnalité de reprise : vous pouvez suivre les fichiers qui ont été chargés. Si un problème survient pendant le chargement, vous pouvez reprendre là où vous vous étiez arrêté. Il se peut que vous deviez retransmettre certaines données vers Amazon RDS, mais dans le cas de petits fichiers, le volume retransmis est minime.
- Chargement parallèle des données : si vous disposez de suffisamment d'IOPS et de bande passante réseau pour le chargement d'un seul fichier, le chargement en parallèle peut vous faire gagner du temps.
- Contrôle du taux de charge : si le chargement de vos données a un impact négatif sur les autres processus, vous pouvez contrôler le taux de charge en augmentant l'intervalle entre les fichiers.

Les transactions importantes réduisent les avantages liés `LOAD DATA LOCAL INFILE` à l'importation de données. Lorsque vous ne pouvez pas diviser une grande quantité de données en fichiers plus petits, pensez à utiliser le langage SQL.

SQL

Le SQL présente un avantage majeur par rapport aux fichiers plats : vous pouvez facilement réduire la taille des transactions. Cependant, le chargement du SQL peut prendre beaucoup plus de temps que celui des fichiers plats. De plus, après un échec, il peut être difficile de déterminer où reprendre : vous ne pouvez pas redémarrer les fichiers mysqldump. En cas d'échec lors du chargement d'un fichier mysqldump, vous devez modifier ou remplacer le fichier avant que le chargement ne puisse reprendre. Ou bien, après avoir corrigé la cause de l'échec, vous pouvez restaurer le fichier à l'endroit où il se trouvait avant le chargement et renvoyer le fichier. Pour de plus amples informations, veuillez consulter Point-in-time rétablissement.

Utilisation des instantanés de base de données Amazon RDS pour les points de contrôle de base de données

Si vous chargez des données sur de longues durées, par exemple des heures ou des jours, sans journalisation binaire, utilisez des instantanés de base de données pour fournir des points de contrôle périodiques en matière de sécurité des données. Chaque instantané de base de données crée une copie cohérente de votre instance de base de données qui sert de point de restauration lors de défaillances du système ou d'événements de corruption de données. Les instantanés de base de données étant rapides, les points de contrôle fréquents ont un impact minimal sur les performances de chargement. Vous pouvez supprimer des instantanés de base de données précédents sans affecter la durabilité ou les capacités de restauration de la base de données. Pour plus d'informations sur les instantanés de base de données, consultezGestion des sauvegardes manuelles.

Réduction des temps de chargement des bases de données

Les éléments suivants constituent des conseils supplémentaires pour réduire les temps de chargement :

- Créez tous les index secondaires avant de charger les données dans les bases de données MySQL. Contrairement aux autres systèmes de base de données, MySQL reconstruit l'intégralité de la table lors de l'ajout ou de la modification d'index secondaires. Ce processus crée une nouvelle table avec les modifications d'index, copie toutes les données et supprime la table d'origine.
- Chargez les données dans l'ordre des clés primaires. Pour les tables InnoDB, cela peut réduire les temps de chargement de 75 % à 80 % et la taille des fichiers de données de 50 %.
- Désactivez les contraintes liées aux clés étrangères en définissant `foreign_key_checks` sur`0`. Cela est souvent nécessaire pour les fichiers plats chargés avec`LOAD DATA LOCAL INFILE`. Quel que soit le chargement, la désactivation des vérifications par clé étrangère accélère le chargement des données. Une fois le chargement terminé, réactivez les contraintes en configurant `1` et `foreign_key_checks` en vérifiant les données.
- Chargez les données en parallèle, sauf si vous approchez d'une limite de ressources. Pour permettre le chargement simultané sur plusieurs segments de table, utilisez des tables partitionnées le cas échéant.
- Pour réduire la charge d'exécution du code SQL, combinez plusieurs `INSERT` instructions en `INSERT` opérations uniques à valeurs multiples. `mysqldump`implémente automatiquement cette optimisation.

- Réduisez les opérations d'E/S du journal InnoDB en `innodb_flush_log_at_trx_commit` réglant sur. `0` Une fois le chargement terminé, restaurez `innodb_flush_log_at_trx_commit` vers`1`.

  > **Warning**
  >
  > Le réglage `innodb_flush_log_at_trx_commit` sur `0` oblige InnoDB à vider ses journaux toutes les secondes plutôt qu'à chaque validation. Ce paramètre améliore les performances mais peut entraîner la perte de transactions en cas de défaillance du système.

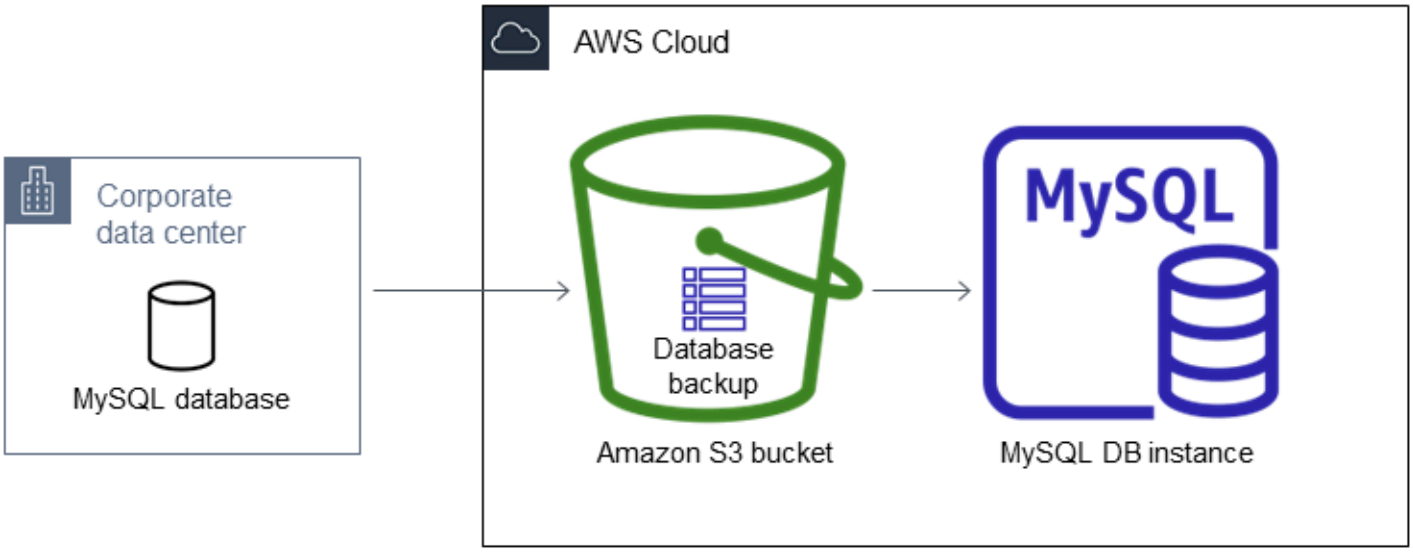
- Si vous chargez des données dans une instance de base de données qui ne possède pas de répliques de lecture, définissez `sync_binlog` cette valeur sur. `0` Une fois le chargement terminé, restaurez `sync_binlog parameter` vers`1`.
- Chargez les données dans une instance de base de données mono-AZ avant de convertir l'instance de base de données en un déploiement multi-AZ. Si l'instance de base de données utilise déjà un déploiement multi-AZ, nous ne recommandons pas de passer à un déploiement mono-AZ pour le chargement des données. Cela n'apporte que des améliorations marginales.

Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL

Amazon RDS prend en charge l'importation de bases de données MySQL avec des fichiers de sauvegarde. Vous pouvez créer une sauvegarde de votre base de données, stocker le fichier de sauvegarde sur Amazon S3, puis restaurer le fichier de sauvegarde sur une nouvelle instance de base de données Amazon RDS exécutant MySQL. Amazon RDS prend en charge l'importation de fichiers de sauvegarde depuis Amazon S3 dans l'ensemble Régions AWS.

Le scénario décrit dans cette section restaure une sauvegarde d'une base de données sur site. Tant que la base de données est accessible, vous pouvez utiliser cette technique pour les bases de données situées sur d'autres sites, tels qu'Amazon EC2 ou d'autres services cloud.

Le schéma suivant illustre le scénario pris en charge.



Si votre base de données locale peut être hors ligne pendant que vous créez, copiez et restaurez des fichiers de sauvegarde, nous vous recommandons d'utiliser des fichiers de sauvegarde pour importer votre base de données sur Amazon RDS. Si votre base de données ne peut pas être hors ligne, vous pouvez utiliser l'une des méthodes suivantes :

- Journaux binaires : importez d'abord des fichiers de sauvegarde depuis Amazon S3 et vers Amazon RDS, comme expliqué dans cette rubrique. Utilisez ensuite la réplication du journal binaire (binlog) pour mettre à jour votre base de données. Pour de plus amples informations, veuillez consulter Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.
- AWS Database Migration Service— AWS Database Migration Service À utiliser pour migrer votre base de données vers Amazon RDS. Pour plus d'informations, voir Qu'est-ce que c'est AWS Database Migration Service ?

Présentation de la configuration pour importer des fichiers de sauvegarde d'Amazon S3 vers Amazon RDS

Pour importer des fichiers de sauvegarde d'Amazon S3 vers Amazon RDS, vous avez besoin des composants suivants :

- Un compartiment Amazon S3 pour stocker vos fichiers de sauvegarde.

Si vous possédez déjà un compartiment Amazon S3, vous pouvez l'utiliser. Si vous n'avez pas de compartiment Amazon S3, créez-en un nouveau. Pour de plus amples informations, veuillez consulter Créer un compartiment.

- Une sauvegarde de votre base de données sur site créée par XtraBackup Percona.

  Pour de plus amples informations, veuillez consulter Création de la sauvegarde de votre base de données.

- Rôle AWS Identity and Access Management (IAM) permettant à Amazon RDS d'accéder au compartiment S3.

  Si vous possédez déjà un rôle IAM, vous pouvez l'utiliser et y associer des politiques de confiance et d'autorisation. Pour de plus amples informations, veuillez consulter Création manuelle d'un rôle IAM.

  Si vous n'avez pas de rôle IAM, deux options s'offrent à vous :

  - Vous pouvez créer manuellement un nouveau rôle IAM. Pour de plus amples informations, veuillez consulter Création manuelle d'un rôle IAM.
  - Vous pouvez choisir qu'Amazon RDS crée un nouveau rôle IAM pour vous. Si vous souhaitez qu'Amazon RDS crée un nouveau rôle IAM pour vous, suivez la procédure qui utilise la section AWS Management Console inPour importer des données à partir d'Amazon S3 vers une nouvelle instance de base de données MySQL.

Création de la sauvegarde de votre base de données

Utilisez le XtraBackup logiciel Percona pour créer votre sauvegarde. Nous vous recommandons d'utiliser la dernière version de Percona XtraBackup. Vous pouvez installer Percona XtraBackup à partir des téléchargements de logiciels sur le site Web de Percona.

> **Warning**
>
> Lors de la création d'une sauvegarde de base de données, XtraBackup vous pouvez enregistrer les informations d'identification dans le fichier xtrabackup_info. Assurez-vous que le `tool_command` paramètre du fichier xtrabackup_info ne contient aucune information sensible.

La XtraBackup version de Percona que vous utilisez dépend de la version de MySQL que vous sauvegardez.

- MySQL 8.4 — Utilisez la XtraBackup version 8.4 de Percona.
- MySQL 8.0 — Utilisez Percona XtraBackup version 8.0.

> **Note**
>
> Percona XtraBackup 8.0.12 et versions supérieures prennent en charge la migration de toutes les versions de MySQL 8.0. Si vous migrez vers RDS pour MySQL 8.0.32 ou supérieur, vous devez utiliser XtraBackup Percona 8.0.12 ou supérieur.

- MySQL 5.7 — Utilisez Percona XtraBackup version 2.4.

Vous pouvez utiliser Percona XtraBackup pour créer une sauvegarde complète de vos fichiers de base de données MySQL. Sinon, si vous utilisez déjà Percona XtraBackup pour sauvegarder les fichiers de votre base de données MySQL, vous pouvez télécharger vos répertoires et fichiers de sauvegarde complets et incrémentiels existants.

Pour plus d'informations sur la sauvegarde de votre base de données avec Percona XtraBackup, consultez Percona XtraBackup - Documentation sur le site Web de Percona.

Création d'une sauvegarde complète avec Percona XtraBackup

Pour créer une sauvegarde complète de vos fichiers de base de données MySQL qu'Amazon RDS peut restaurer depuis Amazon S3, utilisez l' XtraBackup utilitaire Percona ()`xtrabackup`.

Par exemple, la commande suivante crée une sauvegarde d'une base de données MySQL et stocke les fichiers dans le dossier `/on-premises/s3-restore/backup`.

```
xtrabackup --backup --user=myuser --password=password --target-dir=/on-premises/s3-restore/backup
```

Si vous souhaitez compresser votre sauvegarde en un seul fichier, que vous pourrez ensuite diviser en plusieurs fichiers si nécessaire, vous pouvez enregistrer votre sauvegarde dans l'un des formats suivants en fonction de votre version de MySQL :

- Gzip (.gz) — Pour MySQL 5.7 et versions antérieures
- tar (.tar) — Pour MySQL 5.7 et versions antérieures

- Percona xbstream (.xbstream) — Pour toutes les versions de MySQL

> **Note**
>
> Percona XtraBackup 8.0 et versions ultérieures ne prend en charge que Percona xbstream pour la compression.

MySQL 5.7 et versions antérieures

La commande suivante crée une sauvegarde de votre base de données MySQL, divisée en plusieurs fichiers Gzip. Remplacez les valeurs par vos propres informations.

```
xtrabackup --backup --user=my_user --password=password --stream=tar \
   --target-dir=/on-premises/s3-restore/backup | gzip - | split -d --bytes=500MB \
   - /on-premises/s3-restore/backup/backup.tar.gz
```

MySQL 5.7 et versions antérieures

La commande suivante crée une sauvegarde de votre base de données MySQL, divisée en plusieurs fichiers tar. Remplacez les valeurs par vos propres informations.

```
xtrabackup --backup --user=my_user --password=password --stream=tar \
   --target-dir=/on-premises/s3-restore/backup | split -d --bytes=500MB \
   - /on-premises/s3-restore/backup/backup.tar
```

Toutes les versions de MySQL

La commande suivante crée une sauvegarde de votre base de données MySQL, divisée en plusieurs fichiers xbstream. Remplacez les valeurs par vos propres informations.

```
xtrabackup --backup --user=myuser --password=password --stream=xbstream \
   --target-dir=/on-premises/s3-restore/backup | split -d --bytes=500MB \
   - /on-premises/s3-restore/backup/backup.xbstream
```

> **Note**
>
> Si le message d'erreur suivant s'affiche, cela peut être dû au fait que vous avez mélangé des formats de fichiers dans votre commande :

```
ERROR:/bin/tar: This does not look like a tar archive
```

Utilisation de sauvegardes incrémentielles avec Percona XtraBackup

Si vous utilisez déjà Percona XtraBackup pour effectuer des sauvegardes complètes et incrémentielles de vos fichiers de base de données MySQL, vous n'avez pas besoin de créer une sauvegarde complète et de télécharger les fichiers de sauvegarde sur Amazon S3. Pour gagner du temps, copiez plutôt vos répertoires et fichiers de sauvegarde existants dans votre compartiment Amazon S3. Pour plus d'informations sur la création de sauvegardes incrémentielles à l'aide de Percona XtraBackup, voir Création d'une sauvegarde incrémentielle sur le site Web de Percona.

Lorsque vous copiez les fichiers existants des sauvegardes complètes et incrémentielles dans un compartiment Amazon S3, vous devez copier de façon récursive le contenu du répertoire de base. Ce contenu inclut à la fois la sauvegarde complète et tous les répertoires et fichiers de sauvegarde incrémentielle. Cette copie doit conserver la structure de répertoire du compartiment Amazon S3. Amazon RDS effectue une itération sur l'ensemble des fichiers et répertoires. Amazon RDS utilise le `xtrabackup-checkpoints` fichier inclus dans chaque sauvegarde incrémentielle pour identifier le répertoire de base et pour commander les sauvegardes incrémentielles par plage de numéros de séquence de journaux (LSN).

Considérations relatives à la sauvegarde pour Percona XtraBackup

Amazon RDS utilise vos fichiers de sauvegarde sur la base des noms de fichier. Donnez à vos fichiers de sauvegarde l'extension de fichier appropriée en fonction du format de fichier. Par exemple, utilisez-le `.xbstream` pour les fichiers stockés au format Percona xbstream.

Amazon RDS utilise vos fichiers de sauvegarde dans l'ordre alphabétique, ainsi que l'ordre numérique naturel. Pour vous assurer que vos fichiers de sauvegarde sont écrits et nommés dans le bon ordre, utilisez `split` cette option lorsque vous lancez la `xtrabackup` commande.

Amazon RDS ne prend pas en charge les sauvegardes partielles créées à l'aide de XtraBackup Percona. Vous ne pouvez pas utiliser les options suivantes pour créer une sauvegarde partielle lorsque vous sauvegardez les fichiers source de votre base de données :

- `--tables`
- `--tables-exclude`
- `--tables-file`

- `--databases`
- `--databases-exclude`
- `--databases-file`

Création manuelle d'un rôle IAM

Si vous n'avez pas de rôle IAM, vous pouvez en créer manuellement un nouveau. Toutefois, si vous restaurez la base de données à l'aide du AWS Management Console, nous vous recommandons de demander à Amazon RDS de créer ce nouveau rôle IAM pour vous. Pour qu'Amazon RDS crée ce rôle pour vous, suivez la procédure décrite dans la Pour importer des données à partir d'Amazon S3 vers une nouvelle instance de base de données MySQL section.

Pour créer manuellement un nouveau rôle IAM afin d'importer votre base de données depuis Amazon S3, créez un rôle pour déléguer les autorisations d'Amazon RDS à votre compartiment Amazon S3. Lorsque vous créez un rôle IAM, vous attachez des stratégies d'approbation et d'autorisation. Pour importer vos fichiers de sauvegarde depuis Amazon S3, utilisez des politiques de confiance et d'autorisation similaires aux exemples suivants. Pour plus d'informations sur la création du rôle, voir Création d'un rôle pour déléguer des autorisations à un AWS service.

Les stratégies d'approbation et d'autorisation nécessitent que vous fournissiez un Amazon Resource Name (ARN). Pour plus d'informations sur le formatage de l'ARN, consultez Amazon Resource Names (ARNs) et espaces AWS de noms de services.

Example politique de confiance pour l'importation depuis Amazon S3

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AssumeRoleForBackup",
      "Effect": "Allow",
      "Principal": {
        "Service": "rds.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
```

```
  ]
}
```

Example politique d'autorisation pour l'importation depuis Amazon S3 — Autorisations utilisateur IAM

Dans l'exemple suivant, remplacez *iam_user_id* par votre propre valeur.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowS3AccessRole",
            "Effect": "Allow",
            "Action": "iam:PassRole",
            "Resource": "arn:aws:iam::111122223333:role/S3Access"
        }
    ]
}
```

Example politique d'autorisations pour l'importation depuis Amazon S3 — autorisations de rôle

Dans l'exemple suivant, remplacez *amzn-s3-demo-bucket* et *prefix* par vos propres valeurs.

JSON

```
{
    "Version": "2012-10-17",
    "Statement":
    [
        {
        "Effect": "Allow",
        "Action":
            [
                "s3:ListBucket",
                "s3:GetBucketLocation"
            ],
        "Resource": "arn:aws:s3:::amzn-s3-demo-bucket"
```

```
        },
        {
        "Effect": "Allow",
        "Action":
            [
                "s3:GetObject"
            ],
        "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/prefix*"
        },
        {
        "Effect": "Allow",
        "Action":
            [
                "kms:Decrypt"
            ],
        "Resource": [
            "arn:aws:kms:us-east-1:111122223333:key/key_id*"
            ]
        }
    ]
}
```

> **Note**
>
> Si vous incluez un préfixe de nom de fichier, vous devez inclure un astérisque (*) après le préfixe. Si vous ne voulez pas spécifier un préfixe, indiquez uniquement un astérisque.

Pour importer des données à partir d'Amazon S3 vers une nouvelle instance de base de données MySQL

Vous pouvez importer des données depuis Amazon S3 vers une nouvelle instance de base de données MySQL à l'aide de l'API AWS Management Console AWS CLI, ou RDS.

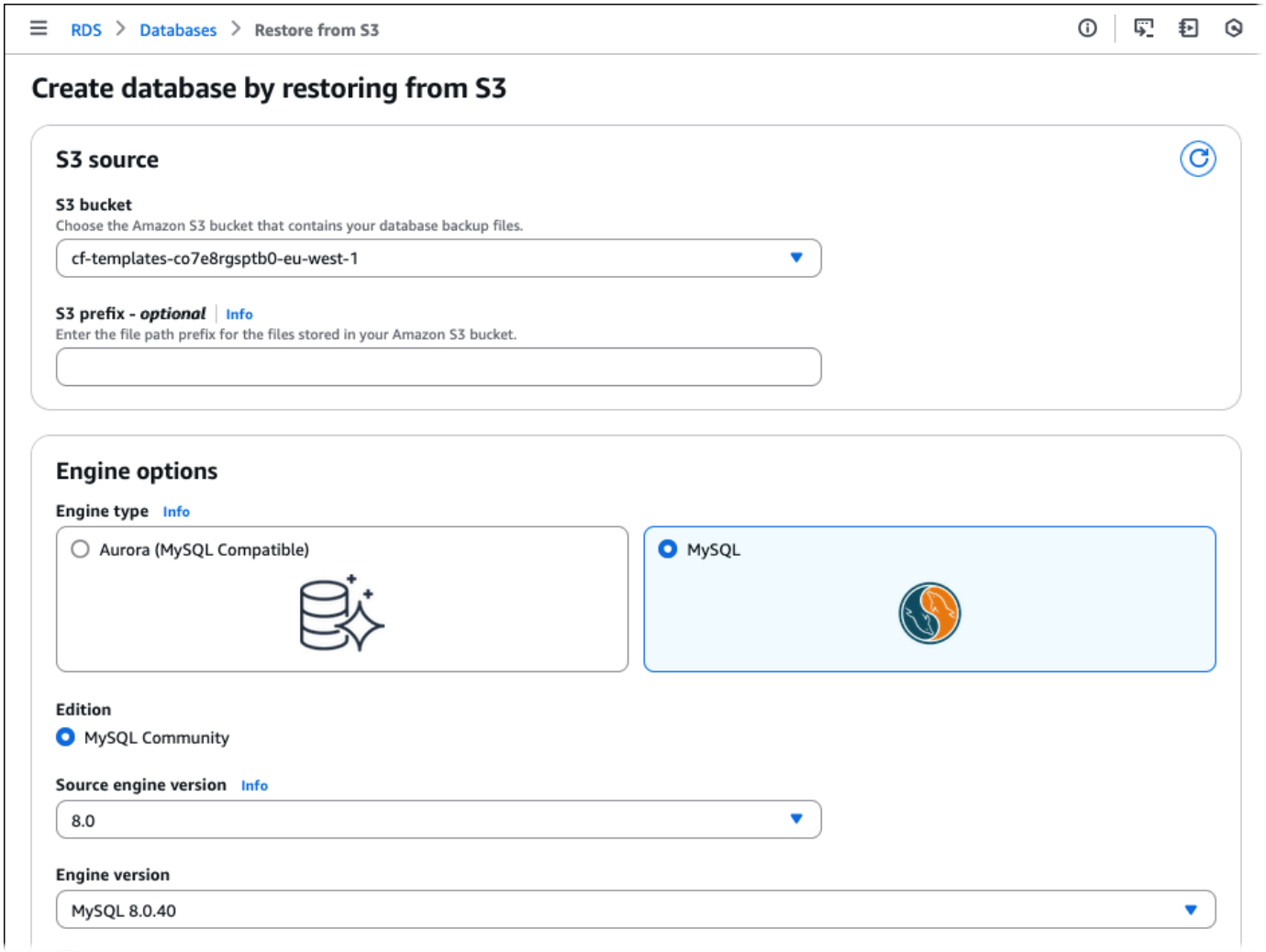
console

Pour importer des données à partir d'Amazon S3 vers une nouvelle instance de base de données MySQL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le coin supérieur droit de la console Amazon RDS, choisissez l' Région AWS endroit où vous souhaitez créer votre instance de base de données. Choisissez le même Région AWS que le compartiment Amazon S3 qui contient la sauvegarde de votre base de données.
3. Dans la panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Restaurer à partir de S3.

   La page Créer une base de données par restauration à partir de S3 s'affiche.



5. Sous la source S3 :

   a. Choisissez le compartiment S3 qui contient la sauvegarde.

   b. (Facultatif) Pour le préfixe S3, entrez le préfixe du chemin de fichier pour les fichiers stockés dans votre compartiment Amazon S3.

      Si vous ne spécifiez aucun préfixe, Amazon RDS crée votre instance de base de données en utilisant tous les fichiers et dossiers du dossier racine du compartiment S3. Si vous

spécifiez un préfixe, Amazon RDS crée votre instance de base de données en utilisant les fichiers et les dossiers du compartiment S3 où le chemin du fichier commence par le préfixe spécifié.

Par exemple, vous stockez vos fichiers de sauvegarde sur S3 dans un sous-dossier nommé backups, et vous disposez de plusieurs ensembles de fichiers de sauvegarde, chacun dans son propre répertoire (gzip_backup1, gzip_backup2, etc.). Dans ce cas, pour effectuer une restauration à partir des fichiers du dossier gzip_backup1, vous devez spécifier le préfixe backups/gzip_backup1.

6. Sous Options du moteur :

   a. Dans le champ Type de moteur, choisissez MySQL.

   b. Dans le champ Version du moteur source, choisissez la version MySQL majeure de votre base de données source.

   c. Pour la version du moteur, choisissez la version mineure par défaut de votre version majeure de MySQL dans votre Région AWS.

      Dans le AWS Management Console, seule la version mineure par défaut est disponible. Une fois l'importation terminée, vous pouvez mettre à niveau votre instance de base de données.

7. Pour le rôle IAM, créez ou choisissez un rôle IAM avec la politique de confiance et la politique d'autorisation requises qui permettent à Amazon RDS d'accéder à votre compartiment Amazon S3. Effectuez l’une des opérations suivantes :

   - (Recommandé) Choisissez Créer un nouveau rôle et entrez le nom du rôle IAM. Avec cette option, Amazon RDS crée automatiquement le rôle avec la politique de confiance et la politique d'autorisation pour vous.
   - Choisissez un rôle IAM existant. Assurez-vous que ce rôle répond à tous les critères dethe section called “Création manuelle d'un rôle IAM”.

8. Spécifiez les informations de votre instance de base de données. Pour plus d'informations sur chaque paramètre, consultez Paramètres des instances de base de données.

   > **Note**
   >
   > Assurez-vous d'allouer suffisamment de stockage pour votre nouvelle instance de base de données afin que l'opération de restauration puisse réussir.

Pour permettre automatiquement la croissance future, sous Configuration du stockage supplémentaire, sélectionnez Activer le dimensionnement automatique du stockage.

9. Choisissez des paramètres supplémentaires selon vos besoins.
10. Choisissez Create database (Créer une base de données).

AWS CLI

Pour importer des données depuis Amazon S3 vers une nouvelle instance de base de données MySQL à l'aide de AWS CLI, exécutez la commande restore-db-instance-from-s3 avec les options suivantes. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Note

Assurez-vous d'allouer suffisamment de stockage pour votre nouvelle instance de base de données afin que l'opération de restauration puisse réussir.
Pour activer le dimensionnement automatique du stockage et permettre une croissance future automatique, utilisez `--max-allocated-storage` cette option.

- `--allocated-storage`
- `--db-instance-identifier`
- `--db-instance-class`
- `--engine`
- `--master-username`
- `--manage-master-user-password`
- `--s3-bucket-name`
- `--s3-ingestion-role-arn`
- `--s3-prefix`
- `--source-engine`
- `--source-engine-version`

Example

Pour LinuxmacOS, ou Unix :

```
aws rds restore-db-instance-from-s3 \
    --allocated-storage 250 \
    --db-instance-identifier my_identifier \
    --db-instance-class db.m5.large \
    --engine mysql \
    --master-username admin \
    --manage-master-user-password \
    --s3-bucket-name amzn-s3-demo-bucket \
    --s3-ingestion-role-arn arn:aws:iam::account-number:role/rolename \
    --s3-prefix bucket_prefix \
    --source-engine my_sql \
    --source-engine-version 8.0.32 \
    --max-allocated-storage 1000
```

Dans Windows :

```
aws rds restore-db-instance-from-s3 ^
    --allocated-storage 250 ^
    --db-instance-identifier my_identifier ^
    --db-instance-class db.m5.large ^
    --engine mysql ^
    --master-username admin ^
    --manage-master-user-password ^
    --s3-bucket-name amzn-s3-demo-bucket ^
    --s3-ingestion-role-arn arn:aws:iam::account-number:role/rolename ^
    --s3-prefix bucket_prefix ^
    --source-engine mysql ^
    --source-engine-version 8.0.32 ^
    --max-allocated-storage 1000
```

API RDS

Pour importer des données d'Amazon S3 vers une nouvelle instance de base de données MySQL à l'aide de l'API Amazon RDS, appelez l'opération Restore DBInstance FromS3.

Limitations et considérations relatives à l'importation de fichiers de sauvegarde d'Amazon S3 vers Amazon RDS

Les limites et considérations suivantes s'appliquent à l'importation de fichiers de sauvegarde depuis Amazon S3 vers une instance de base de données RDS pour MySQL :

- Vous ne pouvez migrer vos données que vers une nouvelle instance de base de données, et non vers une instance de base de données existante.
- Vous devez utiliser Percona XtraBackup pour sauvegarder vos données sur Amazon S3. Pour de plus amples informations, veuillez consulter Création de la sauvegarde de votre base de données.
- Le compartiment Amazon S3 et l'instance de base de données RDS pour MySQL doivent se trouver dans le même Région AWS emplacement.
- Vous ne pouvez pas effectuer de restauration à partir des sources suivantes :
  - Exportation d'un instantané d'instance de base de données vers Amazon S3. Vous ne pouvez pas non plus migrer les données d'une exportation de snapshots d'instance de base de données vers votre compartiment Amazon S3.
  - Une base de données source cryptée. Vous pouvez toutefois chiffrer les données en cours de migration. Vous pouvez également laisser les données non chiffrées pendant le processus de migration.
  - Une base de données MySQL 5.5 ou 5.6.
- RDS pour MySQL ne prend pas en charge Percona Server for MySQL en tant que base de données source car il peut contenir des `compression_dictionary*` tables dans le. `mysql schema`
- RDS for MySQL ne prend pas en charge la rétromigration, que ce soit pour les versions majeures ou mineures. Par exemple, vous ne pouvez pas migrer de MySQL version 8.0 vers RDS pour MySQL 5.7, et vous ne pouvez pas migrer de MySQL version 8.0.32 vers RDS pour MySQL version 8.0.26.
- Amazon RDS ne prend pas en charge l'importation sur la classe d'instance de base de données db.t2.micro depuis Amazon S3. Cependant, vous pouvez effectuer une restauration dans une autre classe d'instance de base de données, puis modifier la classe d'instance de base de données ultérieurement. Pour plus d'informations sur les classes d'instance , consultez Spécifications matérielles pour les classes d'instance de base de données .
- Amazon S3 limite la taille d'un fichier chargé vers un compartiment Amazon S3 à 5 To. Si un fichier de sauvegarde dépasse 5 To, vous devez diviser celui-ci en plusieurs fichiers plus petits.

- Amazon RDS limite le nombre de fichiers chargés vers un compartiment Amazon S3 à 1 million. Si les données de sauvegarde de votre base de données, y compris toutes les sauvegardes complètes et incrémentielles, dépassent 1 million de fichiers, utilisez un fichier Gzip (.gz), tar (.tar.gz) ou Percona xbstream (.xbstream) pour stocker les fichiers des sauvegardes complètes et incrémentielles dans le compartiment Amazon S3. Percona XtraBackup 8.0 prend uniquement en charge Percona xbstream pour la compression.
- Pour fournir des services de gestion pour chaque instance de base de données, Amazon RDS crée l'`rdsadmin`utilisateur lors de la création de l'instance de base de données. Étant donné qu'`rdsamin`il s'agit d'un utilisateur réservé dans Amazon RDS, les restrictions suivantes s'appliquent :
  - Amazon RDS n'importe pas les fonctions, les procédures, les vues, les événements et les déclencheurs avec le `'rdsadmin'@'localhost'` définisseur. Pour plus d'informations, consultez Objets stockés avec 'rdsamin'@'localhost' comme définisseur et Privilèges du compte utilisateur principal.
  - Lors de la création de l'instance de base de données, Amazon RDS crée un utilisateur principal doté du maximum de privilèges pris en charge. Lors de la restauration à partir d'une sauvegarde, Amazon RDS supprime automatiquement tous les privilèges non pris en charge attribués aux utilisateurs importés.

    Pour identifier les utilisateurs susceptibles d'être concernés, consultez Comptes d'utilisateur avec des privilèges non pris en charge. Pour plus d'informations sur les privilèges pris en charge dans RDS pour MySQL, consultezModèle de privilèges basé sur les rôles pour for My RDS SQL.
- Amazon RDS ne migre pas les tables créées par les utilisateurs dans le `mysql` schéma.
- Vous devez configurer le `innodb_data_file_path` paramètre avec un seul fichier de données utilisant le nom de fichier de données par défau`ibdata1:12M:autoextend`. Vous pouvez migrer des bases de données contenant deux fichiers de données ou un fichier de données portant un nom différent en utilisant cette méthode.

  Les exemples suivants sont des noms de fichiers qu'Amazon RDS n'autorise pas :
  - `innodb_data_file_path=ibdata1:50M`
  - `ibdata2:50M:autoextend`
  - `innodb_data_file_path=ibdata01:50M:autoextend`
- Vous ne pouvez pas migrer à partir d'une base de données source dotée de tables définies à l'extérieur du répertoire de données MySQL par défaut.

- La taille maximale prise en charge pour les sauvegardes non compressées utilisant cette méthode est limitée à 64 TiB. Pour les sauvegardes compressées, cette limite est inférieure pour tenir compte des besoins en espace de décompression. Dans de tels cas, la taille de sauvegarde maximale prise en charge est de`64 TiB - compressed backup size`.

  Pour plus d'informations sur la taille de base de données maximale prise en charge par RDS for MySQL, consultez Stockage SSD à usage général etStockage SSD d'IOPS par seconde provisionnées.
- Amazon RDS ne prend pas en charge l'importation de MySQL et d'autres composants et plugins externes.
- Amazon RDS ne restaure pas tout ce qui se trouve dans votre base de données. Nous vous recommandons d'enregistrer le schéma de base de données et les valeurs des éléments suivants à partir de la base de données système MySQL source, puis de les ajouter à votre instance de base de données RDS pour MySQL restaurée après sa création :
  - Comptes utilisateurs
  - Fonctions
  - Procédures stockées
  - Informations de fuseau horaire. Les informations de fuseau horaire sont chargées depuis le système d'exploitation local de votre instance de base de données RDS pour MySQL. Pour de plus amples informations, veuillez consulter Fuseau horaire local pour les instances My SQL DB.

Objets stockés avec 'rdsamin'@'localhost' comme définisseur

Amazon RDS n'importe pas de fonctions, de procédures, de vues, d'événements ou de déclencheurs utilisés `'rdsadmin'@'localhost'` comme outil de définition.

Vous pouvez utiliser le script SQL suivant sur votre base de données MySQL source pour répertorier les objets stockés qui possèdent le définisseur non pris en charge.

```
-- This SQL query lists routines with `rdsadmin`@`localhost` as the definer.

SELECT
    ROUTINE_SCHEMA,
    ROUTINE_NAME
FROM
    information_schema.routines
WHERE
    definer = 'rdsadmin@localhost';
```

```
-- This SQL query lists triggers with `rdsadmin`@`localhost` as the definer.

SELECT
    TRIGGER_SCHEMA,
    TRIGGER_NAME,
    DEFINER
FROM
    information_schema.triggers
WHERE
    DEFINER = 'rdsadmin@localhost';

-- This SQL query lists events with `rdsadmin`@`localhost` as the definer.

SELECT
    EVENT_SCHEMA,
    EVENT_NAME
FROM
    information_schema.events
WHERE
    DEFINER = 'rdsadmin@localhost';

-- This SQL query lists views with `rdsadmin`@`localhost` as the definer.
SELECT
    TABLE_SCHEMA,
    TABLE_NAME
FROM
    information_schema.views
WHERE
    DEFINER = 'rdsadmin@localhost';
```

Comptes d'utilisateur avec des privilèges non pris en charge

Les comptes utilisateur dotés de privilèges non pris en charge par RDS for MySQL sont importés sans les privilèges non pris en charge. Pour obtenir la liste des privilèges pris en charge, consultez Modèle de privilèges basé sur les rôles pour for My RDS SQL.

Vous pouvez exécuter la requête SQL suivante sur votre base de données source pour répertorier les comptes d'utilisateur dotés de privilèges non pris en charge.

```
SELECT
    user,
    host
```

```
FROM
    mysql.user
WHERE
    Shutdown_priv = 'y'
    OR File_priv = 'y'
    OR Super_priv = 'y'
    OR Create_tablespace_priv = 'y';
```

Importation de données depuis une base de données MySQL externe vers une instance de base de données Amazon RDS for MySQL

Vous pouvez importer des données d'une base de données MariaDB ou MySQL existante vers une instance de base de données RDS pour MariaDB ou RDS pour MySQL. Pour ce faire, copiez la base de données avec mysqldump et redirigez-la directement vers l'instance de base de données RDS pour MariaDB ou RDS pour MySQL. L'utilitaire de ligne de commande `mysqldump` est généralement utilisé pour effectuer des sauvegardes et des transferts de données d'un serveur MariaDB ou MySQL vers un autre. Il est inclus dans les logiciels clients MySQL et MariaDB.

Dans MariaDB 10.5, le client s'appelle mariadb-dump. À partir de MariaDB 11.0.1, vous devez utiliser à la place de. `mariadb-dump` `mysqldump`

> **Note**
>
> Si vous importez ou exportez de grandes quantités de données avec une instance de base de données MySQL, le transfert de données vers et depuis Amazon RDS est plus fiable et plus rapide à l'aide de fichiers de `xtrabackup` sauvegarde et d'Amazon S3. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.
> Amazon RDS ne prend pas en charge `xtrabackup` MariaDB ou. `mariabackup` Amazon RDS prend uniquement en charge l'importation depuis Amazon S3 pour MySQL.

Une `mysqldump` commande classique pour déplacer des données d'une base de données externe vers une instance de base de données Amazon RDS ressemble à l'exemple suivant. Remplacez les valeurs par vos propres informations. Pour les versions 11.0.1 et supérieures de MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

```
mysqldump -u local_user \
```

```
    --databases database_name \
    --single-transaction \
    --compress \
    --order-by-primary  \
    --routines=0 \
    --triggers=0 \
    --events=0 \
    -plocal_password | mysql -u RDS_user \
        --port=port_number \
        --host=host_name \
        -pRDS_password
```

> **Important**
>
> Veillez à ne pas laisser d'espace entre l'option `-p` et le mot de passe saisi.
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans cet exemple.

Assurez-vous que vous êtes conscient des recommandations et des considérations suivantes :

- Excluez les schémas suivants du fichier de vidage :
  - `sys`
  - `performance_schema`
  - `information_schema`

  L'`mariadb-dump`utilitaire `mysqldump` and exclut ces schémas par défaut.
- Si vous devez migrer des utilisateurs et des privilèges, pensez à utiliser un outil qui génère le langage de contrôle des données (DCL) pour les recréer, tel que l'pt-show-grantsutilitaire.
- Pour effectuer l'importation, assurez-vous que l'utilisateur qui l'effectue a accès à l'instance de base de données. Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité.

Les paramètres utilisés sont les suivants :

- `-u local_user` – Utilisez ce paramètre pour spécifier un nom d'utilisateur. Lors de la première utilisation de ce paramètre, spécifiez le nom d'un compte utilisateur sur la base de données MariaDB ou MySQL locale que vous identifiez `--databases` avec le paramètre.

- `--databases` *`database_name`* : utilisez ce paramètre pour spécifier le nom de la base de données sur l'instance MariaDB ou MySQL locale que vous souhaitez importer dans Amazon RDS.
- `--single-transaction` – Utilisez ce paramètre pour vérifier que toutes les données chargées depuis la base de données locale sont en cohérence avec un point dans le temps unique. S'il existe d'autres processus qui modifient les données pendant que `mysqldump` les lit, l'utilisation de ce paramètre permet de maintenir l'intégrité des données.
- `--compress` – Utilisez ce paramètre pour réduire la consommation de bande passante réseau par compression des données à partir de la base de données locale avant de les envoyer vers Amazon RDS.
- `--order-by-primary` – Utilisez ce paramètre pour réduire le temps de chargement en triant les données de chaque tableau sur par clé primaire.
- `--routines`— À utiliser si des routines telles que des procédures stockées ou des fonctions existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les routines pendant le processus d'importation. Ensuite, recréez manuellement les routines dans la base de données Amazon RDS.
- `--triggers`— À utiliser si des déclencheurs existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les déclencheurs pendant le processus d'importation. Ensuite, recréez manuellement les déclencheurs dans la base de données Amazon RDS.
- `--events`— À utiliser si des événements existent dans la base de données que vous copiez. Définissez le paramètre sur`0`, ce qui exclut les événements pendant le processus d'importation. Recréez ensuite manuellement les événements dans la base de données Amazon RDS.
- `-p`*`local_password`* – Utilisez ce paramètre pour spécifier un mot de passe. Lors de la première utilisation de ce paramètre, spécifiez le mot de passe du compte utilisateur que vous identifiez avec le premier `-u` paramètre.
- `-u` *`RDS_user`* – Utilisez ce paramètre pour spécifier un nom d'utilisateur. Lors de la deuxième utilisation de ce paramètre, spécifiez le nom d'un compte utilisateur sur la base de données par défaut pour l'instance de base de données MariaDB ou MySQL que vous identifiez `--host` avec le paramètre.
- `--port` *`port_number`* : utilisez ce paramètre pour spécifier le port pour votre instance de base de données MariaDB ou MySQL. Par défaut, il s'agit de 3306, sauf si vous avez modifié la valeur lors de la création de l'instance de base de données.
- `--host` *`host_name`* : utilisez ce paramètre pour spécifier le nom du système de nom de domaine (DNS) du point de terminaison de l'instance de base de données Amazon RDS, par exemple, `myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez trouver la

valeur du point de terminaison dans les détails de l'instance de base de données dans la console Amazon RDS.

- `-p`*`RDS_password`* – Utilisez ce paramètre pour spécifier un mot de passe. Lors de la seconde utilisation de ce paramètre, vous spécifiez le mot de passe du compte utilisateur identifié par le second paramètre `-u`.

Assurez-vous de créer manuellement les procédures stockées, déclencheurs, fonctions ou événements dans votre base de données Amazon RDS. Si la base de données que vous copiez contient l'un de ces objets, excluez-le lorsque vous exécutez `mysqldump` ou`mariadb-dump`. Pour ce faire, incluez les paramètres suivants dans votre `mariadb-dump` commande `mysqldump` or :

- `--routines=0`
- `--triggers=0`
- `--events=0`

Exemple

L'exemple suivant copie l'`world`exemple de base de données sur l'hôte local vers une instance de base de données RDS pour MySQL. Remplacez les valeurs par vos propres informations. Pour copier l'exemple de base de données sur une instance de base de données RDS pour MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

Dans Linux, macOS, ou Unix:

```
sudo mysqldump -u local_user \
    --databases world \
    --single-transaction \
    --compress \
    --order-by-primary  \
    --routines=0 \
    --triggers=0 \
    --events=0 \
    -plocal_password | mysql -u rds_user \
        --port=3306 \
        --host=my_instance.123456789012.us-east-1.rds.amazonaws.com \
        -pRDS_password
```

Dans Windows:

Exécutez la commande suivante dans une invite de commande qui a été ouverte en cliquant avec le bouton droit sur Invite de commandes dans le menu des programmes Windows et en choisissant Exécuter en tant qu'administrateur. Remplacez les valeurs par vos propres informations. Pour copier l'exemple de base de données sur une instance de base de données RDS pour MariaDB, `mysqldump` remplacez par et par. `mariadb-dump` `mysql` `mariadb`

```
mysqldump -u local_user ^
    --databases world ^
    --single-transaction ^
    --compress ^
    --order-by-primary  ^
    --routines=0 ^
    --triggers=0 ^
    --events=0 ^
    -plocal_password | mysql -u RDS_user ^
        --port=3306 ^
        --host=my_instance.123456789012.us-east-1.rds.amazonaws.com ^
        -pRDS_password
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt

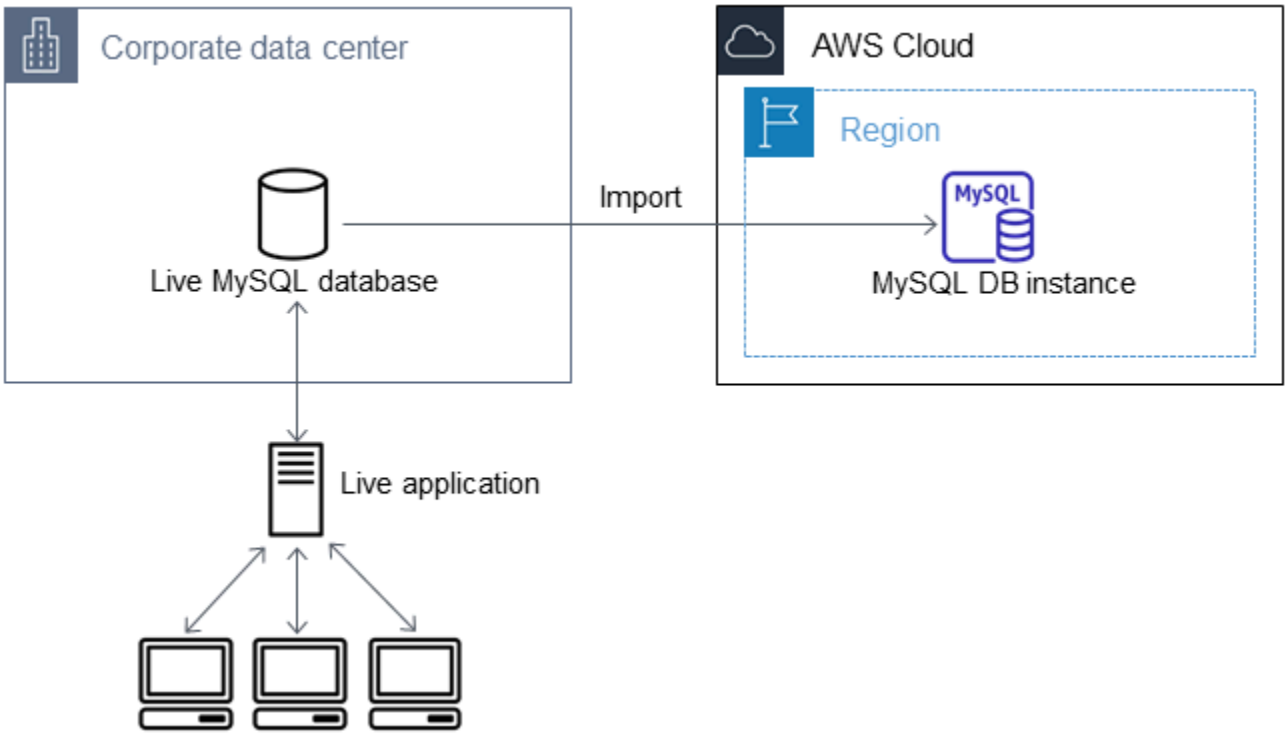
Dans certains cas, vous devrez peut-être importer des données d'une base de données MariaDB ou MySQL externe qui prend en charge une application en direct vers une instance de base de données RDS pour MariaDB, une instance de base de données RDS pour MySQL ou un cluster de base de données multi-AZ RDS pour MySQL. Utilisez la procédure suivante pour réduire l'impact sur la disponibilité des applications. Cette procédure peut s'avérer également utile si vous travaillez avec une base de données très volumineuse. À l'aide de cette procédure, vous pouvez réduire le coût de l'importation en réduisant la quantité de données transmises sur le réseau AWS.

Dans cette procédure, vous transférez une copie des données de votre base de données vers une EC2 instance Amazon et vous importez les données dans une nouvelle base de données Amazon RDS. Vous utilisez ensuite la réplication pour intégrer la base de données Amazon RDS up-to-date

à votre instance externe active, avant de rediriger votre application vers la base de données Amazon RDS. Si l'instance externe est MariaDB 10.0.24 ou version ultérieure et que l'instance cible est RDS pour MariaDB, configurez la réplication MariaDB en fonction des identifiants de transaction globaux (). GTIDs Sinon, configurez la réplication en fonction des coordonnées des journaux binaires. Nous recommandons la réplication GTID si votre base de données externe la prend en charge, car la réplication GTID est une méthode plus fiable. Pour plus d'informations, consultez Identificateurs de transaction mondiaux dans la documentation MariaDB.

> **Note**
>
> Si vous souhaitez importer des données dans une instance de base de données RDS pour MySQL et que votre scénario le permet, nous vous recommandons de déplacer les données vers et depuis Amazon RDS à l'aide de fichiers de sauvegarde et d'Amazon S3. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.

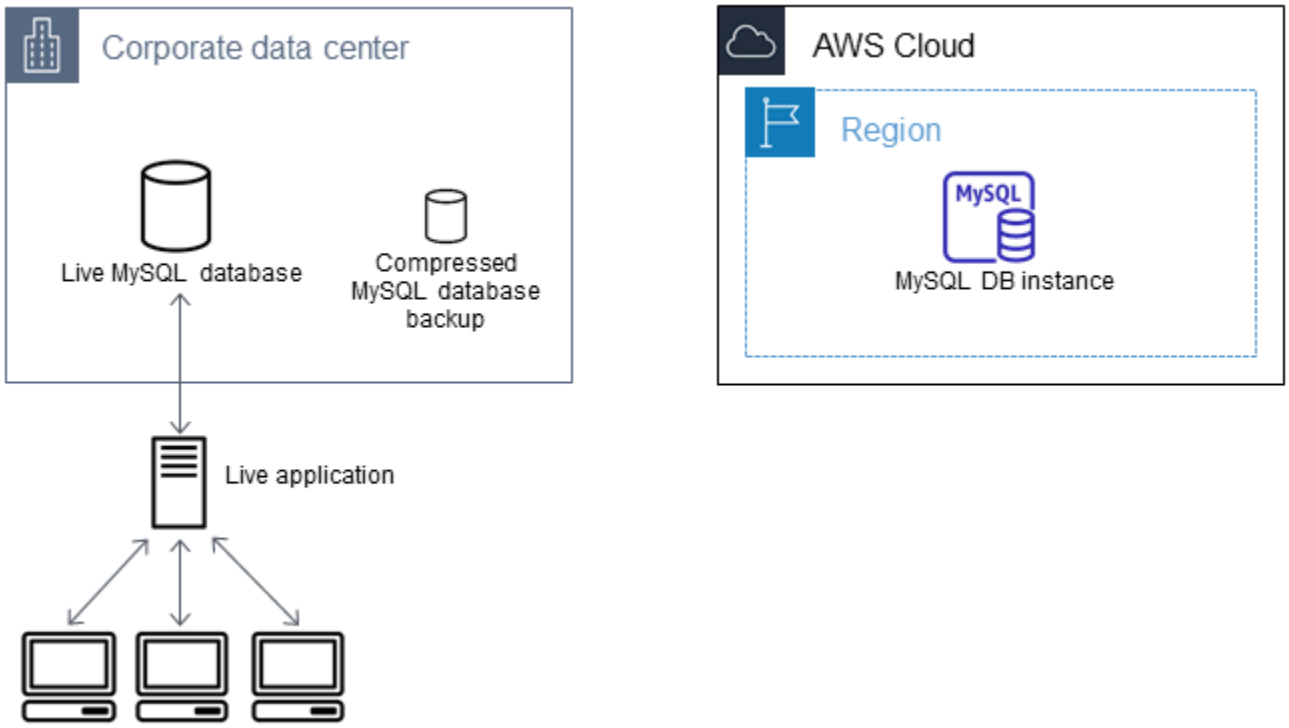
Le schéma suivant montre l'importation d'une base de données MySQL externe dans une base de données MySQL sur Amazon RDS.



Tâche 1 : créer une copie de votre base de données existante

La première étape du processus de migration d'une grande quantité de données vers une base de données RDS for MariaDB ou RDS for MySQL avec un temps d'arrêt minimal consiste à créer une copie des données sources.

Le schéma suivant montre la création d'une sauvegarde de la base de données MySQL.



Vous pouvez recourir à l'utilitaire `mysqldump` pour créer une sauvegarde de base de données au format SQL ou texte délimité. Dans MariaDB 10.5, le client s'appelle mariadb-dump. À partir de MariaDB 11.0.1, vous devez utiliser à la place de. `mariadb-dump` `mysqldump` Nous vous recommandons d'effectuer un test avec chaque format dans un environnement autre que celui de production afin de déterminer la méthode qui minimise le temps d'exécution de mysqldump.

Nous vous recommandons également d'évaluer les `mysqldump` performances par rapport aux avantages offerts en utilisant le format de texte délimité pour le chargement. Une sauvegarde à l'aide du format texte délimité crée un fichier texte séparé par des tabulations pour chaque table vidée. Pour réduire le temps nécessaire à l'importation de votre base de données, vous pouvez charger ces fichiers en parallèle en utilisant la commande `LOAD DATA LOCAL INFILE`. Pour plus d'informations, consultez Charger les données pour MariaDB ou Charger les données pour MySQL.

Avant de commencer l'opération de sauvegarde, assurez-vous de définir les options de réplication sur la base de données MariaDB ou MySQL que vous copiez vers Amazon RDS. Les options de

réplication incluent l'activation de la journalisation binaire et la configuration d'un ID de serveur unique. La définition de ces options oblige votre serveur à démarrer la journalisation des transactions de base de données et le prépare à être une instance de réplication source ultérieurement dans le processus.

Assurez-vous que vous êtes conscient des recommandations et des considérations suivantes :

- Utilisez l'option `--single-transaction` avec `mysqldump`, car elle vide un état cohérent de la base de données. Pour vous assurer de la validité d'un fichier de vidage, n'exécutez pas les instructions DDL (Data Definition Language) pendant l'exécution de `mysqldump`. Vous pouvez planifier une fenêtre de maintenance pour ces opérations.
- Excluez les schémas suivants du fichier de vidage :
  - `sys`
  - `performance_schema`
  - `information_schema`

  L'`mariadb-dump`utilitaire `mysqldump` and exclut ces schémas par défaut.
- Si vous devez migrer des utilisateurs et des privilèges, pensez à utiliser un outil qui génère le langage de contrôle des données (DCL) pour les recréer, tel que l'pt-show-grantsutilitaire.

Pour définir les options de réplication

1. Modifiez le fichier `my.cnf`. Ce fichier se trouve généralement sous`/etc`.

   ```
   sudo vi /etc/my.cnf
   ```

   Ajoutez les options `log_bin` et `server_id` à la section `[mysqld]`. L'option `log_bin` fournit un identifiant de nom de fichier pour les fichiers journaux binaires. L'option `server_id` fournit un identifiant unique pour le serveur dans les relations source/réplica.

   L'exemple suivant montre la `[mysqld]` section mise à jour d'un `my.cnf` fichier :

   ```
   [mysqld]
   log-bin=mysql-bin
   server-id=1
   ```

Pour plus d'informations, consultez la section Définition de la configuration de la source de réplication dans la documentation MySQL.

2. Pour la réplication avec un cluster de base de données multi-AZ, définissez le `GTID_MODE` paramètre `ENFORCE_GTID_CONSISTENCY` et sur`ON`.

```
mysql> SET @@GLOBAL.ENFORCE_GTID_CONSISTENCY = ON;
```

```
mysql> SET @@GLOBAL.GTID_MODE = ON;
```

Ces paramètres ne sont pas requis pour la réplication avec une instance de base de données.

3. Redémarrez le service `mysql`.

```
sudo service mysqld restart
```

Pour créer une copie de sauvegarde de votre base de données existante

1. Créez une sauvegarde de vos données à l'aide de l'`mysqldump`utilitaire ou de l'`mariadb-dump`utilitaire, en spécifiant le format SQL ou le format texte délimité.

   Pour MySQL 8.0.25 et versions antérieures, spécifiez de `--master-data=2` créer un fichier de sauvegarde qui peut être utilisé pour démarrer la réplication entre les serveurs. Pour MySQL 8.0.26 et versions supérieures, spécifiez de `--source-data=2` créer un fichier de sauvegarde qui peut être utilisé pour démarrer la réplication entre les serveurs. Pour plus d'informations, consultez mysqldump — A Database Backup Program dans la documentation MySQL.

   Pour améliorer les performances et garantir l'intégrité des données, utilisez les `--single-transaction` options `--order-by-primary` et pour`mysqldump`.

   Pour éviter d'inclure la base de données système MySQL dans la sauvegarde, n'utilisez pas l'`--all-databases`option with`mysqldump`. Pour plus d'informations, consultez la section Création d'un instantané de données à l'aide de mysqldump dans la documentation MySQL.

   Utilisez`chmod`, si nécessaire, pour vous assurer que le répertoire dans lequel le fichier de sauvegarde est créé est inscriptible.

> **Important**
>
> Sur Windows, exécutez la fenêtre de commande en tant qu'administrateur.

- Pour produire une sortie SQL, utilisez la commande suivante :

  Dans Linux, macOS, ou Unix:

```
sudo mysqldump \
    --databases database_name \
    --master-data=2  \
    --single-transaction \
    --order-by-primary \
    -r backup.sql \
    -u local_user \
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

  Dans Windows:

```
mysqldump ^
    --databases database_name ^
    --master-data=2  ^
    --single-transaction ^
    --order-by-primary ^
    -r backup.sql ^
    -u local_user ^
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.

- Pour produire une sortie en texte délimité, utilisez la commande suivante :

  Dans Linux, macOS, ou Unix:

```
sudo mysqldump \
    --tab=target_directory \
    --fields-terminated-by ',' \
    --fields-enclosed-by '"' \
    --lines-terminated-by 0x0d0a \
    database_name \
    --master-data=2 \
    --single-transaction \
    --order-by-primary \
    -ppassword
```

  Dans Windows:

```
mysqldump ^
    --tab=target_directory ^
    --fields-terminated-by "," ^
    --fields-enclosed-by """ ^
    --lines-terminated-by 0x0d0a ^
    database_name ^
    --master-data=2 ^
    --single-transaction ^
    --order-by-primary ^
    -ppassword
```

> **Note**
>
> Pour des raisons de sécurité, il est recommandé de spécifier des informations d'identification autres que celles indiquées dans l'exemple.
> Assurez-vous de créer manuellement les procédures stockées, déclencheurs, fonctions ou événements dans votre base de données Amazon RDS. Si vous avez l'un de ces

objets dans la base de données que vous copiez, excluez-les lorsque lors de l'exécution de `mysqldump`. Pour ce faire, incluez les arguments suivants dans votre `mysqldump` commande :

- `--routines=0`
- `--triggers=0`
- `--events=0`

Pour MySQL 8.0.22 et versions antérieures et pour les versions MariaDB, lorsque vous exécutez `mysqldump` et spécifiez le format de texte délimité, un commentaire est renvoyé. `CHANGE MASTER TO` Ce commentaire contient le nom du fichier journal maître et son emplacement. Pour MySQL 8.0.23 et versions supérieures, lorsque vous `mysqldump` utilisez le format de texte délimité, un commentaire est renvoyé. `CHANGE REPLICATION SOURCE TO` Ce commentaire contient le nom et la position du fichier journal source. Si l'instance externe est autre chose que MariaDB 10.0.24 et versions supérieures, ou MySQL 8.0.23 et versions supérieures, notez les valeurs pour et. `MASTER_LOG_FILE` `MASTER_LOG_POS` Vous avez besoin de ces valeurs lors de la configuration de la réplication.

La sortie suivante est renvoyée pour les versions de MariaDB et pour MySQL 8.0.22 et versions antérieures :

```
-- Position to start replication or point-in-time recovery from
--
-- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin-changelog.000031',
 MASTER_LOG_POS=107;
```

La sortie suivante est renvoyée pour MySQL 8.0.23 et versions supérieures :

```
-- Position to start replication or point-in-time recovery from
--
-- CHANGE SOURCE TO SOURCE_LOG_FILE='mysql-bin-changelog.000031',
 SOURCE_LOG_POS=107;
```

Pour MySQL 8.0.22 et versions antérieures, si vous utilisez le format SQL, vous pouvez obtenir le nom et la position du fichier journal principal dans le `CHANGE MASTER TO` commentaire du fichier de sauvegarde. Pour MySQL 8.0.23 et versions supérieures, si vous utilisez le format SQL, vous pouvez obtenir le nom et la position du fichier journal source dans le `CHANGE`

`REPLICATION SOURCE TO` commentaire du fichier de sauvegarde. Si l'instance externe est MariaDB version 10.0.24 ou ultérieure, vous pouvez obtenir l'identifiant de transaction global à l'étape suivante.

2. Si l'instance externe que vous utilisez est MariaDB version 10.0.24 ou supérieure, utilisez la réplication basée sur GTID. Exécutez `SHOW MASTER STATUS` sur l'instance externe de MariaDB pour obtenir le nom et la position du fichier journal binaire, puis convertissez-les en GTID en les exécutant `BINLOG_GTID_POS` sur l'instance externe de MariaDB.

```
SELECT BINLOG_GTID_POS('binary_log_file_name', binary_log_file_position);
```

   Notez le GTID renvoyé. Vous avez besoin du GTID pour configurer la réplication.

3. Compressez les données copiées afin de réduire la quantité de ressources réseau nécessaires pour copier vos données sur la base de données Amazon RDS. Notez la taille du fichier de sauvegarde. Vous avez besoin de ces informations pour déterminer la taille d'une EC2 instance Amazon à créer. Lorsque vous avez terminé, compressez le fichier de sauvegarde à l'aide de GZIP ou de votre utilitaire de compression favori.

   - Pour compresser la sortie SQL, utilisez la commande suivante :

```
gzip backup.sql
```

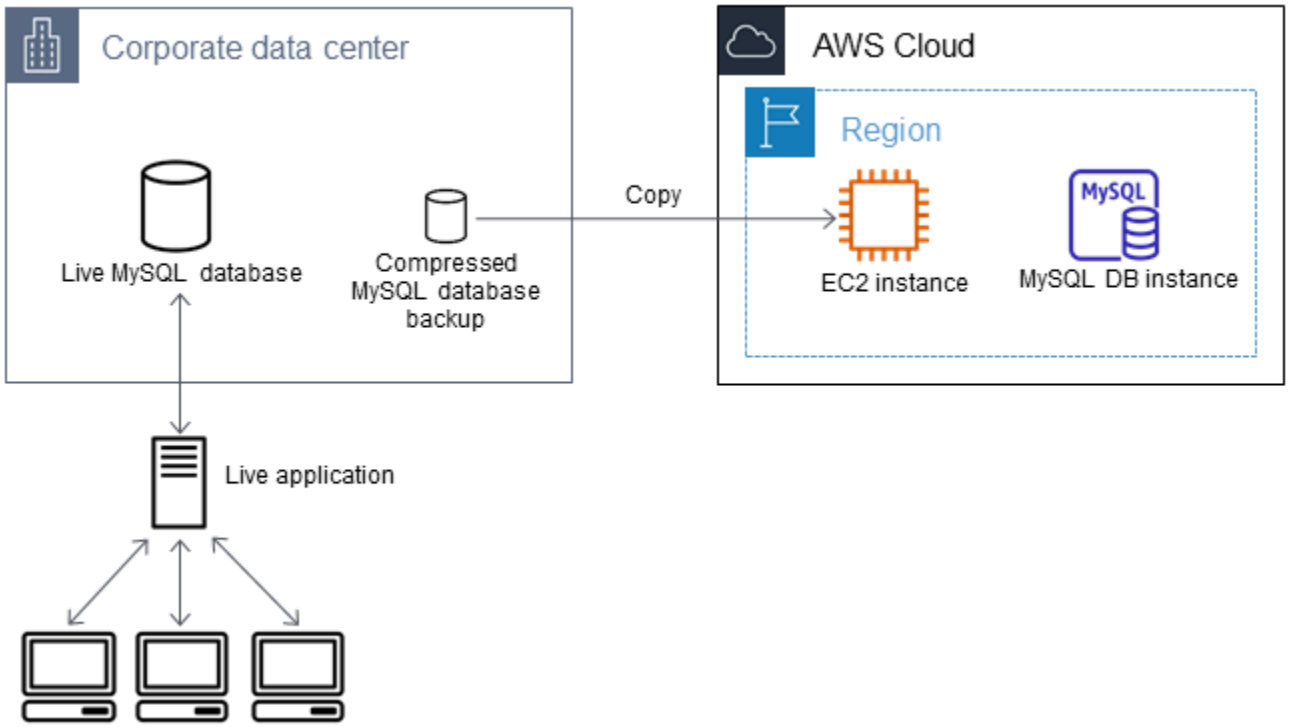
   - Pour compresser une sortie en texte délimité, utilisez la commande suivante :

```
tar -zcvf backup.tar.gz target_directory
```

Tâche 2 : créer une EC2 instance Amazon et copier la base de données compressée

La copie de votre fichier de sauvegarde de base de données compressé sur une EC2 instance Amazon nécessite moins de ressources réseau que la copie directe de données non compressées entre des instances de base de données. Une fois que vos données sont dans Amazon EC2, vous pouvez les copier directement dans votre base de données MariaDB ou MySQL. Pour que vous puissiez économiser sur le coût des ressources réseau, votre EC2 instance Amazon doit être Région AWS identique à votre instance de base de données Amazon RDS. Le fait que l' EC2 instance Amazon soit Région AWS identique à votre base de données Amazon RDS réduit également la latence du réseau lors de l'importation.

Le schéma suivant montre comment copier la sauvegarde de la base de données sur une EC2 instance Amazon.



Pour créer une EC2 instance Amazon et copier vos données

1. Dans l' Région AWS endroit où vous prévoyez de créer la base de données Amazon RDS, créez un cloud privé virtuel (VPC), un groupe de sécurité VPC et un sous-réseau VPC. Assurez-vous que les règles entrantes de votre groupe de sécurité VPC autorisent les adresses IP requises pour que votre application se connecte à AWS. Vous pouvez spécifier une plage d'adresses IP, par exemple, ou `203.0.113.0/24` un autre groupe de sécurité VPC. Vous pouvez utiliser la console Amazon VPC pour créer et gérer des sous-réseaux et VPCs des groupes de sécurité. Pour plus d'informations, consultez Getting started with Amazon VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.
2. Ouvrez la EC2 console Amazon et choisissez celle qui doit contenir Région AWS à la fois votre EC2 instance Amazon et votre base de données Amazon RDS. Lancez une EC2 instance Amazon à l'aide du VPC, du sous-réseau et du groupe de sécurité que vous avez créés à l'étape 1. Vérifiez que vous sélectionnez un type d'instance avec un stockage suffisant pour le fichier de sauvegarde de votre base de données une fois qu'il est décompressé. Pour en savoir plus sur EC2 les instances Amazon, consultez Getting started with Amazon EC2 dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

3. Pour vous connecter à votre base de données Amazon RDS depuis votre EC2 instance Amazon, modifiez votre groupe de sécurité VPC. Ajoutez une règle entrante spécifiant l'adresse IP privée de votre EC2 instance. Vous trouverez l'adresse IP privée dans l'onglet Détails du volet Instance de la fenêtre de EC2 console. Pour modifier le groupe de sécurité VPC et ajouter une règle entrante, choisissez Security Groups dans le volet de navigation de la EC2 console, choisissez votre groupe de sécurité, puis ajoutez une règle entrante pour MySQL ou Aurora spécifiant l'adresse IP privée de votre instance. EC2 Pour savoir comment ajouter une règle entrante à un groupe de sécurité VPC, consultez la section Règles du groupe de sécurité dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.
4. Copiez le fichier de sauvegarde de votre base de données compressée depuis votre système local vers votre EC2 instance Amazon. `chmod`Utilisez-le, si nécessaire, pour vous assurer que vous disposez d'une autorisation d'écriture pour le répertoire cible de l' EC2 instance Amazon. Vous pouvez utiliser `scp` ou un client SSH pour copier le fichier. La commande suivante est un exemple de `scp` commande :

   ```
   scp -r -i key pair.pem backup.sql.gz ec2-user@EC2 DNS:/target_directory/backup.sql.gz
   ```

   > **Important**
   >
   > Lorsque vous copiez des données sensibles, veillez à utiliser un protocole de transfert réseau sécurisé.

5. Connectez-vous à votre EC2 instance Amazon et installez les dernières mises à jour et les outils client MySQL à l'aide des commandes suivantes :

   ```
   sudo yum update -y
   sudo yum install mysql -y
   ```

   Pour plus d'informations, consultez Connect to your instance for Linux dans le guide de l'utilisateur Amazon Elastic Compute Cloud.

   > **Important**
   >
   > Cet exemple installe le client MySQL sur une Amazon Machine Image (AMI) pour une distribution Amazon Linux. Cet exemple n'installe pas le client MySQL sur une autre distribution, telle qu'Ubuntu ou Red Hat Enterprise Linux. Pour plus d'informations sur

l'installation de MySQL, consultez la section Installation de MySQL dans la documentation MySQL.

6. Lorsque vous êtes connecté à votre EC2 instance Amazon, décompressez le fichier de sauvegarde de votre base de données. Les commandes suivantes sont des exemples.
   - Pour décompresser une sortie SQL, utilisez la commande suivante :

     ```
     gzip backup.sql.gz -d
     ```

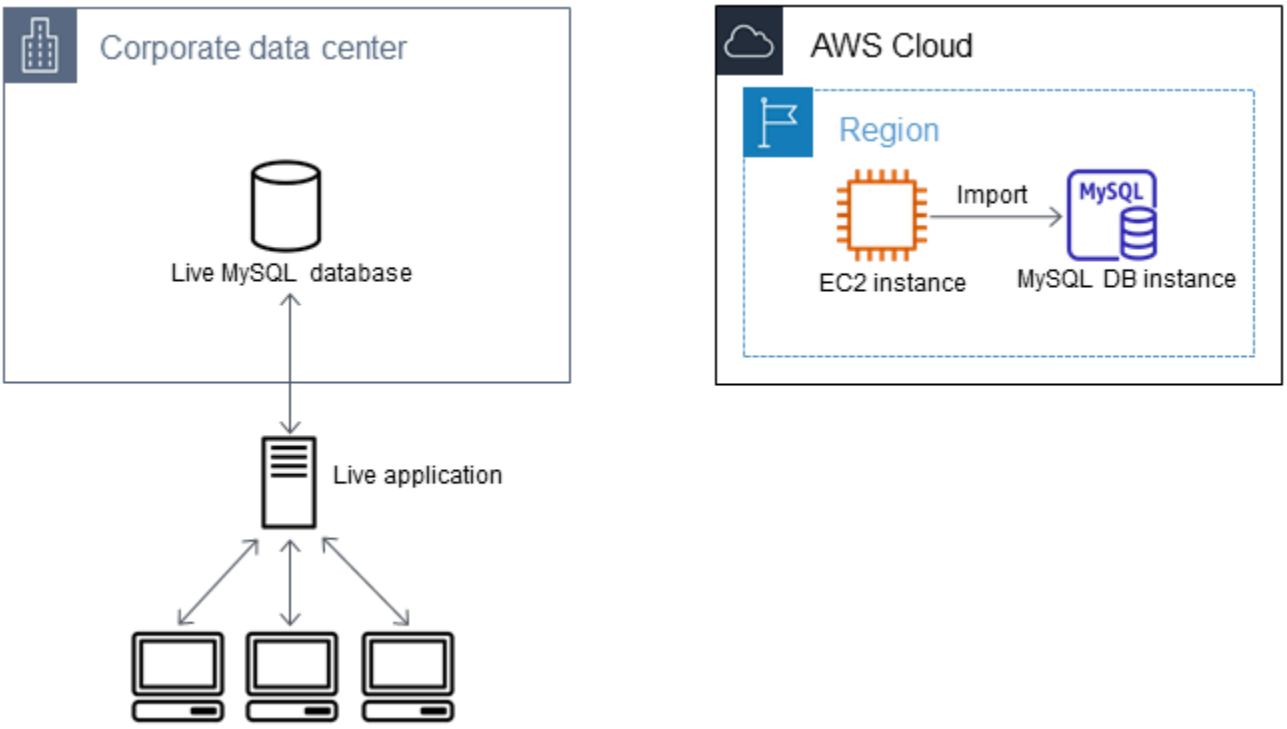
   - Pour décompresser une sortie texte délimité, utilisez la commande suivante :

     ```
     tar xzvf backup.tar.gz
     ```

Tâche 3 : créer une base de données MySQL ou MariaDB et importer des données depuis votre instance Amazon EC2

En créant une instance de base de données RDS pour MariaDB, une instance de base de données RDS pour MySQL ou un cluster de base de données multi-AZ RDS pour MySQL au même Région AWS titre que votre instance Amazon, vous pouvez importer le fichier de sauvegarde de base de données depuis EC2 Amazon plus rapidement que sur Internet. EC2

Le schéma suivant montre l'importation de la sauvegarde depuis une EC2 instance Amazon dans une base de données MySQL.



Pour créer une base de données MariaDB ou MySQL et importer vos données

1. Déterminez quelle classe d'instance de base de données et quelle quantité d'espace de stockage sont nécessaires pour prendre en charge la charge de travail attendue pour cette base de données Amazon RDS. Dans le cadre de ce processus, décidez de l'espace suffisant et de la capacité de traitement qui conviennent à vos procédures de chargement des données. Décidez également de ce qui est nécessaire pour gérer la charge de travail de production. Vous pouvez estimer ces éléments en fonction de la taille et des ressources de la base de données source MariaDB ou MySQL. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .
2. Créez une instance de base de données ou un cluster de base de données multi-AZ dans Région AWS celui qui contient votre EC2 instance Amazon.

   Pour créer un cluster de bases de données multi-AZ RDS pour MySQL, suivez les instructions de. Création d'un cluster de base de données multi-AZ pour Amazon RDS

   Pour créer une instance de base de données RDS pour MariaDB ou RDS pour MySQL, suivez les instructions Création d'une instance de base de données Amazon RDS et utilisez les directives suivantes :

- Spécifiez une version du moteur de base de données compatible avec votre instance de base de données source.
  - Si votre instance source est MySQL 5.5.x, l'instance de base de données Amazon RDS doit être MySQL.
  - Si votre instance source est MySQL 5.6.x ou 5.7.x, l'instance de base de données Amazon RDS doit être MySQL ou MariaDB.
  - Si votre instance source est MySQL 8.0.x, l'instance de base de données Amazon RDS doit être MySQL 8.0.x.
  - Si votre instance source est MySQL 8.4.x, l'instance de base de données Amazon RDS doit être MySQL 8.4.x.
  - Si votre instance source est MariaDB 5.5 ou version ultérieure, l'instance de base de données Amazon RDS doit être MariaDB.
- Spécifiez le même cloud privé virtuel (VPC) et le même groupe de sécurité VPC que pour votre instance Amazon. EC2 Cette approche garantit que votre EC2 instance Amazon et votre instance Amazon RDS sont visibles l'une pour l'autre sur le réseau. Assurez-vous que votre instance de base de données est accessible au public. Pour configurer la réplication avec votre base de données source comme décrit dans la section suivante, votre instance de base de données doit être accessible au public.
- Ne configurez pas plusieurs zones de disponibilité, la rétention des sauvegardes ou les réplicas en lecture tant que vous n'avez pas importé la sauvegarde de la base de données. Lorsque l'importation est terminée, vous pouvez configurer l'option multi-AZ et la rétention des sauvegardes pour l'instance de production.

3. Vérifiez les options de configuration par défaut de la base de données Amazon RDS. Si le groupe de paramètres par défaut pour la base de données ne dispose pas des options de configuration que vous voulez, trouvez un autre groupe qui les possède ou créez un groupe de paramètres. Pour plus d'informations sur la création d'un groupe de paramètres, consultezGroupes de paramètres pour (Amazon RDS).
4. Connectez-vous à la nouvelle base de données Amazon RDS en tant qu'utilisateur principal. Créez les utilisateurs nécessaires pour prendre en charge les administrateurs, les applications et les services qui doivent accéder à l'instance de base de données. Le nom d'hôte de la base de données Amazon RDS est la valeur du point de terminaison de cette instance de base de données sans le numéro de port, par exemple. `mysampledb.123456789012.us-west-2.rds.amazonaws.com` Vous pouvez trouver la valeur du point de terminaison dans les détails de la base de données dans la console Amazon RDS.

5. Connectez-vous à votre EC2 instance Amazon. Pour plus d'informations, consultez Connect to your instance for Linux dans le guide de l'utilisateur Amazon Elastic Compute Cloud.
6. Connectez-vous à votre base de données Amazon RDS en tant qu'hôte distant depuis votre EC2 instance Amazon à l'aide de la `mysql` commande. La commande suivante est un exemple :

   ```
   mysql -h host_name -P 3306 -u db_master_user -p
   ```

   *host_name*Il s'agit du point de terminaison de la base de données Amazon RDS.
7. À l'`mysql`invite, exécutez la `source` commande et transmettez-lui le nom du fichier de vidage de votre base de données. Cette commande charge les données dans l'instance de base de données Amazon RDS.

   - Pour le format SQL, utilisez la commande suivante :

     ```
     mysql> source backup.sql;
     ```

   - Pour le format texte délimité, créez d'abord la base de données, s'il ne s'agit pas de la base de données par défaut que vous avez créée lors de la configuration de la base de données Amazon RDS.

     ```
     mysql> create database database_name;
     mysql> use database_name;
     ```

     Créez ensuite les tables.

     ```
     mysql> source table1.sql
     mysql> source table2.sql
     etc...
     ```

     Enfin, importez les données.

     ```
     mysql> LOAD DATA LOCAL INFILE 'table1.txt' INTO TABLE table1 FIELDS TERMINATED BY
      ',' ENCLOSED BY '"' LINES TERMINATED BY '0x0d0a';
     mysql> LOAD DATA LOCAL INFILE 'table2.txt' INTO TABLE table2 FIELDS TERMINATED BY
      ',' ENCLOSED BY '"' LINES TERMINATED BY '0x0d0a';
     etc...
     ```

Pour améliorer les performances, vous pouvez exécuter ces opérations en parallèle à partir de plusieurs connexions de telle sorte que l'ensemble de vos tables soit créé et chargé simultanément.

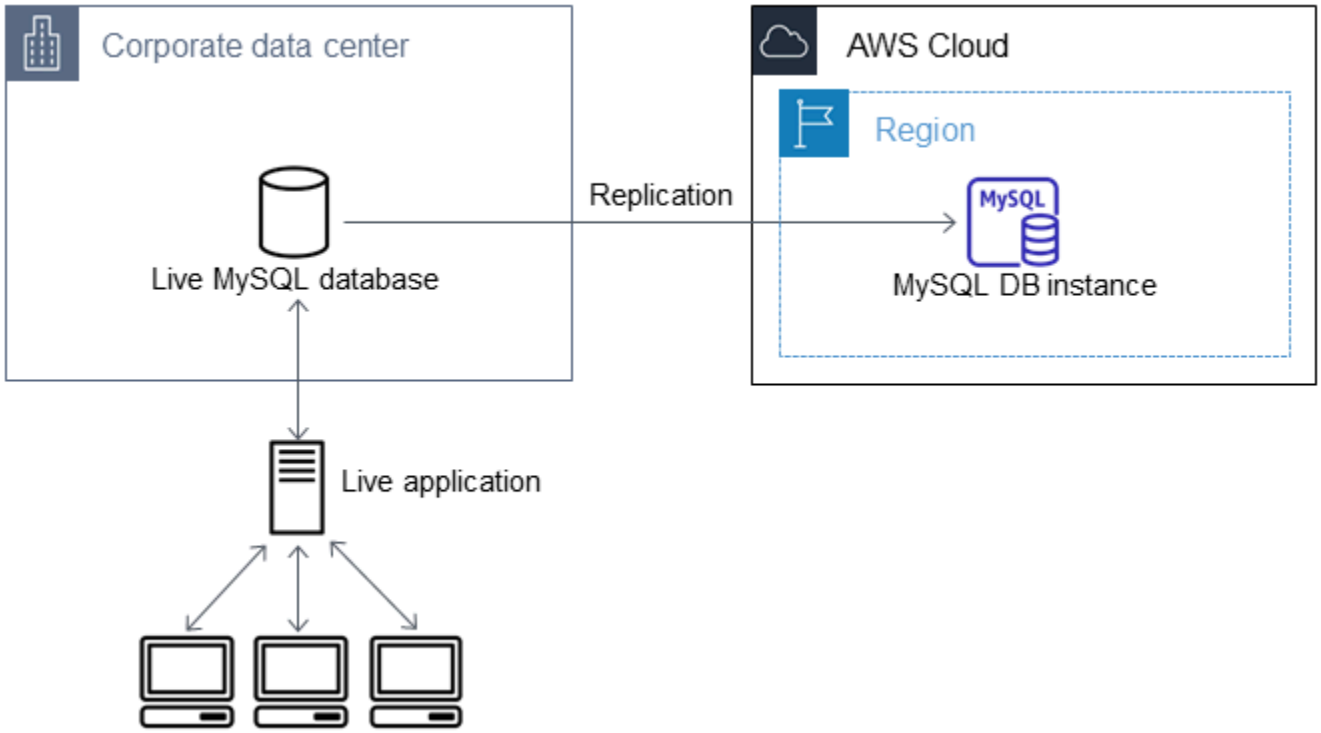
> **Note**
>
> Si vous avez utilisé des options de formatage des données `mysqldump` lorsque vous avez initialement vidé le tableau, veillez à utiliser les mêmes options `LOAD DATA LOCAL INFILE` pour garantir une interprétation correcte du contenu du fichier de données.

8. Exécutez une `SELECT` requête simple sur une ou deux des tables de la base de données importée pour vérifier que l'importation a réussi.

Si vous n'avez plus besoin de l' EC2 instance Amazon utilisée dans cette procédure, mettez-la hors service EC2 afin de réduire votre consommation de AWS ressources. Pour mettre fin à une EC2 instance, consultez la section Résiliation d'une instance dans le guide de l'utilisateur d'Amazon Elastic Compute Cloud.

Tâche 4 : répliquer les données de votre base de données externe vers votre nouvelle base de données Amazon RDS

Votre base de données source a probablement été mise à jour pendant la copie et le transfert des données vers la base de données MariaDB ou MySQL. Ainsi, vous pouvez utiliser la réplication pour intégrer la base de données up-to-date copiée à la base de données source.



Les autorisations requises pour démarrer la réplication sur une base de données Amazon RDS sont limitées et ne sont pas accessibles à votre utilisateur principal Amazon RDS. Pour cette raison, utilisez la procédure stockée Amazon RDS appropriée pour la version principale de votre moteur :

-
-
- mysql.rds_set_external_master_gtidpour configurer la réplication et mysql.rds_start_replication démarrer la réplication

Pour démarrer la réplication

Dans la tâche 1, lorsque vous définissez les options de réplication, vous avez activé la journalisation binaire et défini un ID de serveur unique pour votre base de données source. Maintenant, vous pouvez configurer votre base de données Amazon RDS comme réplica avec votre base de données active comme instance de réplication source.

1. Dans la console Amazon RDS, ajoutez l'adresse IP du serveur qui héberge la base de données source au groupe de sécurité VPC pour la base de données Amazon RDS. Pour plus d'informations sur la configuration d'un groupe de sécurité VPC, consultez la section Configurer les règles du groupe de sécurité dans le guide de l'utilisateur Amazon Virtual Private Cloud.

Vous devrez peut-être également configurer votre réseau local pour autoriser les connexions depuis l'adresse IP de votre base de données Amazon RDS afin qu'elle puisse communiquer avec votre instance source. Pour trouver l'adresse IP de la base de données Amazon RDS, utilisez la `host` commande suivante :

```
host host_name
```

*`host_name`*Il s'agit du nom DNS du point de terminaison de la base de données Amazon RDS, par exemple`myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez trouver la valeur du point de terminaison dans les détails de l'instance de base de données dans la console Amazon RDS.

2. A l'aide du client de votre choix, connectez-vous à l'instance source et créez un utilisateur à utiliser pour la réplication. Ce compte est utilisé exclusivement pour la réplication et doit être limité à votre domaine pour améliorer la sécurité. La commande suivante est un exemple :

```
CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
```

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

3. Pour l'instance source, attribuez les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` à votre utilisateur de réplication. Par exemple, pour accorder les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données à l'utilisateur « `repl_user` » de votre domaine, émettez la commande suivante :

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
```

4. Si vous avez utilisé le format SQL pour créer votre fichier de sauvegarde et que l'instance externe n'est pas MariaDB 10.0.24 ou version ultérieure, examinez le contenu de ce fichier en exécutant la commande suivante :

```
cat backup.sql
```

Le fichier inclut un commentaire `CHANGE MASTER TO` qui contient le nom du fichier journal maître et son emplacement. Ce commentaire est inclus dans le fichier de sauvegarde quand vous utilisez l'option `--master-data` avec `mysqldump`. Notez les valeurs pour `MASTER_LOG_FILE` et `MASTER_LOG_POS`.

```
--
-- Position to start replication or point-in-time recovery from
--

-- CHANGE MASTER TO MASTER_LOG_FILE='mysql-bin-changelog.000031', MASTER_LOG_POS=107;
```

Si vous avez utilisé un format de texte délimité pour créer votre fichier de sauvegarde et que l'instance externe n'est pas MariaDB 10.0.24 ou une version ultérieure, vous devriez déjà disposer des coordonnées de journal binaires indiquées à l'étape 1 de la procédure Pour créer une copie de sauvegarde de votre base de données existante dans le cadre de la tâche 1.

Si l'instance externe est MariaDB 10.0.24 ou version ultérieure, vous devriez déjà disposer du GTID à partir duquel démarrer la réplication, comme indiqué à l'étape 2 de la procédure Pour créer une copie de sauvegarde de votre base de données existante sous Tâche 1.

5. Transformez la base de données Amazon RDS en réplica. Si l'instance externe n'est pas MariaDB 10.0.24 ou version ultérieure, connectez-vous à la base de données Amazon RDS en tant qu'utilisateur principal et identifiez la base de données source en tant qu'instance de réplication source en utilisant la procédure stockée Amazon RDS appropriée :

   •

   •

   Si vous disposez d'un fichier de sauvegarde au format SQL, utilisez le nom du fichier journal principal et la position du journal principal que vous avez déterminés à l'étape 4. Si vous avez utilisé le format texte délimité, utilisez le nom et la position que vous avez déterminés lors de la création des fichiers de sauvegarde. Les commandes suivantes sont des exemples :

   MySQL 8.4 et versions supérieures

   ```
   CALL mysql.rds_set_external_source ('myserver.mydomain.com', 3306,
       'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
   ```

   MariaDB et MySQL 8.0 et versions antérieures

```
CALL mysql.rds_set_external_master ('myserver.mydomain.com', 3306,
    'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

Si l'instance externe est de version MariaDB 10.0.24 ou ultérieure, connectez-vous à la base de données Amazon RDS en tant qu'utilisateur principal et identifiez la base de données source comme instance de réplication source à l'aide de la commande mysql.rds_set_external_master_gtid. Utilisez le GTID que vous avez déterminé à l'étape 2 de la procédure Pour créer une copie de sauvegarde de votre base de données existante dans le cadre de la tâche 1. La commande suivante est un exemple :

```
CALL mysql.rds_set_external_master_gtid ('source_server_ip_address', 3306,
 'ReplicationUser', 'password', 'GTID', 1);
```

source_server_ip_address est l'adresse IP de l'instance de réplication source. Une adresse DNS EC2 privée n'est actuellement pas prise en charge.

> **Note**
>
> Spécifiez des informations d'identification autres que celles affichées ici, en tant que bonne pratique de sécurité.

6. Sur la base de données Amazon RDS, pour démarrer la réplication, exécutez la mysql.rds_start_replication commande suivante :

```
CALL mysql.rds_start_replication;
```

7. Sur la base de données Amazon RDS, pour déterminer à quel moment la réplique est à jour avec l'instance de réplication source, exécutez la commande SHOW REPLICA STATUS. Les résultats de la commande `SHOW REPLICA STATUS` incluent le champ `Seconds_Behind_Master`.

Lorsque le `Seconds_Behind_Master` champ renvoie 0, la réplique est à jour avec l'instance de réplication source.

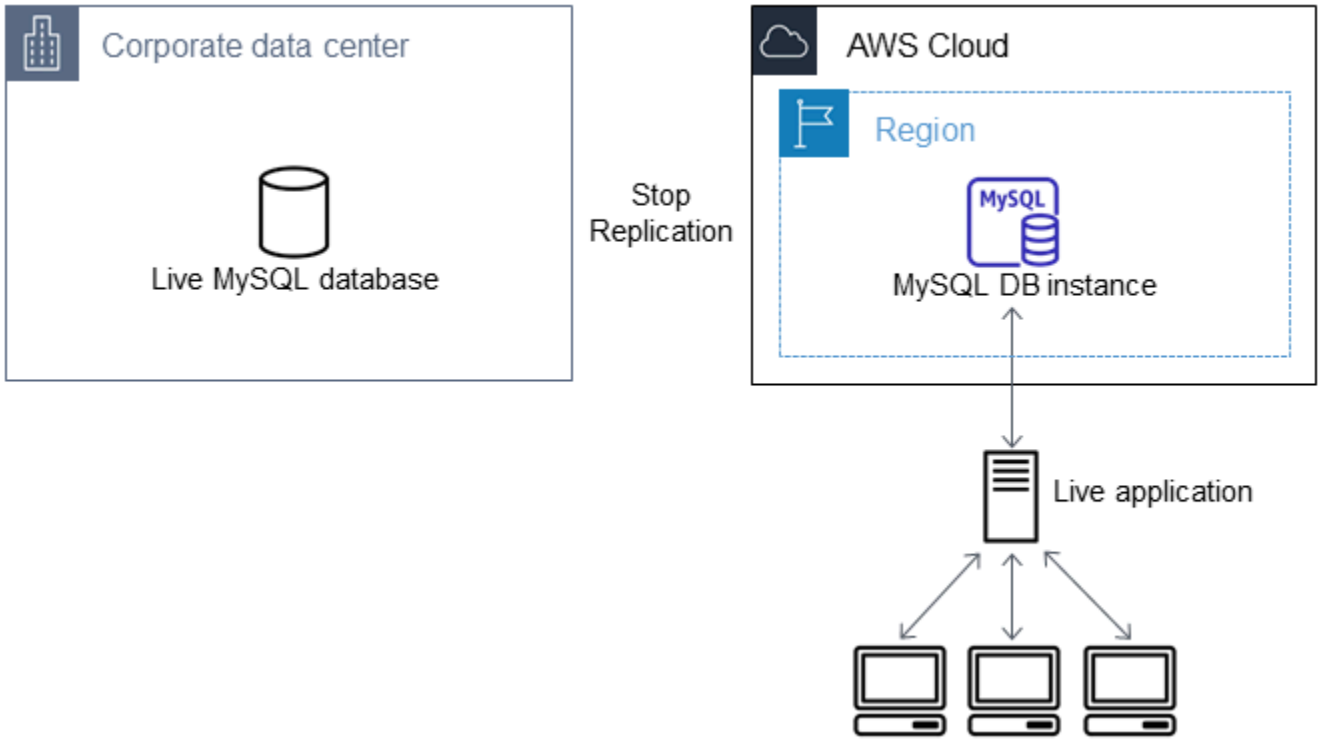
> **Note**
>
> Les versions précédentes de MySQL utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MySQL antérieure à la version 8.0.23, utilisez alors `SHOW SLAVE STATUS`.

Pour une instance de base de données MariaDB 10.5, 10.6 ou 10.11, exécutez la procédure stockée au lieu de mysql.rds_replica_status la commande MySQL.

8. Une fois la base de données Amazon RDS à jour, activez les sauvegardes automatiques afin de pouvoir restaurer cette base de données si nécessaire. Vous pouvez activer ou modifier les sauvegardes automatisées de votre base de données Amazon RDS à l'aide de la console Amazon RDS. Pour de plus amples informations, veuillez consulter Présentation des sauvegardes.

Tâche 5 : rediriger votre application en ligne vers votre instance Amazon RDS

Une fois que la base de données MariaDB ou MySQL est à jour avec l'instance de réplication source, vous pouvez désormais mettre à jour votre application live pour utiliser l'instance Amazon RDS.



Pour rediriger votre application active vers votre base de données MariaDB ou MySQL et arrêter la réplication

1. Pour ajouter le groupe de sécurité VPC pour la base de données Amazon RDS, ajoutez l'adresse IP du serveur qui héberge l'application. Pour plus d'informations sur la modification d'un groupe de sécurité VPC, consultez Configurer les règles du groupe de sécurité dans le guide de l'utilisateur Amazon Virtual Private Cloud.
2. Vérifiez que le `Seconds_Behind_Master` champ des résultats de la commande SHOW REPLICA STATUS est égal à 0, ce qui indique que la réplique est à jour avec l'instance de réplication source.

```
SHOW REPLICA STATUS;
```

> **Note**
>
> Les versions précédentes de MySQL utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MySQL antérieure à la version 8.0.23, utilisez alors `SHOW SLAVE STATUS`.

Pour une instance de base de données MariaDB 10.5, 10.6 ou 10.11, exécutez la procédure mysql.rds_replica_status à la place de la commande MySQL.

3. Fermez toutes les connexions à la source une fois leurs transactions terminées.
4. Mettez à jour votre application pour utiliser la base de données Amazon RDS. Cette mise à jour implique généralement de modifier les paramètres de connexion pour identifier le nom d'hôte et le port de la base de données Amazon RDS, le compte utilisateur et le mot de passe avec lesquels se connecter, et la base de données à utiliser.
5. Connectez-vous à l'instance de base de données.

   Pour un cluster de bases de données multi-AZ, connectez-vous à l'instance de base de données d'écriture.
6. Arrêtez la réplication pour l'instance Amazon RDS à l'aide de la mysql.rds_stop_replication commande suivante :

   ```
   CALL mysql.rds_stop_replication;
   ```
7. Exécutez la procédure stockée Amazon RDS appropriée sur votre base de données Amazon RDS pour réinitialiser la configuration de réplication afin que cette instance ne soit plus identifiée comme une réplique.

   •

   •

   MySQL 8.4 et versions supérieures

   ```
   CALL mysql.rds_reset_external_source;
   ```

   MariaDB et MySQL 8.0 et versions antérieures

   ```
   CALL mysql.rds_reset_external_master;
   ```
8. Activez des fonctions Amazon RDS supplémentaires, telles que la prise en charge Multi-AZ et les réplicas en lecture. Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS et Utilisation des réplicas en lecture d'instance de base de données.

Importation de données depuis n'importe quelle source vers une instance de base de données Amazon RDS for MySQL

Les étapes suivantes vous expliquent comment importer des données vers une instance de base de données Amazon RDS :

Étape 1 : Créer les fichiers plats contenant les données à charger

Utilisez un format courant, tel que CSV (valeurs séparées par des virgules), pour stocker les données à charger. Chaque table doit avoir son propre fichier. Vous ne pouvez pas combiner les données de plusieurs tables dans le même fichier. Attribuez à chaque fichier le même nom que celui de la table à laquelle il correspond. L'extension du fichier est laissée à votre libre choix. Par exemple, si le nom de la table est`sales`, le nom du fichier peut être `sales.csv` ou`sales.txt`.

Si possible, classez les données en fonction de la clé primaire de la table en cours de chargement. Cela améliore de façon spectaculaire les temps de chargement et réduit le stockage disque requis.

Cette procédure est d'autant plus rapide et efficace que les fichiers ont une petite taille. Si la taille non compressée d'un fichier est supérieure à 1 Gio, scindez-le en plusieurs fichiers et chargez chacun d'eux séparément.

Sur les systèmes Unix (Linux inclus), utilisez la commande `split`. Par exemple, la commande suivante fractionne le fichier `sales.csv` en plusieurs fichiers de moins d'1 Gio, le fractionnement n'intervenant qu'aux sauts de ligne (-C 1 024m). Les noms des nouveaux fichiers incluent des suffixes numériques croissants. La commande suivante génère des fichiers portant des noms tels que `sales.part_00` et`sales.part_01`.

```
split -C 1024m -d sales.csv sales.part_
```

Des utilitaires semblables sont disponibles sur les autres systèmes d'exploitation.

Vous pouvez stocker les fichiers plats n'importe où. Toutefois, lorsque vous chargez les données à l'étape 5, vous devez appeler le `mysql` shell depuis le même emplacement où se trouvent les fichiers, ou utiliser le chemin absolu des fichiers lors de l'exécution`LOAD DATA LOCAL INFILE`.

Étape 2 : empêcher les applications d'accéder à l'instance de base de données cible

Avant de démarrer un chargement important, empêchez toute activité de l'application d'accéder à l'instance de base de données cible sur laquelle vous prévoyez de charger. Nous le recommandons

particulièrement quand d'autres sessions sont susceptibles de modifier les tables chargées ou les tables qu'elles référencent. Cela réduit le risque de violation des contraintes intervenant pendant le chargement et améliore les performances de chargement. Dans le même temps, cela permet également de restaurer l'instance de base de données au point juste antérieur au chargement sans perdre les modifications effectuées par les processus non impliqués dans le chargement.

Il est vrai que cela peut ne pas être possible ou pratique. Si vous ne pouvez pas empêcher les applications d'accéder à l'instance de base de données avant le chargement, prenez les mesures nécessaires pour garantir la disponibilité et l'intégrité de vos données. Les étapes spécifiques requises varient grandement en fonction de cas d'utilisation spécifiques et des exigences du site.

Étape 3 : Créer un instantané de base de données

Si vous envisagez de charger des données dans une nouvelle instance de base de données qui ne contient aucune donnée, vous pouvez ignorer cette étape. Dans le cas contraire, nous vous recommandons de créer des instantanés de base de données de l'instance de base de données Amazon RDS cible avant et après le chargement des données. Les instantanés de base de données Amazon RDS sont des sauvegardes complètes de votre instance de base de données que vous pouvez utiliser pour restaurer votre instance de base de données à un état connu. Lorsque vous lancez un instantané de bases de données, les opérations I/O sur votre instance de base de données sont momentanément suspendues pendant la sauvegarde de la base de données.

La création d'un instantané de base de données juste avant le chargement vous permet, si besoin est, de restaurer la base de données à son état avant le chargement. Un instantané de base de données pris immédiatement après le chargement vous évite de devoir charger les données à nouveau en cas d'incident. Vous pouvez également utiliser des instantanés de base de données après le chargement pour importer des données dans de nouvelles instances de base de données.

L'exemple suivant exécute la AWS CLI create-db-snapshotcommande pour créer un instantané de base de données de l'`AcmeRDS`instance et attribuer l'identifiant à l'instantané de base de données`"preload"`.

Dans Linux, macOS, ou Unix:

```
aws rds create-db-snapshot \
    --db-instance-identifier AcmeRDS \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds create-db-snapshot ^
    --db-instance-identifier AcmeRDS ^
    --db-snapshot-identifier preload
```

Vous pouvez aussi utiliser la restauration de la fonctionnalité d'instantané de bases de données pour créer des instances de bases de données de test dans le but de réaliser des essais ou pour annuler les modifications effectuées pendant le chargement.

Gardez à l'esprit que la restauration d'une base de données à partir d'un instantané de bases de données crée une nouvelle instance de base de données qui, comme toutes les instances de base de données, possède un point de terminaison et un identifiant unique. Si vous devez restaurer l'instance de base de données sans modifier le point de terminaison, vous devez d'abord supprimer l'instance de base de données de telle sorte que le point de terminaison puisse être réutilisé.

Par exemple, pour créer une instance de base de données pour les essais ou autres tests, vous attribuez à l'instance de base de données son propre identifiant. Dans cet exemple, l'identifiant est `AcmeRDS-2`. L'exemple se connecte à l'instance de base de données à l'aide du point de terminaison associé à `AcmeRDS-2`. Pour plus d'informations, consultez restore-db-instance-from-db-snapshot.

Dans Linux, macOS, ou Unix:

```
aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier AcmeRDS-2 \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier AcmeRDS-2 ^
    --db-snapshot-identifier preload
```

Pour réutiliser le point de terminaison existant, il faut d'abord supprimer l'instance de base de données, puis donner le même identifiant à la base de données restaurée. Pour de plus amples informations, veuillez consulter delete-db-instance.

L'exemple suivant prend également un instantané de base de données final de l'instance de base de données avant de la supprimer. Cette action est facultative, mais recommandée.

Dans Linux, macOS, ou Unix:

```
aws rds delete-db-instance \
    --db-instance-identifier AcmeRDS \
    --final-db-snapshot-identifier AcmeRDS-Final

aws rds restore-db-instance-from-db-snapshot \
    --db-instance-identifier AcmeRDS \
    --db-snapshot-identifier preload
```

Dans Windows:

```
aws rds delete-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --final-db-snapshot-identifier AcmeRDS-Final

aws rds restore-db-instance-from-db-snapshot ^
    --db-instance-identifier AcmeRDS ^
    --db-snapshot-identifier preload
```

Étape 4 (facultatif) : désactiver les sauvegardes automatisées Amazon RDS

> **Warning**
>
> Ne désactivez pas les sauvegardes automatiques si vous devez effectuer une point-in-time restauration.

La désactivation des sauvegardes automatisées permet d'optimiser les performances et n'est pas nécessaire pour les chargements de données. La désactivation des sauvegardes automatiques efface toutes les sauvegardes existantes. Par conséquent, une fois les sauvegardes automatisées désactivées, la point-in-time restauration n'est pas possible. Les instantanés de bases de données manuels ne sont pas affectés par la désactivation des sauvegardes automatiques. Tous les instantanés manuels de base de données existants demeurent disponibles pour la restauration.

La désactivation des sauvegardes automatiques réduit le temps de chargement de près de 25 %, ainsi que la quantité d'espace de stockage requise pendant le chargement. Si vous envisagez de charger des données dans une nouvelle instance de base de données qui ne contient aucune donnée, la désactivation des sauvegardes constitue un moyen simple d'accélérer le chargement et d'éviter d'utiliser le stockage supplémentaire nécessaire pour les sauvegardes. Cependant, dans certains cas, vous pouvez envisager de charger dans une instance de base de données qui contient

déjà des données. Si tel est le cas, évaluez les avantages de la désactivation des sauvegardes par rapport à l'impact de la perte de performance point-in-time-recovery.

Les instances de bases de données ont les sauvegardes automatiques activées par défaut (avec une période de rétention égale à une journée). Pour désactiver les sauvegardes automatiques, définissez la période de rétention des sauvegardes à 0. Après le chargement, vous pouvez réactiver les sauvegardes en définissant la période de rétention des sauvegardes avec une valeur différente de zéro. Pour activer ou désactiver les sauvegardes, Amazon RDS arrête l'instance de base de données, puis la redémarre pour activer ou désactiver la connexion à MariaDB ou MySQL.

Exécutez la AWS CLI `modify-db-instance` commande pour définir la rétention des sauvegardes sur zéro et appliquez la modification immédiatement. Comme la définition de la période de rétention à la valeur zéro nécessite un redémarrage de l'instance de base de données, attendez que le redémarrage soit terminé avant de poursuivre. Pour de plus amples informations, veuillez consulter modify-db-instance.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier AcmeRDS \
    --apply-immediately \
    --backup-retention-period 0
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --apply-immediately ^
    --backup-retention-period 0
```

Vous pouvez vérifier l'état de votre instance de base de données à l'aide de la AWS CLI describe-db-instancescommande. L'exemple suivant affiche l'état de l'instance de base de données de l'`AcmeRDS`instance de base de données :

```
aws rds describe-db-instances --db-instance-identifier AcmeRDS --query "*[].
{DBInstanceStatus:DBInstanceStatus}"
```

Lorsque le statut de l'instance de base de données est définie`available`, vous êtes prêt à passer à l'étape suivante.

Étape 5 : Charger les données

Pour lire les lignes de vos fichiers plats dans les tables de base de données, utilisez l'`LOAD DATA LOCAL INFILE`instruction MySQL.

> **Note**
>
> Vous devez appeler le `mysql` shell depuis le même emplacement où se trouvent vos fichiers plats, ou utiliser le chemin absolu des fichiers lors de l'exécution`LOAD DATA LOCAL INFILE`.

L'exemple suivant montre comment charger les données d'un fichier nommé `sales.txt` dans une table nommée `Sales` dans la base de données :

```
mysql> LOAD DATA LOCAL INFILE 'sales.txt' INTO TABLE Sales FIELDS TERMINATED BY ' '
 ENCLOSED BY '' ESCAPED BY '\\';
Query OK, 1 row affected (0.01 sec)
Records: 1  Deleted: 0  Skipped: 0  Warnings: 0
```

Pour plus d'informations sur `LOAD DATA` cette instruction, consultez l'instruction LOAD DATA dans la documentation MySQL.

Étape 6 : réactiver les sauvegardes automatisées Amazon RDS

Si vous avez désactivé les sauvegardes automatisées Amazon RDS à l'étape 4, activez les sauvegardes automatisées une fois le chargement terminé en redéfinissant la période de conservation des sauvegardes à sa valeur de préchargement. Comme indiqué à l'étape 4, Amazon RDS redémarre l'instance de base de données. Préparez-vous donc à une brève interruption.

L'exemple suivant exécute la AWS CLI modify-db-instancecommande pour activer les sauvegardes automatiques pour l'`AcmeRDS`instance de base de données et définir la période de rétention sur un jour :

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier AcmeRDS \
    --backup-retention-period 1 \
```

```
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier AcmeRDS ^
    --backup-retention-period 1 ^
    --apply-immediately
```

Utilisation de la réplication MySQL dans Amazon RDS

Vous utilisez généralement des réplicas en lecture pour configurer la réplication entre instances de base de données Amazon RDS. Pour obtenir des informations générales sur les réplicas en lecture, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données. Pour obtenir des informations spécifiques sur l'utilisation des réplicas en lecture sur Amazon RDS pour MySQL, consultez la section Travailler avec les répliques de My SQL read.

Vous pouvez utiliser des identificateurs de transaction globaux (GTIDs) pour la réplication avec RDS for MySQL. Pour de plus amples informations, veuillez consulter Utilisation de GTID la réplication basée.

Vous pouvez également configurer la réplication entre une instance de base de données RDS for MySQL et une instance MariaDB ou MySQL externe à Amazon RDS. Pour plus d'informations sur la réplication de configuration avec une source externe, consultez Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

Pour chacune de ces options de réplication, vous pouvez utiliser la réplication basée sur les lignes, basée sur les instructions ou mixte. La réplication basée sur les lignes réplique uniquement les lignes modifiées à la suite d'une instruction SQL. La réplication basée sur les instructions réplique l'ensemble de l'instruction SQL. La réplication mixte utilise la réplication basée sur les instructions chaque fois que possible, mais bascule vers la réplication basée sur les lignes lorsque des instructions SQL présentant un risque pour la réplication basée sur les instructions sont exécutées. La réplication mixte est recommandée dans la plupart des cas. Le format de journalisation binaire de l'instance de base de données détermine si la réplication est basée sur les lignes, basée sur les instructions ou mixte. Pour plus d'informations sur la définition du format de journalisation binaire, consultez la section Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ.

> **Note**
>
> Vous pouvez configurer la réplication de sorte à importer des bases de données d'une instance MariaDB ou MySQL extérieure à Amazon RDS, ou à exporter des bases de données vers de telles instances. Pour plus d’informations, consultez Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt et Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Après avoir restauré votre instance de base de données à partir d'un instantané ou effectué une point-in-time restauration, vous pouvez consulter la dernière position du journal binaire récupérée depuis la base de données source dans la console RDS. Sous Logs & events, saisissez binlog. La position du binlog apparaît sous Notes système.

Rubriques

- Travailler avec les répliques de My SQL read
- Utilisation de GTID la réplication basée
- Configuration d'une réplication de position de fichier journal binaire avec une instance source externe
- Configuration multi-source-replication pour Amazon RDS for MySQL

Travailler avec les répliques de My SQL read

Vous trouverez ci-dessous des informations spécifiques sur l'utilisation des répliques de lecture sur RDS for MySQL. Pour obtenir des informations générales sur les réplicas en lecture et des instructions pour les utiliser, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Pour plus d'informations sur les répliques My SQL read, consultez les rubriques suivantes.

- Configuration des filtres de réplication avec My SQL
- Configuration de la réplication différée avec My SQL
- Mettre à jour les répliques de lecture avec My SQL
- Utilisation de déploiements de répliques de lecture multi-AZ avec My SQL
- Utilisation de répliques de lecture en cascade avec for My RDS SQL
- Surveillance du délai de réplication pour les répliques My SQL read
- Démarrage et arrêt de la réplication avec My SQL read replicas
- Résolution d'un problème lié à SQL la réplique My read

Configuration des répliques de lecture avec My SQL

Avant qu'une instance My SQL DB puisse servir de source de réplication, assurez-vous d'activer les sauvegardes automatiques sur l'instance de base de données source. Pour cela, vous devez

définir la période de rétention des sauvegardes sur une valeur autre que 0. Cette exigence s'applique également à un réplica en lecture qui serait l'instance de base de données source d'un autre réplica en lecture. Les sauvegardes automatiques sont prises en charge pour les répliques en lecture exécutant n'importe quelle version de MySQL. Vous pouvez configurer la réplication en fonction des coordonnées binaires du journal pour une instance My SQL DB.

Vous pouvez configurer la réplication à l'aide des identificateurs de transaction globaux (GTIDS) sur les versions suivantes :

- RDSpour My SQL version 5.7.44 et versions 5.7 supérieures
- RDSpour My SQL version 8.0.28 et versions 8.0 supérieures
- RDSpour My SQL version 8.4.3 et versions 8.4 supérieures

Pour de plus amples informations, veuillez consulter Utilisation de GTID la réplication basée.

Vous pouvez créer jusqu'à 15 réplicas en lecture à partir d'une seule instance de base de données au sein de la même région. Pour que la réplication fonctionne de façon efficace, chaque réplica en lecture doit avoir la même quantité de ressources de calcul et de stockage que l'instance de base de données source. Si vous mettez à l'échelle l'instance de base de données source, faites-le également pour les réplicas en lecture.

RDSpour My SQL supporte les répliques de lecture en cascade. Pour apprendre à configurer des réplicas en lecture en cascade, consultez Utilisation de répliques de lecture en cascade avec for My RDS SQL.

Vous pouvez exécuter simultanément plusieurs actions de création et suppression de réplicas en lecture qui référencent la même instance de base de données source. Lorsque vous effectuez ces actions, restez dans la limite de 15 réplicas en lecture pour chaque instance source.

Une réplique en lecture d'une instance My SQL DB ne peut pas utiliser une version de moteur de base de données inférieure à celle de son instance de base de données source.

Préparation des instances My SQL DB qui utilisent My ISAM

Si votre instance My SQL DB utilise un moteur non transactionnel tel que MyISAM, vous devez effectuer les étapes suivantes pour configurer correctement votre réplique de lecture. Ces étapes sont nécessaires pour vous assurer que le réplica en lecture dispose d'une copie cohérente de vos données. Ces étapes ne sont pas nécessaires si toutes vos tables utilisent un moteur transactionnel comme InnoDB.

1. Arrêtez toutes les opérations du langage de manipulation des données (DML) et du langage de définition des données (DDL) sur les tables non transactionnelles de l'instance de base de données source et attendez qu'elles soient terminées. SELECTles instructions peuvent continuer à s'exécuter.
2. Videz et verrouillez les tables dans l'instance de bases de données source.
3. Créez le réplica en lecture en suivant l'une des méthodes présentées dans les sections suivantes.
4. Vérifiez la progression de la création de la réplique en lecture à l'aide, par exemple, de l'`DescribeDBInstances`APIopération. Une fois que le réplica en lecture est disponible, déverrouillez les tables de l'instance de base de données source et reprenez les opérations de base de données normales.

Configuration des filtres de réplication avec My SQL

Vous pouvez utiliser des filtres de réplication pour spécifier quelles bases de données et tables sont répliquées avec un réplica en lecture. Les filtres de réplication peuvent inclure des bases de données et des tables dans la réplication ou les exclure de la réplication.

Voici quelques cas d'utilisation pour les filtres de réplication :

- Pour réduire la taille d'un réplica en lecture. Avec le filtrage de réplication, vous pouvez exclure les bases de données et les tables qui ne sont pas nécessaires sur le réplica en lecture.
- Pour exclure des bases de données et des tables des réplicas en lecture, pour des raisons de sécurité.
- Pour répliquer différentes bases de données et tables pour des cas d'utilisation spécifiques au niveau de différents réplicas en lecture. Par exemple, vous pouvez utiliser des réplicas en lecture spécifiques pour l'analyse ou le partage.
- Pour une instance de base de données qui a lu des répliques dans différentes bases de données Régions AWS, pour répliquer différentes bases de données ou tables dans différentes. Régions AWS

> **Note**
>
> Vous pouvez également utiliser des filtres de réplication pour spécifier les bases de données et les tables qui sont répliquées avec une instance My SQL DB principale configurée en tant que réplique dans une topologie de réplication entrante. Pour en savoir plus sur cette

configuration, consultez Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

Rubriques

- Définition des paramètres de filtrage de réplication pour RDS for My SQL
- Limites du filtrage de réplication pour RDS for My SQL
- Exemples de filtrage de réplication pour RDS for My SQL
- Affichage des filtres de réplication pour un réplica en lecture

Définition des paramètres de filtrage de réplication pour RDS for My SQL

Pour configurer des filtres de réplication, définissez les paramètres de filtrage de réplication suivants sur le réplica en lecture :

- `replicate-do-db` – Répliquer les modifications apportées aux bases de données spécifiées. Lorsque vous définissez ce paramètre pour un réplica en lecture, seules les bases de données spécifiées dans le paramètre sont répliquées.
- `replicate-ignore-db` – Ne pas répliquer les modifications apportées aux bases de données spécifiées. Lorsque le paramètre `replicate-do-db` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.
- `replicate-do-table` – Répliquer les modifications apportées aux tables spécifiées. Lorsque vous définissez ce paramètre pour un réplica en lecture, seules les tables spécifiées dans le paramètre sont répliquées. En outre, lorsque le paramètre `replicate-do-db` ou `replicate-ignore-db` est défini, assurez-vous d'inclure la base de données qui comprend les tables spécifiées dans la réplication avec le réplica en lecture.
- `replicate-ignore-table` – Ne pas répliquer les modifications apportées aux tables spécifiées. Lorsque le paramètre `replicate-do-table` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.
- `replicate-wild-do-table` – Répliquer les tables en fonction des modèles de nom de base de données et nom de table spécifiés. Les caractères génériques `%` et `_` sont pris en charge. Lorsque le paramètre `replicate-do-db` ou `replicate-ignore-db` est défini, assurez-vous d'inclure la base de données qui comprend les tables spécifiées dans la réplication avec le réplica en lecture.
- `replicate-wild-ignore-table` – Ne pas répliquer les tables en fonction des modèles de nom de base de données et de nom de table spécifiés. Les caractères génériques `%` et `_` sont pris en

charge. Lorsque le paramètre `replicate-do-table` ou `replicate-wild-do-table` est défini pour un réplica en lecture, ce paramètre n'est pas évalué.

Les paramètres sont évalués dans l'ordre dans lequel ils sont répertoriés. Pour plus d'informations sur le fonctionnement de ces paramètres, consultez la section Ma SQL documentation :

- Pour plus d'informations générales, voir Options et variables du serveur de réplication.
- Pour plus d'informations sur la façon dont les paramètres de filtrage de réplication de base de données sont évalués, voir Évaluation des options de réplication au niveau de la base de données et des options de la journalisation binaire.
- Pour plus d'informations sur l'évaluation des paramètres de filtrage de réplication de table, reportez-vous à la section Évaluation des options de réplication au niveau de la table.

Par défaut, chacun de ces paramètres a une valeur vide. Sur chaque réplica en lecture, vous pouvez utiliser ces paramètres pour définir, modifier et supprimer des filtres de réplication. Lorsque vous définissez l'un de ces paramètres, séparez chaque filtre des autres par une virgule.

Vous pouvez utiliser les caractères génériques `%` et _ dans les paramètres `replicate-wild-do-table` et `replicate-wild-ignore-table`. Le caractère générique `%` correspond à un nombre quelconque de caractères, et le caractère générique _ ne correspond qu'à un seul caractère.

Le format de journalisation binaire de l'instance de base de données source est important pour la réplication, car il détermine l'enregistrement des modifications de données. Le réglage du paramètre `binlog_format` détermine si la réplication est basée sur les lignes ou les instructions. Pour de plus amples informations, veuillez consulter Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ.

> **Note**
>
> Toutes les instructions du langage de définition de données (DDL) sont répliquées sous forme d'instructions, quel que soit le `binlog_format` paramètre de l'instance de base de données source.

Limites du filtrage de réplication pour RDS for My SQL

Les limites suivantes s'appliquent au filtrage de réplication pour RDS for My SQL :

- Chaque paramètre de filtrage de réplication a une limite de 2 000 caractères.
- Les virgules ne sont pas prises en charge dans les filtres de réplication pour les valeurs des paramètres. Dans une liste de paramètres, les virgules ne peuvent être utilisées que comme séparateurs de valeurs. Par exemple, `ParameterValue='`a,b`'` n'est pas pris en charge, mais l'`ParameterValue='a,b'`est.
- Les `--binlog-ignore-db` options My SQL `--binlog-do-db` et pour le filtrage binaire des journaux ne sont pas prises en charge.
- Le filtrage de réplication ne prend pas en charge les transactions XA.

  Pour plus d'informations, consultez la section Restrictions relatives aux transactions XA dans la section Ma SQL documentation.

Exemples de filtrage de réplication pour RDS for My SQL

Pour configurer le filtrage de réplication pour un réplica en lecture, modifiez les paramètres de filtrage de réplication dans le groupe de paramètres associé au réplica en lecture.

> **Note**
>
> Vous ne pouvez pas modifier un groupe de paramètres par défaut. Si le réplica en lecture utilise un groupe de paramètres par défaut, créez un nouveau groupe de paramètres et associez-le au réplica en lecture. Pour plus d'informations sur les groupes de paramètres de base de données, consultez Groupes de paramètres pour (Amazon RDS).

Vous pouvez définir les paramètres d'un groupe de paramètres à l'aide de AWS Management Console AWS CLI, ou RDSAPI. Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora). Lorsque vous définissez des paramètres dans un groupe de paramètres, toutes les instances de base de données associées au groupe de paramètres utilisent les réglages des paramètres. Si vous définissez les paramètres de filtrage de réplication dans un groupe de paramètres, assurez-vous que le groupe de paramètres est associé uniquement aux réplicas en lecture. Laissez les paramètres de filtrage de réplication vides pour les instances de base de données source.

Les exemples suivants définissent les paramètres à l'aide de la AWS CLI. Ces exemples sont définis `ApplyMethod` de `immediate` telle sorte que les modifications de paramètres se produisent

immédiatement après la fin de la CLI commande. Si vous souhaitez qu'une modification en attente soit appliquée après le redémarrage du réplica en lecture, définissez `ApplyMethod` sur `pending-reboot`.

Les exemples suivants définissent des filtres de réplication :

- Including databases in replication
- Including tables in replication
- Including tables in replication with wildcard characters
- Excluding databases from replication
- Excluding tables from replication
- Excluding tables from replication using wildcard characters

Example Inclusion de bases de données dans la réplication

L'exemple suivant inclut les bases de données `mydb1` et `mydb2` dans la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-do-
db,ParameterValue='mydb1,mydb2',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-do-
db,ParameterValue='mydb1,mydb2',ApplyMethod=immediate"
```

Example Inclusion de tables dans la réplication

L'exemple suivant inclut les tables `table1` et `table2` dans la base de données `mydb1` dans la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-do-
table,ParameterValue='mydb1.table1,mydb1.table2',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-do-
table,ParameterValue='mydb1.table1,mydb1.table2',ApplyMethod=immediate"
```

Example Inclusion de tables dans la réplication à l'aide de caractères génériques

L'exemple suivant inclut des tables dont les noms commencent par `order` et `return` dans la base de données `mydb` dans la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-wild-do-table,ParameterValue='mydb.order
%,mydb.return%',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-wild-do-table,ParameterValue='mydb.order
%,mydb.return%',ApplyMethod=immediate"
```

Example Exclusion de bases de données de la réplication

L'exemple suivant exclut les bases de données `mydb5` et `mydb6` de la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
```

```
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-ignore-
db,ParameterValue='mydb5,mydb6',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-ignore-
db,ParameterValue='mydb5,mydb6',ApplyMethod=immediate"
```

Example Exclusion de tables de la réplication

L'exemple suivant exclut les tables `table1` dans la base de données `mydb5` et `table2` dans la base de données `mydb6` de la réplication.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-ignore-
table,ParameterValue='mydb5.table1,mydb6.table2',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-ignore-
table,ParameterValue='mydb5.table1,mydb6.table2',ApplyMethod=immediate"
```

Example Exclusion de tables de la réplication à l'aide des caractères génériques

L'exemple suivant exclut de la réplication les tables dont les noms commencent par `order` et `return` dans la base de données `mydb7`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
```

```
  --db-parameter-group-name myparametergroup \
  --parameters "ParameterName=replicate-wild-ignore-table,ParameterValue='mydb7.order
%,mydb7.return%',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myparametergroup ^
  --parameters "ParameterName=replicate-wild-ignore-table,ParameterValue='mydb7.order
%,mydb7.return%',ApplyMethod=immediate"
```

Affichage des filtres de réplication pour un réplica en lecture

Vous pouvez afficher les filtres de réplication pour un réplica en lecture de la manière suivante :

- Vérifiez les réglages des paramètres de filtrage de réplication dans le groupe de paramètres associé au réplica en lecture.

  Pour obtenir des instructions, consultez Affichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon.
- Dans un SQL client My, connectez-vous à la réplique lue et exécutez l'`SHOW REPLICA STATUS`instruction.

  Dans la sortie, les champs suivants affichent les filtres de réplication pour le réplica en lecture :
  - `Replicate_Do_DB`
  - `Replicate_Ignore_DB`
  - `Replicate_Do_Table`
  - `Replicate_Ignore_Table`
  - `Replicate_Wild_Do_Table`
  - `Replicate_Wild_Ignore_Table`

  Pour plus d'informations sur ces champs, consultez la section Vérification de l'état de réplication dans la section Ma SQL documentation.

Configuration de la réplication différée avec My SQL

Vous pouvez utiliser la réplication retardée comme stratégie pour la reprise après sinistre. Avec la réplication retardée, vous spécifiez la durée minimale, en secondes, pour retarder la réplication de la

source vers la réplique de lecture. En cas de sinistre, par exemple la suppression accidentelle d'une table, vous appliquez la procédure suivante pour reprendre rapidement après le sinistre :

- Arrêtez la réplication vers le réplica en lecture avant que lui soit envoyée la modification qui a provoqué le sinistre.

  Utilisez la procédure stockée mysql.rds_stop_replication pour arrêter la réplication.
- Arrêtez la réplication et précisez qu'elle doit s'arrêter automatiquement à une position donnée dans un fichier journal.

  Vous indiquez une position juste avant le sinistre grâce à la procédure stockée .
- Effectuez la promotion du réplica en lecture pour qu'il devienne la nouvelle instance de base de données source, en suivant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

> **Note**
>
> - Sur RDS My SQL 8.4, la réplication différée est prise en charge pour My SQL 8.4.3 et versions ultérieures. Sur RDS My SQL 8.0, la réplication différée est prise en charge pour My SQL 8.0.28 et versions ultérieures. Sur RDS My SQL 5.7, la réplication différée est prise en charge pour My SQL 5.7.44 et versions ultérieures.
> - Utilisez des procédures stockées pour configurer la réplication retardée. Vous ne pouvez pas configurer la réplication différée avec le AWS Management Console AWS CLI, le ou Amazon RDSAPI.
> - Vous pouvez utiliser la réplication basée sur les identificateurs de transaction globaux (GTIDs) dans une configuration de réplication différée pour les versions suivantes :
>   - RDSpour My SQL version 5.7.44 et versions 5.7 supérieures
>   - RDSpour My SQL version 8.0.28 et versions 8.0 supérieures
>   - RDSpour My SQL version 8.4.3 et versions 8.4 supérieures
>
>   Si vous utilisez la réplication GTID basée, utilisez la procédure stockée au lieu de la procédure stockée. Pour plus d'informations sur la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

Rubriques

- Configuration de la réplication retardée pendant la création du réplica en lecture
- Modification de la réplication retardée pour un réplica en lecture existant
- Définition d'une position où arrêter la réplication vers un réplica en lecture
- Promotion d'un réplica en lecture

Configuration de la réplication retardée pendant la création du réplica en lecture

Pour configurer la réplication retardée pour tout réplica en lecture à venir créé à partir d'une instance de base de données, exécutez la procédure stockée mysql.rds_set_configuration avec le paramètre `target delay`.

Pour configurer la réplication retardée pendant la création du réplica en lecture

1. À l'aide d'un SQL client My, connectez-vous à l'instance My SQL DB pour être la source des répliques de lecture en tant qu'utilisateur principal.
2. Exécutez la procédure stockée mysql.rds_set_configuration avec le paramètre `target delay`.

   Par exemple, exécutez la procédure stockée suivante pour indiquer que la réplication est retardée d'au moins une heure (3 600 secondes) pour tout réplica en lecture créé à partir de l'instance de base de données actuelle.

   ```
   call mysql.rds_set_configuration('target delay', 3600);
   ```

   > **Note**
   >
   > Après avoir exécuté cette procédure stockée, toute réplique en lecture que vous créez à l'aide de AWS CLI ou Amazon RDS API est configurée avec un délai de réplication du nombre de secondes spécifié.

Modification de la réplication retardée pour un réplica en lecture existant

Pour modifier la réplication retardée pour un réplica en lecture existant, exécutez la procédure stockée mysql.rds_set_source_delay.

Pour modifier la réplication retardée pour un réplica en lecture existant

1. À l'aide d'un SQL client My, connectez-vous à la réplique lue en tant qu'utilisateur principal.

2. Utilisez la procédure stockée mysql.rds_stop_replication pour arrêter la réplication.
3. Exécutez la procédure stockée mysql.rds_set_source_delay.

   Par exemple, exécutez la procédure stockée suivante pour indiquer que la réplication vers le réplica en lecture est retardée d'au moins une heure (3 600 secondes).

   ```
   call mysql.rds_set_source_delay(3600);
   ```
4. Utilisez la procédure stockée mysql.rds_start_replication pour lancer la réplication.

Définition d'une position où arrêter la réplication vers un réplica en lecture

Après avoir arrêté la réplication vers le réplica en lecture, vous pouvez démarrer la réplication, puis l'arrêter à la position spécifiée dans le fichier journal binaire en utilisant la procédure stockée .

Pour démarrer la réplication vers un réplica en lecture et l'arrêter à une position donnée

1. À l'aide d'un SQL client My, connectez-vous à l'instance My SQL DB source en tant qu'utilisateur principal.
2. Exécutez la procédure stockée .

   L'exemple suivant lance la réplication et réplique les modifications jusqu'à ce qu'il atteigne la position `120` dans le fichier journal binaire `mysql-bin-changelog.000777`. Dans un scénario de reprise après sinistre, nous supposons que cette position `120` est juste avant le sinistre.

   ```
   call mysql.rds_start_replication_until(
     'mysql-bin-changelog.000777',
     120);
   ```

La réplication s'arrête automatiquement lorsque le point d'arrêt est atteint. L'RDSévénement suivant est généré :`Replication has been stopped since the replica reached the stop point specified by the rds_start_replication_until stored procedure`.

Promotion d'un réplica en lecture

Après l'arrêt de la réplication, dans un scénario de reprise après sinistre, vous pouvez promouvoir un réplica en lecture comme nouvelle instance de base de données source. Pour de plus amples informations sur la promotion d'un réplica en lecture, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.

Mettre à jour les répliques de lecture avec My SQL

Les réplicas en lecture sont conçus pour prendre en charge les requêtes de lecture, mais vous pouvez avoir besoin de mises à jour ponctuelles. À titre d'exemple, vous pouvez avoir besoin d'ajouter un index, pour optimiser les types spécifiques de requêtes qui accèdent au réplica.

Bien que vous puissiez activer les mises à jour en définissant le paramètre `read_only` sur `0` dans le groupe de paramètres de base de données pour le réplica en lecture, nous vous recommandons de ne pas le faire car cela peut poser des problèmes si le réplica en lecture devient incompatible avec l'instance de base de données source. Pour les opérations de maintenance, nous vous recommandons d'utiliser des déploiements bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements bleu/vert pour les mises à jour de base de données.

Si vous désactivez la lecture seule sur un réplica en lecture, modifiez la valeur du paramètre 1 pour rétablir `read_only` dès que possible.

Utilisation de déploiements de répliques de lecture multi-AZ avec My SQL

Vous pouvez créer un réplica en lecture à partir de déploiements d'instance de base de données mono-AZ ou multi-AZ. Vous utilisez des déploiements multi-AZ pour améliorer la durabilité et la disponibilité des données critiques, mais vous ne pouvez pas utiliser une instance secondaire multi-AZ pour servir les requêtes en lecture seule. À la place, vous pouvez créer des réplicas en lecture à partir d'instances de base de données multi-AZ à trafic élevé pour décharger les requêtes en lecture seule. Si l'instance source d'un déploiement multi-AZ bascule vers l'instance secondaire, tous les réplicas en lecture associés se mettent automatiquement à utiliser l'instance secondaire (désormais principale) comme source de réplication. Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Vous pouvez créer un réplica en lecture en tant qu'instance de base de données Multi-AZ. Amazon RDS met en veille votre réplique dans une autre zone de disponibilité afin de prendre en charge le basculement de la réplique. La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.

Utilisation de répliques de lecture en cascade avec for My RDS SQL

RDSpour My SQL supporte les répliques de lecture en cascade. Avec les répliques de lecture en cascade, vous pouvez redimensionner les lectures sans surcharger votre source RDS pour l'instance My SQL DB.

Avec les répliques de lecture en cascade, votre instance RDS for My SQL DB envoie des données à la première réplique de lecture de la chaîne. Ce réplica en lecture envoie ensuite les données au deuxième réplica de la chaîne, etc. Le résultat final est que toutes les répliques de lecture de la chaîne présentent les mêmes modifications que celles de l'instance RDS for My SQL DB, mais sans la surcharge uniquement sur l'instance de base de données source.

Vous pouvez créer une série de trois répliques de lecture au maximum dans une chaîne à partir d'une source RDS pour l'instance My SQL DB. Supposons, par exemple, que vous disposiez d'une instance RDS for My SQL DB,`mysql-main`. Vous pouvez effectuer les actions suivantes :

- À partir de `mysql-main`, créez le premier réplica en lecture de la chaîne, `read-replica-1`.
- Ensuite, à partir de `read-replica-1`, créez le réplica en lecture suivant dans la chaîne, `read-replica-2`.
- Enfin, à partir de `read-replica-2`, créez le troisième réplica en lecture de la chaîne, `read-replica-3`.

Vous ne pouvez pas créer un autre réplica en lecture au-delà de ce troisième réplica en lecture en cascade dans la série pour `mysql-main`. Une série complète d'instances allant d'une instance de base de données RDS for My SQL source à la fin d'une série de répliques de lecture en cascade peut comprendre au maximum quatre instances de base de données.

Pour que les répliques de lecture en cascade fonctionnent, les sauvegardes automatiques doivent être RDS activées pour chaque source de l'instance My SQL DB. Pour activer les sauvegardes automatiques sur un réplica en lecture, commencez par créer le réplica en lecture, puis modifiez-le pour activer les sauvegardes automatiques. Pour de plus amples informations, veuillez consulter Création d'un réplica en lecture.

Comme pour tout réplica en lecture, vous pouvez promouvoir un réplica en lecture faisant partie d'une cascade. La promotion d'un réplica en lecture depuis une chaîne de réplicas en lecture retire ce réplica de la chaîne. Par exemple, supposons que vous souhaitez déplacer une partie de la charge de travail de votre instance de base de données `mysql-main` vers une nouvelle instance destinée uniquement au service comptable. En prenant pour hypothèse la chaîne de trois réplicas en lecture de l'exemple, vous décidez de promouvoir `read-replica-2`. La chaîne est affectée comme suit :

- La promotion de `read-replica-2` le retire de la chaîne de réplication.
  - Il s'agit désormais d'une instance de base de données en lecture/écriture complète.
  - La réplication continue sur `read-replica-3`, tout comme avant la promotion.

- Votre `mysql-main` continue la réplication sur `read-replica-1`.

Pour plus d'informations sur la promotion des réplicas en lecture, consultez Promotion d'un réplica en lecture en instance de bases de données autonome.

Surveillance du délai de réplication pour les répliques My SQL read

Pour les répliques My SQL read, vous pouvez surveiller le délai de réplication sur Amazon en CloudWatch consultant la RDS `ReplicaLag` métrique Amazon. La métrique `ReplicaLag` contient la valeur du champ `Seconds_Behind_Master` de la commande `SHOW REPLICA STATUS`.

Les causes courantes du retard de réplication pour My SQL sont les suivantes :

- Une indisponibilité du réseau.
- L'écriture dans des tables avec des index différents sur un réplica en lecture. Si le paramètre `read_only` est défini sur `0` sur le réplica en lecture, la réplication peut être rompue si le réplica en lecture devient incompatible avec l'instance de base de données source. Une fois que vous avez effectué les tâches de maintenance sur le réplica en lecture, nous vous recommandons de définir à nouveau le paramètre `read_only` sur `1`.
- En utilisant un moteur de stockage non transactionnel tel que My. ISAM La réplication n'est prise en charge que pour le moteur de stockage InnoDB sur My. SQL

Lorsque la métrique `ReplicaLag` atteint 0, le réplica a rattrapé l'instance de bases de données source. Si la métrique `ReplicaLag` retourne -1, la réplication n'est actuellement pas active. `ReplicaLag`= -1 est équivalent à `Seconds_Behind_Master` = `NULL`.

Démarrage et arrêt de la réplication avec My SQL read replicas

Vous pouvez arrêter et redémarrer le processus de réplication sur une RDS instance de base de données Amazon en appelant les procédures stockées du système mysql.rds_stop_replication etmysql.rds_start_replication. Vous pouvez le faire lors de la réplication entre deux RDS instances Amazon pour des opérations de longue durée, telles que la création d'index volumineux. Vous devez également arrêter et démarrer la réplication lors de l'importation ou de l'exportation de bases de données. Pour plus d’informations, consultez Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt et Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Si la réplication est arrêtée pendant plus de 30 jours consécutifs, soit manuellement, soit en raison d'une erreur de réplication, Amazon RDS met fin à la réplication entre l'instance de base de données source et toutes les répliques en lecture. Cela permet d'éviter l'augmentation des besoins en stockage sur l'instance de bases de données source et d'importants délais de basculement. L'instance de base de données du réplica en lecture est toujours disponible. En revanche, la réplication ne peut pas être reprise, car les journaux binaires requis par le réplica en lecture sont supprimés de l'instance de base de données source une fois la réplication terminée. Vous pouvez créer un nouveau réplica en lecture pour l'instance de base de données source afin de rétablir la réplication.

Résolution d'un problème lié à SQL la réplique My read

Pour les instances My SQL DB, dans certains cas, les répliques en lecture présentent des erreurs de réplication ou des incohérences de données (ou les deux) entre la réplique en lecture et son instance de base de données source. Ce problème survient quand des événements de journaux binaires ou des journaux redo InnoDB ne sont pas vidés lors d'une panne du réplica en lecture ou de l'instance de base de données source. Dans ces situations, supprimez et recréez manuellement les réplicas en lecture. Vous pouvez réduire la probabilité que cela se produise en définissant les valeurs de paramètre suivantes : `sync_binlog=1` et `innodb_flush_log_at_trx_commit=1`. Ces paramètres peuvent réduire les performances. Testez donc leur impact avant d'implémenter les modifications dans un environnement de production.

> **Warning**
>
> Dans le groupe de paramètres associé à l'instance de base de données source, nous recommandons de conserver ces valeurs de paramètres : `sync_binlog=1` et `innodb_flush_log_at_trx_commit=1`. Ces paramètres sont dynamiques. Si vous ne souhaitez pas utiliser ces paramètres, nous vous recommandons de définir temporairement ces valeurs avant d'exécuter toute opération sur l'instance de base de données source susceptible de provoquer son redémarrage. Ces opérations incluent, sans s'y limiter, le redémarrage, le redémarrage avec basculement, la mise à niveau de la version de la base de données et la modification de la classe d'instance de base de données ou de son stockage. La même recommandation s'applique à la création de nouveaux réplicas en lecture pour l'instance de base de données source.
>
> Le non-respect de ces instructions augmente le risque que les réplicas en lecture présentent des erreurs ou des incohérences de données (ou les deux) entre le réplica en lecture et son instance de base de données source.

Les technologies de réplication pour My SQL sont asynchrones. Par conséquent, des augmentations `BinLogDiskUsage` sur l'instance de base de données source et `ReplicaLag` sur le réplica en lecture sont prévisibles. Par exemple, un volume élevé d'opérations d'écriture sur l'instance de bases de données source peut se produire en parallèle. Tandis que les opérations d'écritures sur le réplica en lecture sont sérialisées à l'aide d'un seul thread d'I/O, ce qui peut conduire à un retard entre l'instance source et le réplica. Pour plus d'informations sur les répliques en lecture seule dans ma SQL documentation, consultez la section Détails de l'implémentation de la réplication.

Vous pouvez effectuer plusieurs opérations pour réduire le retard entre les mises à jour d'une instance de base de données source et les mises à jour suivantes appliquées au réplica en lecture, telles que les opérations suivantes :

- Dimensionnement d'un réplica en lecture pour qu'il ait une taille de stockage et une classe d'instance de base de données comparables à celles de l'instance de base de données source.
- Garantie que les réglages des paramètres dans les groupes de paramètres de base de données utilisés par l'instance de base de données source et le réplica en lecture sont compatibles. Pour obtenir plus d'informations et un exemple, reportez-vous à la présentation du paramètre `max_allowed_packet`, plus loin dans cette section.

Amazon RDS surveille l'état de réplication de vos répliques de lecture et met à jour le `Replication State` champ de l'instance de réplication de lecture en `Error` cas d'arrêt de la réplication pour une quelconque raison. Par exemple, si les DML requêtes exécutées sur votre réplique de lecture entrent en conflit avec les mises à jour effectuées sur l'instance de base de données source.

Vous pouvez consulter les détails de l'erreur associée générée par le SQL moteur My en consultant le `Replication Error` champ. Des événements indiquant l'état du réplica en lecture sont également générés, y compris RDS-EVENT-0045, RDS-EVENT-0046 et RDS-EVENT-0047. Pour plus d'informations sur les événements et l'abonnement aux événements, consultez Utilisation des notifications d'RDSévénements Amazon. Si un message Mon SQL erreur est renvoyé, consultez le numéro d'erreur dans la documentation Mon message SQL d'erreur.

Un problème courant susceptible de causer des erreurs de réplication se pose lorsque la valeur du paramètre `max_allowed_packet` d'un réplica en lecture est inférieure à celle du paramètre `max_allowed_packet` de l'instance de base de données source. Le paramètre `max_allowed_packet` est un paramètre personnalisé que vous pouvez définir dans un groupe de paramètres de base de données. Vous l'utilisez `max_allowed_packet` pour spécifier la taille maximale du DML code pouvant être exécuté sur la base de données. Dans certains cas, la valeur `max_allowed_packet` du groupe de paramètres de base de données associé à un réplica en

lecture est inférieure à la valeur `max_allowed_packet` du groupe de paramètres de base de données associé à l'instance de base de données source. Dans ces cas, le processus de réplication peut lancer l'erreur `Packet bigger than 'max_allowed_packet' bytes` et arrêter la réplication. Pour corriger cette erreur, faites en sorte que l'instance de base de données source et le réplica en lecture utilisent des groupes de paramètres de base de données avec les mêmes valeurs pour le paramètre `max_allowed_packet`.

Voici d'autres situations courantes susceptibles de causer des erreurs de réplication :

- Écriture sur les tables d'un réplica en lecture. Dans certains cas, vous pouvez créer des index sur un réplica en lecture différents des index sur l'instance de base de données source. Vous devez alors définir le paramètre `read_only` sur `0` pour créer les index. Si vous écrivez dans des tables sur le réplica en lecture, cela peut interrompre la réplication si le réplica en lecture devient incompatible avec l'instance de base de données source. Une fois que vous avez effectué les tâches de maintenance sur le réplica en lecture, nous vous recommandons de définir à nouveau le paramètre `read_only` sur `1`.
- En utilisant un moteur de stockage non transactionnel tel que My. ISAM Les réplicas en lecture nécessitent un moteur de stockage transactionnel. La réplication n'est prise en charge que pour le moteur de stockage InnoDB sur My. SQL
- Utilisation de requêtes non déterministes non sécurisées telles que `SYSDATE()`. Pour de plus amples informations, consultez Determination of safe and unsafe statements in binary logging.

Si vous décidez que vous pouvez ignorer une erreur en toute sécurité, vous pouvez suivre la procédure décrite dans la section Ignorer l'erreur de réplication actuelle pour RDS for My SQL. Sinon, vous pouvez d'abord supprimer le réplica en lecture. Vous créez ensuite une instance à l'aide du même identifiant d'instance de base de données, de telle sorte que le point de terminaison demeure le même que celui de votre ancien réplica en lecture. Si une erreur de réplication est corrigée, le champ `Replication State` prend la valeur replicating (réplication en cours).

Utilisation de GTID la réplication basée

Le contenu suivant explique comment utiliser les identificateurs de transaction globaux (GTIDs) avec la réplication du journal binaire (binlog) entre les instances Amazon RDS for My SQL DB.

Si vous utilisez la réplication binlog et que vous n'êtes pas familiarisé avec la réplication GTID basée sur MySQL, consultez la section Réplication avec des identifiants de transaction globaux dans la documentation MySQL.

GTIDla réplication basée sur la réplication est prise en charge pour les versions suivantes :

- Tout RDS pour les versions My SQL 8.4
- Tout RDS pour les versions My SQL 8.0
- Tout RDS pour les versions My SQL 5.7

Toutes les instances My SQL DB d'une configuration de réplication doivent répondre à cette exigence de version.

Rubriques

- Vue d'ensemble des identifiants de transactions globaux () GTIDs
- Paramètres pour la réplication GTID basée
- Activation GTID de la réplication basée pour les nouvelles répliques de lecture pour RDS for My SQL
- Activation GTID de la réplication basée sur les répliques de lecture existantes pour RDS for My SQL
- Désactivation de la réplication GTID basée pour une instance My SQL DB avec des répliques en lecture et un DB

Vue d'ensemble des identifiants de transactions globaux () GTIDs

Les identifiants de transaction globaux (GTIDs) sont des identifiants uniques générés pour les transactions Mes SQL transactions validées. Vous pouvez l'utiliser GTIDs pour simplifier la réplication des journaux binaires et faciliter le dépannage.

My SQL utilise deux types de transactions différents pour la réplication des journaux binaires :

- GTIDtransactions — Transactions identifiées par unGTID.
- Transactions anonymes — Transactions pour lesquelles aucun identifiant n'a GTID été attribué.

Dans une configuration de réplication, GTIDs ils sont uniques dans toutes les instances de base de données. GTIDssimplifiez la configuration de la réplication, car lorsque vous les utilisez, vous n'avez pas à vous référer aux positions des fichiers journaux. GTIDsfacilitent également le suivi des transactions répliquées et permettent de déterminer si l'instance source et les répliques sont cohérentes.

Vous pouvez utiliser la réplication GTID basée pour répliquer les données avec RDS for My SQL read replicas. Vous pouvez configurer la réplication GTID basée lorsque vous créez de nouvelles répliques de lecture, ou vous pouvez convertir des répliques de lecture existantes en une réplication GTID basée sur l'utilisation.

Vous pouvez également utiliser la réplication GTID basée dans une configuration de réplication différée avec RDS for MySQL. Pour de plus amples informations, veuillez consulter Configuration de la réplication différée avec My SQL.

Paramètres pour la réplication GTID basée

Utilisez les paramètres suivants pour configurer la réplication GTID basée.

Paramètre	Valeurs valides	Description
`gtid_mode`	`OFF`, `OFF_PERMISSIVE` , `ON_PERMISSIVE` , `ON`	`OFF`indique que les nouvelles transactions sont des transactions anonymes (c'est-à-dire qu'elles n'en ont pasGTIDs) et qu'une transaction doit être anonyme pour être répliquée. `OFF_PERMISSIVE` spécifie que les nouvelles transactions sont des transactions anonymes, mais que toutes les transactions peuvent être répliquées. `ON_PERMISSIVE` indique que les nouvelles transactions sont des GTID transactions, mais que toutes les transactions peuvent être répliquées. `ON`indique que les nouvelles transactions sont GTID des transactions et qu'une transaction doit être une GTID transaction pour être répliquée.
`enforce_gtid_consistency`	`OFF`, `ON`, `WARN`	`OFF`permet aux transactions de violer GTID la cohérence. `ON`empêche les transactions de violer GTID la cohérence.

Paramètre	Valeurs valides	Description
		`WARN`autorise les transactions à violer GTID la cohérence, mais génère un avertissement en cas de violation.

> **Note**
>
> Dans le AWS Management Console, le `gtid_mode` paramètre apparaît sous la forme`gtid-mode`.

Pour la réplication GTID basée, utilisez ces paramètres pour le groupe de paramètres de votre instance de base de données ou lisez la réplique :

- `ON`Net `ON_PERMISSIVE` s'appliquent uniquement à la réplication sortante à partir d'une RDS instance de base de données. Ces deux valeurs obligent votre RDS instance de base de données à être utilisée GTIDs pour les transactions répliquées. `ON`nécessite que la base de données cible utilise également la réplication GTID basée. `ON_PERMISSIVE`rend la réplication GTID basée facultative sur la base de données cible.
- `OFF_PERMISSIVE`, s'il est défini, signifie que vos RDS instances de base de données peuvent accepter la réplication entrante à partir d'une base de données source. Ils peuvent le faire indépendamment du fait que la base de données source utilise ou non la réplication GTID basée.
- `OFF`, s'il est défini, signifie que votre RDS instance de base de données n'accepte que la réplication entrante provenant de bases de données sources qui n'utilisent pas de réplication GTID basée.

Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Activation GTID de la réplication basée pour les nouvelles répliques de lecture pour RDS for My SQL

Lorsque la réplication GTID basée est activée pour une instance RDS for My SQL DB, la réplication GTID basée est configurée automatiquement pour les répliques en lecture de l'instance de base de données.

Pour activer la réplication GTID basée sur les nouvelles répliques de lecture

1. Assurez-vous que le groupe de paramètres associé à l'instance de base de données contient la configuration de paramètres suivante :

   - `gtid_mode` – `ON` ou `ON_PERMISSIVE`
   - `enforce_gtid_consistency` – `ON`

   Pour plus d'informations sur la définition des paramètres de configuration à l'aide de groupes de paramètres, veuillez consulter Groupes de paramètres pour (Amazon RDS).
2. Si vous avez modifié le groupe de paramètres de l'instance de base de données, redémarrez celle-ci. Pour en savoir plus à ce sujet, veuillez consulter Redémarrage d'une instance de base de données.
3. Créez un ou plusieurs réplicas en lecture de l'instance de base de données. Pour en savoir plus à ce sujet, veuillez consulter Création d'un réplica en lecture.

Amazon RDS tente d'établir une réplication GTID basée entre l'instance My SQL DB et les répliques de lecture à l'aide du`MASTER_AUTO_POSITION`. Si la tentative échoue, Amazon RDS utilise les positions des fichiers journaux pour la réplication avec les répliques lues. Pour plus d'informations sur le`MASTER_AUTO_POSITION`, consultez la section GTIDPositionnement automatique dans la section Ma SQL documentation.

Activation GTID de la réplication basée sur les répliques de lecture existantes pour RDS for My SQL

Pour une instance My SQL DB existante avec des répliques en lecture qui n'utilisent pas de réplication GTID basée, vous pouvez configurer la réplication GTID basée entre l'instance de base de données et les répliques en lecture.

Pour activer la réplication GTID basée sur les répliques de lecture existantes

1. Si l'instance de base de données ou toute réplique de lecture utilise une version 8.0 de RDS for My inférieure à 8.0.26, mettez à niveau l'instance de base de données ou la réplique de lecture vers la SQL version 8.0.26 ou une version ultérieure de My 8.0. SQL Toutes RDS les versions SQL 8.4 et 5.7 de My prennent en charge la réplication GTID basée sur la réplication.

Pour de plus amples informations, veuillez consulter Mises à niveau du moteur de base de données RDS pour MySQL.

2. (Facultatif) Réinitialisez les GTID paramètres, testez le comportement de l'instance de base de données et lisez les répliques :

   a. Assurez-vous que le groupe de paramètres associé à l'instance de base de données et à chaque réplica en lecture contient le paramètre `enforce_gtid_consistency` défini sur `WARN`.

      Pour plus d'informations sur la définition des paramètres de configuration à l'aide de groupes de paramètres, veuillez consulter Groupes de paramètres pour (Amazon RDS).

   b. Si vous modifiez le groupe de paramètres de l'instance de base de données, redémarrez celle-ci. Si vous modifiez le groupe de paramètres pour un réplica en lecture, redémarrez celui-ci.

      Pour plus d'informations, consultez Redémarrage d'une instance de base de données.

   c. Exécutez votre instance de base de données et vos réplicas en lecture avec votre charge de travail normale et surveillez les fichiers journaux.

      Si des avertissements concernant des transactions GTID incompatibles s'affichent, ajustez votre application pour qu'elle n'utilise que des fonctionnalités GTID compatibles. Assurez-vous que l'instance de base de données ne génère aucun avertissement concernant les transactions GTID incompatibles avant de passer à l'étape suivante.

3. Réinitialisez les GTID paramètres de la réplication GTID basée qui autorise les transactions anonymes jusqu'à ce que les répliques en lecture les aient toutes traitées.

   a. Assurez-vous que le groupe de paramètres associé à l'instance de base de données et à chaque réplica en lecture contient la configuration de paramètres suivante :

      - `gtid_mode` – `ON_PERMISSIVE`
      - `enforce_gtid_consistency` – `ON`

   b. Si vous modifiez le groupe de paramètres de l'instance de base de données, redémarrez celle-ci. Si vous modifiez le groupe de paramètres pour un réplica en lecture, redémarrez celui-ci.

4. Attendez que toutes vos transactions anonymes soient répliquées. Pour vérifier qu'elles ont été répliquées, procédez comme suit :

a. Exécutez l'instruction suivante sur votre instance de base de données source.

Mon SQL 8,4

```
SHOW BINARY LOG STATUS;
```

Mes SQL 5.7 et 8.0

```
SHOW MASTER STATUS;
```

Notez les valeurs dans les colonnes `File` et `Position`.

b. Sur chaque réplica en lecture, utilisez les informations de fichier et de position de instance source lors de l'étape précédente pour exécuter la requête suivante.

```
SELECT MASTER_POS_WAIT('file', position);
```

Par exemple, si votre fichier se nomme `mysql-bin-changelog.000031` et que sa position est `107`, exécutez l'instruction suivante.

```
SELECT MASTER_POS_WAIT('mysql-bin-changelog.000031', 107);
```

Si le réplica en lecture a dépassé la position spécifiée, la requête renvoie immédiatement un résultat. Sinon, la fonction attend. Passez à l'étape suivante lorsque la requête a renvoyé un résultat pour tous les réplicas en lecture.

5. Réinitialisez les GTID paramètres pour la réplication GTID basée uniquement.

a. Assurez-vous que le groupe de paramètres associé à l'instance de base de données et à chaque réplica en lecture contient la configuration de paramètres suivante :

- `gtid_mode` – `ON`
- `enforce_gtid_consistency` – `ON`

b. Redémarrez l'instance de base de données et chaque réplica en lecture.

6. Sur chaque réplica en lecture, exécutez la procédure suivante.

Mes versions majeures SQL 8.4 et supérieures

```
CALL mysql.rds_set_source_auto_position(1);
```

Mes SQL versions majeures 8.0 et antérieures

```
CALL mysql.rds_set_master_auto_position(1);
```

Désactivation de la réplication GTID basée pour une instance My SQL DB avec des répliques en lecture et un DB

Vous pouvez désactiver la réplication GTID basée pour une instance My SQL DB avec des répliques de lecture.

Pour désactiver la réplication GTID basée sur , une instance My SQL DB avec des répliques de lecture

1. Sur chaque réplique lue, exécutez la procédure suivante :

   Mes versions majeures SQL 8.4 et supérieures

   ```
   CALL mysql.rds_set_source_auto_position(0);
   ```

   Mes SQL versions majeures 8.0 et antérieures

   ```
   CALL mysql.rds_set_master_auto_position(0);
   ```

2. Réinitialisez `gtid_mode` sur `ON_PERMISSIVE`.

   a. Assurez-vous que le groupe de paramètres associé à l'instance My SQL DB et à chaque réplique de lecture est `gtid_mode` défini sur`ON_PERMISSIVE`.

      Pour plus d'informations sur la définition des paramètres de configuration à l'aide de groupes de paramètres, veuillez consulter Groupes de paramètres pour (Amazon RDS).

   b. Redémarrez l'instance My SQL DB et chaque réplique lue. Pour de plus amples informations sur le redémarrage, veuillez consulter Redémarrage d'une instance de base de données.

3. Réinitialisez `gtid_mode` sur `OFF_PERMISSIVE`.

a. Assurez-vous que le groupe de paramètres associé à l'instance My SQL DB et à chaque réplique de lecture est `gtid_mode` défini sur`OFF_PERMISSIVE`.

b. Redémarrez l'instance My SQL DB et chaque réplique lue.

4. Attendez que toutes les GTID transactions soient appliquées à toutes les répliques lues. Pour vérifier qu'elles sont appliquées, procédez comme suit :

a. Sur l'instance My SQL DB, exécutez la commande suivante :

Mon SQL 8,4

```
SHOW BINARY LOG STATUS
```

Mes SQL 5.7 et 8.0

```
SHOW MASTER STATUS
```

Votre sortie doit être similaire à la sortie suivante.

```
File                        Position
-------------------------------------
mysql-bin-changelog.000031      107
-------------------------------------
```

Notez le fichier et la position dans votre sortie.

b. Sur chaque réplique lue, utilisez le fichier et les informations de position de son instance source à l'étape précédente pour exécuter la requête suivante :

Mes SQL versions 8.4 et My SQL 8.0.26 et supérieures Mes versions 8.0 SQL

```
SELECT SOURCE_POS_WAIT('file', position);
```

Mon SQL 5.7

```
SELECT MASTER_POS_WAIT('file', position);
```

Par exemple, si le nom du fichier est `mysql-bin-changelog.000031` et sa position l'est`107`, exécutez l'instruction suivante :

Mes SQL versions 8.4 et My SQL 8.0.26 et supérieures Mes versions 8.0 SQL

```
SELECT SOURCE_POS_WAIT('mysql-bin-changelog.000031', 107);
```

Mon SQL 5.7

```
SELECT MASTER_POS_WAIT('mysql-bin-changelog.000031', 107);
```

5. Réinitialisez les GTID paramètres pour désactiver la réplication GTID basée.

   a. Assurez-vous que le groupe de paramètres associé à l'instance My SQL DB et à chaque réplique de lecture possède les paramètres suivants :

      - `gtid_mode` – `OFF`
      - `enforce_gtid_consistency` – `OFF`

   b. Redémarrez l'instance My SQL DB et chaque réplique lue.

Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Vous pouvez configurer la réplication entre une instance de base de données RDS for My SQL ou MariaDB et une instance My SQL ou MariaDB externe à Amazon à RDS l'aide de la réplication de fichiers journaux binaires.

Rubriques

- Avant de commencer
- Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Avant de commencer

Vous pouvez configurer la réplication en utilisant la position du fichier journal binaire des transactions répliquées.

Les autorisations requises pour démarrer la réplication sur une RDS instance de base de données Amazon sont limitées et ne sont pas accessibles à votre utilisateur RDS principal Amazon. Pour cette raison, assurez-vous d'utiliser les mysql.rds_start_replication commandes Amazon RDS ou et pour configurer la réplication entre votre base de données active et votre base de RDS données Amazon.

Pour définir le format de journalisation binaire pour une base de données My SQL ou MariaDB, mettez à jour le paramètre. `binlog_format` Si votre instance de base de données utilise le groupe de paramètres d'instance de base de données par défaut, créez un nouveau groupe de paramètres de base de données pour modifier le `binlog_format` paramètre. Dans MariaDB et SQL My 8.0 et versions antérieures`binlog_format`, la valeur par défaut est. `MIXED` Cependant, vous pouvez également définir `binlog_format` sur `ROW` ou `STATEMENT` si vous avez besoin d'un format de journaux binaires (binlog) spécifique. Redémarrez votre instance de base de données pour que les modifications prennent effet. Dans My SQL 8.4 et versions supérieures, la `binlog_format` valeur par défaut est. `ROW`

Pour plus d'informations sur la configuration du paramètre `binlog_format`, consultez la section Configuration d'RDS pour la journalisation binaire MySQL pour les bases de données mono-AZ. Pour plus d'informations sur les implications des différents types de SQL réplication My, consultez la section Avantages et inconvénients de la réplication basée sur des instructions et basée sur des lignes dans la documentation My. SQL

Configuration d'une réplication de position de fichier journal binaire avec une instance source externe

Suivez ces directives lorsque vous configurez une instance source externe et une réplique sur Amazon RDS :

- Surveillez les événements de basculement pour l'RDSinstance de base de données Amazon qui est votre réplique. En cas de basculement, l'instance de base de données qui est votre réplica peut alors être recréée sur un nouvel hôte avec une autre adresse réseau. Pour plus d'informations sur la surveillance des événements de basculement, consultez Utilisation des notifications d'RDSévénements Amazon.
- Tenez à jour les journaux binaires sur votre instance source jusqu'à ce que vous ayez vérifié qu'ils ont été appliqués au réplica. Cette maintenance garantit que vous pouvez restaurer votre instance source en cas de défaillance.
- Activez les sauvegardes automatisées sur votre RDS instance de base de données Amazon. L'activation des sauvegardes automatiques garantit que vous pouvez restaurer votre réplica sur un instant donné si vous devez resynchroniser votre instance source et votre réplica. Pour

plus d'informations sur les sauvegardes et les point-in-time restaurations, consultezSauvegarde, restauration et exportation de données.

Pour configurer une réplication de position de fichier journal binaire avec une instance source externe

1. Rendez l'instance source My SQL ou MariaDB en lecture seule.

```
mysql> FLUSH TABLES WITH READ LOCK;
mysql> SET GLOBAL read_only = ON;
```

2. Exécutez la `SHOW MASTER STATUS` commande sur l'instance source My SQL ou MariaDB pour déterminer l'emplacement du journal binaire.

   Vous obtenez une sortie similaire à ce qui suit.

```
File                          Position
-----------------------------------
 mysql-bin-changelog.000031      107
-----------------------------------
```

3. Copiez la base de données de l'instance externe vers l'instance de base de RDS données Amazon à l'aide de`mysqldump`. Pour les bases de données très volumineuses, il se peut que vous vouliez utiliser la procédure décrite dans Importation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt.

   Dans Linux, macOS, ou Unix:

```
mysqldump --databases database_name \
    --single-transaction \
    --compress \
    --order-by-primary \
    -u local_user \
    -plocal_password | mysql \
        --host=hostname \
        --port=3306 \
        -u RDS_user_name \
        -pRDS_password
```

   Dans Windows:

```
mysqldump --databases database_name ^
    --single-transaction ^
    --compress ^
    --order-by-primary ^
    -u local_user ^
    -plocal_password | mysql ^
        --host=hostname ^
        --port=3306 ^
        -u RDS_user_name ^
        -pRDS_password
```

> **Note**
>
> Veillez bien à ce qu'il n'y ait pas d'espace entre l'option `-p` et le mot de passe saisi.

Pour spécifier le nom d'hôte, le nom d'utilisateur, le port et le mot de passe pour vous connecter à votre RDS instance de base de données Amazon `--host--user (-u)`, utilisez les `-p` options, `--port` et dans la `mysql` commande. Le nom d'hôte est le nom du service de nom de domaine (DNS) du point de terminaison de l'RDSinstance de base de données Amazon, par exemple`myinstance.123456789012.us-east-1.rds.amazonaws.com`. Vous pouvez trouver la valeur du point de terminaison dans la AWS Management Console au niveau des détails de l'instance.

4. Rendez l'instance source My SQL ou MariaDB inscriptible à nouveau.

```
mysql> SET GLOBAL read_only = OFF;
mysql> UNLOCK TABLES;
```

Pour plus d'informations sur la création de sauvegardes destinées à la réplication, consultez la section Ma SQL documentation.

5. Dans le AWS Management Console, ajoutez l'adresse IP du serveur qui héberge la base de données externe au groupe de sécurité du cloud privé virtuel (VPC) pour l'RDSinstance de base de données Amazon. Pour plus d'informations sur la modification d'un groupe VPC de sécurité, consultez la section Groupes de sécurité correspondants VPC dans le guide de l'utilisateur d'Amazon Virtual Private Cloud.

   L'adresse IP peut changer lorsque les conditions suivantes sont réunies :

- Vous utilisez une adresse IP publique pour la communication entre l'instance source externe et l'instance de base de données.
- L'instance source externe a été arrêtée et redémarrée.

Si ces conditions sont réunies, vérifiez l'adresse IP avant de l'ajouter.

Vous devrez peut-être également configurer votre réseau local pour autoriser les connexions à partir de l'adresse IP de votre RDS instance de base de données Amazon. Vous procédez ainsi pour que votre réseau local puisse communiquer avec votre instance externe My SQL ou MariaDB. Pour trouver l'adresse IP de l'RDSinstance de base de données Amazon, utilisez la `host` commande.

```
host db_instance_endpoint
```

Le nom d'hôte est le DNS nom du point de terminaison de l'RDSinstance de base de données Amazon.

6. En utilisant le client de votre choix, connectez-vous à l'instance externe et créez un utilisateur à utiliser pour la réplication. Utilisez ce compte exclusivement pour la réplication et limitez-le à votre domaine pour améliorer la sécurité. Voici un exemple.

```
CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

7. Pour l'instance externe, attribuez les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` à votre utilisateur de réplication. Par exemple, pour accorder les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données à l'utilisateur « `repl_user` » de votre domaine, émettez la commande suivante.

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
```

8. Faites de l'RDSinstance de base de données Amazon la réplique. Pour ce faire, connectez-vous d'abord à l'RDSinstance de base de données Amazon en tant qu'utilisateur principal. Identifiez

ensuite la base de données externe My SQL ou MariaDB comme instance source à l'aide de la commande or. Utilisez le nom et la position du fichier journal maître que vous avez déterminés à l'étape 2. Les commandes suivantes sont des exemples.

Mon SQL 8,4

```
CALL mysql.rds_set_external_source ('mysourceserver.mydomain.com', 3306,
 'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

MariaDB et My 8.0 et 5.7 SQL

```
CALL mysql.rds_set_external_master ('mymasterserver.mydomain.com', 3306,
 'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1);
```

> **Note**
>
> Sur RDS MySQL, vous pouvez choisir d'utiliser la réplication différée en exécutant la procédure mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL ou la procédure mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL stockée à la place. On RDS for MySQL, l'une des raisons d'utiliser la réplication différée est d'activer la reprise après sinistre avec la procédure stockée. Actuellement, RDS MariaDB prend en charge la réplication différée mais ne prend pas en charge la procédure. `mysql.rds_start_replication_until`

9. Sur l'RDSinstance de base de données Amazon, émettez la mysql.rds_start_replication commande pour démarrer la réplication.

```
CALL mysql.rds_start_replication;
```

Configuration multi-source-replication pour Amazon RDS for MySQL

Avec la réplication multi-source, vous pouvez configurer une instance de base de données Amazon RDS pour MySQL en tant que réplique qui reçoit les événements du journal binaire de plusieurs instances de base de données source RDS pour MySQL. La réplication multi-source est prise en charge pour les instances de base de données RDS for MySQL exécutant les versions de moteur suivantes :

- Toutes les versions de MySQL 8.4
- Versions mineures 8.0.35 et supérieures
- Versions mineures 5.7.44 et supérieures

Pour plus d'informations sur la réplication multi-source MySQL, consultez la section Réplication multi-source MySQL dans la documentation MySQL. La documentation MySQL contient des informations détaillées sur cette fonctionnalité, tandis que cette rubrique décrit comment configurer et gérer les canaux de réplication multi-sources sur vos instances de base de données RDS pour MySQL.

Cas d'utilisation de la réplication multi-sources

Les cas suivants sont de bons candidats pour l'utilisation de la réplication multi-source sur RDS for MySQL :

- Applications qui doivent fusionner ou combiner plusieurs partitions sur des instances de base de données distinctes en une seule partition.
- Applications qui doivent générer des rapports à partir de données consolidées provenant de sources multiples.
- Exigences relatives à la création de sauvegardes consolidées à long terme des données distribuées entre plusieurs instances de base de données RDS for MySQL.

Conditions préalables à la réplication multi-sources

Avant de configurer la réplication multi-source, remplissez les conditions préalables suivantes.

- Assurez-vous que les sauvegardes automatiques sont activées pour chaque instance de base de données RDS pour MySQL source. L'activation des sauvegardes automatiques active la journalisation binaire. Pour savoir comment activer les sauvegardes automatiques, consultezthe section called “Activation des sauvegardes automatiques”.
- Pour éviter les erreurs de réplication, nous vous recommandons de bloquer les opérations d'écriture sur les instances de base de données source. Vous pouvez le faire en définissant le `read-only` paramètre sur `0N` dans un groupe de paramètres personnalisé attaché à l'instance de base de données source RDS pour MySQL. Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour créer un nouveau groupe de paramètres personnalisés ou pour modifier un groupe existant. Pour plus d’informations, consultez the section called “Création d'un groupe de paramètres de bases de données” et the section called “Modification de paramètres dans un groupe de paramètres de bases de données”.

- Pour chaque instance de base de données source, ajoutez l'adresse IP de l'instance au groupe de sécurité Amazon Virtual Private Cloud (VPC) pour l'instance de base de données multi-source. Pour identifier l'adresse IP d'une instance de base de données source, vous pouvez exécuter la commande`dig` *`RDS Endpoint`*. Exécutez la commande depuis une EC2 instance Amazon dans le même VPC que l'instance de base de données multi-source de destination.
- Pour chaque instance de base de données source, utilisez un client pour vous connecter à l'instance de base de données et créez un utilisateur de base de données doté des privilèges requis pour la réplication, comme dans l'exemple suivant.

```
CREATE USER 'repl_user' IDENTIFIED BY 'password';
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user';
```

> **Note**
>
> À partir de MySQL 8.4, le `REPLICATION SLAVE` privilège est devenu obsolète et remplacé par. `REPLICATION REPLICA` Pour MySQL 8.4 et versions ultérieures, utilisez plutôt la syntaxe suivante :
>
> ```
> CREATE USER 'repl_user' IDENTIFIED BY 'password';
> GRANT REPLICATION CLIENT, REPLICATION REPLICA ON *.* TO 'repl_user';
> ```

Configuration de canaux de réplication multi-sources sur RDS pour les instances de base de données MySQL

La configuration de canaux de réplication à sources multiples est similaire à la configuration de la réplication à source unique. Pour la réplication multi-source, vous devez d'abord activer la journalisation binaire sur l'instance source. Ensuite, vous importez les données des sources vers la réplique multi-sources. Ensuite, vous lancez la réplication à partir de chaque source en utilisant les coordonnées binaires du journal ou en utilisant le positionnement automatique GTID.

Pour configurer une instance de base de données RDS pour MySQL en tant que réplique multi-source de deux instances de base de données RDS pour MySQL ou plus, effectuez les étapes suivantes.

Rubriques

- Étape 1 : Importer les données des instances de base de données source vers la réplique multi-source
- Étape 2 : démarrer la réplication depuis les instances de base de données source vers la réplique multi-source

Étape 1 : Importer les données des instances de base de données source vers la réplique multi-source

Effectuez les étapes suivantes sur chaque instance de base de données source.

Avant d'importer les données d'une source vers la réplique multi-source, déterminez le fichier journal binaire actuel et sa position en exécutant la `SHOW MASTER STATUS` commande. Prenez note de ces informations pour les utiliser à l'étape suivante. Dans cet exemple de sortie, le fichier est `mysql-bin-changelog.000031` et la position est`107`.

> **Note**
>
> À partir de MySQL 8.4, la `SHOW MASTER STATUS` commande est devenue obsolète et remplacée par. `SHOW BINARY LOG STATUS` Pour MySQL 8.4 et versions ultérieures, utilisez `SHOW BINARY LOG STATUS` plutôt.

```
File                        Position
-----------------------------------
mysql-bin-changelog.000031      107
-----------------------------------
```

Copiez maintenant la base de données de l'instance de base de données source vers la réplique multi-source en utilisant`mysqldump`, comme dans l'exemple suivant.

```
mysqldump --databases database_name \
 --single-transaction \
 --compress \
 --order-by-primary \
 -u RDS_user_name \
 -p RDS_password \
 --host=RDS Endpoint | mysql \
 --host=RDS Endpoint \
 --port=3306 \
```

```
 -u RDS_user_name \
-p RDS_password
```

Après avoir copié la base de données, vous pouvez définir le paramètre en lecture seule 0FF sur l'instance de base de données source.

Étape 2 : démarrer la réplication depuis les instances de base de données source vers la réplique multi-source

Pour chaque instance de base de données source, utilisez les informations d'identification de l'utilisateur administratif pour vous connecter à l'instance et exécutez les deux procédures stockées suivantes. Ces procédures stockées configurent la réplication sur un canal et démarrent la réplication. Cet exemple utilise le nom et la position du fichier binlog indiqués dans l'exemple de sortie de l'étape précédente.

```
CALL mysql.rds_set_external_source_for_channel('mysourcehost.example.com', 3306,
 'repl_user', 'password', 'mysql-bin-changelog.000031', 107, 1, 'channel_1');
CALL mysql.rds_start_replication_for_channel('channel_1');
```

Pour plus d'informations sur l'utilisation de ces procédures stockées et d'autres pour configurer et gérer vos canaux de réplication, consultezthe section called “Gestion de la réplication multi-sources”.

Utilisation de filtres avec réplication multi-sources

Vous pouvez utiliser des filtres de réplication pour spécifier les bases de données et les tables qui sont répliquées dans une réplique multi-sources. Les filtres de réplication peuvent inclure des bases de données et des tables dans la réplication ou les exclure de la réplication. Pour plus d'informations sur les filtres de réplication, consultezthe section called “Configuration des filtres de réplication”.

Avec la réplication multi-sources, vous pouvez configurer les filtres de réplication globalement ou au niveau du canal. Le filtrage au niveau du canal n'est disponible qu'avec les instances de base de données prises en charge exécutant la version 8.0 ou la version 8.4. Les exemples suivants montrent comment configurer les filtres globalement ou au niveau du canal.

Notez les exigences et le comportement suivants en matière de filtrage dans le cadre de la réplication multi-sources :

- Les noms des chaînes doivent être placés entre guillemets (``).
- Si vous modifiez les filtres de réplication dans le groupe de paramètres, les répliques multi-sources de tous les canaux mis à jour sont redémarrées `sql_thread` pour appliquer les modifications de

manière dynamique. Si une mise à jour implique un filtre global, tous les canaux de réplication en cours d'exécution sont redémarrés.

- Tous les filtres globaux sont appliqués avant les filtres spécifiques au canal.
- Si un filtre est appliqué globalement et au niveau du canal, seul le filtre au niveau du canal est appliqué. Par exemple, si les filtres le sont`replicate_ignore_db="db1,`channel_22`:db2"`, le paramètre `replicate_ignore_db` défini sur `db1` est appliqué à tous les canaux à l'exception de`channel_22`, et `channel_22` ignore uniquement les modifications de`db2`.

Exemple 1 : définition d'un filtre global

Dans l'exemple suivant, la `temp_data` base de données est exclue de la réplication sur tous les canaux.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group \
--db-parameter-group-name myparametergroup \
--parameters "ParameterName=replicate-ignore-
db,ParameterValue='temp_data',ApplyMethod=immediate"
```

Exemple 2 : définition d'un filtre au niveau du canal

Dans l'exemple suivant, les modifications apportées à la `sample22` base de données ne sont incluses que dans le canal`channel_22`. De même, les modifications apportées à la `sample99` base de données ne sont incluses que dans le canal`channel_99`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group \
--db-parameter-group-name myparametergroup \
--parameters "ParameterName=replicate-do-db,ParameterValue='\`channel_22\`:sample22,
\`channel_99\`:sample99',ApplyMethod=immediate"
```

Surveillance des canaux de réplication multi-sources

Vous pouvez surveiller des canaux individuels dans une réplique multi-sources en utilisant les méthodes suivantes :

- Pour surveiller l'état de tous les canaux ou d'un canal spécifique, connectez-vous à la réplique multi-source et exécutez la `SHOW REPLICA STATUS FOR CHANNEL 'channel_name'` commande `SHOW REPLICA STATUS` or. Pour plus d'informations, consultez la section Vérification de l'état de réplication dans la documentation MySQL.
- Pour recevoir une notification lorsqu'un canal de réplication est démarré, arrêté ou supprimé, utilisez la notification d'événement RDS. Pour de plus amples informations, veuillez consulter the section called “Utilisation des notifications d'RDSévénements Amazon”.
- Pour surveiller le décalage d'un canal spécifique, vérifiez la `ReplicationChannelLag` métrique correspondante. Les points de données pour cette métrique ont une période de 60 secondes (1 minute) et sont disponibles pendant 15 jours. Pour déterminer le décalage d'un canal de réplication, utilisez l'identifiant de l'instance et le nom du canal de réplication. Pour recevoir une notification lorsque ce décalage dépasse un certain seuil, vous pouvez configurer une CloudWatch alarme. Pour de plus amples informations, veuillez consulter the section called “Surveillance du RDS avec CloudWatch”.

Considérations et bonnes pratiques relatives à la réplication multi-sources

Avant d'utiliser la réplication multi-source sur RDS pour MySQL, passez en revue les considérations et les meilleures pratiques suivantes :

- Assurez-vous qu'une instance de base de données configurée en tant que réplique multi-source dispose de ressources suffisantes telles que le débit, la mémoire, le processeur et les IOPS pour gérer la charge de travail provenant de plusieurs instances sources.
- Surveillez régulièrement l'utilisation des ressources sur votre réplique multi-source et ajustez la configuration du stockage ou de l'instance pour gérer la charge de travail sans surcharger les ressources.
- Vous pouvez configurer la réplication multithread sur une réplique multi-source en définissant la variable système sur une valeur `replica_parallel_workers` supérieure à. `0` Dans ce cas, le nombre de threads alloués à chaque canal est la valeur de cette variable, plus un thread coordinateur pour gérer les threads applicateurs.
- Configurez les filtres de réplication de manière appropriée pour éviter les conflits. Pour répliquer une base de données complète vers une autre base de données sur un réplica, vous pouvez utiliser `--replicate-rewrite-db` cette option. Par exemple, vous pouvez répliquer toutes les tables de la base de données A vers la base de données B sur une instance de réplication. Cette approche peut être utile lorsque toutes les instances source utilisent la même convention de

dénomination de schéma. Pour plus d'informations sur `--replicate-rewrite-db` cette option, consultez la section Options et variables du serveur de réplication dans la documentation MySQL.

- Pour éviter les erreurs de réplication, évitez d'écrire sur le réplica. Nous vous recommandons d'activer le `read_only` paramètre sur les répliques multi-sources pour bloquer les opérations d'écriture. Cela permet d'éliminer les problèmes de réplication causés par des opérations d'écriture contradictoires.
- Pour améliorer les performances des opérations de lecture telles que les tris et les jointures à charge élevée exécutées sur la réplique multi-source, pensez à utiliser des lectures optimisées RDS. Cette fonctionnalité peut être utile pour les requêtes qui dépendent de grandes tables temporaires ou de fichiers de tri. Pour de plus amples informations, veuillez consulter the section called “Améliorer les performances des requêtes grâce à des lectures RDS optimisées”.
- Pour minimiser le délai de réplication et améliorer les performances d'une réplique multi-sources, pensez à activer les écritures optimisées. Pour de plus amples informations, veuillez consulter the section called “Améliorer les performances d'écriture grâce à RDS Optimized Writes for My SQL”.
- Effectuez des opérations de gestion (telles que la modification de la configuration) sur un canal à la fois et évitez de modifier plusieurs canaux à partir de plusieurs connexions. Ces pratiques peuvent entraîner des conflits dans les opérations de réplication. Par exemple, l'exécution simultanée `rds_skip_repl_error_for_channel` de `rds_start_replication_for_channel` procédures à partir de plusieurs connexions peut entraîner l'omission d'événements sur un canal différent de celui prévu.
- Vous pouvez activer les sauvegardes sur une instance de réplication multi-sources et exporter les données de cette instance vers un compartiment Amazon S3 afin de les stocker à des fins de long terme. Cependant, il est également important de configurer les sauvegardes avec une rétention appropriée sur les instances sources individuelles. Pour plus d'informations sur l'exportation de données de capture d'écran vers Amazon S3, consultezthe section called “Exportation de données d'instantanés de bases de données vers Amazon S3”.
- Pour répartir la charge de travail de lecture sur un réplica multi-source, vous pouvez créer des répliques de lecture à partir d'un réplica multi-sources. Vous pouvez localiser ces répliques de lecture de différentes manières en Régions AWS fonction des exigences de votre application. Pour plus d'informations sur les réplicas en lecture, consultez the section called “Mes répliques de SQL lecture”.

Limitations de la réplication multi-source sur RDS pour MySQL

Les limitations suivantes s'appliquent à la réplication multi-source sur RDS pour MySQL :

- Actuellement, RDS for MySQL prend en charge la configuration d'un maximum de 15 canaux pour une réplique multi-sources.
- Une instance de réplication en lecture ne peut pas être configurée en tant que réplique multi-source.
- Pour configurer la réplication multi-source sur RDS pour MySQL exécutant le moteur version 5.7, le schéma de performance doit être activé sur l'instance de réplique. L'activation du schéma de performance est facultative sur RDS pour MySQL exécutant le moteur version 8.0 ou 8.4.
- Pour RDS for MySQL exécutant le moteur version 5.7, les filtres de réplication s'appliquent à tous les canaux de réplication. Pour RDS for MySQL exécutant la version 8.0 ou 8.4 du moteur, vous pouvez configurer des filtres qui s'appliquent à tous les canaux de réplication ou à des canaux individuels.
- La restauration d'un instantané RDS ou l'exécution d'un Point-in-time-Restore (PITR) ne restaurent pas les configurations de canaux de réplication multi-sources.
- Lorsque vous créez une réplique en lecture d'une réplique multi-source, elle ne réplique que les données de l'instance multi-source. Il ne restaure aucune configuration de canal.
- MySQL ne prend pas en charge la configuration d'un nombre différent de travailleurs parallèles pour chaque canal. Chaque canal reçoit le même nombre de travailleurs parallèles en fonction de la `replica_parallel_workers` valeur.

Les limitations supplémentaires suivantes s'appliquent si votre cible de réplication multi-source est un cluster de base de données multi-AZ :

- Un canal doit être configuré pour une instance source RDS pour MySQL avant toute écriture sur cette instance.
- La réplication basée sur GTID doit être activée pour chaque instance source RDS pour MySQL.
- Un événement de basculement sur le cluster de base de données supprime la configuration de réplication multi-source. La restauration de cette configuration nécessite de répéter les étapes de configuration.

Configuration de clusters actifs-actifs pour for My RDS SQL

Un cluster actif-actif dans Amazon RDS est une configuration de base de données dans laquelle plusieurs nœuds gèrent activement les opérations de lecture et d'écriture, répartissant la charge de travail entre les instances afin d'améliorer la disponibilité et l'évolutivité. Chaque nœud du cluster est synchronisé pour maintenir la cohérence des données, ce qui permet une haute disponibilité et un basculement plus rapide en cas de défaillance du nœud

Vous pouvez configurer un cluster actif-actif RDS pour My à l'aide du SQL plug-in My SQL Group Replication. Le plugin Group Replication est pris en charge RDS pour les instances My SQL DB exécutant les versions suivantes :

- Toutes les versions de My SQL 8.4
- Mes SQL versions mineures 8.0.35 et supérieures

Pour plus d'informations sur My SQL Group Replication, consultez la section Réplication de groupe dans la SQL documentation My Group. La SQL documentation My contient des informations détaillées sur cette fonctionnalité, tandis que cette rubrique décrit comment configurer et gérer le plugin sur vos instances RDS for My SQL DB.

> **Note**
>
> Par souci de concision, toutes les mentions de cluster « actif-actif » dans cette rubrique font référence aux clusters actifs-actifs utilisant le plugin My Group Replication. SQL

Cas d'utilisation pour les clusters actifs-actifs

Les cas suivants sont de bons candidats pour l'utilisation de clusters actifs-actifs :

- Applications qui ont besoin de toutes les instances de base de données du cluster pour prendre en charge les opérations d'écriture. Le plugin Group Replication assure la cohérence des données sur chaque instance de base de données du cluster actif-actif. Pour plus d'informations sur son fonctionnement, consultez la section Réplication de groupe dans la section Ma SQL documentation.
- Applications nécessitant une disponibilité continue de la base de données. Avec un cluster actif-actif, les données sont conservées sur toutes les instances de base de données du cluster. En

cas de défaillance d'une instance de base de données, l'application peut rediriger le trafic vers une autre instance de base de données du cluster.

- Applications susceptibles de devoir répartir les opérations de lecture et d'écriture entre les différentes instances de base de données du cluster à des fins d'équilibrage de charge. Avec un cluster actif-actif, vos applications peuvent envoyer du trafic de lecture vers des instances de base de données spécifiques et du trafic d'écriture vers d'autres instances. Vous pouvez également changer à tout moment les instances de base de données auxquelles envoyer des lectures ou des écritures.

Rubriques

- Limites et considérations relatives aux clusters actifs-actifs
- Préparation à un cluster à VPC activités croisées
- Réglages de paramètres requis pour les clusters actifs-actifs
- Conversion d'une instance de base de données existante en cluster actif-actif
- Configuration d'un cluster actif-actif avec de nouvelles instances de base de données
- Ajouter une instance de base de données à un cluster actif-actif
- Surveillance des clusters actifs-actifs
- Arrêt de la réplication de groupe sur une instance de base de données dans un cluster actif-actif
- Modification du nom d'une instance de base de données dans un cluster actif-actif
- Supprimer une instance de base de données d'un cluster actif-actif

Limites et considérations relatives aux clusters actifs-actifs

Les clusters actifs d'Amazon RDS offrent une disponibilité et une évolutivité améliorées en répartissant les charges de travail sur plusieurs instances. Cependant, il existe des limites et des considérations importantes à prendre en compte lors de l'utilisation de cette architecture.

Les sections suivantes décrivent les facteurs clés tels que les délais de réplication, la résolution des conflits, l'allocation des ressources et le comportement de basculement. La compréhension de ces considérations peut contribuer à garantir des performances et une fiabilité optimales dans les déploiements de clusters actifs-actifs.

Rubriques

- Limitations RDS pour Mes clusters SQL actifs-actifs

- Considérations et bonnes pratiques relatives à RDS for My SQL active-active clusters

Limitations RDS pour Mes clusters SQL actifs-actifs

Les limitations suivantes s'appliquent aux clusters actifs-actifs pour RDS for My : SQL

- Le nom d'utilisateur principal ne peut pas être `rdsgrprepladmin` destiné aux instances de base de données d'un cluster actif-actif. Ce nom d'utilisateur est réservé aux connexions de réplication de groupe.
- Pour les instances de base de données avec des répliques de lecture dans des clusters actifs-actifs, un état de réplication prolongé `Replicating` peut entraîner le dépassement des limites de stockage des fichiers journaux. Pour plus d'informations sur l'état des répliques de lecture, consultezSupervision de la réplication en lecture.
- Les déploiements bleu/vert ne sont pas pris en charge pour les instances de base de données dans un cluster actif-actif. Pour de plus amples informations, veuillez consulter Utilisation des déploiements Amazon RDS (Blue/Green) pour les mises à jour de bases de données.
- L'authentification Kerberos n'est pas prise en charge pour les instances de base de données dans un cluster actif-actif. Pour de plus amples informations, veuillez consulter Utilisation de Kerberos l'authentification pour Amazon RDS for MySQL.
- Les instances de base de données d'un cluster de base de données multi-AZ ne peuvent pas être ajoutées à un cluster actif-actif. Toutefois, les instances de base de données d'un déploiement d'instance de base de données multi-AZ peuvent être ajoutées à un cluster actif-actif. Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
- Les tables dépourvues de clé primaire ne sont pas répliquées dans un cluster actif-actif car les écritures sont rejetées par le plugin Group Replication.
- Les tables non-InnoDB ne sont pas répliquées dans un cluster actif-actif.
- Les clusters actifs-actifs ne prennent pas en charge les DDL instructions DML simultanées sur les différentes instances de base de données du cluster.
- Vous ne pouvez pas configurer un cluster actif-actif pour utiliser le mode primaire unique pour le mode de réplication du groupe. Pour cette configuration, nous vous recommandons d'utiliser plutôt un cluster de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données multi-AZ pour Amazon RDS.
- La réplication multi-source n'est pas prise en charge pour les instances de base de données dans un cluster actif-actif.

- Un cluster actif-actif interrégional ne peut pas appliquer la vérification de l'autorité de certification (CA) pour les connexions de réplication de groupe.

Considérations et bonnes pratiques relatives à RDS for My SQL active-active clusters

Avant d'utiliser RDS Mes clusters SQL actifs-actifs, passez en revue les considérations et les meilleures pratiques suivantes :

- Les clusters actifs-actifs ne peuvent pas avoir plus de neuf instances de base de données.
- Avec le plugin Group Replication, vous pouvez contrôler les garanties de cohérence des transactions du cluster actif-actif. Pour plus d'informations, consultez la section Garanties de cohérence des transactions dans la section Ma SQL documentation.
- Des conflits sont possibles lorsque différentes instances de base de données mettent à jour la même ligne dans un cluster actif-actif. Pour plus d'informations sur les conflits et leur résolution, consultez la section Réplication de groupe dans la section Ma SQL documentation.
- Pour la tolérance aux pannes, incluez au moins trois instances de base de données dans votre cluster actif-actif. Il est possible de configurer un cluster actif-actif avec une ou deux instances de base de données uniquement, mais le cluster ne tolérera pas les pannes. Pour plus d'informations sur la tolérance aux pannes, consultez la section Tolérance aux pannes dans la section Ma SQL documentation.
- Lorsqu'une instance de base de données rejoint un cluster actif-actif existant et exécute la même version de moteur que la version la plus basse du cluster, l'instance de base de données se joint en mode lecture-écriture.
- Lorsqu'une instance de base de données rejoint un cluster actif-actif existant et exécute une version de moteur supérieure à la version la plus basse du cluster, l'instance de base de données doit rester en mode lecture seule.
- Si vous activez la réplication de groupe pour une instance de base de données en définissant son `rds.group_replication_enabled` paramètre sur `1` dans le groupe de paramètres de base de données, mais que la réplication n'a pas démarré ou n'a pas pu démarrer, l'instance de base de données est placée en super-read-only mode pour éviter les incohérences dans les données. Pour plus d'informations sur super-read-only le mode, consultez la section Ma SQL documentation.
- Vous pouvez mettre à niveau une instance de base de données dans un cluster actif-actif, mais l'instance de base de données est en lecture seule jusqu'à ce que toutes les autres instances de base de données du cluster actif-actif soient mises à niveau vers la même version de moteur ou une version de moteur supérieure. Lorsque vous mettez à niveau une instance de base de

données, celle-ci rejoint automatiquement le même cluster actif-actif une fois la mise à niveau terminée. Pour éviter un passage involontaire en mode lecture seule pour une instance de base de données, désactivez les mises à niveau automatiques des versions mineures pour celle-ci. Pour plus d'informations sur la mise à niveau d'une instance My SQL DB, consultezMises à niveau du moteur de base de données RDS pour MySQL.

- Vous pouvez ajouter une instance de base de données dans un déploiement d'instance de base de données multi-AZ à un cluster actif-actif existant. Vous pouvez également convertir une instance de base de données mono-AZ d'un cluster actif-actif en un déploiement d'instance de base de données multi-AZ. Si une instance de base de données principale échoue dans un déploiement multi-AZ, cette instance principale bascule vers l'instance de secours. La nouvelle instance de base de données principale rejoint automatiquement le même cluster une fois le basculement terminé. Pour plus d'informations sur les déploiements d'instances de base de données multi-AZ, consultez. Déploiements d'instances de base de données multi-AZ pour Amazon RDS
- Nous recommandons que les instances de base de données d'un cluster actif-actif aient des plages de temps différentes pour leurs fenêtres de maintenance. Cette pratique évite que plusieurs instances de base de données du cluster ne soient mises hors ligne pour des raisons de maintenance en même temps. Pour de plus amples informations, veuillez consulter Fenêtre de maintenance Amazon RDS.
- Les clusters actifs-actifs peuvent être utilisés SSL pour les connexions entre les instances de base de données. Pour configurer les SSL connexions, définissez les paramètres group_replication_recovery_use_ssl et group_replication_ssl_mode. Les valeurs de ces paramètres doivent correspondre à toutes les instances de base de données du cluster actif-actif.

  Actuellement, les clusters actifs-actifs ne prennent pas en charge la vérification par l'autorité de certification (CA) pour les connexions entre eux. Régions AWS Le paramètre group_replication_ssl_mode doit donc être défini sur `DISABLED` (valeur par défaut) ou pour les clusters interrégionaux. `REQUIRED`
- Un cluster RDS for My SQL active-active s'exécute en mode multi-primaire. La valeur par défaut de group_replication_enforce_update_everywhere_checks est et le paramètre est statique. `ON` Lorsque ce paramètre est défini sur`ON`, les applications ne peuvent pas l'insérer dans une table soumise à des contraintes de clé étrangère en cascade.
- Un cluster RDS for My SQL active-active utilise la pile de SQL communication My pour la sécurité des connexions au lieu de. XCOM Pour plus d'informations, consultez Communication Stack pour la gestion de la sécurité des connexions dans la section Ma SQL documentation.

- Lorsqu'un groupe de paramètres de base de données est associé à une instance de base de données dans un cluster actif-actif, nous recommandons de n'associer ce groupe de paramètres de base de données qu'aux autres instances de base de données présentes dans le cluster.
- Les clusters actifs-actifs ne prennent en charge que RDS les instances My SQL DB. Ces instances de base de données doivent exécuter des versions prises en charge du moteur de base de données.
- Lorsqu'une instance de base de données d'un cluster actif-actif connaît une défaillance inattendue, la restauration de l'instance de base de données RDS démarre automatiquement. Si l'instance de base de données ne se rétablit pas, nous vous recommandons de la remplacer par une nouvelle instance de base de données en effectuant une point-in-time restauration avec une instance de base de données saine dans le cluster. Pour obtenir des instructions, consultez Ajouter une instance de base de données à un cluster actif-actif à l'aide de la restauration point-in-time.
- Vous pouvez supprimer une instance de base de données dans un cluster actif-actif sans affecter les autres instances de base de données du cluster. Pour plus d'informations sur la création d'une instance de base de données, veuillez consulter Suppression d'une instance DB.
- Lorsqu'une instance de base de données quitte involontairement un cluster actif-actif, le `group_replication_exit_state_action` paramètre passe par défaut à. `OFFLINE_MODE` Dans cet état, l'instance de base de données est inaccessible et vous devez redémarrer l'instance de base de données pour la remettre en ligne et rejoindre le cluster. Vous pouvez modifier ce comportement en modifiant le `group_replication_exit_state_action` paramètre dans un groupe de paramètres personnalisé. En définissant le paramètre sur`READ_ONLY`, lorsque l'instance de base de données quitte involontairement un cluster, elle passe en mode super lecture seule au lieu de se déconnecter.

Préparation à un cluster à VPC activités croisées

Vous pouvez configurer un cluster actif-actif avec des instances Amazon RDS for My SQL DB dans plusieurs instances. VPC Ils VPCs peuvent être identiques Région AWS ou différents Régions AWS.

> **Note**
>
> L'envoi de trafic entre plusieurs Régions AWS sites peut entraîner des coûts supplémentaires. Pour plus d'informations, voir Vue d'ensemble des coûts de transfert de données pour les architectures courantes.

Si vous configurez un cluster actif-actif en un seulVPC, vous pouvez ignorer ces étapes et passer à. Configuration d'un cluster actif-actif avec de nouvelles instances de base de données

Pour préparer un cluster actif-actif avec des instances de base de données dans plusieurs VPC

1. Assurez-vous que les plages d'IPv4adresses dans les CIDR blocs répondent aux exigences suivantes :

   - Les plages d'IPv4adresses dans les CIDR blocs du ne VPCs peuvent pas se chevaucher.
   - Toutes les plages d'IPv4adresses des CIDR blocs doivent être inférieures `128.0.0.0/`*`subnet_mask`* ou supérieures à 128.0.0.0/*`subnet_mask`*.

   Les plages suivantes illustrent ces exigences :

   - `10.1.0.0/16`dans l'un VPC et `10.2.0.0/16` dans l'autre VPC est pris en charge.
   - `172.1.0.0/16`dans l'un VPC et `172.2.0.0/16` dans l'autre VPC est pris en charge.
   - `10.1.0.0/16`dans l'un VPC et `10.1.0.0/16` dans l'autre n'VPCest pas pris en charge car les plages se chevauchent.
   - `10.1.0.0/16`dans l'un VPC et `172.1.0.0/16` dans l'autre n'VPCest pas pris en charge car l'un est en dessous `128.0.0.0/`*`subnet_mask`* et l'autre en haut`128.0.0.0/`*`subnet_mask`*.

   Pour plus d'informations sur les CIDR VPCCIDRblocs, consultez le manuel Amazon VPC User Guide.
2. Dans chacun d'euxVPC, assurez-vous que DNS la résolution et les DNS noms d'hôte sont activés.

   Pour obtenir des instructions, consultez la section Afficher et mettre à jour vos DNS attributs VPC dans le guide de VPC l'utilisateur Amazon.
3. Configurez le VPCs afin de pouvoir acheminer le trafic entre eux de l'une des manières suivantes :

   - Créez une connexion VPC d'appairage entre lesVPCs.

     Pour obtenir des instructions, consultez la section Créer une connexion VPC de peering dans le Amazon VPC Peering Guide. Dans chacun d'euxVPC, assurez-vous que vos groupes de sécurité comportent des règles entrantes qui font référence aux groupes de sécurité du pairVPC. Cela permet au trafic de circuler vers et depuis les instances associées au

groupe de sécurité référencé dans le peeredVPC. Pour obtenir des instructions, consultez la section Mettre à jour vos groupes de sécurité pour faire référence aux groupes de sécurité homologues dans le Amazon VPC Peering Guide.

- Créez une passerelle de transit entre lesVPCs.

  Pour obtenir des instructions, consultez Commencer à utiliser les passerelles de transit dans Amazon VPC Transit Gateways. Dans chaque groupe de sécuritéVPC, assurez-vous que des règles entrantes autorisent le trafic provenant de l'autreVPC, telles que des règles de trafic entrant qui spécifient l'un ou CIDR l'autre. VPC Cela permet au trafic de circuler vers et depuis les instances associées au groupe de sécurité référencé dans le cluster actif-actif. Pour plus d'informations, consultez la section Contrôler le trafic vers vos AWS ressources à l'aide de groupes de sécurité dans le guide de VPC l'utilisateur Amazon.

Réglages de paramètres requis pour les clusters actifs-actifs

La configuration des paramètres des clusters actifs-actifs dans Amazon RDS for My SQL est essentielle pour maintenir des performances constantes et une stabilité opérationnelle. Ce tableau détaille les principaux paramètres qui contrôlent la réplication, la résolution des conflits et la distribution de la charge de travail. Une configuration correcte garantit une synchronisation efficace entre les nœuds, minimise le délai de réplication et optimise l'utilisation des ressources dans les environnements distribués ou à fort trafic.

Paramètre	Description	Paramètre requis
`binlog_format`	Définit le format de journalisation binaire. La valeur par défaut RDS pour les versions de My SQL 8.0 et antérieures est`MIXED`. La valeur par défaut RDS pour les versions My SQL 8.4 est`ROW`. Pour plus d'informations, consultez la section Ma SQL documentation.	`ROW`
`enforce_gtid_consistency`	Assure la GTID cohérence de l'exécution des instructions.	`ON`

Paramètre	Description	Paramètre requis
	La valeur par défaut de RDS for My SQL est`OFF`. Pour plus d'informations, consultez la section Ma SQL documentation.	
`group_replication_group_name`	Définit le nom de réplication de groupe sur unUUID. Le UUID format est`11111111-2222-3333-4444-555555555555` . Vous pouvez générer un My SQL UUID en vous connectant à une instance My SQL DB et en l'exécutant`SELECT UUID()`. La valeur doit être la même pour toutes les instances de base de données du cluster actif-actif. Pour plus d'informations, consultez la section Ma SQL documentation.	Un mois SQL UUID
`gtid-mode`	Journalisation GTID basée sur les contrôles. La valeur par défaut de RDS for My SQL est`OFF_PERMISSIVE` . Pour plus d'informations, consultez la section Ma SQL documentation.	`ON`

Paramètre	Description	Paramètre requis
`rds.custom_dns_resolution`	Spécifie s'il faut autoriser la DNS résolution depuis le DNS serveur Amazon de votreVPC. DNSla résolution doit être activée lorsque la réplication de groupe est activée avec le `rds.group_replication_enabled` paramètre. DNSla résolution ne peut pas être activée lorsque la réplication de groupe est désactivée avec le `rds.group_replication_enabled` paramètre. Pour plus d'informations, consultez le DNSserveur Amazon dans le guide de VPC l'utilisateur Amazon.	1
`rds.group_replication_enabled`	Spécifie si la réplication de groupe est activée pour une instance de base de données. La réplication de groupe doit être activée sur une instance de base de données dans un cluster actif-actif.	1
`replica_preserve_commit_order` (RDSpour My SQL 8.4 et versions supérieures) ou `slave_preserve_commit_order` (RDSpour les versions My SQL 8.0)	Contrôle l'ordre dans lequel les transactions sont validées sur une réplique. La valeur par défaut de RDS for My SQL est0N. Pour plus d'informations, consultez la section Ma SQL documentation.	0N

Conversion d'une instance de base de données existante en cluster actif-actif

La version du moteur de base de données de l'instance de base de données que vous souhaitez migrer vers un cluster actif-actif doit être l'une des versions suivantes :

- Toutes les versions de My SQL 8.4
- Mes SQL versions mineures 8.0.35 et supérieures

Si vous devez mettre à niveau la version du moteur, consultezMises à niveau du moteur de base de données RDS pour MySQL.

Si vous configurez un cluster actif-actif avec plusieurs instances de base de donnéesVPC, assurez-vous de remplir les conditions requises dans. Préparation à un cluster à VPC activités croisées

Procédez comme suit pour migrer une instance de base de données existante vers un cluster actif-actif pour RDS for My. SQL

Rubriques

- Étape 1 : définir les paramètres du cluster actif-actif dans un ou plusieurs groupes de paramètres personnalisés
- Étape 2 : associer l'instance de base de données à un groupe de paramètres de base de données dont les paramètres de réplication de groupe requis sont définis
- Étape 3 : Création du cluster actif-actif
- Étape 4 : créer des instances supplémentaires RDS pour My SQL DB pour le cluster actif-actif
- Étape 5 : Initialiser le groupe sur l'instance de base de données que vous convertissez
- Étape 6 : démarrer la réplication sur les autres instances de base de données du cluster actif-actif
- Étape 7 : (Recommandé) Vérifiez l'état du cluster actif-actif

Étape 1 : définir les paramètres du cluster actif-actif dans un ou plusieurs groupes de paramètres personnalisés

Les instances RDS for My SQL DB d'un cluster actif-actif doivent être associées à un groupe de paramètres personnalisé dont le réglage est correct pour les paramètres requis. Pour plus d'informations sur les paramètres et le réglage requis pour chacun d'entre eux, reportez-vous àRéglages de paramètres requis pour les clusters actifs-actifs.

Vous pouvez définir ces paramètres dans de nouveaux groupes de paramètres ou dans des groupes de paramètres existants. Toutefois, pour éviter d'affecter accidentellement les instances de base de données qui ne font pas partie du cluster actif-actif, nous vous recommandons vivement de créer un nouveau groupe de paramètres personnalisé. Les instances de base de données d'un cluster actif-actif peuvent être associées au même groupe de paramètres de base de données ou à différents groupes de paramètres de base de données.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour créer un nouveau groupe de paramètres personnalisés. Pour de plus amples informations, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon. L'exemple suivant exécute la create-db-parameter-group AWS CLI commande pour créer un groupe de paramètres de base de données personnalisé nommé *myactivepg* RDS pour My SQL 8.0 :

Dans Linux, macOS, ou Unix:

```
aws rds create-db-parameter-group \
  --db-parameter-group-name myactivepg \
  --db-parameter-group-family mysql8.0 \
  --description "Parameter group for active-active clusters"
```

Dans Windows:

```
aws rds create-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --db-parameter-group-family mysql8.0 ^
  --description "Parameter group for active-active clusters"
```

Vous pouvez également utiliser le AWS Management Console ou le AWS CLI pour définir les paramètres du groupe de paramètres personnalisés. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

L'exemple suivant exécute la modify-db-parameter-group AWS CLI commande pour définir les paramètres de My SQL 8.0. RDS Pour utiliser cet exemple avec RDS for My SQL 8.4, remplacez `slave_preserve_commit_order` par`replica_preserve_commit_order`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myactivepg \
```

```
  --parameters
 "ParameterName='rds.group_replication_enabled',ParameterValue='1',ApplyMethod=pending-
reboot" \

 "ParameterName='rds.custom_dns_resolution',ParameterValue='1',ApplyMethod=pending-
reboot" \

 "ParameterName='enforce_gtid_consistency',ParameterValue='ON',ApplyMethod=pending-
reboot" \
              "ParameterName='gtid-mode',ParameterValue='ON',ApplyMethod=pending-
reboot" \

 "ParameterName='binlog_format',ParameterValue='ROW',ApplyMethod=immediate" \

 "ParameterName='slave_preserve_commit_order',ParameterValue='ON',ApplyMethod=immediate"
 \

 "ParameterName='group_replication_group_name',ParameterValue='11111111-2222-3333-4444-55555555
reboot"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --parameters
 "ParameterName='rds.group_replication_enabled',ParameterValue='1',ApplyMethod=pending-
reboot" ^

 "ParameterName='rds.custom_dns_resolution',ParameterValue='1',ApplyMethod=pending-
reboot" ^

 "ParameterName='enforce_gtid_consistency',ParameterValue='ON',ApplyMethod=pending-
reboot" ^
              "ParameterName='gtid-mode',ParameterValue='ON',ApplyMethod=pending-
reboot" ^

 "ParameterName='binlog_format',ParameterValue='ROW',ApplyMethod=immediate" ^

 "ParameterName='slave_preserve_commit_order',ParameterValue='ON',ApplyMethod=immediate"
 ^

 "ParameterName='group_replication_group_name',ParameterValue='11111111-2222-3333-4444-55555555
reboot"
```

Étape 2 : associer l'instance de base de données à un groupe de paramètres de base de données dont les paramètres de réplication de groupe requis sont définis

Associez l'instance de base de données à un groupe de paramètres que vous avez créé ou modifié à l'étape précédente. Pour obtenir des instructions, consultez Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon.

Redémarrez l'instance de base de données pour que les nouveaux paramètres prennent effet. Pour obtenir des instructions, consultez Redémarrage d'une instance de base de données.

Étape 3 : Création du cluster actif-actif

Dans le groupe de paramètres de base de données associé à l'instance de base de données, définissez le `group_replication_group_seeds` paramètre sur le point de terminaison de l'instance de base de données que vous convertissez.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour définir le paramètre. Il n'est pas nécessaire de redémarrer l'instance de base de données après avoir défini ce paramètre. Pour de plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

L'exemple suivant exécute la modify-db-parameter-group AWS CLI commande pour définir les paramètres :

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myactivepg \
  --parameters
 "ParameterName='group_replication_group_seeds',ParameterValue='myactivedb1.123456789012.us-east-1.rds.amazonaws.com:3306',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --parameters
 "ParameterName='group_replication_group_seeds',ParameterValue='myactivedb1.123456789012.us-east-1.rds.amazonaws.com:3306',ApplyMethod=immediate"
```

Étape 4 : créer des instances supplémentaires RDS pour My SQL DB pour le cluster actif-actif

Pour créer des instances de base de données supplémentaires pour le cluster actif-actif, effectuez une point-in-time restauration sur l'instance de base de données que vous convertissez. Pour obtenir des instructions, consultez Ajouter une instance de base de données à un cluster actif-actif à l'aide de la restauration point-in-time.

Un cluster actif-actif peut avoir jusqu'à neuf instances de base de données. Effectuez la point-in-time restauration sur l'instance de base de données jusqu'à ce que vous disposiez du nombre d'instances de base de données que vous souhaitez pour le cluster. Lorsque vous effectuez cette point-in-recovery opération, assurez-vous d'associer l'instance de base de données que vous ajoutez à un groupe de paramètres de base de données `rds.group_replication_enabled` défini sur`1`. Sinon, la réplication de groupe ne démarrera pas sur l'instance de base de données nouvellement ajoutée.

Étape 5 : Initialiser le groupe sur l'instance de base de données que vous convertissez

Initialisez le groupe et lancez la réplication :

1. Connectez-vous à l'instance de base de données que vous êtes en train de convertir en SQL client. Pour plus d'informations sur la connexion à une instance RDS for My SQL DB, consultezConnexion à votre instance de base de données MySQL.
2. Dans le SQL client, exécutez les procédures stockées suivantes et remplacez-les *`group_replication_user_password`* par le mot de passe de l'`rdsgrprepladmin`utilisateur. L'`rdsgrprepladmin`utilisateur est réservé aux connexions de réplication de groupe dans un cluster actif-actif. Le mot de passe de cet utilisateur doit être le même sur toutes les instances de base de données d'un cluster actif-actif.

   ```
   call mysql.rds_set_configuration('binlog retention hours', 168); -- 7 days binlog
   call mysql.rds_group_replication_create_user('group_replication_user_password');
   call
    mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
   call mysql.rds_group_replication_start(1);
   ```

   Cet exemple définit la `binlog retention hours` valeur sur`168`, ce qui signifie que les fichiers journaux binaires sont conservés pendant sept jours sur l'instance de base de données. Vous pouvez ajuster cette valeur en fonction de vos besoins.

Cet exemple indique 1 dans la procédure `mysql.rds_group_replication_start` stockée d'initialiser un nouveau groupe avec l'instance de base de données actuelle.

Pour plus d'informations sur les procédures stockées appelées dans l'exemple, consultezGestion des clusters actifs-actifs.

Étape 6 : démarrer la réplication sur les autres instances de base de données du cluster actif-actif

Pour chacune des instances de base de données du cluster actif-actif, utilisez un SQL client pour vous connecter à l'instance et exécutez les procédures stockées suivantes. *`group_replication_user_password`*Remplacez-le par le mot de passe de l'`rdsgrprepladmin`utilisateur.

```
call mysql.rds_set_configuration('binlog retention hours', 168); -- 7 days binlog
call mysql.rds_group_replication_create_user('group_replication_user_password');
call
 mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
call mysql.rds_group_replication_start(0);
```

Cet exemple définit la `binlog retention hours` valeur sur168, ce qui signifie que les fichiers journaux binaires sont conservés pendant sept jours sur chaque instance de base de données. Vous pouvez ajuster cette valeur en fonction de vos besoins.

Cet exemple indique `0` dans la procédure `mysql.rds_group_replication_start` stockée de joindre l'instance de base de données actuelle à un groupe existant.

> **Tip**
>
> Assurez-vous d'exécuter ces procédures stockées sur toutes les autres instances de base de données du cluster actif-actif.

Étape 7 : (Recommandé) Vérifiez l'état du cluster actif-actif

Pour vous assurer que chaque membre du cluster est correctement configuré, vérifiez l'état du cluster en vous connectant à une instance de base de données du cluster actif-actif et en exécutant la commande suivante : SQL

```
SELECT * FROM performance_schema.replication_group_members;
```

Votre sortie doit s'afficher `ONLINE` pour chaque instance `MEMBER_STATE` de base de données, comme dans l'exemple de sortie suivant :

```
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
| CHANNEL_NAME              | MEMBER_ID                            | MEMBER_HOST    |
 MEMBER_PORT | MEMBER_STATE | MEMBER_ROLE | MEMBER_VERSION | MEMBER_COMMUNICATION_STACK
 |
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
| group_replication_applier | 9854d4a2-5d7f-11ee-b8ec-0ec88c43c251 | ip-10-15-3-137 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | 9e2e9c28-5d7f-11ee-8039-0e5d58f05fef | ip-10-15-3-225 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | a6ba332d-5d7f-11ee-a025-0a5c6971197d | ip-10-15-1-83  |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
3 rows in set (0.00 sec)
```

Pour plus d'informations sur les `MEMBER_STATE` valeurs possibles, consultez la section États des serveurs de réplication de groupe dans la section Ma SQL documentation.

Configuration d'un cluster actif-actif avec de nouvelles instances de base de données

Procédez comme suit pour configurer un cluster actif-actif à l'aide des nouvelles instances Amazon RDS for My SQL DB.

Si vous configurez un cluster actif-actif avec plusieurs instances de base de donnéesVPC, assurez-vous de remplir les conditions requises dans. Préparation à un cluster à VPC activités croisées

Rubriques

- Étape 1 : définir les paramètres du cluster actif-actif dans un ou plusieurs groupes de paramètres personnalisés

- Étape 2 : créer de nouvelles instances RDS for My SQL DB pour le cluster actif-actif
- Étape 3 : Spécifier les instances de base de données dans le cluster actif-actif
- Étape 4 : Initialisation du groupe sur une instance de base de données et démarrage de la réplication
- Étape 5 : démarrer la réplication sur les autres instances de base de données du cluster actif-actif
- Étape 6 : (Recommandé) Vérifiez l'état du cluster actif-actif
- Étape 7 : (Facultatif) Importer des données dans une instance de base de données du cluster actif-actif

Étape 1 : définir les paramètres du cluster actif-actif dans un ou plusieurs groupes de paramètres personnalisés

Les instances RDS for My SQL DB d'un cluster actif-actif doivent être associées à un groupe de paramètres personnalisé dont le réglage est correct pour les paramètres requis. Pour plus d'informations sur les paramètres et le réglage requis pour chacun d'entre eux, consultezRéglages de paramètres requis pour les clusters actifs-actifs.

Vous pouvez définir ces paramètres dans de nouveaux groupes de paramètres ou dans des groupes de paramètres existants. Toutefois, pour éviter d'affecter accidentellement les instances de base de données qui ne font pas partie du cluster actif-actif, nous vous recommandons vivement de créer un nouveau groupe de paramètres personnalisé. Les instances de base de données d'un cluster actif-actif peuvent être associées au même groupe de paramètres de base de données ou à différents groupes de paramètres de base de données.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour créer un nouveau groupe de paramètres personnalisés. Pour de plus amples informations, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon. L'exemple suivant exécute la create-db-parameter-group AWS CLI commande pour créer un groupe de paramètres de base de données personnalisé nommé *myactivepg* RDS pour My SQL 8.0 :

Dans Linux, macOS, ou Unix:

```
aws rds create-db-parameter-group \
  --db-parameter-group-name myactivepg \
  --db-parameter-group-family mysql8.0 \
  --description "Parameter group for active-active clusters"
```

Dans Windows:

```
aws rds create-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --db-parameter-group-family mysql8.0 ^
  --description "Parameter group for active-active clusters"
```

Vous pouvez également utiliser le AWS Management Console ou le AWS CLI pour définir les paramètres du groupe de paramètres personnalisés. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

L'exemple suivant exécute la modify-db-parameter-group AWS CLI commande pour définir les paramètres de My SQL 8.0. RDS Pour utiliser cet exemple avec RDS for My SQL 8.4, remplacez `slave_preserve_commit_order` par`replica_preserve_commit_order`.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myactivepg \
  --parameters
 "ParameterName='rds.group_replication_enabled',ParameterValue='1',ApplyMethod=pending-
reboot" \

 "ParameterName='rds.custom_dns_resolution',ParameterValue='1',ApplyMethod=pending-
reboot" \

 "ParameterName='enforce_gtid_consistency',ParameterValue='ON',ApplyMethod=pending-
reboot" \
               "ParameterName='gtid-mode',ParameterValue='ON',ApplyMethod=pending-
reboot" \

 "ParameterName='binlog_format',ParameterValue='ROW',ApplyMethod=immediate" \

 "ParameterName='slave_preserve_commit_order',ParameterValue='ON',ApplyMethod=immediate"
 \

 "ParameterName='group_replication_group_name',ParameterValue='11111111-2222-3333-4444-55555555
reboot"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --parameters
 "ParameterName='rds.group_replication_enabled',ParameterValue='1',ApplyMethod=pending-
reboot" ^

 "ParameterName='rds.custom_dns_resolution',ParameterValue='1',ApplyMethod=pending-
reboot" ^

 "ParameterName='enforce_gtid_consistency',ParameterValue='ON',ApplyMethod=pending-
reboot" ^
               "ParameterName='gtid-mode',ParameterValue='ON',ApplyMethod=pending-
reboot" ^

 "ParameterName='binlog_format',ParameterValue='ROW',ApplyMethod=immediate" ^

 "ParameterName='slave_preserve_commit_order',ParameterValue='ON',ApplyMethod=immediate"
 ^

 "ParameterName='group_replication_group_name',ParameterValue='11111111-2222-3333-4444-55555555
reboot"
```

Étape 2 : créer de nouvelles instances RDS for My SQL DB pour le cluster actif-actif

Les clusters actifs-actifs sont pris en charge pour les versions suivantes des RDS instances My SQL DB :

- Toutes les versions de My SQL version 8.4
- Ma SQL version 8.0.35 et versions mineures supérieures

Vous pouvez créer jusqu'à neuf nouvelles instances de base de données pour le cluster.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour créer de nouvelles instances de base de données. Pour plus d'informations sur la création d'une instance de base de données, consultez Création d'une instance de base de données Amazon RDS. Lorsque vous créez l'instance de base de données, associez-la à un groupe de paramètres de base de données que vous avez créé ou modifié à l'étape précédente.

Étape 3 : Spécifier les instances de base de données dans le cluster actif-actif

Dans le groupe de paramètres de base de données associé à chaque instance de base de données, définissez le `group_replication_group_seeds` paramètre sur les points de terminaison des instances de base de données que vous souhaitez inclure dans le cluster.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour définir le paramètre. Il n'est pas nécessaire de redémarrer l'instance de base de données après avoir défini ce paramètre. Pour de plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

L'exemple suivant exécute la modify-db-parameter-group AWS CLI commande pour définir les paramètres :

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
  --db-parameter-group-name myactivepg \
  --parameters
 "ParameterName='group_replication_group_seeds',ParameterValue='myactivedb1.123456789012.us-
east-1.rds.amazonaws.com:3306,myactivedb2.123456789012.us-
east-1.rds.amazonaws.com:3306,myactivedb3.123456789012.us-
east-1.rds.amazonaws.com:3306',ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
  --db-parameter-group-name myactivepg ^
  --parameters
 "ParameterName='group_replication_group_seeds',ParameterValue='myactivedb1.123456789012.us-
east-1.rds.amazonaws.com:3306,myactivedb2.123456789012.us-
east-1.rds.amazonaws.com:3306,myactivedb3.123456789012.us-
east-1.rds.amazonaws.com:3306',ApplyMethod=immediate"
```

> **Tip**
>
> Assurez-vous de définir le `group_replication_group_seeds` paramètre dans chaque groupe de paramètres de base de données associé à une instance de base de données dans le cluster actif-actif.

Étape 4 : Initialisation du groupe sur une instance de base de données et démarrage de la réplication

Vous pouvez choisir n'importe quelle nouvelle base de données pour initialiser le groupe et démarrer la réplication. Pour ce faire, exécutez les étapes suivantes :

1. Choisissez une instance de base de données dans le cluster actif-actif et connectez-vous à cette instance de base de données dans un SQL client. Pour plus d'informations sur la connexion à une instance RDS for My SQL DB, consultezConnexion à votre instance de base de données MySQL.
2. Dans le SQL client, exécutez les procédures stockées suivantes et remplacez-les *group_replication_user_password* par le mot de passe de l'`rdsgrprepladmin`utilisateur. L'`rdsgrprepladmin`utilisateur est réservé aux connexions de réplication de groupe dans un cluster actif-actif. Le mot de passe de cet utilisateur doit être le même sur toutes les instances de base de données d'un cluster actif-actif.

   ```
   call mysql.rds_set_configuration('binlog retention hours', 168); -- 7 days binlog
   call mysql.rds_group_replication_create_user('group_replication_user_password');
   call
    mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
   call mysql.rds_group_replication_start(1);
   ```

   Cet exemple définit la `binlog retention hours` valeur sur168, ce qui signifie que les fichiers journaux binaires sont conservés pendant sept jours sur l'instance de base de données. Vous pouvez ajuster cette valeur en fonction de vos besoins.

   Cet exemple indique 1 dans la procédure `mysql.rds_group_replication_start` stockée d'initialiser un nouveau groupe avec l'instance de base de données actuelle.

   Pour plus d'informations sur les procédures stockées appelées dans l'exemple, consultezGestion des clusters actifs-actifs.

Étape 5 : démarrer la réplication sur les autres instances de base de données du cluster actif-actif

Pour chacune des instances de base de données du cluster actif-actif, utilisez un SQL client pour vous connecter à l'instance et exécutez les procédures stockées suivantes.

*group_replication_user_password*Remplacez-le par le mot de passe de l'`rdsgrprepladmin`utilisateur.

```
call mysql.rds_set_configuration('binlog retention hours', 168); -- 7 days binlog
call mysql.rds_group_replication_create_user('group_replication_user_password');
call
 mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
call mysql.rds_group_replication_start(0);
```

Cet exemple définit la `binlog retention hours` valeur sur168, ce qui signifie que les fichiers journaux binaires sont conservés pendant sept jours sur chaque instance de base de données. Vous pouvez ajuster cette valeur en fonction de vos besoins.

Cet exemple indique `0` dans la procédure `mysql.rds_group_replication_start` stockée de joindre l'instance de base de données actuelle à un groupe existant.

> **Tip**
>
> Assurez-vous d'exécuter ces procédures stockées sur toutes les autres instances de base de données du cluster actif-actif.

Étape 6 : (Recommandé) Vérifiez l'état du cluster actif-actif

Pour vous assurer que chaque membre du cluster est correctement configuré, vérifiez l'état du cluster en vous connectant à une instance de base de données du cluster actif-actif et en exécutant la commande suivante : SQL

```
SELECT * FROM performance_schema.replication_group_members;
```

Votre sortie doit s'afficher `ONLINE` pour chaque instance `MEMBER_STATE` de base de données, comme dans l'exemple de sortie suivant :

```
+---------------------------+--------------------------------------
+----------------+-------------+--------------+-------------+----------------
+-----------------------------+
| CHANNEL_NAME              | MEMBER_ID                            | MEMBER_HOST    |
 MEMBER_PORT | MEMBER_STATE | MEMBER_ROLE | MEMBER_VERSION | MEMBER_COMMUNICATION_STACK
 |
```

```
+---------------------------+--------------------------------------
+---------------+-------------+-------------+-------------+---------------
+----------------------------+
| group_replication_applier | 9854d4a2-5d7f-11ee-b8ec-0ec88c43c251 | ip-10-15-3-137 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | 9e2e9c28-5d7f-11ee-8039-0e5d58f05fef | ip-10-15-3-225 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | a6ba332d-5d7f-11ee-a025-0a5c6971197d | ip-10-15-1-83  |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
+---------------------------+--------------------------------------
+---------------+-------------+-------------+-------------+---------------
+----------------------------+
3 rows in set (0.00 sec)
```

Pour plus d'informations sur les `MEMBER_STATE` valeurs possibles, consultez la section États des serveurs de réplication de groupe dans la section Ma SQL documentation.

Étape 7 : (Facultatif) Importer des données dans une instance de base de données du cluster actif-actif

Vous pouvez importer des données d'une SQL base de données My dans une instance de base de données du cluster actif-actif. Une fois les données importées, Group Replication les réplique sur les autres instances de base de données du cluster.

Pour plus d'informations sur l'importation de données, consultezImportation de données vers une base de données Amazon RDS for MySQL avec réduction des temps d'arrêt.

Ajouter une instance de base de données à un cluster actif-actif

Vous pouvez ajouter une instance de base de données à un cluster SQL actif Amazon RDS for My en restaurant un instantané de base de données ou en rétablissant une instance de base de données à un moment donné. Un cluster actif-actif peut inclure jusqu'à neuf instances de base de données.

Lorsque vous restaurez une instance de base de données à un moment donné, elle inclut généralement des transactions plus récentes qu'une instance de base de données restaurée à partir d'un instantané de base de données. Lorsque l'instance de base de données possède des transactions plus récentes, moins de transactions doivent être appliquées lorsque vous démarrez la réplication. Ainsi, l'utilisation de point-in-time la restauration pour ajouter une instance de base de données à un cluster est généralement plus rapide que la restauration à partir d'un instantané de base de données.

Rubriques

- Ajouter une instance de base de données à un cluster actif-actif à l'aide de la restauration point-in-time
- Ajouter une instance de base de données à un cluster actif-actif à l'aide d'un instantané de base de données

Ajouter une instance de base de données à un cluster actif-actif à l'aide de la restauration point-in-time

Vous pouvez ajouter une instance de base de données à un cluster actif-actif en effectuant une point-in-time restauration sur une instance de base de données du cluster.

Pour plus d'informations sur la restauration d'une instance de base de données à un autre moment dans le temps Région AWS, consultezRéplication des sauvegardes automatisées vers une autre Région AWS.

Pour ajouter une instance de base de données à un cluster actif-actif à l'aide de la restauration point-in-time

1. Créez une nouvelle instance de base de données en effectuant une point-in-time restauration sur une instance de base de données du cluster actif-actif.

   Vous pouvez effectuer une point-in-time restauration sur n'importe quelle instance de base de données du cluster pour créer la nouvelle instance de base de données. Pour obtenir des instructions, consultez Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

   > **Important**
   >
   > Pendant point-in-time-recovery, associez la nouvelle instance de base de données à un groupe de paramètres de base de données dont les paramètres de cluster actif-actif sont définis. Sinon, la réplication de groupe ne démarrera pas sur la nouvelle instance de base de données. Pour plus d'informations sur les paramètres et le réglage requis pour chacun d'entre eux, reportez-vous àRéglages de paramètres requis pour les clusters actifs-actifs.

> **Tip**
>
> Si vous prenez un instantané de l'instance de base de données avant de commencer la point-in-time restauration, vous pourrez peut-être réduire le temps nécessaire pour appliquer les transactions sur la nouvelle instance de base de données.

2. Ajoutez l'instance de base de données au `group_replication_group_seeds` paramètre de chaque groupe de paramètres de base de données associé à une instance de base de données du cluster actif-actif, y compris le groupe de paramètres de base de données que vous avez associé à la nouvelle instance de base de données.

   Pour de plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

3. Dans un SQL client, connectez-vous à la nouvelle instance de base de données et appelez la procédure mysql.rds_group_replication_set_recovery_channel stockée. Remplacez *`group_replication_user_password`* avec le mot de passe de l'`rdsgrpreplad`minutilisateur.

   ```
   call
    mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
   ```

4. À l'aide du SQL client, appelez la procédure mysql.rds_group_replication_start stockée pour démarrer la réplication :

   ```
   call mysql.rds_group_replication_start(0);
   ```

Ajouter une instance de base de données à un cluster actif-actif à l'aide d'un instantané de base de données

Vous pouvez ajouter une instance de base de données à un cluster actif-actif en créant un instantané de base de données d'une instance de base de données dans le cluster, puis en restaurant l'instantané de base de données.

Pour plus d'informations sur la copie d'un instantané vers un autre Région AWS, voirthe section called “Copie entre régions”.

Pour ajouter une instance de base de données à un cluster actif-actif à l'aide d'un instantané de base de données

1. Créez un instantané de base de données d'une instance de base de données dans le cluster actif-actif.

   Vous pouvez créer un instantané de base de données de n'importe quelle instance de base de données du cluster. Pour obtenir des instructions, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

2. Restaurez une instance de base de données à partir du snapshot de base de données.

   Au cours de l'opération de restauration des instantanés, associez la nouvelle instance de base de données à un groupe de paramètres de base de données dont les paramètres de cluster actif-actif sont définis. Pour plus d'informations sur les paramètres et le réglage requis pour chacun d'entre eux, reportez-vous àRéglages de paramètres requis pour les clusters actifs-actifs.

   Pour plus d'informations sur la restauration d'une instance de base de données à partir d'un instantané de base de données, consultezRestauration vers une instance de base de données.

3. Ajoutez l'instance de base de données au `group_replication_group_seeds` paramètre de chaque groupe de paramètres de base de données associé à une instance de base de données du cluster actif-actif, y compris le groupe de paramètres de base de données que vous avez associé à la nouvelle instance de base de données.

   Pour de plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

4. Dans un SQL client, connectez-vous à la nouvelle instance de base de données et appelez la procédure mysql.rds_group_replication_set_recovery_channel stockée. Remplacez *`group_replication_user_password`* avec le mot de passe de l'`rdsgrpreplad``min`utilisateur.

   ```
   call
    mysql.rds_group_replication_set_recovery_channel('group_replication_user_password');
   ```

5. À l'aide du SQL client, appelez la procédure mysql.rds_group_replication_start stockée pour démarrer la réplication :

   ```
   call mysql.rds_group_replication_start(0);
   ```

Surveillance des clusters actifs-actifs

La surveillance des clusters actifs-actifs dans Amazon RDS for My SQL est essentielle pour suivre les performances, l'intégrité de la réplication et la synchronisation des nœuds. Vous pouvez surveiller votre cluster actif-actif en vous connectant à une instance de base de données du cluster et en exécutant la commande suivante : SQL

```
SELECT * FROM performance_schema.replication_group_members;
```

Votre sortie doit s'afficher `ONLINE` pour chaque instance `MEMBER_STATE` de base de données, comme dans l'exemple de sortie suivant :

```
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
| CHANNEL_NAME              | MEMBER_ID                            | MEMBER_HOST    |
 MEMBER_PORT | MEMBER_STATE | MEMBER_ROLE | MEMBER_VERSION | MEMBER_COMMUNICATION_STACK
 |
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
| group_replication_applier | 9854d4a2-5d7f-11ee-b8ec-0ec88c43c251 | ip-10-15-3-137 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | 9e2e9c28-5d7f-11ee-8039-0e5d58f05fef | ip-10-15-3-225 |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
| group_replication_applier | a6ba332d-5d7f-11ee-a025-0a5c6971197d | ip-10-15-1-83  |
      3306 | ONLINE       | PRIMARY     | 8.0.35         | MySQL                       |
+---------------------------+--------------------------------------
+---------------+-------------+--------------+-------------+---------------
+----------------------------+
3 rows in set (0.00 sec)
```

Pour plus d'informations sur les `MEMBER_STATE` valeurs possibles, consultez la section États des serveurs de réplication de groupe dans la section Ma SQL documentation.

Arrêt de la réplication de groupe sur une instance de base de données dans un cluster actif-actif

Vous pouvez arrêter la réplication de groupe sur une instance de base de données dans un cluster actif-actif. Lorsque vous arrêtez la réplication de groupe, l'instance de base de données est placée en

super-read-only mode jusqu'à ce que la réplication soit redémarrée ou que cette instance de base de données soit supprimée du cluster actif-actif. Pour plus d'informations sur super-read-only le mode, consultez la section Ma SQL documentation.

Pour arrêter temporairement la réplication de groupe pour un cluster actif-actif

1. Connectez-vous à une instance de base de données dans le cluster actif-actif à l'aide d'un SQL client.

   Pour plus d'informations sur la connexion à une instance RDS for My SQL DB, consultezConnexion à votre instance de base de données MySQL.
2. Dans le SQL client, appelez la procédure mysql.rds_group_replication_stop stockée :

```
call mysql.rds_group_replication_stop();
```

Modification du nom d'une instance de base de données dans un cluster actif-actif

Vous pouvez modifier le nom d'une instance de base de données dans un cluster actif-actif. Pour renommer plusieurs instances de base de données dans un cluster actif-actif, faites-le une instance de base de données à la fois. Renommez donc une instance de base de données et rejoignez-la au cluster avant de renommer l'instance de base de données suivante.

Pour renommer une instance de base de données dans un cluster actif-actif

1. Connectez-vous à l'instance de base de données dans un SQL client et appelez la procédure mysql.rds_group_replication_stop stockée :

```
call mysql.rds_group_replication_stop();
```
2. Renommez l'instance de base de données en suivant les instructions deAffectation d'un nouveau nom à une instance DB.
3. Modifiez le `group_replication_group_seeds` paramètre dans chaque groupe de paramètres de base de données associé à une instance de base de données dans le cluster actif-actif.

   Dans le réglage des paramètres, remplacez l'ancien point de terminaison de l'instance de base de données par le nouveau point de terminaison de l'instance de base de données. Pour de

plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

4. Connectez-vous à l'instance de base de données dans un SQL client et appelez la procédure mysql.rds_group_replication_start stockée :

```
call mysql.rds_group_replication_start(0);
```

Supprimer une instance de base de données d'un cluster actif-actif

Lorsque vous supprimez une instance de base de données d'un cluster actif-actif, elle redevient une instance de base de données autonome.

Pour supprimer une instance de base de données d'un cluster actif-actif

1. Connectez-vous à l'instance de base de données dans un SQL client et appelez la procédure mysql.rds_group_replication_stop stockée :

```
call mysql.rds_group_replication_stop();
```

2. Modifiez le `group_replication_group_seeds` paramètre des instances de base de données qui resteront dans le cluster actif-actif.

   Dans le `group_replication_group_seeds` paramètre, supprimez l'instance de base de données que vous supprimez du cluster actif-actif. Pour de plus amples informations sur la définition des paramètres, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

3. Modifiez les paramètres de l'instance de base de données que vous supprimez du cluster actif-actif afin qu'elle ne fasse plus partie du cluster.

   Vous pouvez associer l'instance de base de données à un autre groupe de paramètres ou modifier les paramètres du groupe de paramètres de base de données associé à l'instance de base de données. Les paramètres à modifier incluent `group_replication_group_namerds.group_replication_enabled`, et`group_replication_group_seeds`. Pour plus d'informations sur les paramètres de cluster actif-actif, consultez. Réglages de paramètres requis pour les clusters actifs-actifs

Si vous modifiez les paramètres d'un groupe de paramètres de base de données, assurez-vous que le groupe de paramètres de base de données n'est pas associé à d'autres instances de base de données du cluster actif-actif.

4. Redémarrez l'instance de base de données que vous avez supprimée du cluster actif-actif pour que les nouveaux paramètres prennent effet.

   Pour obtenir des instructions, consultez Redémarrage d'une instance de base de données.

Exportation de données depuis une instance My SQL DB à l'aide de la réplication

Pour exporter des données d'une instance RDS for My SQL DB vers une SQL instance My exécutée en externe à AmazonRDS, vous pouvez utiliser la réplication. Dans ce scénario, l'instance My SQL DB est l'instance My SQL DB source, et l'SQLinstance My exécutée en externe à Amazon RDS est la My SQL database externe.

La base de données externe My SQL Database peut être exécutée sur site dans votre centre de données ou sur une EC2 instance Amazon. La SQL base de données My DB externe doit exécuter la même version que l'instance My SQL DB source, ou une version ultérieure.

La réplication vers une SQL base de données My DB externe n'est prise en charge que pendant le temps nécessaire pour exporter une base de données depuis l'instance My SQL DB source. La réplication doit être interrompue lorsque les données ont été exportées et que les applications peuvent commencer à accéder à l'SQLinstance externe My.

La liste suivante montre les étapes à suivre. Chaque étape est présentée plus en détail dans les sections ultérieures.

1. Préparez une instance My SQL DB externe.
2. Préparez l'instance My SQL DB source pour la réplication.
3. Utilisez l'utilitaire mysqldump pour transférer la base de données de l'instance My SQL DB source vers l'instance My DB externe. SQL
4. Lancez la réplication vers la base de SQL données My database externe.
5. Une fois l'exportation terminée, arrêtez la réplication.

Préparer une base de SQL données My externe

Procédez comme suit pour préparer la base de données externe Ma SQL base de données.

Pour préparer la base de données externe Ma SQL base de données

1. Installez la base de données externe Ma SQL base de données.
2. Connectez-vous à la base de données externe My SQL Database en tant qu'utilisateur principal. Créez ensuite les utilisateurs requis pour prendre en charge les administrateurs, les applications et les services qui accèdent à la base de données.

3. Suivez les instructions de la section Ma SQL documentation pour préparer la SQL base de données My externe sous forme de réplique. Pour plus d'informations, reportez-vous à la section Configuration de la configuration de la réplique dans la section Ma SQL documentation.
4. Configurez une règle de sortie pour que la base de SQL données My Database externe fonctionne comme une réplique en lecture lors de l'exportation. La règle de sortie permet à la SQL base de données externe My DB de se connecter à l'instance My SQL DB source pendant la réplication. Spécifiez une règle de sortie qui autorise les connexions au protocole de contrôle de transmission (TCP) au port et à l'adresse IP de l'instance My SQL DB source.

   Spécifiez les règles de sortie appropriées pour votre environnement :

   - Si la base de données externe My SQL Database s'exécute dans une EC2 instance Amazon dans un cloud privé virtuel (VPC) basé sur le VPC service Amazon, spécifiez les règles de sortie dans un groupe VPC de sécurité. Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité.
   - Si la base de données externe My SQL database est installée sur site, spécifiez les règles de sortie dans un pare-feu.
5. Si la base de données externe My SQL database s'exécute dans unVPC, configurez les règles de la liste de contrôle d'VPCaccès (ACL) en plus de la règle de sortie du groupe de sécurité :

   - Configurez une règle d'ACLentrée autorisant le TCP trafic vers les ports 1024 à 65535 à partir de l'adresse IP de l'instance My DB source. SQL
   - Configurez une ACL règle de sortie autorisant le TCP trafic sortant vers le port et l'adresse IP de l'instance My SQL DB source.

   Pour plus d'informations sur le VPC réseau AmazonACLs, consultez la section Réseau ACLs dans le guide de VPC l'utilisateur Amazon.
6. (Facultatif) Définissez le paramètre `max_allowed_packet` sur la taille maximale pour éviter les erreurs de réplication. Nous recommandons ce paramètre.

Préparer l'instance My SQL DB source

Procédez comme suit pour préparer l'instance My SQL DB source en tant que source de réplication.

Pour préparer l'instance My SQL DB source

1. Assurez-vous que votre ordinateur client possède assez d'espace disque pour enregistrer les journaux binaires lors de la configuration de la réplication.
2. Connectez-vous à l'instance My SQL DB source et créez un compte de réplication en suivant les instructions de la section Création d'un utilisateur pour la réplication dans la SQL documentation Ma.
3. Configurez les règles d'entrée sur le système exécutant l'instance My SQL DB source pour permettre à la SQL base de données My DB externe de se connecter pendant la réplication. Spécifiez une règle d'entrée qui autorise TCP les connexions au port utilisé par l'instance My SQL DB source à partir de l'adresse IP de la SQL base de données My DB externe.
4. Spécifiez les règles de sortie :

   - Si l'instance My SQL DB source s'exécute dans unVPC, spécifiez les règles d'entrée dans un groupe VPC de sécurité. Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité.
5. Si l'instance source My SQL DB s'exécute dans unVPC, configurez des VPC ACL règles en plus de la règle d'entrée du groupe de sécurité :

   - Configurez une règle d'ACLentrée pour autoriser TCP les connexions au port utilisé par l'RDSinstance Amazon à partir de l'adresse IP de la base de SQL données My database externe.
   - Configurez une règle de ACL sortie pour autoriser TCP les connexions entre les ports 1024 et 65535 à l'adresse IP de la base de données My database externe. SQL

   Pour plus d'informations sur le VPC réseau AmazonACLs, consultez la section Réseau ACLs dans le guide de VPC l'utilisateur Amazon.
6. Assurez-vous que la période de rétention des sauvegardes soit assez longue pour qu'aucun journal binaire ne soit purgé pendant l'exportation. Si l'un des journaux est purgé avant la fin de l'exportation, vous devez redémarrer la réplication depuis le début. Pour plus d'informations sur la configuration de la période de rétention des sauvegardes, consultez Présentation des sauvegardes.
7. Utilisez la procédure stockée `mysql.rds_set_configuration` pour définir une période de conservation du journal binaire suffisamment longue pour que les journaux binaires ne soient pas purgés pendant l'exportation. Pour de plus amples informations, veuillez consulter Accès aux journaux binaires MySQL.

8. Créez une réplique Amazon RDS read à partir de l'instance My SQL DB source pour garantir que les journaux binaires de l'instance My SQL DB source ne sont pas purgés. Pour de plus amples informations, veuillez consulter Création d'un réplica en lecture.
9. Une fois la réplique Amazon RDS read créée, appelez la procédure `mysql.rds_stop_replication` stockée pour arrêter le processus de réplication. L'instance My SQL DB source ne purge plus ses fichiers journaux binaires, ils sont donc disponibles pour le processus de réplication.
10. (Facultatif) Définissez le paramètre `max_allowed_packet` et le paramètre `slave_max_allowed_packet` sur la taille maximale pour éviter les erreurs de réplication. La taille maximale pour les deux paramètres est de 1 Go. Nous recommandons ces valeurs pour les deux paramètres. Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Copier la base de données

Effectuez les étapes suivantes pour copier la base de données.

Pour copier la base de données

1. Connectez-vous à la réplique en RDS lecture de l'instance My SQL DB source et exécutez l'SQL`SHOW REPLICA STATUS\G`instruction My. Notez les valeurs pour les éléments suivants :
   - `Master_Host`
   - `Master_Port`
   - `Master_Log_File`
   - `Exec_Master_Log_Pos`

   > **Note**
   >
   > Les versions précédentes de My SQL utilisaient à la `SHOW SLAVE STATUS` place de`SHOW REPLICA STATUS`. Si vous utilisez une SQL version My antérieure à 8.0.23, utilisez. `SHOW SLAVE STATUS`
2. Utilisez l'utilitaire mysqldump pour créer un instantané qui copie les données d'Amazon sur votre ordinateur RDS client local. Assurez-vous que votre ordinateur client possède assez

d'espace disque pour contenir les fichiers `mysqldump` des bases de données à répliquer. Ce processus peut prendre plusieurs heures pour les bases de données très volumineuses. Suivez les instructions de la section Création d'un instantané de données à l'aide de mysqldump dans la section Ma documentation. SQL

L'exemple suivant exécute `mysqldump` sur un client et écrit le vidage dans un fichier.

Dans Linux, macOS, ou Unix:

```
mysqldump -h source_MySQL_DB_instance_endpoint \
    -u user \
    -ppassword \
    --port=3306 \
    --single-transaction \
    --routines \
    --triggers \
    --databases  database database2 > path/rds-dump.sql
```

Dans Windows:

```
mysqldump -h source_MySQL_DB_instance_endpoint ^
    -u user ^
    -ppassword ^
    --port=3306 ^
    --single-transaction ^
    --routines ^
    --triggers ^
    --databases  database database2 > path\rds-dump.sql
```

Vous pouvez charger le fichier de sauvegarde dans la base de données externe Ma SQL base de données. Pour plus d'informations, consultez la section Reloading SQL -Format Backups dans la section Ma SQL documentation. Vous pouvez exécuter un autre utilitaire pour charger les données dans la base de données externe Ma SQL base de données.

Terminer l'exportation

Effectuez les étapes suivantes pour terminer l'exportation.

Pour terminer l'exportation

1. Utilisez l'SQL`CHANGE MASTER`instruction My pour configurer la base de SQL données My externe. Spécifiez l'ID et le mot de passe de l'utilisateur auquel ont été attribuées les autorisations `REPLICATION SLAVE`. Spécifiez les `Exec_Master_Log_Pos` valeurs `Master_HostMaster_Port`,`Relay_Master_Log_File`, et que vous avez obtenues à partir de l'SQL`SHOW REPLICA STATUS\G`instruction My que vous avez exécutée sur la réplique RDS lue. Pour plus d'informations, consultez CHANGEMASTERla section TO Statement dans la section Ma SQL documentation.

   > **Note**
   >
   > Les versions précédentes de My SQL utilisaient à la `SHOW SLAVE STATUS` place de`SHOW REPLICA STATUS`. Si vous utilisez une SQL version My antérieure à 8.0.23, utilisez. `SHOW SLAVE STATUS`

2. Utilisez la SQL `START REPLICA` commande My pour lancer la réplication depuis l'instance My SQL DB source vers la SQL base de données My DB externe.

   Cela démarre la réplication à partir de l'instance My SQL DB source et exporte toutes les modifications de source survenues après l'arrêt de la réplication à partir de la réplique Amazon RDS read.

   > **Note**
   >
   > Les versions précédentes de My SQL utilisaient à la `START SLAVE` place de`START REPLICA`. Si vous utilisez une SQL version My antérieure à 8.0.23, utilisez. `START SLAVE`

3. Exécutez la SQL `SHOW REPLICA STATUS\G` commande My sur la SQL base de données My externe pour vérifier qu'elle fonctionne comme une réplique en lecture. Pour plus d'informations sur l'interprétation des résultats, voir SHOWSLAVE| REPLICA STATUS Déclaration dans ma SQL documentation.
4. Une fois que la réplication sur l'instance My SQL DB externe a rattrapé l'instance My SQL DB source, utilisez la SQL `STOP REPLICA` commande My pour arrêter la réplication depuis l'instance My SQL DB source.

> **Note**
>
> Les versions précédentes de My SQL utilisaient à la `STOP SLAVE` place de`STOP REPLICA`. Si vous utilisez une SQL version My antérieure à 8.0.23, utilisez. `STOP SLAVE`

5. Sur la réplique Amazon RDS read, appelez la procédure `mysql.rds_start_replication` stockée. Cela permet RDS à Amazon de commencer à purger les fichiers journaux binaires de l'instance My SQL DB source.

Options pour les instances My SQL DB

Vous trouverez ci-dessous une description des options ou des fonctionnalités supplémentaires disponibles pour les RDS instances Amazon exécutant le moteur My SQL DB. Pour activer ces options, vous pouvez les ajouter à un groupe d'options personnalisé, puis associer ce dernier à votre instance de base de données. Pour plus d'informations sur l'utilisation de groupes d'options, consultez Utilisation de groupes d'options.

Amazon RDS prend en charge les options suivantes pour My SQL :

Option	ID d'option	Versions du moteur
Support du plugin d'audit MariaDB pour My SQL	`MARIADB_AUDIT_PLUGIN`	Toutes les versions de My SQL 8.4 Mes SQL versions 8.0.28 et supérieures 8.0 Toutes les versions de My SQL 5.7
Prise en charge memcached MySQL	`MEMCACHED`	Toutes les versions de My SQL 5.7 et 8.0

Support du plugin d'audit MariaDB pour My SQL

Amazon RDS propose un plugin d'audit pour les instances SQL de My database basé sur le plugin d'audit open source MariaDB. Pour plus d'informations, consultez le plug-in d'audit pour le GitHub référentiel My SQL Server.

> **Note**
>
> Le plugin d'audit pour My SQL est basé sur le plugin d'audit MariaDB. Tout au long de cet article, nous l'appelons Plug-in d'audit MariaDB.

Le plugin d'audit MariaDB enregistre l'activité de la base de données, y compris la connexion des utilisateurs à la base de données et les requêtes exécutées sur la base de données. L'enregistrement de l'activité de la base de données est stocké dans un fichier journal.

Paramètres de l'option du plugin d'audit

Amazon RDS prend en charge les paramètres suivants pour l'option du plug-in d'audit MariaDB.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
`SERVER_AUDIT_FILE_PATH`	`/rdsdbdata/log/audit/`	`/rdsdbdata/log/audit/`	Emplacement du fichier journal. Le fichier journal contient l'enregistrement de l'activité spécifiée dans `SERVER_AUDIT_EVENTS` . Pour de plus amples informations, veuillez consulter Liste et affichage des fichiers journaux de base de données et Fichiers journaux de base de données MySQL.
`SERVER_AUDIT_FILE_ROTATE_SIZE`	1–1000000000	1000000	Taille en octets qui, lorsqu'elle est atteinte, entraîne la rotation du fichier. Pour plus d'informations, consultez Présentation des journaux de base de données RDS for MySQL.
`SERVER_AUDIT_FILE_`	0–100	9	Nombre de rotations de journaux à enregistrer quand `server_audit_output_type=fi`

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
ROTATIONS			`le` . S'il est défini sur 0, le fichier journal ne pivote jamais. Pour plus d'informations, consultez Présentation des journaux de base de données RDS for MySQL et Télécharg ement d'un fichier journal de base de données.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_EVENTS	CONNECT, QUERY, QUERY_DDL, QUERY_DML, QUERY_DML_NO_SELECT, QUERY_DCL	CONNECT, QUERY	Types d'activités à enregistrer dans le journal. L'installation du plugin d'audit MariaDB est elle-même enregistrée. • CONNECT : Permet d'enregistrer les connexions à la base de données, réussies ou non, et les déconnexions de la base de données. • QUERY : Permet d'enregistrer le texte de toutes les requêtes exécutées sur la base de données. • QUERY_DDL : Similaire à l'QUERYévénement, mais renvoie uniquement les requêtes du langage de définition des données (DDL) (CREATEALTER,, etc.). • QUERY_DML : Similaire à l'QUERYévénement, mais renvoie uniquement les requêtes du langage de manipulation des données (DMLINSERT) (,, etc., etc.SELECT). UPDATE • QUERY_DML_NO_SELECT : Similaire à l'événement QUERY_DML, mais ne journalise pas les requêtes SELECT. • QUERY_DCL : Similaire à l'QUERYévénement, mais renvoie uniquement des requêtes en langage de contrôle des données (DCL) (GRANTREVOKE,, etc.). For MySQL, n'TABLEest pas pris en charge.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_INCL_USERS	Plusieurs valeurs séparées par des virgules	Aucune	Incluez uniquement l'activité des utilisateurs spécifiés. Par défaut, l'activité est enregistrée pour tous les utilisateurs. SERVER_AUDIT_INCL_USERS et SERVER_AUDIT_EXCL_USERS sont mutuellement exclusifs. Si vous ajoutez des valeurs à SERVER_AUDIT_INCL_USERS, assurez-vous qu'aucune valeur n'est ajoutée à SERVER_AUDIT_EXCL_USERS.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_EXCL_USERS	Plusieurs valeurs séparées par des virgules	Aucune	Excluez l'activité des utilisateurs spécifiés. Par défaut, l'activité est enregistrée pour tous les utilisateurs. SERVER_AUDIT_INCL_USERS et SERVER_AUDIT_EXCL_USERS sont mutuellement exclusifs. Si vous ajoutez des valeurs à SERVER_AUDIT_EXCL_USERS , assurez-vous qu'aucune valeur n'est ajoutée à SERVER_AUDIT_INCL_USERS . L'utilisateur rdsadmin interroge la base de données par seconde pour vérifier l'intégrité de la base de données. En fonction de vos autres paramètres, cette activité peut éventuellement provoquer un accroissement considérable et rapide de la taille de votre fichier journal. Si vous n'avez pas besoin d'enregistrer cette activité, ajoutez l'utilisateur rdsadmin à la liste SERVER_AUDIT_EXCL_USERS . **Note** CONNECTL'activité est toujours enregistrée pour tous les utilisateurs, même si l'utilisateur est spécifié pour ce paramètre d'option.
SERVER_AUDIT_LOGGING	ON	ON	La journalisation est active. La seule valeur valide est ON. Amazon RDS ne prend pas en charge la désactivation de la journalisation. Si vous souhaitez désactiver la journalisation, supprimez le plugin d'audit MariaDB. Pour plus d'informations, consultez Suppression du plugin d'audit MariaDB.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
SERVER_AUDIT_QUERY_LOG_LIMIT	0–2147483647	1 024	Limite de longueur de la chaîne de requête dans un enregistrement.

Ajout du plugin d'audit MariaDB

Le processus général pour ajouter le plug-in d'audit MariaDB à une instance de base de données est le suivant :

- Créez un groupe d'options ou copiez ou modifiez un groupe d'options existant.
- Ajouter l'option au groupe d'options
- Associer un groupe d'options à une instance de base de données

Une fois que vous ajoutez le plug-in d'audit MariaDB, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, l'audit commence immédiatement.

> **Important**
>
> L'ajout du plug-in d'audit MariaDB à une instance de base de données peut entraîner une interruption de service. Nous vous recommandons d'ajouter le plug-in d'audit MariaDB pendant une fenêtre de maintenance ou lorsque la charge de travail de base de données est faible.

Pour ajouter le plug-in d'audit MariaDB

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options de base de données personnalisé. Choisissez mysql pour le moteur, et choisissez 5.7, 8.0 ou 8.4 pour la version majeure du moteur. Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'PLUGINoption MARIADBAUDIT_ _ au groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options. Pour plus d'informations sur chaque paramètre, consultez Paramètres de l'option du plugin d'audit.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante.
   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Format des journaux d'audit

Les fichiers journaux sont représentés sous forme de fichiers de variables (CSV) séparés par des virgules au format UTF -8.

> **Tip**
>
> Les entrées de fichier journal ne sont pas classées par ordre séquentiel. Pour ordonner les entrées, utilisez la valeur d'horodatage. Pour consulter les derniers événements, vous devrez peut-être passer en revue tous les fichiers journaux. Pour plus de flexibilité dans le tri et la recherche des données des journaux, activez le paramètre permettant de télécharger les journaux d'audit CloudWatch et de les consulter à l'aide de l' CloudWatch interface.
> Pour afficher les données d'audit avec plusieurs types de champs et avec une sortie au JSON format, vous pouvez également utiliser la fonctionnalité Database Activity Streams. Pour de plus amples informations, veuillez consulter Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données.

Les fichiers journaux d'audit incluent les informations séparées par des virgules suivantes en lignes, dans l'ordre indiqué :

Champ	Description
timestamp	YYYYMMDD suivi de HH:MI:SS (format 24 heures) correspondant à l'événement enregistré.
serverhost	Le nom de l'instance pour laquelle*** l'événement est consigné.
username	Le nom d'utilisateur connecté de l'utilisateur.
hôte	L'hôte à partir duquel** l'utilisateur s'est connecté.
connectionid	Le numéro d'identification de la connexion pour l'opération consignée.
queryid	Le numéro d'identification de la requête qui peut être utilisé pour trouver les événements de la table relationnelle et les requêtes liées. Pour les événements TABLE, plusieurs lignes sont ajoutées.
fonctionnement	Le type d'action enregistrée. Les valeurs possibles sont : CONNECT, QUERY, READ, WRITE, CREATE, ALTER, RENAME et DROP.
database	La base de données active, telle que définie par la commande USE.
objet	Pour les événements QUERY, cette valeur indique la demande effectuée par la base de données. Pour les événements TABLE, cette valeur indique le nom de la table.
retcode	Le code de retour de l'opération consignée.
connection_type	État de sécurité de la connexion au serveur. Les valeurs possibles sont : • 0 : non défini • 1— TCP /IP • 2 : socket • 3 : canal nommé • 4 – SSL/TLS • 5 : mémoire partagée

Affichage et téléchargement du journal du plugin d'audit MariaDB

Une fois que vous activez le plugin d'audit MariaDB, vous accéder aux résultats dans les fichiers journaux de la même manière que tous les autres fichiers journaux texte. Les fichiers journaux d'audit se trouvent dans `/rdsdbdata/log/audit/`. Pour plus d'informations sur l'affichage du fichier journal dans la console, consultez Liste et affichage des fichiers journaux de base de données. Pour plus d'informations sur le téléchargement du fichier journal, consultez Téléchargement d'un fichier journal de base de données.

Modification des paramètres de plug-in d'audit MariaDB

Une fois que vous activez le plug-in d'audit MariaDB, vous pouvez modifier les paramètres. Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option. Pour plus d'informations sur chaque paramètre, consultez Paramètres de l'option du plugin d'audit.

Suppression du plugin d'audit MariaDB

Amazon RDS ne prend pas en charge la désactivation de la connexion dans le plug-in d'audit MariaDB. Toutefois, vous pouvez supprimer le plugin dans une instance de base de données. Lorsque vous supprimez le plugin d'audit MariaDB, l'instance de base de données est automatiquement redémarrée pour cesser l'audit.

Pour supprimer le plugin d'audit MariaDB d'une instance de base de données, effectuez l'une des actions suivantes :

- Supprimez l'option de plugin d'audit MariaDB du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options
- Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas le plugin. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Prise en charge memcached MySQL

Amazon RDS prend en charge l'utilisation de l'interface memcached pour les tableaux InnoDB introduits dans MySQL 5.6. L'API memcached permet aux applications d'utiliser les tables InnoDB de la même façon que les magasins de données clé-valeur NoSQL.

> **Note**
>
> L'interface memcached n'est plus disponible dans MySQL 8.4. Lorsque vous mettez à niveau vos instances de base de données vers MySQL 8.4, vous devez les désactiver memcached dans les groupes d'options existants.

L'interface memcached est un cache simple basé sur les clés. Les applications utilisent memcached pour insérer, manipuler et récupérer les paires de données clé-valeur du cache. MySQL 5.6 a présenté un plug-in qui implémente un service démon exposant les données des tables InnoDB via le protocole memcached. Pour de plus amples informations sur le plug-in MySQL memcached, veuillez consulter InnoDB Integration with memcached.

Pour activer la prise en charge memcached d'une instance de base de données RDS for MySQL

1. Déterminez le groupe de sécurité à utiliser pour contrôler l'accès à l'interface memcached. Si l'ensemble d'applications qui utilise déjà l'interface SQL est identique à celui qui accède à l'interface memcached, vous pouvez utiliser le groupe de sécurité VPC existant utilisé par l'interface SQL. Si un ensemble différent d'applications accède à l'interface memcached, définissez un nouveau groupe de sécurité VPC ou DB. Pour plus d'informations sur la gestion des groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité
2. Créez un groupe d'options de base de données personnalisé, en sélectionnant MySQL comme type et version du moteur. Pour plus d'informations sur la création d'un groupe d'options, consultez Création d'un groupe d'options.
3. Ajoutez l'option MEMCACHED au groupe d'options. Spécifiez le port que l'interface memcached utilisera, et le groupe de sécurité à utiliser pour contrôler l'accès à l'interface. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
4. Modifiez les options pour configurer les paramètres memcached, le cas échéant. Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option.

5. Appliquez le groupe d'options à une instance. Amazon RDS active la prise en charge de memcached pour cette instance lorsque le groupe d'options est appliqué :

   - Vous activez la prise en charge memcached pour une nouvelle instance en spécifiant le groupe d'options personnalisé lorsque vous lancez l'instance. Pour plus d'informations sur le lancement d'une instance MySQL, consultez Création d'une instance de base de données Amazon RDS.
   - Vous activez la prise en charge memcached pour une instance existante en spécifiant le groupe d'options personnalisé lorsque vous modifiez l'instance. Pour de plus amples informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.
6. Spécifiez les colonnes de vos tables MySQL accessibles via l'interface memcached. Le plug-in memcached crée une table de catalogue appelée containers dans une base de données dédiée appelée innodb_memcache. Vous insérez une ligne dans la table containers pour mapper une table InnoDB et y accéder via memcached. Vous spécifiez une colonne dans la table InnoDB qui est utilisée pour stocker les valeurs de clé memcached, et une ou plusieurs colonnes qui sont utilisées pour stocker les valeurs de données associées à la clé. Vous spécifiez également un nom qu'une application memcached utilise pour faire référence à cet ensemble de colonnes. Pour de plus amples informations sur l'insertion de lignes dans la table containers, veuillez consulter Internals of the InnoDB memcached Plugin. Pour obtenir un exemple de mappage d'une table InnoDB et y accéder via memcached, veuillez consulter Writing Applications for the InnoDB memcached Plugin.
7. Si les applications accédant à l'memcachedinterface se trouvent sur des ordinateurs ou des EC2 instances différents de ceux des applications utilisant l'interface SQL, ajoutez les informations de connexion de ces ordinateurs au groupe de sécurité VPC associé à l'instance MySQL. Pour plus d'informations sur la gestion des groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité.

Vous désactivez la prise en charge memcached pour une instance en modifiant l'instance et en spécifiant le groupe d'options par défaut pour votre version MySQL. Pour de plus amples informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Considérations de sécurité memcached MySQL

Le protocole memcached ne prend pas en charge l'authentification utilisateur. Pour plus d'informations sur les considérations de sécurité relatives à MySQL memcached, consultez Security

Considerations for the InnoDB memcached Plugin (Considérations de sécurité relatives au plug-in InnoDB memcached) dans la documentation MySQL.

Vous pouvez prendre les mesures suivantes pour aider à augmenter la sécurité de l'interface `memcached` :

- Spécifiez un port différent du port par défaut 11211 lorsque vous ajoutez l'option `MEMCACHED` au groupe d'options.
- Assurez-vous d'associer l'`memcached`interface à un groupe de sécurité VPC qui limite l'accès aux adresses clients et aux instances connues et EC2 fiables. Pour plus d'informations sur la gestion des groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité.

Informations de connexion memcached MySQL

Pour accéder à l'interface `memcached`, une application doit spécifier le nom DNS de l'instance Amazon RDS et le numéro de port `memcached`. Par exemple, si une instance possède un nom DNS de `my-cache-instance.cg034hpkmmjt.region.rds.amazonaws.com` et l'interface memcached utilise le port 11212, les informations de connexion spécifiées dans PHP seront :

```
<?php

$cache = new Memcache;
$cache->connect('my-cache-instance.cg034hpkmmjt.region.rds.amazonaws.com',11212);
?>
```

Pour trouver le nom DNS et le port memcached d'une instance de base de données MySQL

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit du AWS Management Console, sélectionnez la région qui contient l'instance de base de données.
3. Dans la panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez le nom de l'instance de base de données MySQL pour afficher ses détails.
5. Dans la section Connexion, notez la valeur du champ Point de terminaison. Le nom DNS est le même que le point de terminaison. Veuillez également noter que le port dans la section Connexion n'est pas utilisé pour accéder à l'interface `memcached`.

6. Dans la section Détails, notez le nom répertorié dans le champ Groupe d'options.
7. Dans le panneau de navigation, choisissez Groupes d'options.
8. Choisissez le nom du groupe d'options utilisé par l'instance de base de données MySQL pour afficher les détails du groupe d'options. Dans la section Options, notez la valeur du paramètre Port pour l'option MEMCACHED.

Paramètres d'option memcached MySQL

Amazon RDS expose les paramètres `memcached` MySQL comme paramètres d'option dans l'option Amazon RDS `MEMCACHED`.

Paramètres memcached MySQL

- `DAEMON_MEMCACHED_R_BATCH_SIZE` – Nombre entier qui spécifie combien d'opérations de lecture (get) `memcached` doivent être effectuées avant d'exécuter un COMMIT pour lancer une nouvelle transaction. Les valeurs autorisées sont comprises entre 1 et 4294967295, et celle par défaut est 1. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- `DAEMON_MEMCACHED_W_BATCH_SIZE` – Nombre entier qui spécifie combien d'opérations d'écriture `memcached` comme add, set ou incr doivent être effectuées avant d'exécuter un COMMIT pour lancer une nouvelle transaction. Les valeurs autorisées sont comprises entre 1 et 4294967295, et celle par défaut est 1. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- `INNODB_API_BK_COMMIT_INTERVAL` – Nombre entier qui spécifie la fréquence d'auto-commit des connexions inactives qui utilisent l'interface `memcached` InnoDB. Les valeurs autorisées sont comprises entre 1 et 1073741824, et celle par défaut est 5. L'option prend effet immédiatement, sans que vous ayez besoin de redémarrer l'instance.
- `INNODB_API_DISABLE_ROWLOCK` – Valeur booléenne qui désactive (1 (vrai)) ou active (0 (faux)) l'utilisation des verrouillages de ligne lorsque vous utilisez l'interface `memcached` InnoDB. La valeur par défaut est 0 (faux). L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- `INNODB_API_ENABLE_MDL` – Valeur booléenne qui, lorsqu'elle est configurée sur 0 (faux), verrouille la table utilisée par le plug-in `memcached` InnoDB pour ne pas qu'il puisse être abandonné ou modifié par une instruction DDL via l'interface SQL. La valeur par défaut est 0 (faux). L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- `INNODB_API_TRX_LEVEL` – Nombre entier qui spécifie le niveau d'isolation de la transaction pour les requêtes traitées par l'interface `memcached`. Les valeurs autorisées sont comprises entre 0 et 3. La valeur par défaut est 0. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.

Amazon RDS configure ces paramètres memcached MySQL, ils ne peuvent pas être modifiés : DAEMON_MEMCACHED_LIB_NAME, DAEMON_MEMCACHED_LIB_PATH et INNODB_API_ENABLE_BINLOG. Les paramètres que les administrateurs MySQL configurent en utilisant daemon_memcached_options sont disponibles comme paramètres d'options MEMCACHED individuels dans Amazon RDS.

Paramètres MySQL daemon_memcached_options

- BINDING_PROTOCOL– Chaîne qui spécifie le protocole de liaison à utiliser. Les valeurs autorisées sont auto, ascii ou binary. La valeur par défaut est auto, ce qui signifie que le serveur négocie automatiquement le protocole avec le client. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- BACKLOG_QUEUE_LIMIT – Nombre entier qui spécifie combien de connexions réseau peuvent être en attente de traitement par memcached. L'augmentation de cette limite peut réduire les erreurs reçues par un client qui ne peut pas se connecter à l'instance memcached, mais n'améliore pas les performances du serveur. Les valeurs autorisées sont comprises entre 1 et 2048, et celle par défaut est 1024. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- CAS_DISABLED – Valeur booléenne qui active (1 (vrai)) ou désactive (0 (faux)) l'utilisation de la fonction CAS (Compare and Swap), ce qui réduit la taille par élément de 8 octets. La valeur par défaut est 0 (faux). L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- CHUNK_SIZE – Nombre entier qui spécifie la taille minimum du bloc, en octets, à attribuer à la clé, à la valeur et aux indicateurs de l'élément le plus petit. Les valeurs autorisées sont comprises entre 1 et 48. La valeur par défaut est 48 et vous pouvez considérablement améliorer l'efficacité de la mémoire avec une valeur inférieure. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- CHUNK_SIZE_GROWTH_FACTOR – Nombre flottant qui contrôle la taille des nouveaux blocs. La taille d'un nouveau bloc correspond à la taille du bloc précédent multipliée par CHUNK_SIZE_GROWTH_FACTOR. Les valeurs autorisées sont comprises entre 1 et 2, et celle par défaut est 1.25. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- ERROR_ON_MEMORY_EXHAUSTED – Valeur booléenne qui, lorsqu'elle est configurée sur 1 (vrai), spécifie que memcached renverra une erreur plutôt que d'expulser les éléments lorsqu'il n'y a plus de mémoire pour les stocker. S'il est configuré sur 0 (faux), memcached expulse les éléments s'il n'y a plus de mémoire. La valeur par défaut est 0 (faux). L'option ne prend pas effet tant que l'instance n'est pas redémarrée.
- MAX_SIMULTANEOUS_CONNECTIONS – Nombre entier qui spécifie le nombre maximum de connexions simultanées. La configuration de cette valeur sur n'importe quel chiffre inférieur à 10

empêche MySQL de démarrer. Les valeurs autorisées sont comprises entre 10 et 1024, et celle par défaut est 1024. L'option ne prend pas effet tant que l'instance n'est pas redémarrée.

- VERBOSITY – Chaîne qui spécifie le niveau d'informations consignées dans le journal d'erreurs MySQL par le service memcached. La valeur par défaut est v. L'option ne prend pas effet tant que l'instance n'est pas redémarrée. Les valeurs autorisées sont :
  - v – Journalise les erreurs et avertissements pendant l'exécution de la boucle principale d'évènements.
  - vv – Outre les informations consignées par v, journalise également la commande de chaque client et la réponse.
  - vvv – Outre les informations consignées par vv, journalise également les transitions d'état interne.

Amazon RDS configure ces paramètres MySQL DAEMON_MEMCACHED_OPTIONS, ils ne peuvent pas être modifiés : DAEMON_PROCESS, LARGE_MEMORY_PAGES, MAXIMUM_CORE_FILE_LIMIT, MAX_ITEM_SIZE, LOCK_DOWN_PAGE_MEMORY, MASK, IDFILE, REQUESTS_PER_EVENT, SOCKET et USER.

Paramètres pour My SQL

Par défaut, une instance My SQL DB utilise un groupe de paramètres DB spécifique à My SQL database. Ce groupe de paramètres contient les paramètres du moteur My SQL database. Pour de plus amples informations sur l'utilisation des groupes de paramètres et sur la définition des paramètres, veuillez consulter Groupes de paramètres pour (Amazon RDS).

RDScar Mes SQL paramètres sont définis sur les valeurs par défaut du moteur de stockage que vous avez sélectionné. Pour plus d'informations sur Mes SQL paramètres, consultez la SQLdocumentation Mes. Pour plus d'informations sur Mes moteurs SQL de stockage, consultezMoteurs de stockage pris en charge pour RDS for MySQL.

Vous pouvez consulter les paramètres disponibles pour une SQL version spécifique RDS de My à l'aide de la RDS console ou du AWS CLI. Pour plus d'informations sur l'affichage des paramètres d'un groupe Mes SQL paramètres dans la RDS console, consultezAffichage des valeurs de paramètres pour un groupe de paramètres de base de données dans RDSAmazon.

À l'aide de AWS CLI, vous pouvez afficher les paramètres d'une version RDS pour Ma SQL version en exécutant la `describe-engine-default-parameters`commande. Spécifiez l'une des valeurs suivantes pour l'option `--db-parameter-group-family` :

- `mysql8.4`
- `mysql8.0`
- `mysql5.7`

Par exemple, pour afficher les paramètres de RDS for My SQL version 8.0, exécutez la commande suivante.

```
aws rds describe-engine-default-parameters --db-parameter-group-family mysql8.0
```

Votre résultat ressemble à ce qui suit.

```
{
    "EngineDefaults": {
        "Parameters": [
            {
                "ParameterName": "activate_all_roles_on_login",
                "ParameterValue": "0",
```

```
                "Description": "Automatically set all granted roles as active after the
 user has authenticated successfully.",
                "Source": "engine-default",
                "ApplyType": "dynamic",
                "DataType": "boolean",
                "AllowedValues": "0,1",
                "IsModifiable": true
            },
            {
                "ParameterName": "allow-suspicious-udfs",
                "Description": "Controls whether user-defined functions that have only
 an xxx symbol for the main function can be loaded",
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "boolean",
                "AllowedValues": "0,1",
                "IsModifiable": false
            },
            {
                "ParameterName": "auto_generate_certs",
                "Description": "Controls whether the server autogenerates SSL key and
 certificate files in the data directory, if they do not already exist.",
                "Source": "engine-default",
                "ApplyType": "static",
                "DataType": "boolean",
                "AllowedValues": "0,1",
                "IsModifiable": false
            },
        ...
```

Pour répertorier uniquement les paramètres modifiables RDS pour My SQL version 8.0, exécutez la commande suivante.

Dans Linux, macOS, ou Unix:

```
aws rds describe-engine-default-parameters --db-parameter-group-family mysql8.0 \
   --query 'EngineDefaults.Parameters[?IsModifiable==`true`]'
```

Dans Windows:

```
aws rds describe-engine-default-parameters --db-parameter-group-family mysql8.0 ^
   --query "EngineDefaults.Parameters[?IsModifiable==`true`]"
```

DBATâches courantes pour les instances My SQL DB

Dans le contenu suivant, vous trouverez des descriptions des implémentations RDS spécifiques à Amazon de certaines DBA tâches courantes pour les instances de base de données exécutant le moteur de base de SQL données My database. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui requièrent des privilèges avancés.

Pour plus d'informations sur l'utilisation de Mes fichiers SQL journaux sur AmazonRDS, consultezFichiers journaux de base de données MySQL.

Comprendre les utilisateurs prédéfinis

Amazon crée RDS automatiquement plusieurs utilisateurs prédéfinis avec de nouvelles instances RDS pour My SQL DB. Les utilisateurs prédéfinis et leurs privilèges ne peuvent pas être modifiés. Vous ne pouvez pas supprimer, renommer ou modifier les privilèges de ces utilisateurs prédéfinis. Toute tentative de ce type génère une erreur.

- rdsadmin — Utilisateur créé pour gérer de nombreuses tâches de gestion que l'administrateur doté de `superuser` privilèges exécuterait sur une base de données My database autonome. SQL Cet utilisateur est utilisé en interne par RDS for My SQL pour de nombreuses tâches de gestion.
- rdsrepladmin — Utilisateur utilisé en interne par Amazon pour prendre en charge les activités RDS de réplication sur RDS les instances et clusters My SQL DB.

Pour plus d'informations sur les autres DBA tâches courantes, consultez les rubriques suivantes.

Rubriques

- Modèle de privilèges basé sur les rôles pour for My RDS SQL
- Privilèges dynamiques pour RDS for My SQL
- Fin d'une session ou d'une requête RDS pour My SQL
- Ignorer l'erreur de réplication actuelle pour RDS for My SQL
- Utilisation des tablespaces InnoDB pour améliorer les temps de reprise après incident pour for My RDS SQL
- Gestion de l'historique global du statut de RDS for My SQL
- Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL

Modèle de privilèges basé sur les rôles pour for My RDS SQL

À partir RDS de la SQL version 8.0.36 de My, vous ne pouvez pas modifier directement les tables de la `mysql` base de données. En particulier, vous ne pouvez pas créer d'utilisateurs de base de données en effectuant des opérations du langage de manipulation des données (DML) sur les `grant` tables. Vous utilisez plutôt les instructions de SQL gestion de mon compte telles que `CREATE USER`, `GRANT`, et `REVOKE` pour accorder des privilèges basés sur les rôles aux utilisateurs. Vous ne pouvez pas non plus créer d'autres types d'objets tels que des procédures stockées dans la base de données `mysql`. Vous pouvez toujours interroger les tables `mysql`. Si vous utilisez la réplication binaire des journaux, les modifications apportées directement aux `mysql` tables de l'instance de base de données source ne sont pas répliquées sur le cluster cible.

Dans certains cas, votre application peut utiliser des raccourcis pour créer des utilisateurs ou d'autres objets en les insérant dans les tables `mysql`. Le cas échéant, modifiez le code de votre application pour utiliser les instructions correspondantes telles que `CREATE USER`.

Pour exporter des métadonnées destinées aux utilisateurs de la base de données lors de la migration depuis une SQL base de données My database externe, appliquez l'une des méthodes suivantes :

- Utilisez l'utilitaire de vidage d'instance de My SQL Shell avec un filtre pour exclure les utilisateurs, les rôles et les autorisations. L'exemple suivant montre la syntaxe de commande à utiliser. Assurez-vous qu'il `outputUrl` est vide.

  ```
  mysqlsh user@host -- util.dumpInstance(outputUrl,{excludeSchemas:['mysql'],users:
   true})
  ```

  Pour plus d'informations, consultez les rubriques Utilitaire Instance Dump, Schema Dump Utility et Table Dump Utility dans le My SQL Reference Manual.

- Utilisez l'utilitaire `mysqlpump` client. Cet exemple inclut toutes les tables à l'exception des tables de la base de données `mysql` système. Il inclut également `CREATE USER` des `GRANT` instructions pour reproduire tous les SQL utilisateurs My dans la base de données migrée.

  ```
  mysqlpump --exclude-databases=mysql --users
  ```

  L'utilitaire `mysqlpump` client n'est plus disponible avec My SQL 8.4. Utilisez à la place `mysqldump`.

Pour simplifier la gestion des autorisations pour de nombreux utilisateurs ou applications, vous pouvez utiliser l'instruction `CREATE ROLE` pour créer un rôle doté d'un ensemble d'autorisations. Vous pouvez ensuite utiliser les instructions `GRANT` et `SET ROLE`, et la fonction `current_role` pour attribuer des rôles à des utilisateurs ou des applications, changer le rôle actuel et vérifier les rôles en vigueur. Pour plus d'informations sur le système d'autorisation basé sur les rôles dans My SQL 8.0, consultez la section Utilisation des rôles dans le manuel My SQL Reference.

> **Important**
>
> Nous vous recommandons vivement de ne pas avoir recours au rôle d'utilisateur principal directement dans vos applications. Au lieu de cela, respectez la bonne pratique qui consiste à avoir recours à un utilisateur de base de données doté des privilèges minimum requis pour votre application.

À partir de la version 8.0.36, RDS for My SQL inclut un rôle spécial doté de tous les privilèges suivants. Ce rôle est nommé `rds_superuser_role`. Ce rôle est déjà accordé à l'utilisateur administratif principal de chaque instance de base de données. Le rôle `rds_superuser_role` inclut les privilèges suivants pour tous les objets de base de données :

- `ALTER`
- `APPLICATION_PASSWORD_ADMIN`
- `ALTER ROUTINE`
- `CREATE`
- `CREATE ROLE`
- `CREATE ROUTINE`
- `CREATE TEMPORARY TABLES`
- `CREATE USER`
- `CREATE VIEW`
- `DELETE`
- `DROP`
- `DROP ROLE`
- `EVENT`
- `EXECUTE`

- INDEX
- INSERT
- LOCK TABLES
- PROCESS
- REFERENCES
- RELOAD
- REPLICATION CLIENT
- REPLICATION SLAVE
- ROLE_ADMIN
- SET_USER_ID
- SELECT
- SHOW DATABASES
- SHOW VIEW
- TRIGGER
- UPDATE
- XA_RECOVER_ADMIN

La définition du rôle inclut également `WITH GRANT OPTION` afin qu'un utilisateur administratif puisse accorder ce rôle à d'autres utilisateurs. En particulier, l'administrateur doit accorder tous les privilèges nécessaires pour effectuer la réplication des journaux binaires avec le My SQL cluster comme cible.

> **Tip**
>
> Pour voir tous les détails des autorisations, utilisez l'instruction suivante.
>
> ```
> SHOW GRANTS FOR rds_superuser_role@'%';
> ```

Lorsque vous accordez l'accès à l'aide RDS de rôles dans My SQL version 8.0.36 ou ultérieure, vous activez également le rôle à l'aide de l'instruction `SET ROLE` *`role_name`* or`SET ROLE ALL`. L'exemple suivant montre comment procéder. Remplacez le nom de rôle approprié par `CUSTOM_ROLE`.

```
# Grant role to user
mysql> GRANT CUSTOM_ROLE TO 'user'@'domain-or-ip-address'

# Check the current roles for your user. In this case, the CUSTOM_ROLE role has not
 been activated.
# Only the rds_superuser_role is currently in effect.
mysql> SELECT CURRENT_ROLE();
+--------------------------+
| CURRENT_ROLE()           |
+--------------------------+
| `rds_superuser_role`@`%` |
+--------------------------+
1 row in set (0.00 sec)

# Activate all roles associated with this user using SET ROLE.
# You can activate specific roles or all roles.
# In this case, the user only has 2 roles, so we specify ALL.
mysql> SET ROLE ALL;
Query OK, 0 rows affected (0.00 sec)

# Verify role is now active
mysql> SELECT CURRENT_ROLE();
+-------------------------------------------------+
| CURRENT_ROLE()                                  |
+-------------------------------------------------+
| `CUSTOM_ROLE`@`%`,`rds_superuser_role`@`%` |
+-------------------------------------------------+
```

Privilèges dynamiques pour RDS for My SQL

Les privilèges dynamiques sont Mes SQL privilèges que vous pouvez explicitement accorder à l'aide de l'GRANTinstruction. Selon votre version de RDS for MySQL, RDS vous pouvez uniquement accorder des privilèges dynamiques spécifiques. RDSrefuse certains de ces privilèges car ils peuvent interférer avec des opérations de base de données spécifiques, telles que la réplication et la sauvegarde.

Le tableau suivant indique quels privilèges vous pouvez accorder pour les différentes SQL versions de My. Si vous effectuez une mise à niveau d'une SQL version My inférieure à 8.0.36 vers une version 8.0.36 ou supérieure, vous devrez peut-être mettre à jour le code de votre application si l'octroi d'un privilège particulier n'est plus autorisé.

Privilège	My SQL 8.0.35 et versions antérieures	Mes SQL versions mineures 8.0.36 et supérieures	My SQL 8.4.3 et versions ultérieures
ALLOW_NON EXISTENT_DEFINER	Non disponible	Non disponible	Interdit
APPLICATI ON_PASSWO RD_ADMIN	Autorisé	Autorisé	Autorisé
AUDIT_ABO RT_EXEMPT	Autorisé	Interdit	Interdit
AUDIT_ADMIN	Interdit	Interdit	Interdit
AUTHENTIC ATION_POL ICY_ADMIN	Autorisé	Interdit	Interdit
BACKUP_ADMIN	Autorisé	Interdit	Interdit
BINLOG_ADMIN	Autorisé	Interdit	Interdit
BINLOG_EN CRYPTION_ADMIN	Interdit	Interdit	Interdit
CLONE_ADMIN	Interdit	Interdit	Interdit
CONNECTIO N_ADMIN	Autorisé	Interdit	Interdit
ENCRYPTIO N_KEY_ADMIN	Interdit	Interdit	Interdit
FIREWALL_ADMIN	Interdit	Interdit	Interdit
FIREWALL_EXEMPT	Autorisé	Interdit	Interdit
FIREWALL_USER	Interdit	Interdit	Interdit

Privilège	My SQL 8.0.35 et versions antérieures	Mes SQL versions mineures 8.0.36 et supérieures	My SQL 8.4.3 et versions ultérieures
FLUSH_OPTIMIZER_COSTS	Autorisé	Autorisé	Autorisé
FLUSH_PRIVILEGES	Non disponible	Non disponible	Autorisé
FLUSH_STATUS	Autorisé	Autorisé	Autorisé
FLUSH_TABLES	Autorisé	Autorisé	Autorisé
FLUSH_USER_RESOURCES	Autorisé	Autorisé	Autorisé
GROUP_REPLICATION_ADMIN	Interdit	Interdit	Interdit
GROUP_REPLICATION_STREAM	Interdit	Interdit	Interdit
INNODB_REDO_LOG_ARCHIVE	Interdit	Interdit	Interdit
INNODB_REDO_LOG_ENABLE	Interdit	Interdit	Interdit
MASKING_DICTIONARIES_ADMIN	Interdit	Interdit	Interdit
NDB_STORED_USER	Interdit	Interdit	Interdit
OPTIMIZE_LOCAL_TABLE	Non disponible	Non disponible	Interdit
PASSWORDLESS_USER_ADMIN	Interdit	Interdit	Interdit

Privilège	My SQL 8.0.35 et versions antérieures	Mes SQL versions mineures 8.0.36 et supérieures	My SQL 8.4.3 et versions ultérieures
PERSIST_RO_ _ VARIABLES ADMIN	Interdit	Interdit	Interdit
REPLICATI ON_APPLIER	Autorisé	Interdit	Interdit
REPLICATI ON_SLAVE_ADMIN	Interdit	Interdit	Interdit
RESOURCE_ GROUP_ADMIN	Autorisé	Interdit	Interdit
RESOURCE_ GROUP_USER	Autorisé	Interdit	Interdit
ROLE_ADMIN	Autorisé	Autorisé	Autorisé
SENSITIVE _VARIABLE S_OBSERVER	Autorisé	Autorisé	Autorisé
SERVICE_C ONNECTION_ADMIN	Autorisé	Interdit	Interdit
SESSION_V ARIABLES_ADMIN	Autorisé	Autorisé	Autorisé
SET_ANY_DEFINER	Non disponible	Non disponible	Autorisé
SET_ USER _IDENTIFIANT	Autorisé	Autorisé	Non disponible
SHOW_ROUTINE	Autorisé	Autorisé	Autorisé
SKIP_QUER Y_REWRITE	Interdit	Interdit	Interdit

Privilège	My SQL 8.0.35 et versions antérieures	Mes SQL versions mineures 8.0.36 et supérieures	My SQL 8.4.3 et versions ultérieures
SYSTEM_USER	Interdit	Interdit	Interdit
SYSTEM_VARIABLES_ADMIN	Interdit	Interdit	Interdit
TABLE_ENCRYPTION_ADMIN	Interdit	Interdit	Interdit
TELEMETRY_LOG_ADMIN	Autorisé	Interdit	Interdit
TP _ _ CONNECTION ADMIN	Interdit	Interdit	Interdit
TRANSACTION_GTID_TAG	Non disponible	Non disponible	Interdit
VERSION_TOKEN_ADMIN	Interdit	Interdit	Interdit
XA__ RECOVER ADMIN	Autorisé	Autorisé	Autorisé

Fin d'une session ou d'une requête RDS pour My SQL

Vous pouvez mettre fin aux sessions d'utilisateur ou requêtes sur les instances de base de données à l'aide des commandes `rds_kill` et `rds_kill_query`. Connectez-vous d'abord à votre instance My SQL DB, puis émettez la commande appropriée comme indiqué ci-dessous. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données MySQL.

```
CALL mysql.rds_kill(thread-ID)
CALL mysql.rds_kill_query(thread-ID)
```

Par exemple, pour arrêter la session qui s'exécute sur le thread 99, entrez la commande suivante :

```
CALL mysql.rds_kill(99);
```

Pour arrêter la requête qui s'exécute sur le thread 99, entrez la commande suivante :

```
CALL mysql.rds_kill_query(99);
```

Ignorer l'erreur de réplication actuelle pour RDS for My SQL

Amazon RDS fournit un mécanisme qui vous permet d'ignorer une erreur sur vos réplicas en lecture, si l'erreur entraîne une absence de réponse du réplica en lecture et qu'elle n'affecte pas l'intégrité de vos données.

> **Note**
>
> D'abord, vérifiez que l'erreur concernée peut être ignorée en toute sécurité. Dans un SQL utilitaire My, connectez-vous à la réplique lue et exécutez la SQL commande My suivante.
>
> ```
> SHOW REPLICA STATUS\G
> ```
>
> Pour plus d'informations sur les valeurs renvoyées, consultez la section Ma SQL documentation.
> Les versions précédentes de et My SQL utilisaient à la `SHOW SLAVE STATUS` place de`SHOW REPLICA STATUS`. Si vous utilisez une SQL version My antérieure à la version 8.0.23, utilisez. `SHOW SLAVE STATUS`

Vous pouvez ignorer une erreur sur votre réplica en lecture de la manière suivante.

Rubriques

- Appel de la procédure mysql.rds_skip_repl_error
- Définition du paramètre slave_skip_errors

Appel de la procédure mysql.rds_skip_repl_error

Amazon RDS fournit une procédure stockée que vous pouvez appeler pour éviter une erreur sur vos répliques de lecture. Connectez-vous d'abord à votre réplica en lecture, puis émettez les commandes

appropriées comme illustré ci-après. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données MySQL.

Pour ignorer l'erreur, émettez la commande suivante.

```
CALL mysql.rds_skip_repl_error;
```

Cette commande n'a aucun effet si vous l'exécutez sur l'instance de base de données source ou sur un réplica en lecture qui n'a rencontré aucune erreur de réplication.

Pour plus d'informations, telles que les versions de My SQL compatibles`mysql.rds_skip_repl_error`, consultezmysql.rds_skip_repl_error.

> **Important**
>
> Si vous tentez d'appeler `mysql.rds_skip_repl_error` et que vous rencontrez l'erreur suivante :`ERROR 1305 (42000): PROCEDURE mysql.rds_skip_repl_error does not exist`, mettez à niveau votre instance My SQL DB vers la dernière version mineure ou vers l'une des versions mineures minimales répertoriées dansmysql.rds_skip_repl_error.

Définition du paramètre slave_skip_errors

Pour ignorer une ou plusieurs erreurs, vous pouvez définir le paramètre statique `slave_skip_errors` sur le réplica en lecture. Vous pouvez définir ce paramètre pour ignorer un ou plusieurs codes d'erreur de réplication spécifiques. Actuellement, vous ne pouvez définir ce paramètre que RDS pour les instances de base de données My SQL 5.7. Après avoir modifié ce paramètre, veillez à redémarrer votre instance de base de données pour que le nouveau paramètre prenne effet. Pour plus d'informations sur la définition de ce paramètre, consultez la section Ma SQL documentation.

Nous vous recommandons de définir ce paramètre dans un groupe de paramètres de base de données distinct. Vous pouvez associer ce groupe de paramètres de base de données aux réplicas en lecture qui doivent ignorer les erreurs. Le suivi de cette bonne pratique réduit l'impact potentiel sur d'autres instances de base de données et réplicas en lecture.

> **Important**
>
> La définition d'une valeur autre que par défaut pour ce paramètre peut entraîner une incohérence de la réplication. Ne définissez ce paramètre sur une valeur autre que par défaut

que si vous avez épuisé d'autres options pour résoudre le problème et que vous êtes sûr de l'impact potentiel sur les données de votre réplica en lecture.

Utilisation des tablespaces InnoDB pour améliorer les temps de reprise après incident pour for My RDS SQL

Chaque table de My SQL comprend une définition de table, des données et des index. Le moteur My SQL storage InnoDB stocke les données des tables et les index dans un tablespace. InnoDB crée un espace de table global partagé qui contient un dictionnaire de données et autres métadonnées pertinentes, et peut contenir des données de table et des index. InnoDB peut aussi créer des espaces de table distincts pour chaque table et partition. Ces espaces de table distincts sont stockés dans des fichiers ayant .ibd comme extension et l'en-tête de chaque espace de table contient un numéro qui l'identifie de façon unique.

Amazon RDS fournit un paramètre dans un groupe My SQL parameter appelé`innodb_file_per_table`. Ce paramètre contrôle le fait qu'InnoDB ajoute ou non de nouvelles données et de nouveaux index de tables au tablespace partagé (en définissant la valeur du paramètre du 0) ou à des tablespaces individuels (en définissant la valeur du paramètre sur 1). Amazon RDS définit la valeur par défaut du `innodb_file_per_table` paramètre sur 1, ce qui vous permet de supprimer des tables InnoDB individuelles et de récupérer le stockage utilisé par ces tables pour l'instance de base de données. Dans la plupart des cas d'utilisation, la définition du paramètre `innodb_file_per_table` à la valeur 1 est celle recommandée.

Vous devez définir le `innodb_file_per_table` paramètre sur 0 lorsque vous avez un grand nombre de tables, par exemple plus de 1 000 tables lorsque vous utilisez un SSD stockage standard (magnétique) ou à usage général ou plus de 10 000 tables lorsque vous utilisez le IOPS stockage provisionné. Lorsque vous définissez ce paramètre à la valeur 0, les espaces de table individuels ne sont pas créés et cela peut améliorer le temps nécessaire pour la récupération sur incident de base de données.

My SQL traite chaque fichier de métadonnées, y compris les tablespaces, pendant le cycle de reprise après incident. Le temps nécessaire SQL à My pour traiter les informations de métadonnées dans l'espace disque logique partagé est négligeable par rapport au temps nécessaire au traitement de milliers de fichiers d'espace disque logique lorsqu'il existe plusieurs espaces disque logiques. Comme le nombre d'espaces de table est stocké au sein de l'en-tête de chaque fichier, le temps total nécessaire pour lire tous les fichiers d'espace de table peut prendre jusqu'à plusieurs heures. Par exemple, un million d'espaces de table InnoDB sur un stockage standard peut nécessiter entre cinq

et huit heures de traitement pendant un cycle de récupération sur incident. Dans certains, InnoDB peut déterminer qu'il a besoin d'un nettoyage supplémentaire après un cycle de récupération sur incident et, par conséquent, entamera un autre cycle de récupération sur incident, ce qui augmente le temps total de récupération. Gardez à l'esprit qu'un cycle de récupération sur incident implique aussi la restauration de transactions, la correction des pages rompues et autres opérations en plus du traitement des informations sur les espaces de table.

Comme le paramètre `innodb_file_per_table` réside dans un groupe de paramètres, vous pouvez modifier la valeur du paramètre en modifiant le groupe de paramètres utilisé par votre instance de base de données sans avoir à redémarrer celle-ci. Une fois que la valeur est modifiée, de la valeur 1 (créer des tables individuelles) à la valeur 0 (utiliser un espace de table partagé), par exemple, les nouvelles tables InnoDB sont ajoutées à l'espace de table partagé, pendant que les tables existantes continuent d'avoir des espaces de table individuels. Pour déplacer une table InnoDB vers l'espace de table partagé, vous devez utiliser la commande `ALTER TABLE`.

Migration de plusieurs espaces de table vers l'espace de table partagé

Vous pouvez déplacer les métadonnées d'une table InnoDB de son propre espace de table vers l'espace de table partagé, ce qui recrée les métadonnées de la table selon la valeur du paramètre `innodb_file_per_table`. Connectez-vous d'abord à votre instance My SQL DB, puis émettez les commandes appropriées comme indiqué ci-dessous. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données MySQL.

```
ALTER TABLE table_name ENGINE = InnoDB, ALGORITHM=COPY;
```

Par exemple, la requête suivante retourne une instruction `ALTER TABLE` pour chaque table InnoDB qui ne figure pas dans l'espace de table partagé.

Pour les instances de base de données My SQL 5.7 :

```
SELECT CONCAT('ALTER TABLE `',
REPLACE(LEFT(NAME , INSTR((NAME), '/') - 1), '`', '``'), '`.`',
REPLACE(SUBSTR(NAME FROM INSTR(NAME, '/') + 1), '`', '``'), '` ENGINE=InnoDB,
 ALGORITHM=COPY;') AS Query
FROM INFORMATION_SCHEMA.INNODB_SYS_TABLES
WHERE SPACE <> 0 AND LEFT(NAME, INSTR((NAME), '/') - 1) NOT IN ('mysql','');
```

Pour les instances de base de données My SQL 8.4 et 8.0 :

```
SELECT CONCAT('ALTER TABLE `',
```

```
REPLACE(LEFT(NAME , INSTR((NAME), '/') - 1), '`', '``'), '`.`',
REPLACE(SUBSTR(NAME FROM INSTR(NAME, '/') + 1), '`', '``'), '` ENGINE=InnoDB,
 ALGORITHM=COPY;') AS Query
FROM INFORMATION_SCHEMA.INNODB_TABLES
WHERE SPACE <> 0 AND LEFT(NAME, INSTR((NAME), '/') - 1) NOT IN ('mysql','');
```

La reconstruction d'une SQL table My pour déplacer les métadonnées de la table vers le tablespace partagé nécessite temporairement un espace de stockage supplémentaire pour reconstruire la table. L'instance de base de données doit donc disposer d'un espace de stockage disponible. Pendant la reconstruction, la table est verrouillée et inaccessible aux requêtes. Pour les petites tables ou les tables qui ne sont pas fréquemment consultées, ce n'est pas nécessairement un problème. Pour les tables volumineuses ou fréquemment consultées dans un environnement fortement concurrentiel, vous pouvez reconstruire les tables sur un réplica en lecture.

Vous pouvez créer un réplica en lecture et migrer les métadonnées de la table vers l'espace de table partagé du réplica en lecture. Bien que l'ALTERTABLEinstruction bloque l'accès à la réplique en lecture, l'instance de base de données source n'est pas affectée. L'instance de base de données source continue à générer ses journaux binaires, tandis que le réplica en lecture ralentit pendant le processus de reconstruction de la table. Étant donné que la reconstruction exige un espace de stockage supplémentaire et que le fichier journal de relecture peut devenir volumineux, vous devriez créer un réplica en lecture dont la capacité de stockage allouée est supérieure à l'instance de base de données source.

Pour créer un réplica en lecture et reconstruire les tables InnoDB afin d'utiliser l'espace de table partagé, procédez comme suit :

1. Assurez-vous que la rétention des sauvegardes est activée sur l'instance de base de données source de sorte que la journalisation binaire soit activée.
2. Utilisez le AWS Management Console ou AWS CLI pour créer une réplique en lecture pour l'instance de base de données source. Étant donné que la création d'un réplica en lecture implique un grand nombre de processus semblables à ceux de la récupération sur incident, le processus de création peut prendre un certain temps si le nombre d'espaces de table InnoDB est élevé. Allouez plus d'espace de stockage sur le réplica en lecture qu'il n'en est actuellement utilisé sur l'instance de base de données source.
3. Lorsque le réplica en lecture a été créé, créez un groupe de paramètres avec les valeurs de paramètre `read_only = 0` et `innodb_file_per_table = 0`. Associez ensuite le groupe de paramètres au réplica en lecture.
4. Émettez l'SQLinstruction suivante pour toutes les tables que vous souhaitez migrer sur le réplica :

```
ALTER TABLE name ENGINE = InnoDB
```

5. Une fois que toutes vos instructions `ALTER TABLE` sont terminées sur le réplica en lecture, vérifiez que celui-ci est connecté à l'instance de base de données source et que les deux instances sont synchronisées.
6. Utilisez la console ou CLI pour promouvoir la réplique lue en tant qu'instance. Assurez-vous que le groupe de paramètres utilisé pour la nouvelle instance de base de données autonome a le paramètre `innodb_file_per_table` défini sur 0. Modifiez le nom de la nouvelle instance de base de données autonome et pointez toutes les applications vers la nouvelle instance de base de données autonome.

Gestion de l'historique global du statut de RDS for My SQL

> **Tip**
>
> Pour analyser les performances des bases de données, vous pouvez également utiliser Performance Insights sur AmazonRDS. Pour de plus amples informations, veuillez consulter Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

My SQL gère de nombreuses variables d'état qui fournissent des informations sur son fonctionnement. Leur valeur peut vous aider à détecter les problèmes de verrouillage ou de mémoire d'une instance de base de données. Les valeurs de ces variables d'état se cumulent depuis le dernier démarrage de l'instance de base de données. Vous pouvez réinitialiser à la valeur 0 la plupart des variables d'état à l'aide de la commande `FLUSH STATUS`.

Pour permettre le suivi de ces valeurs au fil du temps, Amazon RDS fournit un ensemble de procédures qui permettent de capturer les valeurs de ces variables d'état au fil du temps et de les écrire dans une table, ainsi que les modifications apportées depuis le dernier instantané. Cette infrastructure, appelée Global Status History (GoSH), est installée sur toutes les instances My SQL DB à partir des versions 5.5.23. GoSH est désactivé par défaut.

Pour activer GoSH, vous devez d'abord activer le planificateur d'événement à partir d'un groupe de paramètres de base de données en définissant le paramètre `event_scheduler` sur `ON`. Pour les instances My SQL DB exécutant My SQL 5.7, définissez également le paramètre

`show_compatibility_56` sur1. Pour plus d'informations sur la création et la modification d'un groupe de paramètres DB, consultez Groupes de paramètres pour (Amazon RDS). Pour plus d'informations sur les effets secondaires liés à l'activation de ce paramètre, consultez show_compatibility_56 dans le manuel de référence My 5.7. SQL

Vous pouvez ensuite utiliser les procédures du tableau suivant pour activer et configurer GoSH. Connectez-vous d'abord à votre instance My SQL DB, puis émettez les commandes appropriées comme indiqué ci-dessous. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données MySQL. Pour chaque procédure, exécutez la commande suivante et remplacez *`procedure-name`*:

```
CALL procedure-name;
```

Le tableau suivant répertorie toutes les procédures que vous pouvez utiliser *`procedure-name`*dans la commande précédente.

Procédure	Description
`mysql.rds_enable_gsh_collector`	Active l'infrastructure GoSH pour prendre des instantanés par défaut à intervalles spécifiés par `rds_set_gsh_collector` .
`mysql.rds_set_gsh_collector`	Spécifie l'intervalle, en minutes, entre les instantanés. La valeur par défaut est 5.
`mysql.rds_disable_gsh_collector`	Désactive les instantanés.
`mysql.rds_collect_global_status_history`	Prend un instantané sur demande.
`mysql.rds_enable_gsh_rotation`	Active la rotation du contenu de la table `mysql.rds_global_status_history` en `mysql.rds_global_status_history_old` à intervalles spécifiés par `rds_set_gsh_rotation` .
`mysql.rds_set_gsh_rotation`	Spécifie l'intervalle, en jours, entre deux rotations de table. La valeur par défaut est 7.

Procédure	Description
`mysql.rds_disable_gsh_rotation`	Désactive la rotation de table.
`mysql.rds_rotate_global_status_history`	Effectue une rotation du contenu de la table `mysql.rds_global_status_history` en `mysql.rds_global_status_history_old` à la demande.

Lorsque l'infrastructure GoSH est en cours d'exécution, vous pouvez interroger les tables sur lesquelles elle écrit. Par exemple, pour interroger le taux d'accès du groupe de tampons Innodb, vous devez émettre la requête suivante :

```
select a.collection_end, a.collection_start, (( a.variable_Delta-b.variable_delta)/
a.variable_delta)*100 as "HitRatio"
    from mysql.rds_global_status_history as a join mysql.rds_global_status_history as b
 on a.collection_end = b.collection_end
    where a. variable_name = 'Innodb_buffer_pool_read_requests' and b.variable_name =
 'Innodb_buffer_pool_reads'
```

Configuration de la taille du pool de mémoire tampon et de la capacité du redo log dans My 8.4 SQL

Dans My SQL 8.4, Amazon RDS active le `innodb_dedicated_server` paramètre par défaut. Avec le `innodb_dedicated_server` paramètre, le moteur de base de données calcule les `innodb_redo_log_capacity` paramètres `innodb_buffer_pool_size` et. Pour plus d'informations sur le mode de calcul de ces paramètres, consultez Configuration de la taille du pool de mémoire tampon InnoDB et Redo Log dans la documentation Ma. SQL

Lorsque cette option est `innodb_dedicated_server` activée, le `innodb_buffer_pool_size` paramètre est calculé en fonction de la mémoire de classe d'instance de base de données. Le tableau suivant indique la mémoire du serveur détectée et la taille du pool de mémoire tampon correspondant.

Mémoire du serveur détectée	Taille du pool tampon
< 1 Go	Valeur par défaut de 128 Mo

Mémoire du serveur détectée	Taille du pool tampon
1 Go à 4 Go	*Detected server memory** 0,5
> 4 GO	*Detected server memory** 0,75

Le `innodb_redo_log_capacity` paramètre évolue automatiquement avec la classe d'instance jusqu'à (nombre de vCPUs /2) Go jusqu'à un maximum de 16 Go. Les classes d'instances plus importantes ont une capacité de journalisation plus importante, ce qui peut améliorer les performances et la résilience pour les charges de travail intensives en écriture.

Avant de passer de My SQL 8.0 à My SQL 8.4, assurez-vous d'augmenter votre espace de stockage pour faire face à une éventuelle augmentation de la taille des redo logs qui pourrait survenir une fois la mise à niveau terminée. Pour de plus amples informations, veuillez consulter Augmentation de la capacité de stockage d'une instance de base de données.

Si vous ne souhaitez pas que le `innodb_dedicated_server` paramètre calcule les valeurs des `innodb_redo_log_capacity` paramètres `innodb_buffer_pool_size` et, vous pouvez remplacer ces valeurs en leur attribuant des valeurs spécifiques dans un groupe de paramètres personnalisé. Vous pouvez également désactiver le `innodb_dedicated_server` paramètre et définir des valeurs pour les `innodb_redo_log_capacity` paramètres `innodb_buffer_pool_size` et dans un groupe de paramètres personnalisé. Pour de plus amples informations, veuillez consulter Groupes de paramètres par défaut et personnalisés.

Si vous désactivez le `innodb_dedicated_server` paramètre en le définissant sur `0` et ne définissez pas de valeurs pour les `innodb_redo_log_capacity` paramètres `innodb_buffer_pool_size` et, Amazon RDS définit les deux derniers paramètres à 128 Mo et 100 Mo, respectivement. Ces valeurs par défaut se traduisent par des performances médiocres sur des classes d'instance plus importantes.

Fuseau horaire local pour les instances My SQL DB

Par défaut, le fuseau horaire d'une instance My SQL DB est Universal Time Coordinated (UTC). Vous pouvez à la place définir le fuseau horaire de votre instance de base de données sur le fuseau horaire local de votre application.

Pour définir le fuseau horaire local d'une instance de base de données, définissez le paramètre `time_zone` du groupe de paramètres de votre instance de base de données avec l'une des valeurs prises en charge et répertoriées plus bas dans cette section. Lorsque vous définissez le paramètre `time_zone` d'un groupe de paramètres, toutes les instances de base de données et tous les réplicas en lecture qui ont recours à ce groupe de paramètres sont modifiés de façon à utiliser le nouveau fuseau horaire local. Pour plus d'informations sur la définition des paramètres d'un groupe de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Une fois que vous avez défini le fuseau horaire local, toutes les nouvelles connexions à la base de données reflètent la modification. Si des connexions à votre base de données sont ouvertes lorsque vous modifiez le fuseau horaire local, la mise à jour du fuseau horaire local n'apparaît pas tant que vous n'avez pas fermé la connexion et n'en avez pas ouvert une nouvelle.

Vous pouvez définir un fuseau horaire local différent pour une instance de base de données et un ou plusieurs de ses réplicas en lecture. Pour ce faire, utilisez un autre groupe de paramètres pour l'instance de base de données et les réplicas, et définissez le paramètre `time_zone` de chaque groupe de paramètres avec un autre fuseau horaire local.

Si vous effectuez une réplication Régions AWS, l'instance de base de données source et la réplique en lecture utilisent des groupes de paramètres différents (les groupes de paramètres sont uniques à un Région AWS). Pour que chaque instance utilise le même fuseau horaire local, vous devez définir le paramètre `time_zone` dans les groupes de paramètres de l'instance et du réplica en lecture.

Lorsque vous restaurez une instance de base de données à partir d'un instantané de base de données, le fuseau horaire local est défini surUTC. Vous pouvez mettre à jour le fuseau horaire sur votre fuseau horaire local une fois la restauration terminée. Si vous restaurez une instance de base de données à un instant dans le passé, le fuseau horaire local de l'instance de base de données restaurée est le paramètre de fuseau horaire du groupe de paramètres de l'instance de base de données restaurée.

L'Internet Assigned Numbers Authority (IANA) publie de nouveaux fuseaux horaires sur https://www.iana.org/time-zones plusieurs fois par an. Chaque fois RDS qu'une nouvelle version de maintenance mineure de My est SQL publiée, celle-ci est livrée avec les dernières données de

fuseau horaire au moment de la publication. Lorsque vous utilisez la dernière version RDS pour Mes SQL versions, vous disposez de données de fuseau horaire récentes provenant deRDS. Pour vous assurer que votre instance de base de données dispose de données de fuseau horaire récentes, nous vous recommandons de passer à une version supérieure du moteur de base de données. Vous pouvez également modifier manuellement les tables de fuseaux horaires dans les instances de base de données MariaDB. Pour ce faire, vous pouvez utiliser des SQL commandes ou exécuter l'outil mysql_tzinfo_to_sql dans un client. SQL Après avoir mis à jour les données de fuseau horaire manuellement, redémarrez votre instance de base de données afin que les modifications prennent effet. RDSne modifie ni ne réinitialise les données de fuseau horaire des instances de base de données en cours d'exécution. Les nouvelles données de fuseau horaire ne sont installées que lorsque vous effectuez une mise à niveau de la version du moteur de base de données.

Vous pouvez définir votre fuseau horaire local avec l'une des valeurs suivantes.

disponibilité	Fuseau horaire
Afrique	Africa/Cairo, Africa/Casablanca, Africa/Harare, Africa/Monrovia, Africa/Nairobi, Africa/Tripoli, Africa/Windhoek
Amérique	America/Araguaina, America/Asuncion, America/Bogota, America/Buenos_Aires, America/Caracas, America/Chihuahua, America/Cuiaba, America/Denver, America/Fortaleza, America/Guatemala, America/Halifax, America/Manaus, America/Matamoros, America/Monterrey, America/Montevideo, America/Phoenix, America/Santiago, America/Tijuana
Asie	Asia/Amman, Asia/Ashgabat, Asia/Baghdad, Asia/Baku, Asia/Bangkok, Asia/Beirut, Asia/Calcutta, Asia/Damascus, Asia/Dhaka, Asia/Irkutsk, Asia/Jerusalem, Asia/Kabul, Asia/Karachi, Asia/Kathmandu, Asia/Krasnoyarsk, Asia/Magadan, Asia/Muscat, Asia/Novosibirsk, Asia/Riyadh, Asia/Seoul, Asia/Shanghai, Asia/Singapore, Asia/Taipei, Asia/Tehran, Asia/Tokyo, Asia/Ulaanbaatar, Asia/Vladivostok, Asia/Yakutsk, Asia/Yerevan
Atlantique	Atlantique/Açores
Australie	Australia/Adelaide, Australia/Brisbane, Australia/Darwin, Australia/Hobart, Australia/Perth, Australia/Sydney
Brésil	Brazil/DeNoronha, Brazil/East

disponibilité	Fuseau horaire
Canada	Canada/Newfoundland, Canada/Saskatchewan, Canda/Yukon
Europe	Europe/Amsterdam, Europe/Athens, Europe/Dublin, Europe/Helsinki, Europe/Istanbul, Europe/Kaliningrad Europe/Moscow, Europe/Paris, Europe/Prague, Europe/Sarajevo
Pacifique	Pacific/Auckland, Pacific/Fiji, Pacific/Guam, Pacific/Honolulu, Pacific/Samoa
ETATS-UNIS	US/Alaska, US/Central, US/East-Indiana, US/Eastern, US/Pacific
UTC	UTC

Limites et problèmes connus pour Amazon RDS for MySQL

Les limites et les problèmes connus liés à l'utilisation de Amazon RDS for MySQL sont répertoriés ci-dessous.

Rubriques

- Mot réservé InnoDB
- Comportement de stockage plein pour Amazon RDS for MySQL
- Taille de pool de mémoires tampons InnoDB incohérente
- L'optimisation de la fusion d'index renvoie des résultats incorrects
- Exceptions des paramètres MySQL pour les instances de base de données Amazon RDS
- Limites de taille des fichiers MySQL dans Amazon RDS
- Plug-in MySQL Keyring non pris en charge
- Ports personnalisés
- Limitations des procédures stockées MySQL
- Réplication basée sur GTID avec une instance source externe
- Plugin d'authentification par défaut MySQL
- Remplacer innodb_buffer_pool_size
- Mise à niveau de MySQL 5.7 vers MySQL 8.4
- Compression de page InnoDB

Mot réservé InnoDB

`InnoDB` est un mot réservé pour RDS for MySQL. Vous ne pouvez pas utiliser ce nom pour une base de données MySQL.

Comportement de stockage plein pour Amazon RDS for MySQL

Lorsque le stockage devient plein pour une instance de base de données MySQL, cela peut entraîner des incohérences de métadonnées, des incohérences de dictionnaire et des tables orphelines. Pour éviter ces problèmes, Amazon RDS arrête automatiquement une instance de base de données qui atteint l'état `storage-full`.

Une instance de base de données MySQL atteint l'état `storage-full` dans les cas suivants :

- L'instance de base de données possède moins de 20 000 Mio de stockage et le stockage disponible atteint 200 Mio ou moins.
- L'instance de base de données possède plus de 102 400 Mio de stockage et le stockage disponible atteint 1024 Mio ou moins.
- L'instance de base de données possède entre 20 000 Mio et 102 400 Mio de stockage et dispose de moins de 1 % du stockage disponible.

Après l'arrêt automatique par Amazon RDS d'une instance de base de données car elle a atteint l'état `storage-full`, vous pouvez toujours la modifier. Pour redémarrer l'instance de base de données, effectuez au moins l'une des opérations suivantes :

- Modifiez l'instance de base de données pour activer le dimensionnement automatique du stockage.

  Pour plus d'informations sur le dimensionnement automatique du stockage, consultez Gestion automatique de la capacité avec le dimensionnement automatique du stockage Amazon RDS.
- Modifiez l'instance de base de données pour augmenter sa capacité de stockage.

  Pour plus d'informations sur l'augmentation de la capacité de stockage, consultez Augmentation de la capacité de stockage d'une instance de base de données.

Après avoir effectué l'une de ces modifications, l'instance de base de données est automatiquement redémarrée. Pour plus d'informations sur la modification d'une instance de base de données , consultez Modification d'une RDS instance de base de données Amazon.

Taille de pool de mémoires tampons InnoDB incohérente

Pour MySQL 5.7, il y a actuellement un bogue dans la manière dont la taille du pool de mémoires tampons InnoDB est gérée. MySQL 5.7 peut ajuster la valeur du paramètre `innodb_buffer_pool_size` sur une valeur importante qui peut entraîner un développement trop important du pool de mémoires tampons InnoDB et l'utilisation d'un trop gros volume de mémoire. Cet effet peut entraîner l'arrêt de l'exécution du moteur de base de données MySQL ou empêcher son démarrage. Ce problème est plus courant pour des classes d'instance de base de données dont l'espace mémoire disponible est moindre.

Pour résoudre ce problème, définissez la valeur du paramètre `innodb_buffer_pool_size` sur un multiple du produit des valeurs de paramètre `innodb_buffer_pool_instances` et `innodb_buffer_pool_chunk_size`. Par exemple, vous pouvez définir la valeur de

paramètre `innodb_buffer_pool_size` sur un multiple de huit fois le produit des valeurs de paramètre `innodb_buffer_pool_instances` et `innodb_buffer_pool_chunk_size`, comme illustré dans l'exemple suivant.

```
innodb_buffer_pool_chunk_size = 536870912
innodb_buffer_pool_instances = 4
innodb_buffer_pool_size = (536870912 * 4) * 8 = 17179869184
```

Pour plus de détails sur ce bogue MySQL 5.7, consultez le https://bugs.mysql.comfichier /bug.php ?id=79379 dans la documentation MySQL.

L'optimisation de la fusion d'index renvoie des résultats incorrects

Les requêtes qui utilisent l'optimisation par fusion d'index peuvent renvoyer des résultats incorrects en raison d'un bogue dans l'optimiseur de requêtes MySQL introduit dans MySQL 5.5.37. Lorsque vous exécutez une requête sur une table avec plusieurs index, l'optimiseur analyse les plages de lignes selon les multiples index, mais il ne fusionne pas correctement les résultats. Pour plus d'informations sur le bogue de l'optimiseur de requête, consultez http://bugs.mysql.com/bug.php?id=72745 et http://bugs.mysql.com/bug.php?id=68194 dans la base de données des bogues MySQL.

Par exemple, imaginons une requête sur une table avec deux index où les arguments de la recherche font référence aux colonnes indexées.

```
SELECT * FROM table1
WHERE indexed_col1 = 'value1' AND indexed_col2 = 'value2';
```

Dans ce cas, le moteur de recherche analysera les deux index. Cependant, à cause du bogue, les résultats fusionnés sont incorrects.

Pour résoudre ce problème, vous pouvez procéder de l'une des manières suivantes :

- Définissez le paramètre `optimizer_switch` sur `index_merge=off` dans le groupe de paramètres DB de votre instance de base de données MySQL. Pour plus d'informations sur la définition des paramètres d'un groupe de paramètres DB, consultez Groupes de paramètres pour (Amazon RDS).
- Mettez à niveau votre instance de base de données MySQL vers MySQL version 5.7 ou 8.0. Pour de plus amples informations, veuillez consulter Mises à niveau du moteur de base de données RDS pour MySQL.

- Si vous ne pouvez pas mettre à niveau votre instance ou modifier le paramètre `optimizer_switch`, vous pouvez contourner le bogue en identifiant explicitement un index pour la requête, par exemple :

```
SELECT * FROM table1
USE INDEX covering_index
WHERE indexed_col1 = 'value1' AND indexed_col2 = 'value2';
```

Pour en savoir plus, consultez Index merge optimization (Optimisation de la fusion d'index) dans la documentation MySQL.

Exceptions des paramètres MySQL pour les instances de base de données Amazon RDS

Certains paramètres MySQL nécessitent des considérations spéciales lors d'une utilisation avec une instance de base de données Amazon RDS.

lower_case_table_names

Comme Amazon RDS utilise un système de fichiers qui respecte la casse, la définition de la valeur du paramètre de serveur `lower_case_table_names` sur 2 (noms stockés tels qu'ils ont été fournis mais comparés en minuscules) n'est pas prise en charge. Voici les valeurs prises en charge pour les instances de base de données Amazon RDS for MySQL :

- 0 (les noms stockés tels quels et les comparaisons sont sensibles à la casse) est pris en charge pour toutes les versions de RDS for MySQL.
- 1 (les noms sont stockés en minuscules et les comparaisons ne font pas la distinction entre majuscules et minuscules) est pris en charge pour RDS pour MySQL version 5.7, version 8.0.28 et versions 8.0 supérieures, et version 8.4.

Définissez le paramètre `lower_case_table_names` dans un groupe de paramètres de base de données personnalisé avant de créer une instance de base de données. Ensuite, spécifiez le groupe de paramètres de base de données personnalisé lorsque vous créez l'instance de base de données.

Quand un groupe de paramètres est associé à une instance de base de données MySQL dont la version est antérieure à la version 8.0, nous vous recommandons d'éviter de modifier le paramètre `lower_case_table_names` dans le groupe de paramètres. Sa modification peut entraîner des

incohérences entre les sauvegardes de point-in-time restauration et la lecture des instances de base de données répliquées.

Lorsqu'un groupe de paramètres est associé à une instance de base de données MySQL version 8.0 ou 8.4, vous ne pouvez pas modifier le `lower_case_table_names` paramètre dans le groupe de paramètres.

Les réplicas en lecture doivent toujours utiliser la même valeur de paramètre `lower_case_table_names` que l'instance de base de données source.

long_query_time

Vous pouvez définir le paramètre `long_query_time` sur une valeur à virgule flottante afin de pouvoir consigner les requêtes lentes dans le journal des requêtes lentes MySQL avec une résolution en microsecondes. Vous pouvez définir une valeur telle que 0,1 seconde, ce qui correspondrait à 100 millisecondes, pour aider lors du débogage de transactions lentes qui durent moins d'une seconde.

Limites de taille des fichiers MySQL dans Amazon RDS

Pour les instances de base de données MySQL 8.0 et supérieures, la taille de fichier maximale est de 16 TiB. Lorsque vous utilisez des file-per-table tablespaces, la taille de fichier maximale limite la taille d'une table InnoDB à 16 TiB. Les file-per-table tablespaces InnoDB (avec des tables chacune dans leur propre tablespace) sont définis par défaut pour les instances de base de données MySQL. Pour plus d'informations, consultez InnoDB Limits dans la documentation MySQL.

> **Note**
>
> Certaines instances de bases de données existantes ont une limite inférieure. Par exemple, les instances de base de données MySQL créées avant avril 2014 ont une limite de taille de fichier et de table de 2 To. Cette limite de taille de fichier de 2 To s'applique également aux instances de bases de données ou aux réplicas en lecture créés à partir d'instantanés de bases de données pris avant avril 2014, quel que soit le moment de la création de l'instance de base de données.

L'utilisation des file-per-table tablespaces InnoDB présente des avantages et des inconvénients, en fonction de votre application. Pour déterminer la meilleure approche pour votre application, consultez les File-per-table espaces de table dans la documentation MySQL.

Il est déconseillé d'autoriser les tables à dépasser la taille maximale de fichier. En général, une meilleure pratique consiste à partitionner les données en tables plus petites, ce qui peut améliorer la performance et les temps de récupération.

Vous pouvez utiliser l'option de partitionnement pour diviser une grande table en tables plus petites. Le partitionnement répartit des portions de votre grande table en fichiers distincts basés sur des règles que vous spécifiez. Par exemple, si vous stockez des transactions par date, vous pouvez créer des règles de partitionnement qui répartissent des transactions plus anciennes en fichiers distincts en utilisant le partitionnement. Ensuite, vous pouvez archiver régulièrement les données de transaction historiques qui n'ont pas besoin d'être rapidement utilisables par votre application. Pour de plus amples informations, veuillez consulter Partitionnement dans la documentation MySQL.

Comme aucune table ou vue système ne fournit la taille de toutes les tables et de l'espace de table système InnoDB, vous devez interroger plusieurs tables pour déterminer la taille des espaces de table.

Pour déterminer la taille de l'espace de table système InnoDB et de l'espace de table du dictionnaire de données

- Utilisez la commande SQL suivante pour déterminer si certains de vos espaces de table sont trop volumineux et pourraient faire l'objet d'un partitionnement.

  > **Note**
  >
  > Le tablespace du dictionnaire de données est spécifique à MySQL 8.0 et aux versions supérieures.

```
select FILE_NAME,TABLESPACE_NAME, ROUND(((TOTAL_EXTENTS*EXTENT_SIZE)
/1024/1024/1024), 2)  as "File Size (GB)" from information_schema.FILES
where tablespace_name in ('mysql','innodb_system');
```

Pour déterminer la taille des tables utilisateur InnoDB en dehors de l'espace de table système InnoDB (pour les versions MySQL 5.7)

- Utilisez la commande SQL suivante pour déterminer si certaines de vos tables sont trop volumineuses et peuvent faire l'objet d'un partitionnement.

```
SELECT SPACE,NAME,ROUND((ALLOCATED_SIZE/1024/1024/1024), 2)
as "Tablespace Size (GB)"
FROM information_schema.INNODB_SYS_TABLESPACES ORDER BY 3 DESC;
```

Pour déterminer la taille des tables utilisateur InnoDB en dehors de l'espace de table du système InnoDB (pour MySQL 8.0 et versions supérieures)

- Utilisez la commande SQL suivante pour déterminer si certaines de vos tables sont trop volumineuses et peuvent faire l'objet d'un partitionnement.

```
SELECT SPACE,NAME,ROUND((ALLOCATED_SIZE/1024/1024/1024), 2)
as "Tablespace Size (GB)"
FROM information_schema.INNODB_TABLESPACES ORDER BY 3 DESC;
```

Pour déterminer la taille des tables utilisateur non-InnoDB

- Utilisez la commande SQL suivante pour déterminer si certaines de vos tables utilisateur non-InnoDB sont trop volumineuses.

```
SELECT TABLE_SCHEMA, TABLE_NAME, round(((DATA_LENGTH + INDEX_LENGTH+DATA_FREE)
/ 1024 / 1024/ 1024), 2) As "Approximate size (GB)" FROM information_schema.TABLES
WHERE TABLE_SCHEMA NOT IN ('mysql', 'information_schema', 'performance_schema')
and ENGINE<>'InnoDB';
```

Pour activer les tablespaces InnoDB file-per-table

- Définissez le paramètre innodb_file_per_table sur 1 dans le groupe de paramètres pour l'instance de base de données.

Pour désactiver les tablespaces InnoDB file-per-table

- Définissez le paramètre innodb_file_per_table sur `0` dans le groupe de paramètres pour l'instance de base de données.

Pour plus d'informations sur la mise à jour d'un groupe de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Lorsque vous avez activé ou désactivé les file-per-table tablespaces InnoDB, vous pouvez émettre une `ALTER TABLE` commande pour déplacer une table du tablespace global vers son propre tablespace, ou de son propre tablespace vers le tablespace global, comme indiqué dans l'exemple suivant :

```
ALTER TABLE table_name TABLESPACE=innodb_file_per_table;
```

Plug-in MySQL Keyring non pris en charge

Actuellement, Amazon RDS for MySQL ne prend pas en charge le plug-in MySQL `keyring_aws` Amazon Web Services Keyring.

Ports personnalisés

Amazon RDS bloque les connexions au port personnalisé 33060 pour le moteur MySQL. Choisissez un port différent pour votre moteur MySQL.

Limitations des procédures stockées MySQL

Les procédures stockées mysql.rds_kill et mysql.rds_kill_query ne peuvent pas mettre fin à des sessions ou à des requêtes appartenant à des utilisateurs MySQL dont le nom d'utilisateur comporte plus de 16 caractères sur les versions suivantes de RDS for MySQL :

- 8.0.32 et versions 8 antérieures
- 5.7.41 et versions 5.7 antérieures

Réplication basée sur GTID avec une instance source externe

Amazon RDS prend en charge la réplication basée sur des identifiants de transaction globaux (GTIDs) depuis une instance MySQL externe vers une instance de base de données Amazon RDS for MySQL qui nécessite de définir GTID_PURGED lors de la configuration. Toutefois, seules les versions 8.0.37 et supérieures de RDS for MySQL prennent en charge cette fonctionnalité.

Plugin d'authentification par défaut MySQL

RDS pour MySQL version 8.0.34 et versions 8.0 supérieures utilisent le plugin. `mysql_native_password` Vous ne pouvez pas modifier le paramètre `default_authentication_plugin`.

RDS for MySQL version 8.4 et versions supérieures utilisent le `caching_sha2_password` plugin comme plugin d'authentification par défaut. Vous pouvez modifier le plugin d'authentification par défaut pour MySQL 8.4. Le `mysql_native_password` plugin fonctionne toujours avec MySQL 8.4, mais le support de ce plugin prend fin avec MySQL 8.4. Pour modifier le plug-in d'authentification par défaut, créez un groupe de paramètres personnalisé et modifiez la valeur du `authentication_policy` paramètre. Pour de plus amples informations, veuillez consulter the section called “Groupes de paramètres par défaut et personnalisés”.

Remplacer innodb_buffer_pool_size

Dans le cas de classes d'instance de base de données de petite ou de petite taille, la valeur par défaut du `innodb_buffer_pool_size` paramètre peut être différente de la valeur renvoyée en exécutant la commande suivante :

```
mysql> SELECT @@innodb_buffer_pool_size;
```

Cette différence peut se produire lorsqu'Amazon RDS doit remplacer la valeur par défaut dans le cadre de la gestion des classes d'instances de base de données. Si nécessaire, vous pouvez remplacer la valeur par défaut et la définir sur une valeur prise en charge par votre classe d'instance de base de données. Pour déterminer une valeur valide, ajoutez l'utilisation de la mémoire et la mémoire totale disponible sur votre instance de base de données. Pour plus d'informations, consultez la section Types d'instances Amazon RDS.

Si votre instance de base de données ne dispose que de 4 Go de mémoire, vous ne pouvez pas la `innodb_buffer_pool_size` définir sur 8 Go, mais vous pouvez peut-être la définir sur 3 Go, en fonction de la quantité de mémoire que vous avez allouée aux autres paramètres.

Si la valeur que vous saisissez est trop élevée, Amazon RDS la réduit aux limites suivantes :

- Classes d'instances Micro DB : 256 Mo
- Classes d'instance de base de données db.t4g.micro : 128 Mo

Mise à niveau de MySQL 5.7 vers MySQL 8.4

Vous ne pouvez pas passer directement de MySQL 5.7 à MySQL 8.4. Vous devez d'abord effectuer une mise à niveau de MySQL 5.7 vers MySQL 8.0, puis de MySQL 8.0 vers MySQL 8.4. Pour de plus amples informations, veuillez consulter Mises à niveau des versions majeures de RDS pour MySQL.

Compression de page InnoDB

La compression de page InnoDB ne fonctionne pas avec les instances de base de données Amazon RDS dont la taille de bloc de système de fichiers est de 16 ko, car la taille de bloc du système de fichiers doit être inférieure à la taille de page InnoDB. À compter de février 2024, toutes les instances de base de données nouvellement créées ont une taille de bloc de système de fichiers de 16 000 octets, ce qui augmente le débit et réduit la consommation d'IOPS lors du vidage des pages.

RDSpour Ma référence de procédure SQL stockée

Ces rubriques décrivent les procédures stockées dans le système disponibles pour les RDS instances Amazon exécutant le moteur My SQL DB. L'utilisateur principal doit exécuter ces procédures.

Rubriques

- Collecte et mise à jour de l'historique des statuts mondiaux
- Configuration, démarrage et arrêt de la réplication du journal binaire (binlog)
- Mettre fin à une session ou à une requête
- Gestion des clusters actifs-actifs
- Gestion de la réplication multi-sources
- Répliquer des transactions à l'aide de GTIDs
- Rotation des journaux de requêtes
- Configuration et affichage de la configuration du journal binaire
- Réchauffement du cache InnoDB

Collecte et mise à jour de l'historique des statuts mondiaux

Amazon RDS fournit un ensemble de procédures qui prennent des instantanés des valeurs des variables d'état au fil du temps et les écrivent dans une table, ainsi que les modifications apportées depuis le dernier instantané. Cette infrastructure porte le nom d'historique global des statuts. Pour plus d'informations, consultez Gestion de l'historique global du statut de RDS for My SQL.

Les procédures stockées suivantes gèrent la manière dont l'historique global des statuts est collecté et conservé.

Rubriques

- mysql.rds_collect_global_status_history
- mysql.rds_disable_gsh_collector
- mysql.rds_disable_gsh_rotation
- mysql.rds_enable_gsh_collector
- mysql.rds_enable_gsh_rotation
- mysql.rds_rotate_global_status_history
- mysql.rds_set_gsh_collector
- mysql.rds_set_gsh_rotation

mysql.rds_collect_global_status_history

Prend un instantané sur demande pour l'historique global des statuts.

Syntaxe

```
CALL mysql.rds_collect_global_status_history;
```

mysql.rds_disable_gsh_collector

Désactive les instantanés pris par l'historique global des statuts.

Syntaxe

```
CALL mysql.rds_disable_gsh_collector;
```

mysql.rds_disable_gsh_rotation

Désactive la rotation de la table `mysql.global_status_history`.

Syntaxe

```
CALL mysql.rds_disable_gsh_rotation;
```

mysql.rds_enable_gsh_collector

Active l'historique global des statuts pour prendre des instantanés par défaut aux intervalles spécifiés par `rds_set_gsh_collector`.

Syntaxe

```
CALL mysql.rds_enable_gsh_collector;
```

mysql.rds_enable_gsh_rotation

Active la rotation du contenu de la table `mysql.global_status_history` en `mysql.global_status_history_old` aux intervalles spécifiés par `rds_set_gsh_rotation`.

Syntaxe

```
CALL mysql.rds_enable_gsh_rotation;
```

mysql.rds_rotate_global_status_history

Effectue une rotation du contenu de la table `mysql.global_status_history` en `mysql.global_status_history_old` à la demande.

Syntaxe

```
CALL mysql.rds_rotate_global_status_history;
```

mysql.rds_set_gsh_collector

Spécifie l'intervalle, en minutes, entre les instantanés pris par l'historique global des statuts.

Syntaxe

```
CALL mysql.rds_set_gsh_collector(intervalPeriod);
```

Paramètres

intervalPeriod

Intervalle, en minutes, entre les instantanés. La valeur par défaut est 5.

mysql.rds_set_gsh_rotation

Spécifie l'intervalle, en jours, entre deux rotations de la table `mysql.global_status_history`.

Syntaxe

```
CALL mysql.rds_set_gsh_rotation(intervalPeriod);
```

Paramètres

intervalPeriod

Intervalle, en jours, entre deux rotations de table. La valeur par défaut est 7.

Configuration, démarrage et arrêt de la réplication du journal binaire (binlog)

Les procédures stockées suivantes contrôlent la manière dont les transactions sont répliquées depuis une base de données externe vers RDS for MySQL, ou depuis RDS for My SQL vers une base de données externe.

Lorsque vous utilisez ces procédures stockées pour gérer la réplication avec un utilisateur de réplication configuré avec`caching_sha2_password`, vous devez configurer TLS en spécifiant`SOURCE_SSL=1`. `caching_sha2_password`est le plugin d'authentification par défaut RDS pour My SQL 8.4. Pour plus d'informations, consultezChiffrer avec/SSLTLS.

Pour plus d'informations sur la configuration, l'utilisation et la gestion des répliques de lecture, consultezthe section called “Mes répliques de SQL lecture”.

Rubriques

-
-
-
-
-
-
- mysql.rds_set_external_master_with_auto_position (pour mes versions majeures 8.0 et antérieures) RDS SQL
- mysql.rds_set_external_source_with_auto_position (pour mes versions majeures 8.4 et supérieures) RDS SQL
- mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL
- mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL
- mysql.rds_set_external_source_gtid_purged
-
-
- mysql.rds_set_source_delay

- mysql.rds_skip_repl_error
- mysql.rds_start_replication
-
- mysql.rds_stop_replication

Modifie la position du journal de l'instance de base de données source au début du journal binaire suivant sur l'instance de base de données source. N'utilisez cette procédure que si vous recevez une erreur 1236 d'I/O de réplication sur un réplica en lecture.

Syntaxe

```
CALL mysql.rds_next_master_log(
curr_master_log
);
```

Paramètres

curr_master_log

Index du fichier journal maître actif. Par exemple, si le fichier en cours se nomme `mysql-bin-changelog.012345`, l'index est 12345. Pour déterminer le nom du fichier journal maître actif, exécutez la commande `SHOW REPLICA STATUS` et affichez le champ `Master_Log_File`.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_next_master_log`.

> **Warning**
>
> Appelez `mysql.rds_next_master_log` uniquement si la réplication échoue après le basculement d'une instance de base de données multi-AZ qui est la source de la réplication, et que le champ `Last_IO_Errno` de `SHOW REPLICA STATUS` signale une erreur d'I/O 1236.
> L'appel de `mysql.rds_next_master_log` peut se traduire par une perte de données dans le réplica en lecture si les transactions de l'instance source n'ont pas été écrites dans le

journal binaire sur disque avant que l'événement de basculement se produise. Vous pouvez réduire le risque que cela se produise en définissant les paramètres de l'instance source `sync_binlog` et `innodb_support_xa` en le 1 faisant, même si cela peut réduire les performances. Pour de plus amples informations, veuillez consulter Résolution d'un problème lié à SQL la réplique My read.

Exemples

Supposons que la réplication échoue sur une RDS réplique de My SQL read. L'exécution de `SHOW REPLICA STATUS\G` sur le réplica en lecture renvoie le résultat suivant :

```
*************************** 1. row ***************************
             Replica_IO_State:
                  Source_Host: myhost.XXXXXXXXXXXXXX.rr-rrrr-1.rds.amazonaws.com
                  Source_User: MasterUser
                  Source_Port: 3306
                Connect_Retry: 10
              Source_Log_File: mysql-bin-changelog.012345
          Read_Source_Log_Pos: 1219393
               Relay_Log_File: relaylog.012340
                Relay_Log_Pos: 30223388
        Relay_Source_Log_File: mysql-bin-changelog.012345
           Replica_IO_Running: No
          Replica_SQL_Running: Yes
              Replicate_Do_DB:
          Replicate_Ignore_DB:
           Replicate_Do_Table:
       Replicate_Ignore_Table:
      Replicate_Wild_Do_Table:
  Replicate_Wild_Ignore_Table:
                   Last_Errno: 0
                   Last_Error:
                 Skip_Counter: 0
          Exec_Source_Log_Pos: 30223232
              Relay_Log_Space: 5248928866
              Until_Condition: None
               Until_Log_File:
                Until_Log_Pos: 0
           Source_SSL_Allowed: No
           Source_SSL_CA_File:
           Source_SSL_CA_Path:
```

```
              Source_SSL_Cert:
            Source_SSL_Cipher:
               Source_SSL_Key:
        Seconds_Behind_Master: NULL
Source_SSL_Verify_Server_Cert: No
                Last_IO_Errno: 1236
                Last_IO_Error: Got fatal error 1236 from master when reading data from
 binary log: 'Client requested master to start replication from impossible position;
 the first event 'mysql-bin-changelog.013406' at 1219393, the last event read from
 '/rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4, the last byte read from '/
rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4.'
               Last_SQL_Errno: 0
               Last_SQL_Error:
  Replicate_Ignore_Server_Ids:
             Source_Server_Id: 67285976
```

Le champ `Last_IO_Errno` montre que l'instance reçoit une erreur 1236 d'I/O. Le champ `Master_Log_File` montre que le nom du fichier est `mysql-bin-changelog.012345`, ce qui signifie que l'index du fichier journal est `12345`. Pour résoudre l'erreur, vous pouvez appeler `mysql.rds_next_master_log` avec le paramètre suivant :

```
CALL mysql.rds_next_master_log(12345);
```

Modifie la position du journal de l'instance de base de données source au début du journal binaire suivant sur l'instance de base de données source. N'utilisez cette procédure que si vous recevez une erreur 1236 d'I/O de réplication sur un réplica en lecture.

Syntaxe

```
CALL mysql.rds_next_source_log(
curr_source_log
);
```

Paramètres

curr_source_log

Index du fichier journal source actuel. Par exemple, si le fichier en cours se nomme `mysql-bin-changelog.012345`, l'index est 12345. Pour déterminer le nom du fichier journal actuel, exécutez la commande `SHOW REPLICA STATUS` et affichez le champ `Source_Log_File`.

Notes d’utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_next_source_log`.

> **Warning**
>
> Appelez `mysql.rds_next_source_log` uniquement si la réplication échoue après le basculement d'une instance de base de données multi-AZ qui est la source de la réplication, et que le champ `Last_IO_Errno` de `SHOW REPLICA STATUS` signale une erreur d'I/O 1236.
>
> L'appel de `mysql.rds_next_source_log` peut se traduire par une perte de données dans le réplica en lecture si les transactions de l'instance source n'ont pas été écrites dans le journal binaire sur disque avant que l'événement de basculement se produise. Vous pouvez réduire le risque que cela se produise en définissant les paramètres de l'instance source `sync_binlog` et `innodb_support_xa` en le 1 faisant, même si cela peut réduire les performances. Pour de plus amples informations, veuillez consulter Résolution d'un problème lié à SQL la réplique My read.

Exemples

Supposons que la réplication échoue sur une RDS réplique de My SQL read. L'exécution de `SHOW REPLICA STATUS\G` sur le réplica en lecture renvoie le résultat suivant :

```
*************************** 1. row ***************************
             Replica_IO_State:
                  Source_Host: myhost.XXXXXXXXXXXXXX.rr-rrrr-1.rds.amazonaws.com
                  Source_User: MasterUser
                  Source_Port: 3306
                Connect_Retry: 10
              Source_Log_File: mysql-bin-changelog.012345
          Read_Source_Log_Pos: 1219393
               Relay_Log_File: relaylog.012340
                Relay_Log_Pos: 30223388
        Relay_Source_Log_File: mysql-bin-changelog.012345
           Replica_IO_Running: No
          Replica_SQL_Running: Yes
              Replicate_Do_DB:
          Replicate_Ignore_DB:
           Replicate_Do_Table:
```

```
       Replicate_Ignore_Table:
      Replicate_Wild_Do_Table:
  Replicate_Wild_Ignore_Table:
                  Last_Errno: 0
                  Last_Error:
                Skip_Counter: 0
         Exec_Source_Log_Pos: 30223232
             Relay_Log_Space: 5248928866
             Until_Condition: None
              Until_Log_File:
               Until_Log_Pos: 0
          Source_SSL_Allowed: No
          Source_SSL_CA_File:
          Source_SSL_CA_Path:
             Source_SSL_Cert:
           Source_SSL_Cipher:
              Source_SSL_Key:
        Seconds_Behind_Source: NULL
Source_SSL_Verify_Server_Cert: No
               Last_IO_Errno: 1236
               Last_IO_Error: Got fatal error 1236 from source when reading data from
 binary log: 'Client requested source to start replication from impossible position;
 the first event 'mysql-bin-changelog.013406' at 1219393, the last event read from
 '/rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4, the last byte read from '/
rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4.'
              Last_SQL_Errno: 0
              Last_SQL_Error:
  Replicate_Ignore_Server_Ids:
             Source_Server_Id: 67285976
```

Le champ `Last_IO_Errno` montre que l'instance reçoit une erreur 1236 d'I/O. Le champ `Source_Log_File` montre que le nom du fichier est `mysql-bin-changelog.012345`, ce qui signifie que l'index du fichier journal est `12345`. Pour résoudre l'erreur, vous pouvez appeler `mysql.rds_next_source_log` avec le paramètre suivant :

```
CALL mysql.rds_next_source_log(12345);
```

Reconfigure une RDS instance DB afin qu'elle ne soit plus une réplique en lecture d'une instance de My SQL exécutée en dehors d'Amazon. RDS

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur `1`. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_reset_external_master;
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_reset_external_master`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être supprimée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Pour plus d'informations sur l'utilisation de la réplication pour importer des données depuis une instance de My SQL exécutée en dehors d'AmazonRDS, consultezConfiguration d'une réplication de position de fichier journal binaire avec une instance source externe.

Reconfigure une RDS instance DB afin qu'elle ne soit plus une réplique en lecture d'une instance de My SQL exécutée en dehors d'Amazon. RDS

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur `1`. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_reset_external_source;
```

Notes d'utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_reset_external_source`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être supprimée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS.
> Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données. Pour plus d'informations sur l'utilisation de la réplication pour importer des données depuis une instance de My SQL exécutée en dehors d'AmazonRDS, consultezConfiguration d'une réplication de position de fichier journal binaire avec une instance source externe.

Configure une RDS instance My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'AmazonRDS.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur `1`. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

> **Note**
>
> Vous pouvez utiliser la procédure stockée mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL pour configurer une instance de base de données source externe et une réplication différée.

Syntaxe

```
CALL mysql.rds_set_external_master (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS pour devenir l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Nom du journal binaire sur l'instance de base de données source qui contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier journal binaire en exécutant `SHOW MASTER STATUS` sur l'instance de base de données source.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `MASTER_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_master`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Avant d'exécuter`mysql.rds_set_external_master`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour

vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous devez spécifier `replication_user_name` des `replication_user_password` valeurs indiquant un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   Mon SQL 5,7

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
   ```

   Mon SQL 8.0

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED WITH mysql_native_password BY
    'password';
   ```

   > **Note**
   >
   > Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur « repl_user » de votre domaine.

   Mon SQL 5,7

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com'
    IDENTIFIED BY 'password';
   ```

   Mon SQL 8.0

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
   ```

Pour utiliser la réplication chiffrée, configurez l'instance de base de données source pour qu'elle utilise SSL des connexions.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Après avoir appelé `mysql.rds_set_external_master` pour configurer une instance Amazon RDS DB en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsqu'il `mysql.rds_set_external_master` est appelé, Amazon RDS enregistre l'heure, l'utilisateur et une action `set master` dans les `mysql.rds_replication_status` tables `mysql.rds_history` et.

Exemples

Lorsqu'elle est exécutée sur une instance My SQL DB, l'exemple suivant configure l'instance DB comme une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS.

```
call mysql.rds_set_external_master(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'password',
  'mysql-bin-changelog.0777',
  120,
  1);
```

Configure une RDS instance My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'AmazonRDS.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS pour devenir l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Nom du journal binaire sur l'instance de base de données source qui contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier journal binaire en exécutant `SHOW MASTER STATUS` sur l'instance de base de données source.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `SOURCE_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

Notes d'utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_set_external_source`. Cette procédure doit être exécutée sur l'instance de base de SQL données for My pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Avant d'exécuter`mysql.rds_set_external_source`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour

vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous devez spécifier `replication_user_name` des `replication_user_password` valeurs indiquant un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'password';
   ```

   > **Note**
   >
   > Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur « repl_user » de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com';
   ```

Pour utiliser la réplication chiffrée, configurez l'instance de base de données source pour qu'elle utilise SSL des connexions. De plus, importez le certificat de l'autorité de certification, le certificat du client et la clé du client dans l'instance de base de données ou le cluster de bases de données à l'aide de la procédure mysql.rds_import_binlog_ssl_material.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS.

Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Après avoir appelé `mysql.rds_set_external_source` pour configurer une instance de base de SQL données for My en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsqu'il `mysql.rds_set_external_source` est appelé, Amazon RDS enregistre l'heure, l'utilisateur et une action `set master` dans les `mysql.rds_replication_status` tables `mysql.rds_history` et.

Exemples

Lorsqu'elle est exécutée sur une instance de base de SQL données for My, l'exemple suivant configure l'instance de base de données pour qu'elle soit une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS.

```
call mysql.rds_set_external_source(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'password',
  'mysql-bin-changelog.0777',
  120,
  1);
```

mysql.rds_set_external_master_with_auto_position (pour mes versions majeures 8.0 et antérieures) RDS SQL

Configure une instance RDS for My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'AmazonRDS. Cette procédure configure également la réplication différée et la réplication en fonction des identificateurs de transaction globaux ()GTIDs.

Important

Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification

des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_master_with_auto_position (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , ssl_encryption
  , delay
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS pour devenir l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `MASTER_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

delay

Nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_master_with_auto_position`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Cette procédure est prise en charge RDS pour toutes les versions de My SQL 5.7, ainsi que RDS pour les versions 8.0 de My SQL 8.0.26 et supérieures.

Avant d'exécuter`mysql.rds_set_external_master_with_auto_position`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous devez spécifier des valeurs pour `replication_user_name` et`replication_user_password`. Ces valeurs doivent indiquer un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'SomePassW0rd'
   ```

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur `'repl_user'` de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com'
   IDENTIFIED BY 'SomePassW0rd'
   ```

Pour de plus amples informations, veuillez consulter Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Avant d'appelermysql.rds_set_external_master_with_auto_position, assurez-vous d'appeler the section called “mysql.rds_set_external_source_gtid_purged” pour définir la variable `gtid_purged` système avec une GTID plage spécifiée à partir d'une source externe.

Après avoir appelé `mysql.rds_set_external_master_with_auto_position` pour configurer une instance Amazon RDS DB en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsque vous appelezmysql.rds_set_external_master_with_auto_position, Amazon RDS enregistre l'heure, l'utilisateur et une action set master dans les mysql.rds_replication_status tables mysql.rds_history et.

Pour la reprise après sinistre, vous pouvez utiliser cette procédure avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure mysql.rds_set_external_master_with_auto_position. Une fois que la procédure mysql.rds_start_replication_until_gtid a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser mysql.rds_rds_start_replication_until_gtid cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur l'utilisation de la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

Exemples

Lorsqu'elle est exécutée sur une instance My SQL DB, l'exemple suivant configure l'instance DB comme une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS. Il définit le délai de réplication minimal à une heure (3 600 secondes) sur l'instance My SQL DB. Une modification apportée à l'instance de base de données My SQL source exécutée en externe à Amazon RDS n'est pas appliquée à la réplique lue de l'instance My SQL DB pendant au moins une heure.

```
call mysql.rds_set_external_master_with_auto_position(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'SomePassW0rd',
  1,
  3600);
```

mysql.rds_set_external_source_with_auto_position (pour mes versions majeures 8.4 et supérieures) RDS SQL

Configure une instance RDS for My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'AmazonRDS. Cette procédure configure également la réplication différée et la réplication en fonction des identificateurs de transaction globaux ()GTIDs.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source_with_auto_position (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , ssl_encryption
  , delay
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS pour devenir l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `SOURCE_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

delay

Nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

Notes d'utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_set_external_source_with_auto_position`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Avant d'exécuter`mysql.rds_set_external_source_with_auto_position`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous

devez spécifier des valeurs pour `replication_user_name` et`replication_user_password`. Ces valeurs doivent indiquer un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'SomePassW0rd'
   ```

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur `'repl_user'` de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com'
   IDENTIFIED BY 'SomePassW0rd'
   ```

Pour de plus amples informations, veuillez consulter Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Avant d'appelermysql.rds_set_external_source_with_auto_position, assurez-vous d'appeler the section called “mysql.rds_set_external_source_gtid_purged” pour définir la variable `gtid_purged` système avec une GTID plage spécifiée à partir d'une source externe.

Après avoir appelé `mysql.rds_set_external_source_with_auto_position` pour configurer une instance Amazon RDS DB en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsque vous appelez`mysql.rds_set_external_source_with_auto_position`, Amazon RDS enregistre l'heure, l'utilisateur et une action `set master` dans les `mysql.rds_replication_status` tables `mysql.rds_history` et.

Pour la reprise après sinistre, vous pouvez utiliser cette procédure avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure `mysql.rds_set_external_source_with_auto_position`. Une fois que la procédure `mysql.rds_start_replication_until_gtid` a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser `mysql.rds_rds_start_replication_until_gtid` cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur l'utilisation de la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

Exemples

Lorsqu'elle est exécutée sur une instance My SQL DB, l'exemple suivant configure l'instance DB comme une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS. Il définit le délai de réplication minimal à une heure (3 600 secondes) sur l'instance My SQL DB. Une modification apportée à l'instance de base de données My SQL source exécutée en externe à Amazon RDS n'est pas appliquée à la réplique lue de l'instance My SQL DB pendant au moins une heure.

```
call mysql.rds_set_external_source_with_auto_position(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'SomePassW0rd',
  1,
  3600);
```

mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL

Configure une instance RDS de My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'Amazon RDS et configure la réplication différée.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_master_with_delay(
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
  , delay
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS qui deviendra l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de SSH port qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Le nom du journal binaire sur l'instance de base de données source contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier journal binaire en exécutant `SHOW MASTER STATUS` sur l'instance de base de données source.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `MASTER_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

delay

Nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_master_with_delay`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Avant d'exécuter`mysql.rds_set_external_master_with_delay`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous devez spécifier des valeurs pour `replication_user_name` et`replication_user_password`. Ces valeurs doivent indiquer un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'SomePassW0rd'
   ```

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur `'repl_user'` de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com'
   IDENTIFIED BY 'SomePassW0rd'
   ```

Pour de plus amples informations, veuillez consulter Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement

pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Après avoir appelé `mysql.rds_set_external_master_with_delay` pour configurer une instance Amazon RDS DB en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsque vous appelez`mysql.rds_set_external_master_with_delay`, Amazon RDS enregistre l'heure, l'utilisateur et une action `set master` dans les `mysql.rds_replication_status` tables `mysql.rds_history` et.

Pour la reprise après sinistre, vous pouvez utiliser cette procédure avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure `mysql.rds_set_external_master_with_delay`. Une fois que la procédure `mysql.rds_start_replication_until` a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser `mysql.rds_rds_start_replication_until_gtid` cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur l'utilisation de la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

La `mysql.rds_set_external_master_with_delay` procédure est disponible dans les versions suivantes de RDS for My SQL :

- Mes SQL versions 8.0.26 et supérieures 8.0
- Toutes les versions 5.7

Exemples

Lorsqu'elle est exécutée sur une instance My SQL DB, l'exemple suivant configure l'instance DB comme une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS.

Il définit le délai de réplication minimal à une heure (3 600 secondes) sur l'instance My SQL DB. Une modification apportée à l'instance de base de données My SQL source exécutée en externe à Amazon RDS n'est pas appliquée à la réplique lue de l'instance My SQL DB pendant au moins une heure.

```
call mysql.rds_set_external_master_with_delay(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'SomePassW0rd',
  'mysql-bin-changelog.000777',
  120,
  1,
  3600);
```

mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL

Configure une instance RDS de My SQL DB comme une réplique en lecture d'une instance de My SQL exécutée en dehors d'Amazon RDS et configure la réplication différée.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source_with_delay (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
  , delay
```

```
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'SQLinstance My exécutée en dehors d'Amazon RDS qui deviendra l'instance de base de données source.

host_port

Port utilisé par l'SQLinstance My exécutée en externe à Amazon RDS pour être configurée en tant qu'instance de base de données source. Si votre configuration réseau inclut une réplication de SSH port qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'identifiant `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur doté de l'SQLinstance My exécutée en dehors d'AmazonRDS. Nous vous recommandons de fournir un compte qui soit utilisé uniquement pour la réplication avec l'instance externe.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Le nom du journal binaire sur l'instance de base de données source contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier journal binaire en exécutant `SHOW MASTER STATUS` sur l'instance de base de données source.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option SOURCE_SSL_VERIFY_SERVER_CERT n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

delay

Nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

Notes d'utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_set_external_source_with_delay`. Cette procédure doit être exécutée sur l'instance My SQL DB pour être configurée en tant que réplique en lecture d'une SQL instance My exécutée en externe à AmazonRDS.

Avant d'exécuter`mysql.rds_set_external_source_with_delay`, vous devez configurer l'instance de My SQL running external RDS to Amazon comme instance de base de données source. Pour vous connecter à l'SQLinstance My exécutée en externe à AmazonRDS, vous devez spécifier des valeurs pour `replication_user_name` et`replication_user_password`. Ces valeurs doivent indiquer un utilisateur de réplication `REPLICATION SLAVE` disposant `REPLICATION CLIENT` d'autorisations sur l'instance externe de MySQL.

Pour configurer une instance externe de My en SQL tant qu'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance externe de My SQL et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   ```
   CREATE USER 'repl_user'@'mydomain.com' IDENTIFIED BY 'SomePassW0rd'
   ```

2. Sur l'instance externe de MySQL, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur `'repl_user'` de votre domaine.

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'mydomain.com'
IDENTIFIED BY 'SomePassW0rd'
```

Pour de plus amples informations, veuillez consulter Configuration d'une réplication de position de fichier journal binaire avec une instance source externe.

> **Note**
>
> Nous vous recommandons d'utiliser des répliques de lecture pour gérer la réplication entre deux instances Amazon RDS DB lorsque cela est possible. Dans ce cas, nous vous conseillons d'utiliser seulement cette procédure et d'autres procédures stockées liées à la réplication. Ces pratiques permettent des topologies de réplication plus complexes entre les instances Amazon RDS DB. Nous proposons ces procédures stockées principalement pour permettre la réplication avec My SQL instances exécutées en dehors d'AmazonRDS. Pour plus d'informations sur la gestion de la réplication entre les instances Amazon RDS DB, consultezUtilisation des réplicas en lecture d'instance de base de données.

Après avoir appelé `mysql.rds_set_external_source_with_delay` pour configurer une instance Amazon RDS DB en tant que réplique en lecture, vous pouvez faire appel mysql.rds_start_replication à la réplique en lecture pour démarrer le processus de réplication. Vous pouvez appeler pour supprimer la configuration du réplica en lecture.

Lorsque vous appelez`mysql.rds_set_external_source_with_delay`, Amazon RDS enregistre l'heure, l'utilisateur et une action `set master` dans les `mysql.rds_replication_status` tables `mysql.rds_history` et.

Pour la reprise après sinistre, vous pouvez utiliser cette procédure avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure `mysql.rds_set_external_source_with_delay`. Une fois que la procédure `mysql.rds_start_replication_until` a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser `mysql.rds_rds_start_replication_until_gtid` cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur l'utilisation de la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

Exemples

Lorsqu'elle est exécutée sur une instance My SQL DB, l'exemple suivant configure l'instance DB comme une réplique en lecture d'une instance de My SQL exécutée en externe à AmazonRDS. Il définit le délai de réplication minimal à une heure (3 600 secondes) sur l'instance My SQL DB. Une modification apportée à l'instance de base de données My SQL source exécutée en externe à Amazon RDS n'est pas appliquée à la réplique lue de l'instance My SQL DB pendant au moins une heure.

```
call mysql.rds_set_external_source_with_delay(
  'Externaldb.some.com',
  3306,
  'repl_user',
  'SomePassW0rd',
  'mysql-bin-changelog.000777',
  120,
  1,
  3600);
```

mysql.rds_set_external_source_gtid_purged

Définit la variable système gtid_purged avec une GTID plage spécifiée provenant d'une source externe. La `gtid_purged` valeur est requise pour configurer la réplication GTID basée afin de reprendre la réplication à l'aide du positionnement automatique.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source_gtid_purged(
  server_uuid
  , start_pos
  , end_pos
);
```

Paramètres

server_uuid

Identifiant unique universel (UUID) du serveur externe à partir duquel la GTID plage est importée.

start_pos

Position de départ de la GTID plage à définir.

end_pos

Position finale de la GTID plage à définir.

Notes d'utilisation

La `mysql.rds_set_external_source_gtid_purged` procédure n'est disponible qu'avec My SQL 8.0.37 et versions 8.0 supérieures.

Appelez `mysql.rds_set_external_source_gtid_purged` avant d'appeler mysql.rds_set_external_master_with_auto_position (pour mes versions majeures 8.0 et antérieures) RDS SQLmysql.rds_set_external_source_with_auto_position (pour mes versions majeures 8.4 et supérieures) RDS SQL, oumysql.rds_set_external_source_with_auto_position_for_channel.

Avant d'appelermysql.rds_set_external_source_gtid_purged, assurez-vous d'arrêter tous les canaux de réplication actifs pour la base de données. Pour vérifier le statut d'une chaîne, utilisez le formulaire `SHOW REPLICA STATUS` Mon SQL relevé. Pour arrêter la réplication sur un canal, appelezthe section called "mysql.rds_stop_replication_for_channel".

La GTID plage que vous spécifiez doit être un sur-ensemble de la `GTID_PURGED` valeur existante. Cette procédure stockée vérifie les valeurs suivantes avant de définir la `GTID_PURGED` valeur :

- Le `server_uuid` est valide.
- La valeur de `start_pos` est supérieure `0` et inférieure à la valeur deend_pos.

- La valeur de `end_pos` est supérieure ou égale à la valeur de`start_pos`.

Si l'GTIDensemble sur votre serveur externe contient plusieurs plages de valeurs, pensez à appeler la procédure plusieurs fois avec des valeurs GTID définies différentes.

Lorsque vous appelez`mysql.rds_set_external_source_gtid_purged`, Amazon RDS enregistre l'heure, l'utilisateur et une action de `set gtid_purged` dans le `mysql.rds_history` tableau.

Si vous ne définissez pas la `gtid_purged` valeur appropriée pour la sauvegarde que vous utilisez pour la réplication, des transactions peuvent être manquantes ou dupliquées pendant le processus de réplication. Procédez comme suit pour définir la `gtid_purged` valeur correcte.

Pour définir la valeur gtid_purged sur la réplique

1. Déterminez le moment ou le fichier de sauvegarde spécifique à utiliser comme point de départ pour la réplication. Il peut s'agir d'une sauvegarde logique (un fichier mysqldump) ou physique (un instantané AmazonRDS).
2. Déterminez la `gtid_executed` valeur. Cette valeur représente l'ensemble de tous ceux GTIDs qui ont été validés sur le serveur. Pour récupérer cette valeur, sur l'instance source, effectuez l'une des opérations suivantes :

   - Exécutez l'SQLinstruction `SELECT @@GLOBAL.GTID_EXECUTED;` au moment où la sauvegarde a été effectuée.
   - Si des options associées sont incluses dans l'utilitaire de sauvegarde correspondant, extrayez la valeur du fichier de sauvegarde. Pour plus d'informations, consultez l'set-gtid-purgedoption dans Ma SQL documentation.
3. Déterminez la `gtid_purged` valeur à utiliser pour l'appel à`mysql.rds_set_external_source_gtid_purged`. La `gtid_purged` valeur doit inclure tous ceux GTIDs qui ont été exécutés sur l'instance source et qui ne sont plus nécessaires pour la réplication. Par conséquent, la `gtid_purged` valeur doit être un sous-ensemble de la `gtid_executed` valeur que vous avez récupérée à l'étape précédente.

   Pour déterminer la `gtid_purged` valeur, identifiez ceux GTIDs qui ne sont pas inclus dans la sauvegarde et qui ne sont plus nécessaires pour la réplication. Vous pouvez le faire en analysant les journaux binaires ou en utilisant un outil tel que mysqlbinlog pour trouver ceux GTIDs qui ont été purgés des journaux binaires.

Sinon, si vous disposez d'une sauvegarde cohérente qui inclut tous les journaux binaires jusqu'au point de sauvegarde, vous pouvez définir la `gtid_purged` même `gtid_executed` valeur que celle du point de sauvegarde.

4. Après avoir déterminé la `gtid_purged` valeur appropriée qui correspond à votre sauvegarde, appelez la procédure `mysql.rds_set_external_source_gtid_purged` stockée sur votre instance RDS for My SQL DB pour définir la valeur.

Exemples

Lorsqu'il est exécuté sur une instance My SQL DB, l'exemple suivant définit la GTID plage à partir d'un SQL serveur My Server externe avec les UUID `12345678-abcd-1234-efgh-123456789abc` positions de 1 départ et de fin de`100`. La GTID valeur résultante est définie sur`+12345678-abcd-1234-efgh-123456789abc:1-100`.

```
CALL mysql.rds_set_external_source_gtid_purged('12345678-abcd-1234-efgh-123456789abc',
 1, 100);
```

Définit le mode de réplication en fonction des positions du fichier journal binaire ou des identificateurs de transaction globaux (GTIDs).

Syntaxe

```
CALL mysql.rds_set_master_auto_position (
auto_position_mode
);
```

Paramètres

`auto_position_mode`

Une valeur qui indique s'il faut utiliser la réplication de position du fichier journal ou la réplication GTID basée sur la réplication :

- `0` – Utiliser la méthode de réplication basée sur la position du fichier journal binaire. La valeur par défaut est `0`.
- 1— Utilisez la méthode GTID de réplication basée.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_master_auto_position`.

Cette procédure est prise en charge RDS pour toutes les versions de My SQL 5.7 et RDS pour les versions 8.0 de My SQL 8.0.26 et supérieures.

Définit le mode de réplication en fonction des positions du fichier journal binaire ou des identificateurs de transaction globaux (GTIDs).

Syntaxe

```
CALL mysql.rds_set_source_auto_position (auto_position_mode);
```

Paramètres

auto_position_mode

Une valeur qui indique s'il faut utiliser la réplication de position du fichier journal ou la réplication GTID basée sur la réplication :

- `0` – Utiliser la méthode de réplication basée sur la position du fichier journal binaire. La valeur par défaut est `0`.
- 1— Utilisez la méthode GTID de réplication basée.

Notes d'utilisation

L'utilisateur administratif doit exécuter la procédure `mysql.rds_set_source_auto_position`.

mysql.rds_set_source_delay

Définit le nombre minimum de secondes pour retarder la réplication de l'instance de base de données source vers le réplica en lecture actuel. Utilisez cette procédure lorsque vous êtes connecté à un réplica en lecture afin de retarder la réplication à partir de l'instance de base de données source.

Syntaxe

```
CALL mysql.rds_set_source_delay(
delay
);
```

Paramètres

delay

Nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_source_delay`.

Pour la reprise après sinistre, vous pouvez utiliser cette procédure avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure `mysql.rds_set_source_delay`. Une fois que la procédure `mysql.rds_start_replication_until` ou `mysql.rds_start_replication_until_gtid` a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser `mysql.rds_rds_start_replication_until_gtid` cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

La `mysql.rds_set_source_delay` procédure est disponible dans les versions suivantes de RDS for My SQL :

- Tout RDS pour les versions My SQL 8.4
- Mes SQL versions 8.0.26 et supérieures 8.0
- Toutes les versions 5.7

Exemples

Pour retarder la réplication à partir de l'instance de base de données source vers le réplica en lecture actuel pendant au moins un heure (3 600 secondes), vous pouvez appeler `mysql.rds_set_source_delay` avec le paramètre suivant :

```
CALL mysql.rds_set_source_delay(3600);
```

mysql.rds_skip_repl_error

Ignore et supprime une erreur de réplication sur une réplique de lecture My SQL DB.

Syntaxe

```
CALL mysql.rds_skip_repl_error;
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_skip_repl_error` sur un réplica en lecture. Pour plus d'informations sur cette procédure, consultez Appel de la procédure mysql.rds_skip_repl_error.

Pour déterminer s'il existe des erreurs, exécutez la SQL `SHOW REPLICA STATUS\G` commande My. Si une erreur de réplication n'est pas critique, vous pouvez exécuter `mysql.rds_skip_repl_error` pour ignorer l'erreur. S'il y a plusieurs erreurs, `mysql.rds_skip_repl_error` supprime la première erreur, puis avertit qu'il y a d'autres erreurs. Vous pouvez alors utiliser `SHOW REPLICA STATUS\G` pour déterminer l'action appropriée pour l'erreur suivante. Pour plus d'informations sur les valeurs renvoyées, voir la SHOWREPLICASTATUSdéclaration dans la section Ma SQL documentation.

Pour plus d'informations sur la résolution des erreurs de réplication avec AmazonRDS, consultezRésolution d'un problème lié à SQL la réplique My read.

Erreur d'arrêt de réplication

Lorsque vous appelez la procédure `mysql.rds_skip_repl_error`, un message d'erreur peut s'afficher pour indiquer que le réplica a rencontré une erreur ou est désactivé.

Ce message d'erreur s'affiche si vous exécutez la procédure sur l'instance principale plutôt que sur le réplica en lecture. Vous devez exécuter cette procédure sur le réplica en lecture pour que la procédure fonctionne.

Ce message d'erreur peut également s'afficher si vous exécutez la procédure sur le réplica en lecture, mais que la réplication ne peut pas être redémarrée correctement.

Si vous avez besoin d'ignorer un grand nombre d'erreurs, le retard de réplication peut augmenter et dépasser la période de rétention par défaut pour les fichiers journaux binaires (binlog). Dans ce cas, il se peut que vous rencontriez une erreur fatale car les fichiers binlog ont été purgés avant d'être rejoués sur la réplique lue. Cette purge entraîne l'arrêt de la réplication et vous ne pouvez plus appeler la commande `mysql.rds_skip_repl_error` pour ignorer les erreurs de réplication.

Vous pouvez atténuer ce problème en augmentant le nombre d'heures pendant lequel les fichiers journaux binaires sont conservés sur votre instance de base de données source. Une fois que vous avez augmenté le temps de rétention de journaux binaires, vous pouvez redémarrer la réplication et appeler la commande `mysql.rds_skip_repl_error` en fonction des besoins.

Pour définir la période de rétention des journaux binaires, utilisez la procédure mysql.rds_set_configuration et spécifiez un paramètre de configuration `'binlog retention hours'`, ainsi que le nombre d'heures pendant lequel conserver les fichiers journaux binaires sur le cluster de bases de données. L'exemple suivant définit la période de rétention des fichiers journaux binaires à 48 heures.

```
CALL mysql.rds_set_configuration('binlog retention hours', 48);
```

mysql.rds_start_replication

Lance la réplication à partir d'un .

> **Note**
>
> Vous pouvez utiliser la procédure stockée ou RDSpour lancer la réplication à partir d'une instance My SQL DB et arrêter la réplication à l'emplacement du fichier journal binaire spécifié.

Syntaxe

```
CALL mysql.rds_start_replication;
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication`.

Pour importer des données depuis une instance de My SQL external vers AmazonRDS, faites appel `mysql.rds_start_replication` à la réplique en lecture pour démarrer le processus de réplication après l'appel ou pour créer la configuration de réplication. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.

Pour exporter des données vers une instance de My SQL external to AmazonRDS, appelez `mysql.rds_start_replication` et `mysql.rds_stop_replication` activez la réplique en lecture pour contrôler certaines actions de réplication, telles que la purge des journaux binaires. Pour de plus amples informations, veuillez consulter Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Vous pouvez aussi appeler `mysql.rds_start_replication` sur le réplica en lecture pour redémarrer un processus de réplication que vous avez précédemment arrêté en appelant `mysql.rds_stop_replication`. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Lance la réplication à partir RDSd'un cluster Aurora My SQL DB d'instance et arrête la réplication à l'emplacement du fichier journal binaire spécifié.

Syntaxe

```
CALL mysql.rds_start_replication_until (
replication_log_file
  , replication_stop_point
);
```

Paramètres

replication_log_file

Nom du journal binaire sur l'instance de base de données source qui contient les informations de réplication.

replication_stop_point

Position dans le journal binaire `replication_log_file` à laquelle la réplication s'arrêtera.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication_until`.

La `mysql.rds_start_replication_until` procédure est disponible dans les versions suivantes de RDS for My SQL :

- Tout RDS pour les versions My SQL 8.4
- Mes SQL versions 8.0.26 et supérieures 8.0
- Toutes les versions 5.7

Vous pouvez utiliser cette procédure avec la réplication retardée pour la reprise après sinistre. Si vous avez configuré la réplication retardée, vous pouvez utiliser cette procédure pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre. Une fois que cette procédure a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Vous pouvez configurer la réplication retardée en utilisant les procédures stockées suivantes :

- mysql.rds_set_configuration
- mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL
- mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL
- mysql.rds_set_source_delay

Le nom de fichier spécifié pour le paramètre `replication_log_file` doit correspondre au nom du fichier binlog de l'instance de base de données source.

Lorsque le paramètre `replication_stop_point` spécifie une position d'arrêt survenant dans le passé, la réplication est arrêtée immédiatement.

Exemples

L'exemple suivant lance la réplication et réplique les modifications jusqu'à ce qu'il atteigne la position `120` dans le fichier journal binaire `mysql-bin-changelog.000777`.

```
call mysql.rds_start_replication_until(
  'mysql-bin-changelog.000777',
  120);
```

mysql.rds_stop_replication

Arrête la réplication à partir d'une instance My SQL DB.

Syntaxe

```
CALL mysql.rds_stop_replication;
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_stop_replication`.

Si vous configurez la réplication pour importer des données depuis une instance de My SQL exécutée en dehors d'AmazonRDS, vous devez faire appel `mysql.rds_stop_replication` à la réplique en lecture pour arrêter le processus de réplication une fois l'importation terminée. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.

Si vous configurez la réplication pour exporter des données vers une instance de My SQL external to AmazonRDS, vous devez appeler `mysql.rds_start_replication` et `mysql.rds_stop_replication` activer la réplique en lecture pour contrôler certaines actions de réplication, telles que la purge des journaux binaires. Pour de plus amples informations, veuillez consulter Exportation de données depuis une instance My SQL DB à l'aide de la réplication.

Vous pouvez également l'utiliser `mysql.rds_stop_replication` pour arrêter la réplication entre deux instances Amazon RDS DB. Vous arrêtez généralement la réplication pour exécuter une longue opération sur le réplica en lecture, comme la création d'un index volumineux sur le réplica en lecture. Vous pouvez redémarrer tout processus de réplication que vous avez arrêté en appelant mysql.rds_start_replication sur le réplica en lecture. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Mettre fin à une session ou à une requête

Les procédures stockées suivantes mettent fin à une session ou à une requête.

Rubriques

- mysql.rds_kill
- mysql.rds_kill_query

mysql.rds_kill

Termine une connexion au serveur MySQL.

Syntaxe

```
CALL mysql.rds_kill(processID);
```

Paramètres

processID

Identité du thread de connexion à terminer.

Notes d'utilisation

Chaque connexion au serveur MySQL s'exécute dans un thread distinct. Pour terminer une connexion, utilisez la procédure `mysql.rds_kill` et transmettez-lui l'ID de thread de cette connexion. Pour obtenir l'ID de thread, utilisez la commande MySQL SHOW PROCESSLIST.

Pour plus d'informations sur les limites, consultez Limitations des procédures stockées MySQL.

Exemples

L'exemple suivant termine une connexion avec l'ID de thread 4243 :

```
CALL mysql.rds_kill(4243);
```

mysql.rds_kill_query

Termine une requête s'exécutant sur le serveur MySQL.

Syntaxe

```
CALL mysql.rds_kill_query(processID);
```

Paramètres

processID

Identité du processus ou du thread qui exécute la requête à terminer.

Notes d'utilisation

Pour arrêter une requête en cours d'exécution sur le serveur MySQL, utilisez la procédure `mysql_rds_kill_query` et transmettez l'ID de connexion du thread qui exécute la requête. La procédure met alors fin à la connexion.

Pour obtenir l'ID, interrogez la table MySQL INFORMATION_SCHEMA PROCESSLIST ou utilisez la commande MySQL SHOW PROCESSLIST. La valeur figurant dans la colonne ID de `SHOW PROCESSLIST` ou `SELECT * FROM INFORMATION_SCHEMA.PROCESSLIST` est le *processID*.

Pour plus d'informations sur les limites, consultez Limitations des procédures stockées MySQL.

Exemples

L'exemple suivant arrête une requête dont l'ID de thread de requête est 230040 :

```
CALL mysql.rds_kill_query(230040);
```

Gestion des clusters actifs-actifs

Les procédures stockées suivantes permettent de configurer et de gérer RDS Mes clusters SQL actifs-actifs. Pour de plus amples informations, veuillez consulter the section called “Configuration de clusters actifs-actifs”.

Ces procédures stockées ne sont disponibles que RDS pour les instances My SQL DB exécutant les versions suivantes :

- Toutes les versions de My SQL 8.4
- Mes SQL versions mineures 8.0.35 et supérieures

Rubriques

- mysql.rds_group_replication_advance_gtid
- mysql.rds_group_replication_create_user
- mysql.rds_group_replication_set_recovery_channel
- mysql.rds_group_replication_start
- mysql.rds_group_replication_stop

mysql.rds_group_replication_advance_gtid

Crée un espace réservé GTIDs sur l'instance de base de données actuelle.

Syntaxe

```
CALL mysql.rds_group_replication_advance_gtid(
begin_id
, end_id
, server_uuid
);
```

Paramètres

begin_id

L'ID de transaction de départ à créer.

end_id

L'ID de transaction finale à créer.

begin_id

Le `group_replication_group_name` pour la transaction à créer. Le `group_replication_group_name` est spécifié UUID en tant que dans le groupe de paramètres de base de données associé à l'instance de base de données.

Notes d'utilisation

Dans un cluster actif-actif, pour qu'une instance de base de données rejoigne un groupe, toutes les GTID transactions exécutées sur la nouvelle instance de base de données doivent exister sur les autres membres du cluster. Dans des cas inhabituels, une nouvelle instance de base de données peut contenir davantage de transactions lorsque les transactions sont exécutées avant de joindre l'instance au groupe. Dans ce cas, vous ne pouvez supprimer aucune transaction existante, mais vous pouvez utiliser cette procédure pour créer l'espace réservé correspondant GTIDs sur les autres instances de base de données du groupe. Avant cela, vérifiez que les transactions n'affectent pas les données répliquées.

Lorsque vous appelez cette procédure, GTID les transactions de `server_uuid:begin_id-end_id` sont créées avec un contenu vide. Pour éviter les problèmes de réplication, n'utilisez pas cette procédure dans d'autres conditions.

> **Important**
>
> Évitez d'appeler cette procédure lorsque le cluster actif-actif fonctionne normalement. N'appelez cette procédure que si vous comprenez les conséquences possibles des transactions que vous créez. L'appel de cette procédure peut entraîner des données incohérentes.

Exemple

L'exemple suivant crée un espace réservé GTIDs sur l'instance de base de données actuelle. :

```
CALL mysql.rds_group_replication_advance_gtid(5, 6,
 '11111111-2222-3333-4444-555555555555');
```

mysql.rds_group_replication_create_user

Crée l'utilisateur de réplication `rdsgrprepladmin` pour la réplication de groupe sur l'instance de base de données.

Syntaxe

```
CALL mysql.rds_group_replication_create_user(
replication_user_password
);
```

Paramètres

`replication_user_password`

Le mot de passe de l'utilisateur de réplication`rdsgrprepladmin`.

Notes d'utilisation

- Le mot de passe de l'utilisateur de réplication `rdsgrprepladmin` doit être le même sur toutes les instances de base de données d'un cluster actif-actif.
- Le nom `rdsgrprepladmin` d'utilisateur est réservé aux connexions de réplication de groupe. Aucun autre utilisateur, y compris l'utilisateur principal, ne peut avoir ce nom d'utilisateur.

Exemple

L'exemple suivant crée l'utilisateur de réplication `rdsgrprepladmin` pour la réplication de groupe sur l'instance de base de données :

```
CALL mysql.rds_group_replication_create_user('password');
```

mysql.rds_group_replication_set_recovery_channel

Définit le `group_replication_recovery` canal d'un cluster actif-actif. La procédure utilise l'`rdsgrprepladmin`utilisateur réservé pour configurer le canal.

Syntaxe

```
CALL mysql.rds_group_replication_set_recovery_channel(
```

```
replication_user_password);
```

Paramètres

replication_user_password

Le mot de passe de l'utilisateur de réplication`rdsgrprepladmin`.

Notes d'utilisation

Le mot de passe de l'utilisateur de réplication `rdsgrprepladmin` doit être le même sur toutes les instances de base de données d'un cluster actif-actif. Un appel au permet de `mysql.rds_group_replication_create_user` spécifier le mot de passe.

Exemple

L'exemple suivant définit le `group_replication_recovery` canal d'un cluster actif-actif :

```
CALL mysql.rds_group_replication_set_recovery_channel('password');
```

mysql.rds_group_replication_start

Démarre la réplication de groupe sur l'instance de base de données actuelle.

Syntaxe

```
CALL mysql.rds_group_replication_start(
bootstrap
);
```

Paramètres

bootstrap

Valeur qui indique s'il faut initialiser un nouveau groupe ou rejoindre un groupe existant. `1`initialise un nouveau groupe avec l'instance de base de données actuelle. `0`joint l'instance de base de données actuelle à un groupe existant en se connectant aux points de terminaison définis en `group_replication_group_seeds` paramètre dans le groupe de paramètres de base de données associé à l'instance de base de données.

Exemple

L'exemple suivant initialise un nouveau groupe avec l'instance de base de données actuelle :

```
CALL mysql.rds_group_replication_start(1);
```

mysql.rds_group_replication_stop

Arrête la réplication de groupe sur l'instance de base de données actuelle.

Syntaxe

```
CALL mysql.rds_group_replication_stop();
```

Notes d’utilisation

Lorsque vous arrêtez la réplication sur une instance de base de données, cela n'affecte aucune autre instance de base de données du cluster actif-actif.

Gestion de la réplication multi-sources

Les procédures stockées suivantes permettent de configurer et de gérer les canaux de réplication sur une réplique SQL multi-source RDS for My. Pour de plus amples informations, veuillez consulter the section called “Configuration de la réplication multi-sources”.

Ces procédures stockées ne sont disponibles que RDS pour les instances My SQL DB exécutant les versions de moteur suivantes :

- Toutes les versions 8.4
- Versions mineures 8.0.35 et supérieures
- Versions mineures 5.7.44 et supérieures

Lorsque vous utilisez des procédures stockées pour gérer la réplication avec un utilisateur de réplication configuré avec`caching_sha2_passwword`, vous devez configurer TLS en spécifiant`SOURCE_SSL=1`. `caching_sha2_password`est le plugin d'authentification par défaut RDS pour My SQL 8.4.

> **Note**
>
> Bien que cette documentation fasse référence aux instances de base de données source comme RDS aux instances My SQL DB, ces procédures fonctionnent également pour les SQL instances My exécutées en externe à AmazonRDS.

Rubriques

- mysql.rds_next_source_log_for_channel
- mysql.rds_reset_external_source_for_channel
- mysql.rds_set_external_source_for_channel
- mysql.rds_set_external_source_with_auto_position_for_channel
- mysql.rds_set_external_source_with_delay_for_channel
- mysql.rds_set_source_auto_position_for_channel
- mysql.rds_set_source_delay_for_channel
- mysql.rds_skip_repl_error_for_channel
- mysql.rds_start_replication_for_channel

- mysql.rds_start_replication_until_for_channel
- mysql.rds_start_replication_until_gtid_for_channel
- mysql.rds_stop_replication_for_channel

mysql.rds_next_source_log_for_channel

Remplace la position du journal de l'instance de base de données source au début du journal binaire suivant sur l'instance de base de données source pour le canal. Utilisez cette procédure uniquement si vous recevez l'erreur d'E/S de réplication 1236 sur une réplique multi-sources.

Syntaxe

```
CALL mysql.rds_next_source_log_for_channel(
curr_master_log,
channel_name
);
```

Paramètres

curr_master_log

Index du fichier journal source actuel. Par exemple, si le fichier en cours se nomme `mysql-bin-changelog.012345`, l'index est 12345. Pour déterminer le nom du fichier journal actuel, exécutez la commande `SHOW REPLICA STATUS FOR CHANNEL '`*channel_name*`'` et affichez le champ `Source_Log_File`.

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_next_source_log_for_channel`. En cas d'erreur IO_Thread, par exemple, vous pouvez utiliser cette procédure pour ignorer tous les événements du fichier journal binaire actuel et reprendre la réplication à partir du fichier journal binaire suivant pour le canal spécifié dans. `channel_name`

Exemple

Supposons que la réplication échoue sur un canal d'une réplique multi-sources. L'exécution `SHOW REPLICA STATUS FOR CHANNEL 'channel_1'\G` sur le réplica multi-source renvoie le résultat suivant :

```
mysql> SHOW REPLICA STATUS FOR CHANNEL 'channel_1'\G
*************************** 1. row ***************************
             Replica_IO_State: Waiting for source to send event
                  Source_Host: myhost.XXXXXXXXXXXXXX.rr-rrrr-1.rds.amazonaws.com
                  Source_User: ReplicationUser
                  Source_Port: 3306
                Connect_Retry: 60
              Source_Log_File: mysql-bin-changelog.012345
          Read_Source_Log_Pos: 1219393
               Relay_Log_File: replica-relay-bin.000003
                Relay_Log_Pos: 30223388
        Relay_Source_Log_File: mysql-bin-changelog.012345
           Replica_IO_Running: No
          Replica_SQL_Running: Yes
              Replicate_Do_DB:.
              .
              .
                Last_IO_Errno: 1236
                Last_IO_Error: Got fatal error 1236 from master when reading data from
 binary log: 'Client requested master to start replication from impossible position;
 the first event 'mysql-bin-changelog.013406' at 1219393, the last event read from
 '/rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4, the last byte read from '/
rdsdbdata/log/binlog/mysql-bin-changelog.012345' at 4.'
               Last_SQL_Errno: 0
               Last_SQL_Error:
               .
               .
                 Channel_name: channel_1
              .
              .
 -- Some fields are omitted in this example output
```

Le champ `Last_IO_Errno` montre que l'instance reçoit une erreur 1236 d'I/O. Le champ `Source_Log_File` montre que le nom du fichier est `mysql-bin-changelog.012345`, ce

qui signifie que l'index du fichier journal est 12345. Pour résoudre l'erreur, vous pouvez appeler `mysql.rds_next_source_log_for_channel` avec les paramètres suivants :

```
CALL mysql.rds_next_source_log_for_channel(12345,'channel_1');
```

mysql.rds_reset_external_source_for_channel

Arrête le processus de réplication sur le canal spécifié et supprime le canal et les configurations associées de la réplique multi-source.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_reset_external_source_for_channel (channel_name);
```

Paramètres

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_reset_external_source_for_channel`. Cette procédure supprime tous les journaux de relais appartenant au canal à supprimer.

mysql.rds_set_external_source_for_channel

Configure un canal de réplication sur une instance RDS pour My SQL DB afin de répliquer les données depuis une autre instance RDS pour My SQL DB.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

> **Note**
>
> Vous pouvez plutôt utiliser la procédure the section called “mysql.rds_set_external_source_with_delay_for_channel” stockée pour configurer ce canal avec une réplication différée.

Syntaxe

```
CALL mysql.rds_set_external_source_for_channel (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
  , channel_name
);
```

Paramètres

`host_name`

Le nom d'hôte ou l'adresse IP de l'instance de base de données RDS for My SQL source.

host_port

Port utilisé par l'instance de base de données RDS for My SQL source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'ID `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur disposant de l'instance de base de données RDS for My SQL source. Nous vous recommandons de fournir un compte utilisé uniquement pour la réplication avec l'instance de base de données source.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Nom du journal binaire de l'instance de base de données source qui contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier binlog en l'exécutant `SHOW BINARY LOG STATUS` sur l'instance de base de données source.

> **Note**
>
> Les versions précédentes de My SQL utilisaient à la `SHOW MASTER STATUS` place de`SHOW BINARY LOG STATUS`. Si vous utilisez une SQL version My antérieure à la version 8.4, utilisez`SHOW MASTER STATUS`.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option SOURCE_SSL_VERIFY_SERVER_CERT n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

channel_name

Nom du canal de réplication. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_source_for_channel`. Cette procédure doit être exécutée sur la cible RDS de l'instance My SQL DB sur laquelle vous créez le canal de réplication.

Avant d'exécuter`mysql.rds_set_external_source_for_channel`, configurez un utilisateur de réplication sur l'instance de base de données source avec les privilèges requis pour la réplique multi-source. Pour connecter la réplique multi-source à l'instance de base de données source, vous devez spécifier `replication_user_name` et `replication_user_password` les valeurs d'un utilisateur de réplication disposant d'`REPLICATION SLAVE`autorisations sur l'instance de base de données source. `REPLICATION CLIENT`

Pour configurer un utilisateur de réplication sur l'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance de base de données source et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   > **Important**
   >
   > Pour des raisons de sécurité, il est recommandé de spécifier un mot de passe autre que la valeur d'espace réservé indiquée dans les exemples suivants.

   ```
   CREATE USER 'repl_user'@'example.com' IDENTIFIED BY 'password';
   ```

2. Sur l'instance de base de données source, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur « repl_user » de votre domaine.

```
GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'example.com';
```

Pour utiliser la réplication chiffrée, configurez l'instance de base de données source pour qu'elle utilise SSL des connexions.

Après avoir appelé `mysql.rds_set_external_source_for_channel` pour configurer ce canal de réplication, vous pouvez faire appel mysql.rds_start_replication_for_channel à la réplique pour démarrer le processus de réplication sur le canal. Vous pouvez appeler the section called “mysql.rds_reset_external_source_for_channel” pour arrêter la réplication sur le canal et supprimer la configuration du canal de la réplique.

Lorsque vous appelez`mysql.rds_set_external_source_for_channel`, Amazon RDS enregistre l'heure, l'utilisateur et une action de `set channel source` dans le `mysql.rds_history` tableau sans informations spécifiques à la chaîne, et dans le `mysql.rds_replication_status` tableau, avec le nom de la chaîne. Ces informations sont enregistrées uniquement à des fins d'utilisation interne et de surveillance. Pour enregistrer l'appel de procédure complet à des fins d'audit, pensez à activer les journaux d'audit ou les journaux généraux, en fonction des exigences spécifiques de votre application.

Exemples

Lorsqu'il est exécuté sur une instance RDS for My SQL DB, l'exemple suivant configure un canal de réplication nommé `channel_1` sur cette instance de base de données pour répliquer les données à partir de la source spécifiée par l'hôte `sourcedb.example.com` et le port. `3306`

```
call mysql.rds_set_external_source_for_channel(
  'sourcedb.example.com',
  3306,
  'repl_user',
  'password',
  'mysql-bin-changelog.0777',
  120,
  0,
  'channel_1');
```

mysql.rds_set_external_source_with_auto_position_for_channel

Configure un canal de réplication sur une instance RDS for My SQL DB avec un délai de réplication facultatif. La réplication est basée sur des identifiants de transaction globaux (GTIDs).

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source_with_auto_position_for_channel (
  host_name
  , host_port
  , replication_user_name
  , replication_user_password
  , ssl_encryption
  , delay
  , channel_name
);
```

Paramètres

`host_name`

Le nom d'hôte ou l'adresse IP de l'instance de base de données RDS for My SQL source.

`host_port`

Port utilisé par l'instance de base de données RDS for My SQL source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

`replication_user_name`

L'ID `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur disposant de l'instance de base de données RDS for My SQL source. Nous vous recommandons de fournir un compte utilisé uniquement pour la réplication avec l'instance de base de données source.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

ssl_encryption

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `SOURCE_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

delay

Le nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

channel_name

Nom du canal de réplication. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_source_with_auto_position_for_channel`. Cette procédure doit être exécutée sur la cible RDS de l'instance My SQL DB sur laquelle vous créez le canal de réplication.

Avant d'exécuter`rds_set_external_source_with_auto_position_for_channel`, configurez un utilisateur de réplication sur l'instance de base de données source avec les privilèges requis pour la réplique multi-source. Pour connecter la réplique multi-source à l'instance de base de données source, vous devez spécifier `replication_user_name` et `replication_user_password` les

valeurs d'un utilisateur de réplication disposant d'`REPLICATION SLAVE`autorisations sur l'instance de base de données source. `REPLICATION CLIENT`

Pour configurer un utilisateur de réplication sur l'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance de base de données source et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   > **Important**
   >
   > Pour des raisons de sécurité, il est recommandé de spécifier un mot de passe autre que la valeur d'espace réservé indiquée dans les exemples suivants.

   ```
   CREATE USER 'repl_user'@'example.com' IDENTIFIED BY 'password';
   ```

2. Sur l'instance de base de données source, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur « repl_user » de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'example.com';
   ```

Pour utiliser la réplication chiffrée, configurez l'instance de base de données source pour qu'elle utilise SSL des connexions.

Avant d'appeler`mysql.rds_set_external_source_with_auto_position_for_channel`, assurez-vous d'appeler the section called “mysql.rds_set_external_source_gtid_purged” pour définir la variable `gtid_purged` système avec une GTID plage spécifiée à partir d'une source externe.

Après avoir appelé `mysql.rds_set_external_source_with_auto_position_for_channel` pour configurer une instance Amazon RDS DB en tant que réplique en lecture sur un canal spécifique, vous pouvez faire appel the section called “mysql.rds_start_replication_for_channel” à la réplique en lecture pour démarrer le processus de réplication sur ce canal.

Après avoir appelé `mysql.rds_set_external_source_with_auto_position_for_channel` pour configurer ce canal de réplication, vous pouvez faire appel mysql.rds_start_replication_for_channel à la réplique pour démarrer le processus de réplication sur le

canal. Vous pouvez appeler the section called “mysql.rds_reset_external_source_for_channel” pour arrêter la réplication sur le canal et supprimer la configuration du canal de la réplique.

Exemples

Lorsqu'il est exécuté sur une instance RDS for My SQL DB, l'exemple suivant configure un canal de réplication nommé `channel_1` sur cette instance de base de données pour répliquer les données à partir de la source spécifiée par l'hôte `sourcedb.example.com` et le port. `3306` Il définit le délai de réplication minimum à une heure (3 600 secondes). Cela signifie qu'une modification par rapport à la source RDS pour l'instance My SQL DB n'est pas appliquée à la réplique multi-source pendant au moins une heure.

```
call mysql.rds_set_external_source_with_auto_position_for_channel(
  'sourcedb.example.com',
  3306,
  'repl_user',
  'password',
  1,
  3600,
  'channel_1');
```

mysql.rds_set_external_source_with_delay_for_channel

Configure un canal de réplication sur une instance RDS for My SQL DB avec un délai de réplication spécifié.

> **Important**
>
> Pour exécuter cette procédure, `autocommit` doit être activé. Pour l'activer, définissez le paramètre `autocommit` sur 1. Pour de plus amples informations sur la modification des paramètres d'instance, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Syntaxe

```
CALL mysql.rds_set_external_source_with_delay_for_channel (
  host_name
```

```
  , host_port
  , replication_user_name
  , replication_user_password
  , mysql_binary_log_file_name
  , mysql_binary_log_file_location
  , ssl_encryption
  , delay
  , channel_name
);
```

Paramètres

host_name

Le nom d'hôte ou l'adresse IP de l'instance de base de données RDS for My SQL source.

host_port

Port utilisé par l'instance de base de données RDS for My SQL source. Si votre configuration réseau inclut une réplication de port Secure Shell (SSH) qui convertit le numéro de port, spécifiez le numéro de port exposé parSSH.

replication_user_name

L'ID `REPLICATION CLIENT` et les `REPLICATION SLAVE` autorisations d'un utilisateur disposant de l'instance de base de données RDS for My SQL source. Nous vous recommandons de fournir un compte utilisé uniquement pour la réplication avec l'instance de base de données source.

replication_user_password

Mot de passe de l'ID utilisateur spécifié dans `replication_user_name`.

mysql_binary_log_file_name

Le nom du journal binaire sur l'instance de base de données source contient les informations de réplication.

mysql_binary_log_file_location

Emplacement dans le journal binaire `mysql_binary_log_file_name` à partir duquel la réplication commence à lire les informations de réplication.

Vous pouvez déterminer le nom et l'emplacement du fichier journal binaire en exécutant `SHOW BINARY LOG STATUS` sur l'instance de base de données source.

> **Note**
>
> Les versions précédentes de My SQL utilisaient à la `SHOW MASTER STATUS` place de`SHOW BINARY LOG STATUS`. Si vous utilisez une SQL version My antérieure à la version 8.4, utilisez`SHOW MASTER STATUS`.

`ssl_encryption`

Valeur qui indique si le chiffrement Secure Socket Layer (SSL) est utilisé sur la connexion de réplication. 1 indique d'utiliser le SSL chiffrement, 0 indique de ne pas utiliser de chiffrement. La valeur par défaut est 0.

> **Note**
>
> L'option `SOURCE_SSL_VERIFY_SERVER_CERT` n'est pas prise en charge. Cette option est définie sur 0, ce qui signifie que la connexion est chiffrée, mais que les certificats ne sont pas vérifiés.

`delay`

Le nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

`channel_name`

Nom du canal de réplication. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d’utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_external_source_with_delay_for_channel`. Cette procédure doit être exécutée sur la cible RDS de l'instance My SQL DB sur laquelle vous créez le canal de réplication.

Avant d'exécuter`mysql.rds_set_external_source_with_delay_for_channel`, configurez un utilisateur de réplication sur l'instance de base de données source avec les privilèges requis pour

la réplique multi-source. Pour connecter la réplique multi-source à l'instance de base de données source, vous devez spécifier `replication_user_name` et `replication_user_password` les valeurs d'un utilisateur de réplication disposant d'`REPLICATION SLAVE`autorisations sur l'instance de base de données source. `REPLICATION CLIENT`

Pour configurer un utilisateur de réplication sur l'instance de base de données source

1. À l'aide du SQL client My de votre choix, connectez-vous à l'instance de base de données source et créez un compte utilisateur à utiliser pour la réplication. Voici un exemple.

   > **Important**
   >
   > Pour des raisons de sécurité, il est recommandé de spécifier un mot de passe autre que la valeur d'espace réservé indiquée dans les exemples suivants.

   ```
   CREATE USER 'repl_user'@'example.com' IDENTIFIED BY 'password';
   ```

2. Sur l'instance de base de données source, accordez `REPLICATION CLIENT` des `REPLICATION SLAVE` privilèges à votre utilisateur de réplication. L'exemple suivant accorde les privilèges `REPLICATION CLIENT` et `REPLICATION SLAVE` sur toutes les bases de données pour l'utilisateur « repl_user » de votre domaine.

   ```
   GRANT REPLICATION CLIENT, REPLICATION SLAVE ON *.* TO 'repl_user'@'example.com';
   ```

Pour utiliser la réplication chiffrée, configurez l'instance de base de données source pour qu'elle utilise SSL des connexions.

Après avoir appelé `mysql.rds_set_external_source_with_delay_for_channel` pour configurer ce canal de réplication, vous pouvez faire appel mysql.rds_start_replication_for_channel à la réplique pour démarrer le processus de réplication sur le canal. Vous pouvez appeler the section called “mysql.rds_reset_external_source_for_channel” pour arrêter la réplication sur le canal et supprimer la configuration du canal de la réplique.

Lorsque vous appelez`mysql.rds_set_external_source_with_delay_for_channel`, Amazon RDS enregistre l'heure, l'utilisateur et une action de `set channel source` dans le `mysql.rds_history` tableau sans informations spécifiques à la chaîne, et dans le `mysql.rds_replication_status` tableau, avec le nom de la chaîne. Ces informations sont

enregistrées uniquement à des fins d'utilisation interne et de surveillance. Pour enregistrer l'appel de procédure complet à des fins d'audit, pensez à activer les journaux d'audit ou les journaux généraux, en fonction des exigences spécifiques de votre application.

Exemples

Lorsqu'il est exécuté sur une instance RDS for My SQL DB, l'exemple suivant configure un canal de réplication nommé `channel_1` sur cette instance de base de données pour répliquer les données à partir de la source spécifiée par l'hôte `sourcedb.example.com` et le port. `3306` Il définit le délai de réplication minimum à une heure (3 600 secondes). Cela signifie qu'une modification par rapport à la source RDS pour l'instance My SQL DB n'est pas appliquée à la réplique multi-source pendant au moins une heure.

```
call mysql.rds_set_external_source_with_delay_for_channel(
  'sourcedb.example.com',
  3306,
  'repl_user',
  'password',
  'mysql-bin-changelog.000777',
  120,
  1,
  3600,
  'channel_1');
```

mysql.rds_set_source_auto_position_for_channel

Définit le mode de réplication pour le canal spécifié en fonction des positions du fichier journal binaire ou des identificateurs de transaction globaux (GTIDs).

Syntaxe

```
CALL mysql.rds_set_source_auto_position_for_channel (
auto_position_mode
 , channel_name
);
```

Paramètres

auto_position_mode

Une valeur qui indique s'il faut utiliser la réplication de position du fichier journal ou la réplication GTID basée sur la réplication :

- `0` – Utiliser la méthode de réplication basée sur la position du fichier journal binaire. La valeur par défaut est `0`.
- 1— Utilisez la méthode GTID de réplication basée.

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_source_auto_position_for_channel`. Cette procédure redémarre la réplication sur le canal spécifié pour appliquer le mode de positionnement automatique spécifié.

Exemples

L'exemple suivant définit le mode de positionnement automatique pour channel_1 afin d'utiliser la méthode de réplication GTID basée.

```
call mysql.rds_set_source_auto_position_for_channel(1,'channel_1');
```

mysql.rds_set_source_delay_for_channel

Définit le nombre minimal de secondes pour retarder la réplication de l'instance de base de données source vers la réplique multi-source pour le canal spécifié.

Syntaxe

```
CALL mysql.rds_set_source_delay_for_channel(delay, channel_name);
```

Paramètres

delay

Le nombre minimum de secondes pour retarder la réplication à partir de l'instance de base de données source.

La limite de ce paramètre est une journée (soit 86 400 secondes).

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_set_source_delay_for_channel`. Pour utiliser cette procédure, appelez d'abord `mysql.rds_stop_replication_for_channel` pour arrêter la réplication. Appelez ensuite cette procédure pour définir la valeur du délai de réplication. Lorsque le délai est défini, appelez `mysql.rds_start_replication_for_channel` pour redémarrer la réplication.

Exemples

L'exemple suivant définit le délai de réplication depuis l'instance de base `channel_1` de données source sur le réplica multi-source pendant au moins une heure (3 600 secondes).

```
CALL mysql.rds_set_source_delay_for_channel(3600,'channel_1');
```

mysql.rds_skip_repl_error_for_channel

Ignore un événement de journal binaire et supprime une erreur de réplication sur une réplique multi-source My SQL DB pour le canal spécifié.

Syntaxe

```
CALL mysql.rds_skip_repl_error_for_channel(channel_name);
```

Paramètres

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_skip_repl_error_for_channel` sur un réplica en lecture. Vous pouvez utiliser cette procédure de la même manière `mysql.rds_skip_repl_error` que pour ignorer une erreur lors de la lecture d'une réplique. Pour de plus amples informations, veuillez consulter Appel de la procédure mysql.rds_skip_repl_error.

> **Note**
>
> Pour éviter les erreurs liées à la réplication GTID basée, nous vous recommandons d'utiliser cette procédure à la the section called "mysql.rds_skip_transaction_with_gtid" place.

Pour déterminer s'il existe des erreurs, exécutez la SQL `SHOW REPLICA STATUS FOR CHANNEL 'channel_name'\G` commande My. Si une erreur de réplication n'est pas critique, vous pouvez exécuter `mysql.rds_skip_repl_error_for_channel` pour ignorer l'erreur. En cas d'erreurs multiples, `mysql.rds_skip_repl_error_for_channel` supprime la première erreur sur le canal de réplication spécifié, puis avertit de la présence d'autres erreurs. Vous pouvez alors utiliser `SHOW REPLICA STATUS FOR CHANNEL 'channel_name'\G` pour déterminer l'action appropriée pour l'erreur suivante. Pour plus d'informations sur les valeurs renvoyées, voir la SHOWREPLICASTATUSdéclaration dans la section Ma SQL documentation.

mysql.rds_start_replication_for_channel

Lance la réplication depuis une instance RDS for My SQL DB vers une réplique multi-source sur le canal spécifié.

> **Note**
>
> Vous pouvez utiliser la procédure mysql.rds_start_replication_until_gtid_for_channel stockée mysql.rds_start_replication_until_for_channel ou pour lancer la réplication à partir d'une

instance RDS for My SQL DB et arrêter la réplication à l'emplacement du fichier journal binaire spécifié.

Syntaxe

```
CALL mysql.rds_start_replication_for_channel(channel_name);
```

Paramètres

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication_for_channel`. Après avoir importé les données depuis la source RDS pour l'instance My SQL DB, exécutez cette commande sur la réplique multi-source pour démarrer la réplication sur le canal spécifié.

Exemples

L'exemple suivant démarre la réplication sur `channel_1` le réplica multi-source.

```
CALL mysql.rds_start_replication_for_channel('channel_1');
```

mysql.rds_start_replication_until_for_channel

Lance la réplication à partir d'une instance RDS for My SQL DB sur le canal spécifié et arrête la réplication à l'emplacement du fichier journal binaire spécifié.

Syntaxe

```
CALL mysql.rds_start_replication_until_for_channel (
replication_log_file
  , replication_stop_point
```

```
  , channel_name
);
```

Paramètres

replication_log_file

Le nom du journal binaire sur l'instance de base de données source contient les informations de réplication.

replication_stop_point

Position dans le journal binaire `replication_log_file` à laquelle la réplication s'arrêtera.

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication_until_for_channel`. Avec cette procédure, la réplication démarre puis s'arrête lorsque la position spécifiée du fichier binlog est atteinte. Cette procédure arrête à la fois le `SQL_THREAD` et`IO_THREAD`.

Le nom de fichier spécifié pour le `replication_log_file` paramètre doit correspondre au nom du fichier binlog de l'instance de base de données source.

Lorsque le `replication_stop_point` paramètre indique un emplacement d'arrêt antérieur, la réplication est immédiatement arrêtée.

Exemples

L'exemple suivant lance la réplication et réplique les modifications jusqu'à ce qu'elles atteignent leur emplacement `120` dans le `mysql-bin-changelog.000777` fichier journal binaire. `channel_1`

```
call mysql.rds_start_replication_until_for_channel(
  'mysql-bin-changelog.000777',
  120,
  'channel_1'
```

```
);
```

mysql.rds_start_replication_until_gtid_for_channel

Lance la réplication sur le canal spécifié à partir d'une instance RDS for My SQL DB et arrête la réplication à l'identifiant de transaction global spécifié (GTID).

Syntaxe

```
CALL mysql.rds_start_replication_until_gtid_for_channel(gtid,channel_name);
```

Paramètres

gtid

GTIDAprès quoi arrêter la réplication.

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication_until_gtid_for_channel`. La procédure démarre la réplication sur le canal spécifié et applique toutes les modifications jusqu'à la GTID valeur spécifiée. Ensuite, il arrête la réplication sur le canal.

Lorsque le paramètre `gtid` spécifie une transaction ayant déjà été exécutée par le réplica, la réplication est immédiatement arrêtée.

Avant d'exécuter cette procédure, vous devez désactiver la réplication multithread en définissant la valeur de `replica_parallel_workers` ou `slave_parallel_workers` sur. `0`

Exemples

L'exemple suivant lance la réplication et réplique les modifications jusqu'à ce qu'elles soient atteintes. `channel_1` GTID `3E11FA47-71CA-11E1-9E33-C80AA9429562:23`

```
call mysql.rds_start_replication_until_gtid_for_channel('3E11FA47-71CA-11E1-9E33-
C80AA9429562:23','channel_1');
```

mysql.rds_stop_replication_for_channel

Arrête la réplication depuis une instance My SQL DB sur le canal spécifié.

Syntaxe

```
CALL mysql.rds_stop_replication_for_channel(channel_name);
```

Paramètres

channel_name

Nom du canal de réplication sur le réplica multi-source. Chaque canal de réplication reçoit les événements du journal binaire d'une source unique RDS pour l'instance My SQL DB exécutée sur un hôte et un port spécifiques.

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_stop_replication_for_channel`.

Exemples

L'exemple suivant arrête la réplication sur `channel_1` le réplica multi-source.

```
CALL mysql.rds_stop_replication_for_channel('channel_1');
```

Répliquer des transactions à l'aide de GTIDs

Les procédures stockées suivantes contrôlent la manière dont les transactions sont répliquées à l'aide des identificateurs de transaction globaux (GTIDs) avec RDS for My. SQL Pour plus d'informations sur la réplication basée sur GTIDs with RDS for MySQL, consultezUtilisation de GTID la réplication basée.

Lorsque vous utilisez des procédures stockées pour gérer la réplication avec un utilisateur de réplication configuré avec`caching_sha2_password`, vous devez configurer TLS en spécifiant`SOURCE_SSL=1`. `caching_sha2_password`est le plugin d'authentification par défaut RDS pour My SQL 8.4.

Rubriques

•

•

Ignore la réplication d'une transaction avec l'identifiant de transaction global spécifié (GTID) sur une instance My SQL DB.

Vous pouvez utiliser cette procédure pour la reprise après sinistre lorsqu'une GTID transaction spécifique est connue comme étant à l'origine d'un problème. Utilisez cette procédure stockée pour ignorer la transaction problématique. Les transactions problématiques sont par exemple celles qui désactivent la réplication, suppriment des données importantes ou entraînent l'indisponibilité de l'instance de base de données.

Syntaxe

```
CALL mysql.rds_skip_transaction_with_gtid (
gtid_to_skip
);
```

Paramètres

`gtid_to_skip`

La transaction GTID de réplication à ignorer.

Notes d’utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_skip_transaction_with_gtid`.

Cette procédure est prise en charge RDS pour toutes les versions My SQL 5.7, toutes les RDS versions My SQL 8.0 et toutes RDS les versions My SQL 8.4.

Exemples

L'exemple suivant ignore la réplication de la transaction avec le GTID`3E11FA47-71CA-11E1-9E33-C80AA9429562:23`.

```
CALL mysql.rds_skip_transaction_with_gtid('3E11FA47-71CA-11E1-9E33-C80AA9429562:23');
```

Lance la réplication à partir RDSd'un cluster Aurora My SQL DB d'instance et arrête la réplication immédiatement après l'identifiant de transaction global spécifié (GTID).

Syntaxe

```
CALL mysql.rds_start_replication_until_gtid(gtid);
```

Paramètres

`gtid`

GTIDAprès quoi la réplication doit s'arrêter.

Notes d’utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_start_replication_until_gtid`.

Cette procédure est prise en charge RDS pour toutes les versions My SQL 5.7, toutes les RDS versions My SQL 8.0 et toutes RDS les versions My SQL 8.4.

Vous pouvez utiliser cette procédure avec la réplication retardée pour la reprise après sinistre. Si vous avez configuré la réplication retardée, vous pouvez utiliser cette procédure pour restaurer par progression les modifications dans un réplica en lecture retardé au moment précédant un sinistre. Une fois que cette procédure a arrêté la réplication, vous pouvez promouvoir le réplica en lecture

pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Vous pouvez configurer la réplication retardée en utilisant les procédures stockées suivantes :

- mysql.rds_set_configuration
- mysql.rds_set_external_master_with_delay (pour MariaDB et pour My major versions 8.0 et antérieures) RDS RDS SQL
- mysql.rds_set_external_source_with_delay (pour mes versions majeures 8.4 et supérieures) RDS SQL
- mysql.rds_set_source_delay

Lorsque le paramètre `gtid` spécifie une transaction ayant déjà été exécutée par le réplica, la réplication est immédiatement arrêtée.

Exemples

L'exemple suivant lance la réplication et réplique les modifications jusqu'à ce qu'elles soient atteintes. GTID `3E11FA47-71CA-11E1-9E33-C80AA9429562:23`

```
call mysql.rds_start_replication_until_gtid('3E11FA47-71CA-11E1-9E33-C80AA9429562:23');
```

Rotation des journaux de requêtes

Les procédures stockées suivantes font pivoter Mes SQL journaux vers des tables de sauvegarde. Pour de plus amples informations, veuillez consulter Fichiers journaux de base de données MySQL.

Rubriques

- mysql.rds_rotate_general_log
- mysql.rds_rotate_slow_log

mysql.rds_rotate_general_log

Convertit la table `mysql.general_log` en table de sauvegarde.

Syntaxe

```
CALL mysql.rds_rotate_general_log;
```

Notes d'utilisation

Vous pouvez convertir la table `mysql.general_log` en table de sauvegarde en appelant la procédure `mysql.rds_rotate_general_log`. Lors de la rotation des tables de journaux, la table de journal actuelle est copiée vers une table de journal de sauvegarde et les entrées de la table de journal actuelle sont supprimées. Si la table du journal de sauvegarde existe déjà, elle est supprimée avant que la table du journal active ne soit copiée dans la sauvegarde. Si besoin, vous pouvez interroger la table de journal de sauvegarde. La table de journal de sauvegarde de la table `mysql.general_log` est nommée `mysql.general_log_backup`.

Vous ne pouvez exécuter cette procédure que lorsque le paramètre `log_output` est défini sur `TABLE`.

mysql.rds_rotate_slow_log

Convertit la table `mysql.slow_log` en table de sauvegarde.

Syntaxe

```
CALL mysql.rds_rotate_slow_log;
```

Notes d'utilisation

Vous pouvez convertir la table `mysql.slow_log` en table de sauvegarde en appelant la procédure `mysql.rds_rotate_slow_log`. Lors de la rotation des tables de journaux, la table de journal actuelle est copiée vers une table de journal de sauvegarde et les entrées de la table de journal actuelle sont supprimées. Si la table du journal de sauvegarde existe déjà, elle est supprimée avant que la table du journal active ne soit copiée dans la sauvegarde.

Si besoin, vous pouvez interroger la table de journal de sauvegarde. La table de journal de sauvegarde de la table `mysql.slow_log` est nommée `mysql.slow_log_backup`.

Configuration et affichage de la configuration du journal binaire

Les procédures stockées suivantes définissent et affichent les paramètres de configuration, tels que la conservation des fichiers journaux binaires.

Rubriques

- mysql.rds_set_configuration
- mysql.rds_show_configuration

mysql.rds_set_configuration

Spécifie le nombre d'heures pendant lequel les journaux binaires doivent être conservés ou le nombre de secondes pendant lequel retarder la réplication.

Syntaxe

```
CALL mysql.rds_set_configuration(name,value);
```

Paramètres

name

Nom du paramètre de configuration à définir.

value

Valeur du paramètre de configuration.

Notes d'utilisation

La procédure `mysql.rds_set_configuration` prend en charge des paramètres de configuration suivants :

- nombre d'heures de conservation du journal binaire
- retard à la source
- target delay

Les paramètres de configuration sont stockés de manière permanente et survivent à tout redémarrage ou basculement d'une instance de base de données.

nombre d'heures de conservation du journal binaire

Le paramètre `binlog retention hours` est utilisé pour spécifier le nombre d'heures de rétention des fichiers journaux binaires. Amazon purge RDS normalement un journal binaire dès que possible, mais le journal binaire peut toujours être nécessaire pour la réplication avec une base de SQL données My database externe àRDS.

La valeur par défaut de `binlog retention hours` est `NULL`. Pour RDS for MySQL, `NULL` cela signifie que les journaux binaires ne sont pas conservés (0 heure).

Pour spécifier le nombre d'heures pendant lesquelles conserver les journaux binaires sur une instance de base de données, utilisez la procédure stockée `mysql.rds_set_configuration` et spécifiez une période suffisamment longue pour que la réplication se produise, comme illustré dans l'exemple suivant.

```
call mysql.rds_set_configuration('binlog retention hours', 24);
```

> **Note**
>
> Vous ne pouvez pas utiliser la valeur `0` pour `binlog retention hours`.

Pour les instances My SQL DB, la `binlog retention hours` valeur maximale est de 168 (7 jours).

Après avoir défini la période de rétention, surveillez l'utilisation du stockage de l'instance de base de données afin de garantir que les journaux binaires conservés n'utilisent pas un espace de stockage trop grand.

Pour les déploiements de clusters de base de données multi-AZ, vous pouvez uniquement configurer la rétention des journaux binaires à partir de l'instance de base de données du rédacteur, et le paramètre est propagé à toutes les instances de base de données du lecteur de manière asynchrone. Si les journaux binaires du cluster de base de données dépassent la moitié de l'espace de stockage local total, Amazon déplace RDS automatiquement les journaux périmés vers le EBS volume. Cependant, les journaux les plus récents restent dans le stockage local. Ils sont donc susceptibles d'être perdus en cas de panne nécessitant le remplacement d'un hôte, ou si vous augmentez ou diminuez la taille de la base de données.

retard à la source

Utilisez le paramètre `source delay` dans un réplica en lecture pour spécifier le nombre de secondes dont il faut retarder la réplication à partir du réplica en lecture vers son instance de base de données source. Amazon réplique RDS normalement les modifications dès que possible, mais vous souhaiterez peut-être que certains environnements retardent la réplication. Par exemple, lorsque la réplication est retardée, vous pouvez restaurer par progression un réplica en lecture retardé au moment précédant un sinistre. Si une table est supprimée par mégarde, vous pouvez utiliser la réplication retardée pour la récupérer rapidement. La valeur par défaut de `target delay` est `0` (ne pas retarder la réplication).

Lorsque vous utilisez ce paramètre, il s'exécute mysql.rds_set_source_delay et applique la valeur d'entrée CHANGE primaire TO MASTER _ DELAY =. En cas de succès, la procédure enregistre le paramètre `source delay` dans la table `mysql.rds_configuration`.

Pour spécifier le nombre de secondes pendant lequel Amazon doit RDS retarder la réplication vers une instance de base de données source, utilisez la procédure `mysql.rds_set_configuration` stockée et spécifiez le nombre de secondes pendant lesquelles Amazon retarde la réplication. Dans l'exemple suivant, la réplication est retardée d'au moins une heure (3 600 secondes).

```
call mysql.rds_set_configuration('source delay', 3600);
```

La procédure exécute ensuite `mysql.rds_set_source_delay(3600)`.

La limite du paramètre `source delay` est une journée (soit 86 400 secondes).

target delay

Utilisez le `target delay` paramètre pour spécifier le nombre de secondes nécessaires pour retarder la réplication entre une instance de base de données et toute future réplique de RDS lecture gérée créée à partir de cette instance. Ce paramètre est ignoré pour les répliques de non-RDS-managed lecture. Amazon réplique RDS normalement les modifications dès que possible, mais vous souhaiterez peut-être que certains environnements retardent la réplication. Par exemple, lorsque la réplication est retardée, vous pouvez restaurer par progression un réplica en lecture retardé au moment précédant un sinistre. Si une table est supprimée par mégarde, vous pouvez utiliser la réplication retardée pour la récupérer rapidement. La valeur par défaut de `target delay` est `0` (ne pas retarder la réplication).

Pour la reprise après sinistre, vous pouvez utiliser ce paramètre de configuration avec la procédure stockée ou . Pour restaurer par progression les modifications dans un réplica

en lecture retardé au moment précédant un sinistre, vous pouvez exécuter la procédure `mysql.rds_set_configuration` avec ce paramètre défini. Une fois que la procédure `mysql.rds_start_replication_until` ou `mysql.rds_start_replication_until_gtid` a arrêté la réplication, vous pouvez promouvoir le réplica en lecture pour qu'il devienne la nouvelle instance de base de données principale, en utilisant les instructions figurant dans Promotion d'un réplica en lecture en instance de bases de données autonome.

Pour utiliser `mysql.rds_rds_start_replication_until_gtid` cette procédure, la réplication GTID basée doit être activée. Pour ignorer une GTID transaction spécifique connue pour provoquer un sinistre, vous pouvez utiliser la procédure stockée. Pour plus d'informations sur l'utilisation de la réplication GTID basée, consultezUtilisation de GTID la réplication basée.

Pour spécifier le nombre de secondes pendant lequel Amazon doit RDS retarder la réplication vers une réplique en lecture, utilisez la procédure `mysql.rds_set_configuration` stockée et spécifiez le nombre de secondes pendant lequel Amazon retarde la réplication. L'exemple suivant spécifie que la réplication est retardée d'au moins une heure (3 600 secondes).

```
call mysql.rds_set_configuration('target delay', 3600);
```

La limite du paramètre `target delay` est une journée (soit 86 400 secondes).

mysql.rds_show_configuration

Nombre d'heures pendant lequel les journaux binaires sont conservés.

Syntaxe

```
CALL mysql.rds_show_configuration;
```

Notes d'utilisation

Pour vérifier le nombre d'heures pendant lesquelles Amazon RDS conserve les journaux binaires, utilisez la procédure `mysql.rds_show_configuration` stockée.

Exemples

L'exemple suivant affiche la période de rétention :

```
call mysql.rds_show_configuration;
               name                          value     description
```

```
                binlog retention hours       24        binlog retention hours specifies
the duration in hours before binary logs are automatically deleted.
```

Réchauffement du cache InnoDB

Les procédures stockées suivantes enregistrent, chargent ou annulent le chargement du pool de mémoires tampons InnoDB sur les instances de base de données RDS pour MySQL. Pour de plus amples informations, veuillez consulter Préparation du cache InnoDB pour MySQL sur Amazon RDS.

Rubriques

- mysql.rds_innodb_buffer_pool_dump_now
- mysql.rds_innodb_buffer_pool_load_abort
- mysql.rds_innodb_buffer_pool_load_now

mysql.rds_innodb_buffer_pool_dump_now

Vide l'état actuel du pool de tampons sur le disque.

Syntaxe

```
CALL mysql.rds_innodb_buffer_pool_dump_now();
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_innodb_buffer_pool_dump_now`.

mysql.rds_innodb_buffer_pool_load_abort

Annule un chargement en cours de l'état du groupe de tampons enregistré.

Syntaxe

```
CALL mysql.rds_innodb_buffer_pool_load_abort();
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_innodb_buffer_pool_load_abort`.

mysql.rds_innodb_buffer_pool_load_now

Charge l'état enregistré du pool de tampons à partir du disque.

Syntaxe

```
CALL mysql.rds_innodb_buffer_pool_load_now();
```

Notes d'utilisation

L'utilisateur principal doit exécuter la procédure `mysql.rds_innodb_buffer_pool_load_now`.

Amazon RDS for Oracle

Amazon RDS prend en charge les instances de base de données qui exécutent les versions et éditions d'Oracle Database suivantes :

- Oracle Database 21c (21.0.0.0)
- Oracle Database 19c (19.0.0.0)

> **Note**
>
> Oracle Database 11g, Oracle Database 12c et Oracle Database 18c sont des versions héritées qui ne sont plus prises en charge dans Amazon RDS.

Avant de créer une instance de base de données, suivez la procédure de la section Configuration de votre environnement Amazon RDS du présent guide. Lorsque vous créez une instance de base de données à l'aide de votre compte principal, le compte bénéficie des privilèges DBA, avec certaines limitations. Utilisez ce compte pour des tâches administratives telles que la création de comptes de base de données supplémentaires. Vous ne pouvez pas utiliser SYS, SYSTEM ou d'autres comptes administratifs fournis par Oracle.

Vous pouvez créer ce qui suit :

- Instances DB
- Instantanés de base de données
- Point-in-time restaure
- Sauvegardes automatiques
- Sauvegardes manuelles

Vous pouvez utiliser des instances de base de données exécutant Oracle Database dans un VPC. Vous pouvez également ajouter des fonctionnalités à votre instance de base de données en activant diverses options, telles qu'Oracle Spatial ou Oracle Statspack. Amazon RDS prend en charge les déploiements Multi-AZ pour Oracle comme solution de basculement haute disponibilité.

> **Important**
>
> Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui nécessitent des privilèges avancés. Vous pouvez accéder à votre base de données en utilisant des client SQL standard tels qu'Oracle SQL*Plus. Toutefois, vous ne pouvez pas accéder directement à l'hôte en utilisant Telnet ou Secure Shell (SSH).

Rubriques

- Présentation d'Oracle sur Amazon RDS
- Connexion à votre instance de base de données Oracle
- Sécurisation des connexions d'instance de base de données Oracle
- Utilisation des CDB dans RDS for Oracle
- Administration de votre instance de base de données RDS for Oracle
- Configuration des fonctions avancées RDS for Oracle
- Importation de données dans Oracle sur Amazon RDS
- Utilisation de réplicas en lecture pour Amazon RDS for Oracle
- Ajout d'options aux instances de base de données Oracle
- Mise à niveau du moteur de base de données RDS for Oracle
- Utilisation d'un logiciel tiers avec votre instance de base de données RDS for Oracle
- Notes de mise à jour pour le moteur de base de données Oracle

Présentation d'Oracle sur Amazon RDS

Vous pouvez lire les sections suivantes pour obtenir une vue d'ensemble de RDS for Oracle.

Rubriques

- Fonctions RDS for Oracle
- Versions RDS for Oracle
- RDSpour les options de licence Oracle
- Utilisateurs et privilèges RDS for Oracle
- RDS pour les classes d'instance de base de données Oracle

- Architecture de base de données RDS for Oracle
- Paramètres d'initialisation de RDS pour Oracle
- Jeux de caractères RDS for Oracle
- Limitations RDS for Oracle

Fonctions RDS for Oracle

Amazon RDS for Oracle prend en charge la plupart des fonctionnalités et des capacités d'Oracle Database. Certaines fonctions peuvent avoir une prise en charge limitée ou des privilèges restreints. Certaines fonctions sont disponibles uniquement dans Enterprise Edition et certaines fonctions nécessitent des licences supplémentaires. Pour plus d'informations sur les fonctions Oracle Database pour des versions d'Oracle Database spécifiques, consultez le document Oracle Database Licensing Information User Manual pour la version que vous utilisez.

Rubriques

- Nouvelles fonctions RDS for Oracle
- Fonctions prises en charge dans RDS for Oracle
- Fonctions non prises en charge dans RDS for Oracle

Nouvelles fonctions RDS for Oracle

Pour voir les nouvelles fonctionnalités de RDS pour Oracle, Historique du document recherchez le mot clé**`Oracle`**.

Fonctions prises en charge dans RDS for Oracle

Amazon RDS for Oracle prend en charge les fonctions Oracle Database suivantes :

> **Note**
>
> La liste suivante n'est pas exhaustive.

- Advanced Compression
- Oracle Application Express (APEX)

  Pour de plus amples informations, veuillez consulter Oracle Application Express (APEX).

- Automatic Memory Management
- Automatic Undo Management
- Automatic Workload Repository (AWR)

  Pour de plus amples informations, veuillez consulter Génération de rapports de performance avec Automatic Workload Repository (AWR).
- Protection active des données avec des performances maximales dans la même AWS région ou dans plusieurs AWS régions

  Pour de plus amples informations, veuillez consulter Utilisation de réplicas en lecture pour Amazon RDS for Oracle.
- Tables blockchain (Oracle Database 21c et plus)

  Pour plus d'informations, reportez-vous à la section Managing Blockchain Tables (Gestion des tables de blockchain) dans la documentation relative à la base de données Oracle.
- Notification continue des requêtes

  Pour de plus amples informations, veuillez consulter Using Continuous Query Notification (CQN) dans la documentation Oracle.
- Data Redaction
- Notification continue des requêtes

  Pour de plus amples informations, veuillez consulter Database Change Notification dans la documentation Oracle.
- Base de données en mémoire
- Transactions et requêtes distribuées
- Redéfinition basée sur l'édition

  Pour de plus amples informations, veuillez consulter Définition de l'édition par défaut d'une instance de base de données.
- EM Express (version 12c et ultérieures)

  Pour de plus amples informations, veuillez consulter Oracle Enterprise Manager.
- Fine-Grained Auditing
- Flashback Table, Flashback Query, Flashback Transaction Query

- Renouvellement progressif du mot de passe pour les applications (Oracle Database 21c et ultérieures)

  Pour plus d'informations, reportez-vous à la rubrique Managing Gradual Database Password Rollover for Applications (Gestion du transfert progressif du mot de passe de la base de données pour les applications) dans la documentation relative à la base de données Oracle.
- HugePages

  Pour de plus amples informations, veuillez consulter Activation de HugePages pour une instance RDS for Oracle.
- Import/export (existant et Data Pump) et SQL*Loader

  Pour plus d'informations, consultez Importation de données dans Oracle sur Amazon RDS.
- Java Virtual Machine (JVM)

  Pour de plus amples informations, veuillez consulter Oracle Java Virtual Machine.
- JavaScript (Oracle Database 21c et versions ultérieures)

  Pour plus d'informations, consultez DBMS_MLE dans la documentation de la base de données Oracle.
- Sécurité des étiquettes

  Pour de plus amples informations, veuillez consulter Oracle Label Security.
- Locator

  Pour plus d'informations, consultez Oracle Locator.
- Vues matérialisées
- Locataires multiples

  L'architecture multilocataire Oracle est prise en charge pour toutes les versions 19c et supérieures d'Oracle Database. Pour de plus amples informations, veuillez consulter Utilisation des CDB dans RDS for Oracle.
- Chiffrement de réseau

  Pour plus d'informations, consultez Oracle NNE (Native Network Encryption) et Oracle Secure Sockets Layer (SSL).
- Partitioning

- Real Application Testing

  Pour utiliser toutes les fonctionnalités de capture et de rediffusion, vous devez utiliser Amazon Elastic File System (Amazon EFS) pour accéder aux fichiers générés par Oracle Real Application Testing. Pour plus d'informations, consultez EFSIntégration avec Amazon et le billet de blog Utiliser les fonctionnalités d'Oracle Real Application Testing avec Amazon RDS for Oracle.
- Sharding au niveau de l'application (mais pas la fonctionnalité Oracle Sharding)
- Spatial et Graph

  Pour plus d'informations, consultez Oracle Spatial.
- Star Query Optimization
- Streams et Advanced Queuing
- Summary Management – Materialized View Query Rewrite
- Text (les magasins de données de type fichier et URL ne sont pas pris en charge)
- Total Recall
- Transparent Data Encryption (TDE) (Chiffrement transparent des données)

  Pour de plus amples informations, veuillez consulter Oracle Transparent Data Encryption.
- Audit unifié, mode mixte

  Pour plus d'informations, consultez Mixed Mode Auditing dans la documentation Oracle.
- XML DB (sans XML DB Protocol Server)

  Pour plus d'informations, consultez Oracle XML DB.
- Virtual Private Database

Fonctions non prises en charge dans RDS for Oracle

Amazon RDS for Oracle ne prend pas en charge les fonctions Oracle Database suivantes :

> **Note**
>
> La liste suivante n'est pas exhaustive.

- Automatic Storage Management (ASM)

- Database Vault
- Flashback Database

> **Note**
>
> Pour des solutions alternatives, consultez l'article du blog AWS de base de données Alternatives à la fonctionnalité de base de données Oracle Flashback dans Amazon RDS for Oracle.

- FTP et SFTP
- Tables partitionnées hybrides
- Passerelle de messagerie
- Oracle Enterprise Manager Cloud Control Management Repository
- Real Application Clusters (Oracle RAC)
- Real Application Security (RAS)
- Audit unifié, Pure Mode
- Schéma Workspace Manager (WMSYS)

> **Warning**
>
> En général, Amazon RDS ne vous empêche pas de créer des schémas pour des fonctions non prises en charge. Toutefois, si vous créez des schémas pour des fonctions et des composants Oracle nécessitant des privilèges SYSDBA, vous pouvez endommager le dictionnaire de données et affecter la disponibilité de votre instance de base de données. Utilisez uniquement les fonctions et schémas pris en charge et disponibles dans Ajout d'options aux instances de base de données Oracle.

Versions RDS for Oracle

RDS for Oracle for Oracle prend en charge plusieurs versions d'Oracle Database.

> **Note**
>
> Pour plus d'informations sur la mise à jour de vos versions, consultez Mise à niveau du moteur de base de données RDS for Oracle.

Rubriques

- Oracle Database 21c avec Amazon RDS
- Oracle Database 19c avec Amazon RDS

Oracle Database 21c avec Amazon RDS

Amazon RDS prend en charge Oracle Database 21c, qui inclut Oracle Enterprise Edition et Oracle Standard Edition 2. Oracle Database 21c (21.0.0.0) inclut beaucoup de nouvelles fonctions et mises à jour à partir de la version précédente. L'un des principaux changements est que Oracle Database 21c ne prend en charge que l'architecture multilocation : vous ne pouvez plus créer de base de données en tant que base non CDB traditionnelle. Pour en savoir plus sur les différences entre les CDB et les non-CDB, consultez Limites de RDS pour Oracle CDBs.

Cette section couvre les fonctions et modifications importantes liées à l'utilisation d'Oracle Database 21c (21.0.0.0) sur Amazon RDS. Pour obtenir la liste complète des évolutions, veuillez consulter la documentation Oracle Database 21c. Pour obtenir la liste complète des fonctionnalités prises en charge par chaque édition d'Oracle Database 21c, veuillez consulter Permitted Features, Options, and Management Packs by Oracle Database Offering dans la documentation Oracle.

Modifications des paramètres Amazon RDS for Oracle Database 21c (21.0.0.0)

Oracle Database 21c (21.0.0.0) comporte plusieurs nouveaux paramètres ainsi que des paramètres présentant de nouvelles plages et de nouvelles valeurs par défaut.

Rubriques

- Nouveaux paramètres
- Modifications apportées au paramètre compatible
- Paramètres supprimés

Nouveaux paramètres

Le tableau suivant indique les nouveaux paramètres Amazon RDS for Oracle Database 21c (21.0.0.0).

Nom	Plage de valeurs	Valeur par défaut	Adaptabilité	Description
blockchain_table_max_no_drop	NONE | 0	NONE	Y	Vous permet de contrôler la durée maximale d'inactivité pouvant être spécifiée lors de la création d'une table blockchain.
dbnest_enable	NONE | CDB_RESOURCE_PDB_ALL	NONE	N	Permet d'activer ou de désactiver DBNest. DbNest assure l'isolation et la gestion des ressources du système d'exploitation, l'isolation du système de fichiers et le calcul sécurisé pour les PDB.
dbnest_pdb_fs_conf	NONE | *pathname*	NONE	N	Spécifie le fichier de configuration du système de fichiers dbNest pour une PDB.
diagnostics_control	ERROR | WARNING | IGNORE	IGNORE	Y	Vous permet de contrôler et de surveiller les utilisateurs qui effectuent des opérations de diagnostic de la base de données potentiellement dangereuses.
drcp_dedicated_opt	YES | NO	YES	Y	Active ou désactive l'utilisation de l'optimisation dédiée avec le Database Resident Connection

Nom	Plage de valeurs	Valeur par défaut	Adaptabilité	Description
				Pooling (Regroupement des connexions résidant dans la base de données) (DRCP).
enable_per_pdb_drcp	`true | false`	`true`	N	Contrôle si le Database Resident Connection Pooling (Regroupement des connexions résidant dans la base de données) (DRCP) configure un regroupement de connexion pour l'ensemble de la CDB ou un regroupement de connexion isolé pour chaque PDB.
inmemory_deep_vectorization	`true | false`	`true`	Y	Active ou désactive le cadre de vectorisation profonde.
mandatory_user_profile	*`profile_name`*	N/A	N	Spécifie le profil utilisateur obligatoire pour une CDB ou une PDB.
optimizer_capture_sql_quarantine	`true | false`	`false`	Y	Active ou désactive le cadre de vectorisation profonde.
optimizer_use_sql_quarantine	`true | false`	`false`	Y	Active ou désactive la création automatique des configurations de la quarantaine SQL.

Nom	Plage de valeurs	Valeur par défaut	Adaptabilité	Description
result_cache_execution_threshold	0 sur 68719476736	2	Y	Spécifie le nombre maximum de fois qu'une fonction PL/SQL peut être exécutée avant que son résultat ne soit stocké dans le cache des résultats.
result_cache_max_temp_result	0 sur 100	5	Y	Spécifie le pourcentage de RESULT_CACHE_MAX_TEMP_SIZE qu'un seul résultat de requête mis en cache peut consommer.
result_cache_max_temp_size	0 sur 2199023255552	RESULT_CACHE_SIZE * 10	Y	Spécifie la quantité maximale d'espace disque logique temporaire (en octets) qui peut être consommée par la mise en cache des résultats.
sga_min_size	0 à 2199023255552 (la valeur maximale est de 50 % de sga_target)	0	Y	Indique une valeur minimale possible pour l'utilisation de la SGA d'une base de données enfichable (PDB).

Nom	Plage de valeurs	Valeur par défaut	Adaptabilité	Description
tablespace_encryption_default_algorithm	GOST256 | SEED128 | ARIA256 | ARIA192 | ARIA128 | 3DES168 | AES256 | AES192 | AES128	AES128	Y	Spécifie l'algorithme par défaut que la base de données utilise lors du chiffrement d'un espace disque logique.

Modifications apportées au paramètre compatible

Le paramètre `compatible` a une nouvelle valeur maximale pour Oracle Database 21c (21.0.0.0) sur Amazon RDS. Le tableau suivant présente la nouvelle valeur par défaut.

Nom du paramètre	Valeur maximale pour Oracle Database 21c (21.0.0.0)
compatible	21,0,0

Paramètres supprimés

Les paramètres suivants ont été supprimés dans Oracle Database 21c (21.0.0.0) :

- `remote_os_authent`
- `sec_case_sensitive_logon`
- `unified_audit_sga_queue_size`

Oracle Database 19c avec Amazon RDS

Amazon RDS prend en charge Oracle Database 19c, qui inclut Oracle Enterprise Edition et Oracle Standard Edition Two.

Oracle Database 19c (19.0.0.0) inclut beaucoup de nouvelles fonctions et mises à jour à partir de la version précédente. Cette section couvre les fonctions et modifications importantes liées à l'utilisation d'Oracle Database 19c (19.0.0.0) sur Amazon RDS. Pour obtenir la liste complète des évolutions, veuillez consulter la documentation Oracle Database 19c. Pour obtenir la liste complète des fonctionnalités prises en charge par chaque édition d'Oracle Database 19c, veuillez consulter Permitted Features, Options, and Management Packs by Oracle Database Offering dans la documentation Oracle.

Modifications des paramètres Amazon RDS for Oracle Database 19c (19.0.0.0)

Oracle Database 19c (19.0.0.0) comporte plusieurs nouveaux paramètres ainsi que des paramètres présentant de nouvelles plages et de nouvelles valeurs par défaut.

Rubriques

- Nouveaux paramètres
- Modifications apportées au paramètre compatible
- Paramètres supprimés

Nouveaux paramètres

Le tableau suivant indique les nouveaux paramètres Amazon RDS for Oracle Database 19c (19.0.0.0).

Nom	Valeurs	Adaptabilité	Description
lob_signature_enable	TRUE, FALSE (par défaut)	O	Active ou désactive la fonction de signature de localisateur LOB.
max_datapump_parallel_per_job	1 à 1 024 ou AUTO	O	Indique le nombre maximal de processus parallèles autorisés pour chaque tâche Oracle Data Pump.

Modifications apportées au paramètre compatible

Le paramètre `compatible` a une nouvelle valeur maximale pour Oracle Database 19c (19.0.0.0) sur Amazon RDS. Le tableau suivant présente la nouvelle valeur par défaut.

Nom du paramètre	Valeur maximale pour Oracle Database 19c (19.0.0.0)
compatible	19.0.0

Paramètres supprimés

Les paramètres suivants ont été supprimés dans Oracle Database 19c (19.0.0.0) :

- `exafusion_enabled`
- `max_connections`
- `o7_dictionary_access`

RDSpour les options de licence Oracle

Amazon RDS pour Oracle propose deux options de licence : License Included (LI) et Bring Your Own License (BYOL). Après avoir créé une instance de base de données Oracle sur AmazonRDS, vous pouvez modifier le modèle de licence en modifiant l'instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

> **Important**
>
> Assurez-vous de disposer de la licence Oracle Database appropriée, avec licence de mise à jour logicielle et Support, pour votre classe d'instance de base de données et l'édition d'Oracle Database. Assurez-vous également que vous disposez de licences pour toutes les fonctionnalités d'Oracle Database sous licence distincte.

Rubriques

- Modèle de licence inclus pour SE2
- Bring Your Own License (BYOL) pour EE et SE2
- Licences des déploiements multi-AZ Oracle

Modèle de licence inclus pour SE2

Dans le modèle Licence incluse, il n'est pas nécessaire d'acheter des licences Oracle Database séparément. AWS détient la licence du logiciel de base de données Oracle. Le modèle License Included est uniquement pris en charge sur Amazon RDS pour Oracle Database Standard Edition 2 (SE2).

Dans ce modèle, si vous avez un AWS Support compte avec support de dossier, contactez Amazon RDS et Oracle Database Support pour les demandes de service. Votre utilisation de l'option LI RDS pour Oracle est soumise à la section 10.3.1 des conditions de AWS service.

Bring Your Own License (BYOL) pour EE et SE2

Dans le BYOL modèle, vous pouvez utiliser vos licences Oracle Database existantes pour déployer des bases de données sur AmazonRDS. Amazon RDS prend en charge le BYOL modèle uniquement pour Oracle Database Enterprise Edition (EE) et Oracle Database Standard Edition 2 (SE2).

Assurez-vous que vous disposez de la licence Oracle Database appropriée (avec la licence de mise à jour du logiciel et le support) pour la classe d'instance de base de données et l'édition d'Oracle Database que vous souhaitez exécuter. Vous devez aussi suivre les stratégies d'Oracle pour obtenir la licence du logiciel de base de données Oracle dans l'environnement de cloud computing. Pour plus d'informations sur la politique de licence d'Oracle pour AmazonEC2, consultez la section Octroi de licences aux logiciels Oracle dans un environnement de cloud computing.

Dans ce modèle, vous continuez d'utiliser votre compte de support Oracle actif et vous contactez Oracle directement pour les demandes de service propres à Oracle Database. Si vous avez un AWS Support compte avec support de dossier, vous pouvez le contacter Support pour tout RDS problème concernant Amazon. Amazon Web Services et Oracle disposent d'un processus de support multi-vendeurs pour les cas nécessitant une assistance de la part des deux organisations.

Intégration avec AWS License Manager

Pour faciliter le suivi de l'utilisation des licences Oracle dans le BYOL modèle, il AWS License Managers'intègre à Amazon RDS pour Oracle. License Manager prend en charge le suivi RDS des éditions du moteur Oracle et des packs de licences basés sur des cœurs virtuels (vCPUs). Vous pouvez également utiliser License Manager AWS Organizations pour gérer tous les comptes de votre organisation de manière centralisée.

Le tableau suivant présente les filtres d'informations sur les produits RDS pour Oracle.

Filtre	Nom	Description
Engine Edition	`oracle-ee`	Oracle Database Enterprise Edition (EE)
	`oracle-se2`	Oracle Database Standard Edition 2 (SE2)
Pack de licence	`data guard`	Voir Utilisation de réplicas en lecture pour Amazon RDS for Oracle (Oracle Active Data Guard)
	`olap`	Voir Oracle OLAP
	`ols`	Voir Oracle Label Security
	`diagnostic pack sqlt`	Voir Oracle SQLT
	`tuning pack sqlt`	Consultez Oracle SQLT

Pour suivre l'utilisation des licences de vos instances de base de données Oracle, vous pouvez créer une licence autogérée à l'aide AWS License Manager de. Dans ce cas, RDS pour Oracle, les ressources correspondant au filtre d'informations sur le produit sont automatiquement associées à la licence autogérée. La détection des instances de base de données Oracle peut prendre jusqu'à 24 heures. Vous pouvez également suivre une licence sur plusieurs comptes en utilisant AWS Resource Access Manager.

Console

Pour créer une licence autogérée afin de AWS License Manager suivre l'utilisation des licences de vos instances de base RDS de données Oracle

1. Accédez à https://console.aws.amazon.com/license-manager/.
2. Choisissez Créer une licence autogérée.

   Pour obtenir des instructions, voir Création d'une licence autogérée dans le guide de l'AWS License Manager utilisateur.

   Ajoutez une règle pour un filtre d'informations sur le RDS produit dans le panneau d'informations sur le produit.

Pour plus d'informations, reportez-vous ProductInformationà la section AWS License Manager APIRéférence.

3. (Suivi entre comptes uniquement) AWS Resource Access Manager À utiliser pour partager vos licences autogérées avec n'importe quel AWS compte ou via. AWS Organizations Pour plus d'informations, consultez la section Partage de vos AWS ressources.

AWS CLI

Pour créer une licence autogérée à l'aide de AWS CLI, appelez la create-license-configurationcommande. Vous pouvez utiliser les paramètres `--cli-input-json` ou `--cli-input-yaml` pour transmettre les paramètres à la commande.

Example

L'exemple suivant crée une licence autogérée pour Oracle Enterprise Edition.

```
aws license-manager create-license-configuration --cli-input-json file://rds-oracle-ee.json
```

Voici l'exemple de fichier `rds-oracle-ee.json` utilisé dans l'exemple.

```
{
    "Name": "rds-oracle-ee",
    "Description": "RDS Oracle Enterprise Edition",
    "LicenseCountingType": "vCPU",
    "LicenseCountHardLimit": false,
    "ProductInformationList": [
        {
            "ResourceType": "RDS",
            "ProductInformationFilterList": [
                {
                    "ProductInformationFilterName": "Engine Edition",
                    "ProductInformationFilterValue": ["oracle-ee"],
                    "ProductInformationFilterComparator": "EQUALS"
                }
            ]
        }
    ]
}
```

Pour de plus amples informations sur les informations produit, veuillez consulter Détection automatique de l'inventaire des ressources dans le Guide de l'utilisateur AWS License Manager .

Pour plus d'informations sur le `--cli-input` paramètre, consultez la section Génération de AWS CLI squelettes et de paramètres d'entrée à partir d'un fichier YAML d'entrée JSON ou d'un fichier d'entrée dans le Guide de AWS CLI l'utilisateur.

Migration entre les éditions d'Oracle Database

Si vous disposez d'une licence BYOL Oracle Database non utilisée adaptée à l'édition et à la classe d'instance de base de données que vous prévoyez d'exécuter, vous pouvez migrer de l'édition Standard 2 (SE2) vers l'édition Enterprise (EE). Vous ne pouvez pas migrer d'EE vers d'autres éditions.

Pour modifier l'édition de votre base de données Oracle et conserver vos données

1. Créez un snapshot de l'instance DB.

   Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
2. Restaurez l'instantané dans une nouvelle instance de base de données et sélectionnez l'édition de base de données Oracle que vous souhaitez utiliser.

   Pour plus d'informations, consultez Restauration vers une instance de base de données.
3. (Facultatif) Supprimez l'ancienne instance de base de données, sauf si vous souhaitez continuer de l'exécuter et que vous disposez des licences Oracle Database appropriées.

   Pour plus d'informations, consultez Suppression d'une instance DB.

Licences des déploiements multi-AZ Oracle

Amazon RDS prend en charge les déploiements multi-AZ pour Oracle en tant que solution de basculement à haute disponibilité. Le déploiement multi-AZ est conseillé pour les charges de travail de production. Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Si vous utilisez le modèle Réutilisez vos licences, vous devez disposer d'une licence à la fois pour l'instance de base de données principale et l'instance de base de données de secours dans un déploiement multi-AZ.

Utilisateurs et privilèges RDS for Oracle

Lorsque vous créez une instance de base de données Amazon RDS for Oracle, l'utilisateur principal par défaut dispose de la plupart des autorisations utilisateur maximales sur l'instance de base de données. Utilisez ce compte d'utilisateur principal pour toutes les tâches administratives, telles que la création de comptes d'utilisateur supplémentaires dans votre base de données. Comme RDS est un service géré, vous n'êtes pas autorisé à vous connecter en tant que `SYS` ni `SYSTEM`, et n'avez donc pas les privilèges `SYS`.

Rubriques

- Limitations des privilèges Oracle DBA
- Gestion des privilèges sur les objets SYS

Limitations des privilèges Oracle DBA

Dans la base de données, un rôle est un ensemble de privilèges que vous pouvez accorder ou révoquer à un utilisateur. Une base de données Oracle utilise des rôles pour assurer la sécurité. Pour plus d'informations, consultez Configuration de l'autorisation des privilèges et des rôles (langue française non garantie) dans la documentation sur Oracle Database.

Le rôle prédéfini `DBA` autorise normalement tous les privilèges d'administration sur une base de données Oracle. Lorsque vous créez une instance de base de données, votre compte utilisateur principal obtient des privilèges d'administrateur de base de données (avec certaines restrictions). Pour offrir une expérience gérée, une base de données RDS for Oracle ne fournit pas les privilèges suivants au rôle `DBA` :

- `ALTER DATABASE`
- `ALTER SYSTEM`
- `CREATE ANY DIRECTORY`
- `DROP ANY DIRECTORY`
- `GRANT ANY PRIVILEGE`
- `GRANT ANY ROLE`

Pour plus d'informations sur les privilèges et les rôles du système RDS for Oracle, consultez Privilèges du compte utilisateur principal.

Gestion des privilèges sur les objets SYS

Vous pouvez gérer les privilèges sur les objets SYS à l'aide du package `rdsadmin.rdsadmin_util`. Par exemple, si vous créez l'utilisateur de base de données `myuser`, vous pouvez utiliser la procédure `myuser` pour accorder les privilèges `SELECT` sur `V_$SQLAREA` à `myuser`. Pour plus d'informations, consultez les rubriques suivantes :

- Octroi SELECT de EXECUTE privilèges à SYS des objets
- Révocation SELECT de EXECUTE privilèges sur des objets SYS
- Attribution de privilèges à des utilisateurs non-maîtres

RDS pour les classes d'instance de base de données Oracle

La capacité de calcul et de mémoire d'une instance de base de données RDS pour Oracle est déterminée par sa classe d'instance. La classe d'instance de bases de données dont vous avez besoin varie selon vos exigences en mémoire et en puissance de traitement.

RDS pris en charge pour les classes d'instance de base de données Oracle

Les classes d'instance RDS for Oracle prises en charge sont un sous-ensemble des classes d'instance de base de données RDS. Pour obtenir la liste complète des classes d'instances RDS, consultez Classes d'instances de base de données .

Classes d'instances de base de données préconfigurées RDS pour Oracle

RDS pour Oracle propose également des classes d'instances préconfigurées pour les charges de travail nécessitant de la mémoire, du stockage et par I/O vCPU supplémentaires. Ces classes d'instances utilisent la convention de nommage suivante :

```
db.r5b.instance_size.tpcthreads_per_core.memratio
db.r5.instance_size.tpcthreads_per_core.memratio
```

Voici un exemple de classe d'instance préconfigurée pour une mémoire supplémentaire :

```
db.r5b.4xlarge.tpc2.mem2x
```

Les composants du nom de classe d'instance précédent sont les suivants :

- `db.r5b.4xlarge` – Nom de la classe d'instance.
- `tpc2` – Threads par cœur. La valeur 2 signifie que le multithread est activé. La valeur 1 signifie que le multithreading est désactivé.
- `mem2x` – Rapport de la mémoire supplémentaire à la mémoire standard pour la classe d'instance. Dans cet exemple, l'optimisation fournit deux fois plus de mémoire qu'une instance de base de données db.r5.4xlarge standard.

Édition, classe d'instance et combinaisons de licences prises en charge dans RDS pour Oracle

Si vous utilisez la console RDS, vous pouvez savoir si une combinaison d'édition, de classe d'instance et de licence spécifique est prise en charge en choisissant Créer une base de données et en spécifiant une autre option. Dans le AWS CLI, vous pouvez exécuter la commande suivante :

```
aws rds describe-orderable-db-instance-options --engine engine-type --license-
model license-type
```

Le tableau suivant répertorie toutes les éditions, classes d'instances et types de licences pris en charge pour RDS pour Oracle. Pour plus d'informations sur les attributs de mémoire de chaque type, consultez la section RDS for Oracle instance types (Types d'instances RDS for Oracle). Pour plus d'informations sur la tarification, consultez les modèles de tarification Amazon RDS for Oracle.

Édition Oracle	Oracle Database 19c et versions ultérieures
Enterprise Edition (EE)	Classes d'instance standard
Bring Your Own License (Licence à fournir)	db.m7i.large — db.m7i.48 x large db.m6 en largeur — db.m6 en 32 x large db.m6id.large — db.m6id.32xlarge db.m6i.large — db.m6i.32 x large db.m5d.large–db.m5d.24xlarge db.m5.large–db.m5.24xlarge
	Classes d'instances à mémoire optimisée

Édition Oracle	Oracle Database 19c et versions ultérieures
	db.r7i.large — db.r7i.48xlarge db.r6inlarge—db.r6in.32xlarge db.r6id.large — db.r6id.32xlarge db.r6i.large–db.r6i.32xlarge db.r5d.large–db.r5d.24xlarge db.r5b.large–db.r5b.24xlarge db.r5.large–db.r5.24xlarge db.x2iedn.xlarge–db.x2iedn.32xlarge db.x2iezn.2xlarge–db.x2iezn.12xlarge db.x2idn.16xlarge–db.x2idn.32xlarge db.x1e.xlarge–db.x1e.32xlarge db.x1.16xlarge—db.x1.32xlarge 1 db.z1d.large–db.z1d.12xlarge
	Classes d'instances préconfigurées optimisées pour la mémoire

Édition Oracle	Oracle Database 19c et versions ultérieures
	db.r6i.8xlarge.tpc2.mem4x
	db.r6i.8xlarge.tpc2.mem3x
	db.r6i.6xlarge.tpc2.mem4x
	db.r6i.4xlarge.tpc2.mem4x
	db.r6i.4xlarge.tpc2.mem3x
	db.r6i.4xlarge.tpc2.mem2x
	db.r6i.2xlarge.tpc2.mem8x
	db.r6i.2xlarge.tpc2.mem4x
	db.r6i.2xlarge.tpc1.mem2x
	db.r6i.xlarge.tpc2.mem4x
	db.r6i.xlarge.tpc2.mem2x
	db.r6i.large.tpc1.mem2x
	db.r5b.8xlarge.tpc2.mem3x
	db.r5b.6xlarge.tpc2.mem4x
	db.r5b.4xlarge.tpc2.mem4x
	db.r5b.4xlarge.tpc2.mem3x
	db.r5b.4xlarge.tpc2.mem2x
	db.r5b.2xlarge.tpc2.mem8x
	db.r5b.2xlarge.tpc2.mem4x
	db.r5b.2xlarge.tpc1.mem2x
	db.r5b.xlarge.tpc2.mem4x

Édition Oracle	Oracle Database 19c et versions ultérieures
	db.r5b.xlarge.tpc2.mem2x
	db.r5b.large.tpc1.mem2x
	db.r5.12xlarge.tpc2.mem2x
	db.r5.8xlarge.tpc2.mem3x
	db.r5.6xlarge.tpc2.mem4x
	db.r5.4xlarge.tpc2.mem4x
	db.r5.4xlarge.tpc2.mem3x
	db.r5.4xlarge.tpc2.mem2x
	db.r5.2xlarge.tpc2.mem8x
	db.r5.2xlarge.tpc2.mem4x
	db.r5.2xlarge.tpc1.mem2x
	db.r5.xlarge.tpc2.mem4x
	db.r5.xlarge.tpc2.mem2x
	db.r5.large.tpc1.mem2x
	Classes d'instance à capacité extensible
	db.t3.small–db.t3.2xlarge

Édition Oracle	Oracle Database 19c et versions ultérieures
Édition standard (2SE2)	Classes d'instance standard
Bring Your Own License (Licence à fournir)	db.m7i.large — db.m7i.4xlarge db.m6 en large — db.m6 en 4 x large db.m6id.large — db.m6id.4xlarge db.m6i.large — db.m6i.4xlarge db.m5d.large–db.m5d.4xlarge db.m5.large–db.m5.4xlarge
	Classes d'instances à mémoire optimisée
	db.r7i.large — db.r7i.4xlarge db.r6inlarge—db.r6in4xlarge db.r6id.large — db.r6id.4xlarge db.r6i.large–db.r6i.4xlarge db.r5d.large–db.r5d.4xlarge db.r5b.large–db.r5b.4xlarge db.r5.large–db.r5.4xlarge db.x2iedn.xlarge–db.x2iedn.4xlarge db.x2iezn.2xlarge–db.x2iezn.4xlarge db.z1d.large–db.z1d.3xlarge
	Classes d'instances préconfigurées optimisées pour la mémoire

Édition Oracle	Oracle Database 19c et versions ultérieures
	db.r6i.4xlarge.tpc2.mem4x
	db.r6i.4xlarge.tpc2.mem3x
	db.r6i.4xlarge.tpc2.mem2x
	db.r6i.2xlarge.tpc2.mem8x
	db.r6i.2xlarge.tpc2.mem4x
	db.r6i.2xlarge.tpc1.mem2x
	db.r6i.xlarge.tpc2.mem4x
	db.r6i.xlarge.tpc2.mem2x
	db.r6i.large.tpc1.mem2x
	db.r5b.4xlarge.tpc2.mem4x
	db.r5b.4xlarge.tpc2.mem3x
	db.r5b.4xlarge.tpc2.mem2x
	db.r5b.2xlarge.tpc2.mem8x
	db.r5b.2xlarge.tpc2.mem4x
	db.r5b.2xlarge.tpc1.mem2x
	db.r5b.xlarge.tpc2.mem4x
	db.r5b.xlarge.tpc2.mem2x
	db.r5b.large.tpc1.mem2x
	db.r5.4xlarge.tpc2.mem4x
	db.r5.4xlarge.tpc2.mem3x
	db.r5.4xlarge.tpc2.mem2x

Édition Oracle	Oracle Database 19c et versions ultérieures
	db.r5.2xlarge.tpc2.mem8x
	db.r5.2xlarge.tpc2.mem4x
	db.r5.2xlarge.tpc1.mem2x
	db.r5.xlarge.tpc2.mem4x
	db.r5.xlarge.tpc2.mem2x
	db.r5.large.tpc1.mem2x
	Classes d'instance à capacité extensible
	db.t3.small–db.t3.2xlarge
Édition standard (2SE2)	Classes d'instance standard
Licence incluse	db.m5.large–db.m5.4xlarge
	Classes d'instances à mémoire optimisée
	db.r6i.large–db.r6i.4xlarge
	db.r5.large–db.r5.4xlarge
	Classes d'instance à capacité extensible
	db.t3.small–db.t3.2xlarge

[1] Vous ne pouvez plus créer de RDS pour les instances de base de données Oracle à l'aide de la famille de classes d'instance X1.* (les classes d'instance X1e.* sont toujours prises en charge). Si vous utilisez actuellement des classes X1.*, passez à une classe d'instance de nouvelle génération dès que possible. À compter du 22 janvier 2025, RDS lance les mises à niveau automatisées dans le créneau de maintenance que vous avez défini. Au cours de la mise à niveau, RDS choisit le type d'instance X2iEDN équivalent et le met à niveau. Pour plus d'informations, consultez l'article Re:post Amazon RDS for Oracle mettra fin au support des instances de base de données X1 le 22 janvier 2025.

> **Note**
>
> Nous invitions tous les clients qui utilisent leurs propres licences (BYOL) à consulter leur contrat de licence pour évaluer l'impact des obsolescences Amazon RDS for Oracle. Pour obtenir plus d'informations sur la capacité de calcul des classes d'instances de base de données prises en charge par RDS for Oracle, consultez Classes d'instances de base de données et Configuration du processeur pour une classe d'instances de base de données dans RDS for Oracle.

> **Note**
>
> Si vous disposez d'instantanés de bases de données d'instances de base de données qui utilisaient des classes d'instance de base de données obsolètes, vous pouvez choisir une classe d'instance de base de données non obsolète lorsque vous restituez les instantanés de bases de données. Pour de plus amples informations, veuillez consulter Restauration vers une instance de base de données.

RDS obsolète pour les classes d'instance de base de données Oracle

Les classes d'instances de base de données obsolètes pour RDS for Oracle sont les suivantes :

- db.m1, db.m2, db.m3, db.m4
- db.t1, db.t2
- db.r1, db.r2, db.r3, db.r4

Les classes d'instances de base de données précédentes ont été remplacées par des classes d'instances de base de données plus performantes qui sont généralement disponibles à moindre coût. Si vous possédez des instances de base de données qui utilisent des classes d'instance de base de données obsolètes, vous disposez des options suivantes :

- Autorisez Amazon RDS à modifier automatiquement chaque instance de base de données afin d'utiliser une classe d'instance de base de données non obsolète comparable. Pour connaître les délais d'obsolescence, consultez Types de classes d'instance de base de données.

- Changez la classe d'instance de base de données vous-même en modifiant l'instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Si vous disposez d'instantanés de bases de données d'instances de base de données qui utilisaient des classes d'instance de base de données obsolètes, vous pouvez choisir une classe d'instance de base de données non obsolète lorsque vous restituez les instantanés de bases de données. Pour de plus amples informations, veuillez consulter Restauration vers une instance de base de données.

Architecture de base de données RDS for Oracle

L'architecture multilocataire Oracle, également appelée architecture CDB, permet à une base de données Oracle de fonctionner comme une base de données de conteneur (CDB) multilocataire. Un CDB peut inclure des bases de données enfichables créées par le client (). PDBs Une base de données non CDB est une base de données Oracle qui utilise l'architecture traditionnelle, qui ne peut pas contenir. PDBs Pour plus d'informations sur l'architecture multi-locataires, reportez-vous au Guide de l'administrateur d'Oracle Multitenant.

Pour Oracle Database 19c et versions ultérieures, vous pouvez créer une instance de base de données RDS for Oracle qui utilise l'architecture CDB. Dans RDS pour Oracle, PDBs elles sont appelées bases de données mutualisées. Vos applications clientes se connectent au niveau de la base de données client (PDB) plutôt qu'au niveau de la CDB. RDS for Oracle prend en charge les configurations suivantes de l'architecture CDB :

Configuration à locataires multiples

Amazon RDS permet à une instance CDB de contenir entre 1 et 30 bases de données mutualisées, en fonction de l'édition de la base de données et des licences d'option requises. Vous pouvez utiliser RDS APIs pour ajouter, modifier et supprimer des bases de données mutualisées. La configuration multi-locataires de RDS pour Oracle ne prend pas en charge les applications PDBs ou les proxys PDBs, qui sont des types particuliers de. PDBs Pour plus d'informations sur l'application PDBs et le proxy PDBs, consultez la section Types de PDBs dans la documentation de la base de données Oracle.

> **Note**
>
> La configuration Amazon RDS est appelée « multi-tenant » plutôt que « multitenant » car il s'agit d'une fonctionnalité d'Amazon RDS, et pas seulement du moteur de base de

données Oracle. De même, le terme « locataire » du RDS fait référence à tout locataire d'une configuration RDS, et pas seulement à Oracle. PDBs Dans la documentation RDS, le terme sans trait d'union « Oracle multitenant » fait exclusivement référence à l'architecture CDB de la base de données Oracle, qui est compatible avec les déploiements sur site et RDS.

Configuration à locataire unique

Amazon RDS limite une instance RDS pour Oracle CDB à une seule base de données client (PDB). Vous ne pouvez pas en ajouter d'autres PDBs à l'aide de RDS. APIs La configuration à locataire unique utilise le même RDS APIs que l'architecture non CDB. Ainsi, l'expérience de travail avec une CDB dans une configuration à locataire unique est essentiellement la même que celle consistant à travailler avec une architecture non CDB.

Vous pouvez convertir une CDB qui utilise la configuration à locataire unique en configuration multi-locataire, ce qui vous permet d'ajouter PDBs des éléments à votre CDB. Ce changement d'architecture est définitif et irréversible. Pour de plus amples informations, veuillez consulter Conversion de la configuration à locataire unique en configuration à locataires multiples.

> **Note**
>
> Vous ne pouvez pas accéder à la base de données de conteneur (CDB) elle-même.

Dans Oracle Database 21c et versions ultérieures, toutes les bases de données le sont CDBs. En revanche, vous pouvez créer une instance de base de données Oracle Database 19c en tant que base de données CDB ou non-CDB. Vous ne pouvez pas mettre à niveau une base de données non-CDB en CDB, mais vous pouvez convertir une base de données non-CDB Oracle Database 19c en CDB, puis la mettre à niveau. Vous ne pouvez pas convertir une CDB en base de données non-CDB.

Pour plus d’informations, consultez les ressources suivantes :

- Utilisation des CDB dans RDS for Oracle
- Limites de RDS pour Oracle CDBs
- Création d'une instance de base de données Amazon RDS

Paramètres d'initialisation de RDS pour Oracle

Dans Amazon RDS, vous gérez les paramètres à l'aide d'un groupe de paramètres de base de données. Ce groupe vous permet de personnaliser les paramètres d'initialisation. Par exemple, vous pouvez configurer la taille de la zone de tri avec`sort_area_size`. Toutes les instances de base de données RDS pour Oracle associées à un groupe de paramètres de base de données spécifique utilisent les mêmes paramètres. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).

Paramètres d'initialisation pris en charge dans RDS pour Oracle

Les paramètres pris en charge pour votre instance de base de données dépendent de l'édition et de la version de votre base de données Oracle. Pour afficher les paramètres d'initialisation pris en charge pour une édition et une version spécifiques d'Oracle Database, exécutez la AWS CLI commande `describe-engine-default-parameters`. Par exemple, pour répertorier les noms des paramètres d'initialisation pris en charge pour l'édition Enterprise d'Oracle Database 19c, exécutez la commande suivante (exemple de sortie inclus).

```
aws rds describe-engine-default-parameters \
    --db-parameter-group-family oracle-ee-19 \
    --output json | jq -r '.EngineDefaults.Parameters[].ParameterName'

_add_col_optim_enabled
_adg_parselock_timeout
_allow_insert_with_update_check
_allow_level_without_connect_by
_always_semi_join
_autotask_max_window
_awr_disabled_flush_tables
_awr_mmon_cpuusage
_awr_mmon_deep_purge_all_expired
_b_tree_bitmap_plans
_bct_bitmaps_per_file
_bloom_filter_enabled
_buffered_publisher_flow_control_threshold
_bug29394014_allow_triggers_on_vpd_table
_cleanup_rollback_entries
_client_enable_auto_unregister
_clusterwide_global_transactions
_complex_view_merging
_connect_by_use_union_all
```

```
_cost_equality_semi_join
_cursor_features_enabled
_cursor_obsolete_threshold
_datafile_write_errors_crash_instance
_db_block_buffers
...
```

Pour décrire un seul paramètre d'initialisation, utilisez la commande suivante, en le `sga_max_size` remplaçant par le nom de votre paramètre (exemple de sortie inclus).

```
aws rds describe-engine-default-parameters \
    --db-parameter-group-family oracle-ee-19 \
    --query 'EngineDefaults.Parameters[?ParameterName==`sga_max_size`]' \
    --output json

[
    {
        "ParameterName": "sga_max_size",
        "Description": "max total SGA size",
        "Source": "engine-default",
        "ApplyType": "static",
        "DataType": "integer",
        "AllowedValues": "0-2199023255552",
        "IsModifiable": true
    }
]
```

Pour trouver de la documentation générale sur les paramètres d'initialisation de la base de données Oracle, voir Paramètres d'initialisation dans la documentation de la base de données Oracle. Notez que le paramètre `ARCHIVE_LAG_TARGET` fait l'objet de considérations particulières dans RDS pour Oracle. Ce paramètre force le changement de journal redo en ligne une fois le délai spécifié écoulé. Dans RDS pour Oracle, `ARCHIVE_LAG_TARGET` est défini sur `300` parce que l'objectif du point de restauration (RPO) est de 5 minutes. Pour atteindre cet objectif, RDS for Oracle change le journal de journalisation en ligne toutes les 5 minutes et le stocke dans un compartiment Amazon S3.

Si la fréquence du commutateur de journalisation en ligne dégrade les performances de votre base de données RDS pour Oracle, vous pouvez redimensionner votre instance de base de données et votre stockage pour utiliser des IOPS et un débit plus élevés. Sinon, si vous utilisez RDS Custom pour Oracle ou si vous déployez une base de données Oracle sur Amazon EC2, vous pouvez ajuster le réglage du paramètre d'`ARCHIVE_LAG_TARGET`initialisation.

Valeurs de paramètres valides dans RDS pour Oracle

Dans RDS pour Oracle, seuls les caractères suivants sont valides pour les valeurs des paramètres :

- Lettres (`A-Zeta-z`)
- Chiffres (`0-9`)
- Espaces blancs (espaces, tabulations et sauts de ligne)
- Les caractères spéciaux suivants : `_ / . : + = ( ) ' * , % $ -` (tiret)

Jeux de caractères RDS for Oracle

RDS for Oracle prend en charge deux types de jeux de caractères : le jeu de caractères de base de données et le jeu de caractères national.

Jeu de caractères de base de données

Le jeu de caractères de base de données Oracle est utilisé dans les types de données `CHAR`, `VARCHAR2` et `CLOB`. La base de données utilise également ce jeu de caractères pour les métadonnées telles que les noms de table, les noms de colonne et les instructions SQL. Le jeu de caractères de base de données Oracle est généralement appelé jeu de caractères de base de données.

Vous définissez le jeu de caractères lorsque vous créez une instance de base de données. Vous ne pouvez pas modifier le jeu de caractères de base de données après avoir créé la base de données.

Jeux de caractères de base de données pris en charge

La table suivante répertorie les jeux de caractères de base de données Oracle qui sont pris en charge dans Amazon RDS. Vous pouvez utiliser une valeur de cette table avec le paramètre `--character-set-name` de la commande de l'AWS CLI create-db-instance, ou avec le paramètre `CharacterSetName` de l'opération de l'API Amazon RDS CreateDBInstance.

> **Note**
>
> Le jeu de caractères d'une base de données de conteneur (CDB) est toujours AL32UTF8. Vous pouvez définir un jeu de caractères différent uniquement pour la base de données enfichable uniquement (PDB).

Valeur	Description
AL32UTF8	Jeu de caractères universel UTF-8 Unicode 5.0 (par défaut)
AR8ISO8859P6	ISO 8859-6 Latis.o.rabe
AR8MSWIN1256	Page de codes Microsoft Windows 1256 8 bits Latis.o.rabe
BLT8ISO8859P13	ISO 8859-13 Baltique
BLT8MSWIN1257	Page de codes Microsoft Windows 1257 8 bits Baltique
CL8ISO8859P5	ISO 88559-5 Latin/Cyrillique
CL8MSWIN1251	Page de codes Microsoft Windows 1251 8 bits Latin/Cyrillique
EE8ISO8859P2	ISO 8859-2 Europe de l'Est
EL8ISO8859P7	ISO 8859-7 Latin/Grec
EE8MSWIN1250	Page de codes Microsoft Windows 1250 8 bits Europe de l'Est
EL8MSWIN1253	Page de codes Microsoft Windows 1253 8 bits Latin/Grec
IW8ISO8859P8	ISO 8859-8 Latin/Hébreu
IW8MSWIN1255	Page de codes Microsoft Windows 1255 8 bits Latin/Hébreu
JA16EUC	EUC 24 bits Japonais
JA16EUCTILDE	Même chose que JA16EUC, sauf pour le mapping du trait d'union en esse et du tilde vers et depuis Unicode

Valeur	Description
JA16SJIS	Shift-JIS 16 bits Japonais
JA16SJISTILDE	Même chose que JA16SJIS sauf pour le mapping du trait d'union en esse et du tilde vers et depuis Unicode
KO16MSWIN949	Page de codes Microsoft Windows 949 Coréen
NE8ISO8859P10	ISO 8859-10 Europe du Nord
NEE8ISO8859P4	ISO 8859-4 Europe du Nord et du Nord-Est
TH8TISASCII	Thai Industrial Standard 620-2533-ASCII 8 bits
TR8MSWIN1254	Page de codes Microsoft Windows 1254 8 bits Turc
US7ASCII	ASCII 7 bits Américain
UTF8	Jeu de caractères universel UTF-8 Unicode 3.0, compatible CESU-8
VN8MSWIN1258	Page de codes Microsoft Windows 1258 8 bits Vietnamien
WE8ISO8859P1	ISO 8859 Partie 1 Europe de l'Ouest 8 bits
WE8ISO8859P15	ISO 8859-15 Europe de l'Ouest
WE8ISO8859P9	ISO 8859-9 Europe de l'Ouest et Turc
WE8MSWIN1252	Page de codes Microsoft Windows 1252 8 bits Europe de l'Ouest
ZHS16GBK	GBK 16 bits Chinois simplifié

Valeur	Description
ZHT16HKSCS	Page de codes Microsoft Windows 950 avec jeu de caractères supplémentaire Hong Kong HKSCS-2001. La conversion de jeu de caractères est basée sur Unicode 3.0.
ZHT16MSWIN950	Page de codes Microsoft Windows 950 Chinois traditionnel
ZHT32EUC	EUC 32 bits Chinois traditionnel

Variable d'environnement NLS_LANG

Un paramètre régional est un ensemble d'informations répondant aux exigences linguistiques et culturelles qui correspondent à une langue et à un pays donnés. La définition de la variable d'environnement NLS_LANG dans l'environnement de votre client est la manière la plus simple de spécifier le comportement local du logiciel Oracle. Cette variable définit la langue et le territoire utilisés par l'application cliente et le serveur de base de données. Ils indiquent également le jeu de caractères client qui correspond au jeu de caractères pour les données entrées ou affichées par une application cliente. Pour de plus amples informations sur NLS_LANG et les jeux de caractères, veuillez consulter What is a Character set or Code Page? (Qu'est-ce qu'un jeu de caractères ou une page de code ?) dans la documentation Oracle.

Paramètres d'initialisation NLS

Vous pouvez également définir les paramètres d'initialisation NLS (National Language Support) suivants au niveau de l'instance pour une instance de base de données Oracle dans Amazon RDS :

- NLS_DATE_FORMAT
- NLS_LENGTH_SEMANTICS
- NLS_NCHAR_CONV_EXCP
- NLS_TIME_FORMAT
- NLS_TIME_TZ_FORMAT
- NLS_TIMESTAMP_FORMAT
- NLS_TIMESTAMP_TZ_FORMAT

Pour plus d'informations sur la modification des paramètres d'instance, consultez Groupes de paramètres pour (Amazon RDS).

Vous pouvez définir d'autres paramètres d'initialisation NLS dans votre client SQL. Par exemple, l'instruction suivante définit le paramètre d'initialisation NLS LANGUAGE à GERMAN dans un client SQL connecté à une instance de base de données Oracle :

```
ALTER SESSION SET NLS_LANGUAGE=GERMAN;
```

Pour plus d'informations sur la connexion à une instance de base de données Oracle avec un client SQL, consultez Connexion à votre instance de base de données Oracle.

Jeu de caractères national

Le jeu de caractères national est utilisé dans les types de données `NCHAR`, `NVARCHAR2` et `NCLOB`. Le jeu de caractères national est généralement appelé jeu de caractères NCHAR. Contrairement au jeu de caractères de base de données, le jeu de caractères NCHAR n'affecte pas les métadonnées de base de données.

Le jeu de caractères NCHAR prend en charge les jeux de caractères suivants :

- AL16UTF16 (par défaut)
- UTF8

Vous pouvez spécifier l'une des valeurs avec le paramètre `--nchar-character-set-name` de la commande create-db-instance (AWS CLI version 2 seulement). Si vous utilisez l'API Amazon RDS, spécifiez le paramètre `NcharCharacterSetName` de l'opération CreateDBInstance. Vous ne pouvez pas modifier le jeu de caractères national après avoir créé la base de données.

Pour de plus amples informations sur Unicode dans les bases de données Oracle, veuillez consulter Prise en charge des bases de données multilingues avec unicode dans la documentation Oracle.

Limitations RDS for Oracle

Dans les sections suivantes, vous trouverez les limites importantes de l'utilisation de RDS for Oracle. Pour les limitations spécifiques à CDBs, voirLimites de RDS pour Oracle CDBs.

> **Note**
>
> Cette liste n'est pas exhaustive.

Rubriques

- Limites de taille des fichiers Oracle dans Amazon RDS
- Limites de taille de bloc dans RDS pour Oracle
- Synonymes publics des schémas fournis par Oracle
- Schémas pour les fonctionnalités non prises en charge dans RDS pour Oracle
- Limitations relatives aux privilèges DBA dans RDS pour Oracle
- Obsolète de la sécurité de la couche de transport TLS 1.0 et 1.1 dans RDS pour Oracle

Limites de taille des fichiers Oracle dans Amazon RDS

La taille maximale d'un seul fichier sur les instances de bases de données RDS for Oracle est de 16 Tio (tébioctets). Cette limite est imposée par le système de fichiers ext4 utilisé par l'instance. Ainsi, les fichiers de données Oracle bigfile sont limités à 16 Tio. Si vous essayez de redimensionner un fichier de données dans un tablespace bigfile vers une valeur supérieure à la limite, une erreur comme la suivante se produit.

```
ORA-01237: cannot extend datafile 6
ORA-01110: data file 6: '/rdsdbdata/db/mydir/datafile/myfile.dbf'
ORA-27059: could not reduce file size
Linux-x86_64 Error: 27: File too large
Additional information: 2
```

Limites de taille de bloc dans RDS pour Oracle

La taille maximale d'un seul bloc sur RDS pour les instances de base de données Oracle est de 8 Ko. La taille de bloc de 16 Ko n'est pas prise en charge.

Synonymes publics des schémas fournis par Oracle

Ne créez pas de synonymes publics ou ne modifiez pas les synonymes publics des schémas gérés par Oracle, notamment `SYS`, `SYSTEM` et `RDSADMIN`. De telles actions peuvent entraîner l'invalidation

des composants de base de données de base et affecter la disponibilité de votre instance de base de données.

Vous pouvez créer des synonymes publics référençant des objets dans vos propres schémas.

Schémas pour les fonctionnalités non prises en charge dans RDS pour Oracle

En général, Amazon RDS ne vous empêche pas de créer des schémas pour des fonctions non prises en charge. Toutefois, si vous créez des schémas pour des fonctions et des composants Oracle nécessitant des privilèges SYS, vous pouvez endommager le dictionnaire de données et affecter la disponibilité de votre instance. Utilisez uniquement les fonctions et schémas pris en charge et disponibles dans Ajout d'options aux instances de base de données Oracle.

Limitations relatives aux privilèges DBA dans RDS pour Oracle

Dans la base de données, un rôle est un ensemble de privilèges que vous pouvez accorder ou révoquer à un utilisateur. Une base de données Oracle utilise des rôles pour assurer la sécurité.

Le rôle prédéfini `DBA` autorise normalement tous les privilèges d'administration sur une base de données Oracle. Lorsque vous créez une instance de base de données, votre compte utilisateur principal obtient des privilèges d'administrateur de base de données (avec certaines restrictions). Pour offrir une expérience gérée, une base de données RDS for Oracle ne fournit pas les privilèges suivants au rôle `DBA` :

- `ALTER DATABASE`
- `ALTER SYSTEM`
- `CREATE ANY DIRECTORY`
- `DROP ANY DIRECTORY`
- `GRANT ANY PRIVILEGE`
- `GRANT ANY ROLE`

Utilisez ce compte utilisateur principal pour des tâches administratives comme la création de comptes utilisateurs supplémentaires dans la base de données. Vous ne pouvez pas utiliser `SYS`, `SYSTEM` et d'autres comptes administratifs fournis par Oracle.

Obsolète de la sécurité de la couche de transport TLS 1.0 et 1.1 dans RDS pour Oracle

Les versions 1.0 et 1.1 du protocole de sécurité de la couche de transport (TLS 1.0 et TLS 1.1) sont rendues obsolètes. Conformément aux bonnes pratiques de sécurité, Oracle a rendu obsolète l'utilisation de TLS 1.0 et TLS 1.1. Pour répondre à vos exigences de sécurité, nous vous recommandons vivement d'utiliser plutôt le protocole TLS 1.2.

Connexion à votre instance de base de données Oracle

Une fois qu'Amazon RDS a provisionné votre instance de base de données Oracle, vous pouvez utiliser n'importe quelle application cliente SQL standard pour vous connecter à votre instance de base de données. RDS étant un service géré, vous ne pouvez pas vous connecter en tant que SYS ni SYSTEM. Pour de plus amples informations, veuillez consulter Utilisateurs et privilèges RDS for Oracle.

Dans cette rubrique, vous allez apprendre à utiliser Oracle SQL Developer ou SQL*Plus pour vous connecter à une instance de base de données RDS for Oracle. Pour obtenir un exemple qui vous explique le processus de création et de connexion à un exemple d'instance de base de données, consultez Création et connexion à une instance de base de données Oracle.

Rubriques

- Recherche du point de terminaison de votre instance de base de données RDS for Oracle
- Connexion à votre instance de base de données à l'aide d'Oracle SQL Developer
- Connexion à votre instance de base de données à l'aide de SQL*Plus
- Considérations relatives aux groupes de sécurité
- Considérations relatives à l'architecture des processus
- Résolution des problèmes de connexion à votre instance de base de données Oracle
- Modification des propriétés de connexion à l'aide des paramètres sqlnet.ora

Recherche du point de terminaison de votre instance de base de données RDS for Oracle

Chaque instance de base de données Amazon RDS possède un point de terminaison. Chaque point de terminaison possède le nom DNS et le numéro de port de l'instance de base de données. Pour connecter votre instance de base de données à l'aide d'une application cliente SQL, vous avez besoin du nom DNS et du numéro de port de votre instance de base de données.

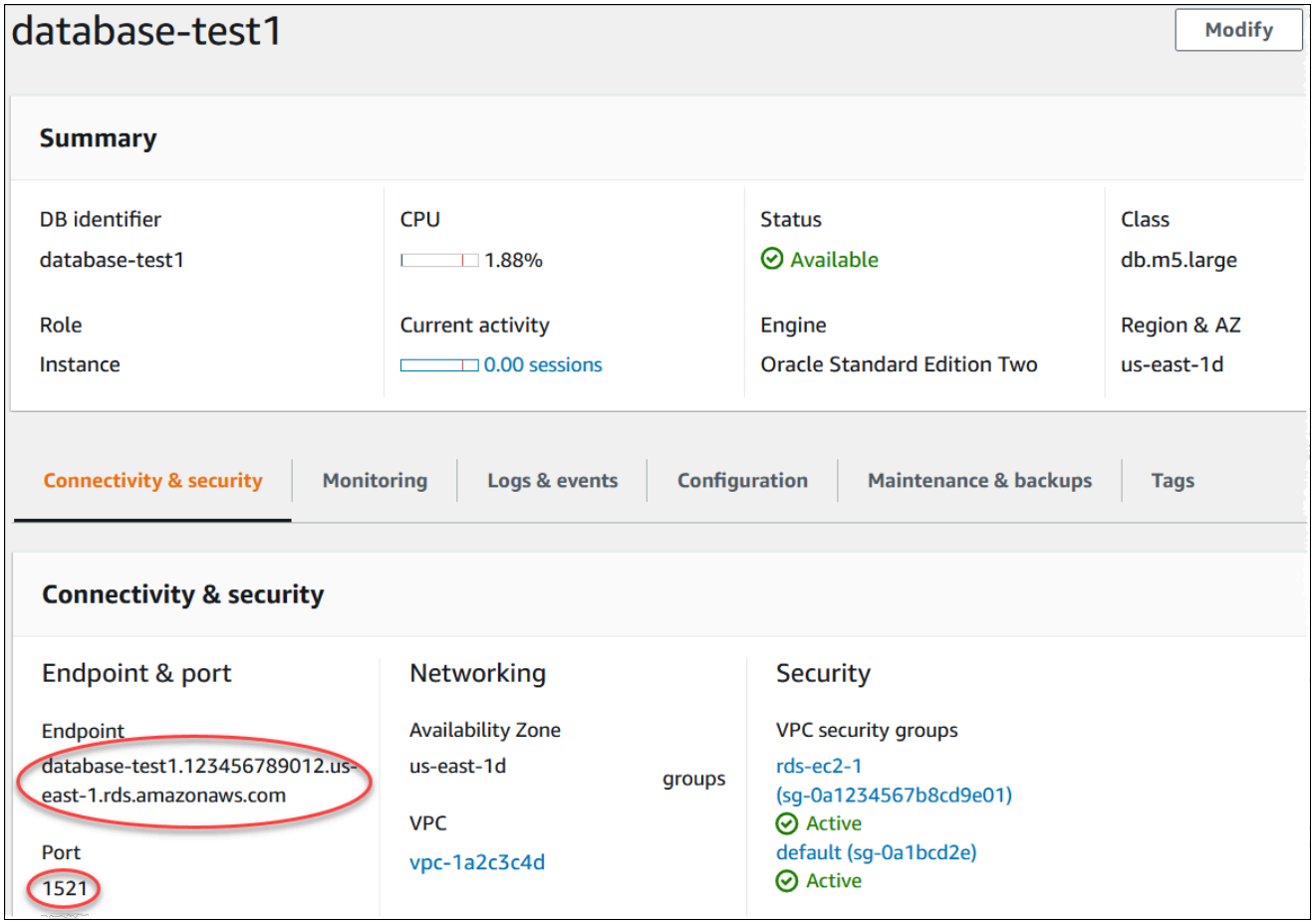
Vous pouvez trouver le point de terminaison d'une instance de base de données à l'aide de la console Amazon RDS ou de l'AWS CLI.

> **Note**
>
> Si vous utilisez l'authentification Kerberos, veuillez consulter Connexion à Oracle avec l'authentification Kerberos.

Console

Pour rechercher le point de terminaison à l'aide de la console

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console, choisissez la région AWS de votre instance de base de données.
3. Trouvez le numéro de port et le nom DNS pour votre instance de base de données.
    a. Choisissez Bases de données pour afficher une liste de vos instances de bases de données.
    b. Choisissez le nom de l'instance de base de données Oracle pour afficher les détails de l'instance.
    c. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



AWS CLI

Pour trouver le point de terminaison d'une instance de base de données Oracle à l'aide de l'AWS CLI, appelez la commande describe-db-instances.

Example Pour rechercher le point de terminaison à l'aide de l AWS CLI

```
aws rds describe-db-instances
```

Recherchez `Endpoint` dans la sortie pour trouver le nom DNS et le numéro de port de votre instance de base de données. La ligne `Address` de la sortie contient le nom DNS. Voici un exemple de sortie de point de terminaison JSON.

```
"Endpoint": {
    "HostedZoneId": "Z1PVIF0B656C1W",
    "Port": 3306,
    "Address": "myinstance.123456789012.us-west-2.rds.amazonaws.com"
```

```
},
```

> **Note**
>
> La sortie peut contenir des informations pour plusieurs instances de bases de données.

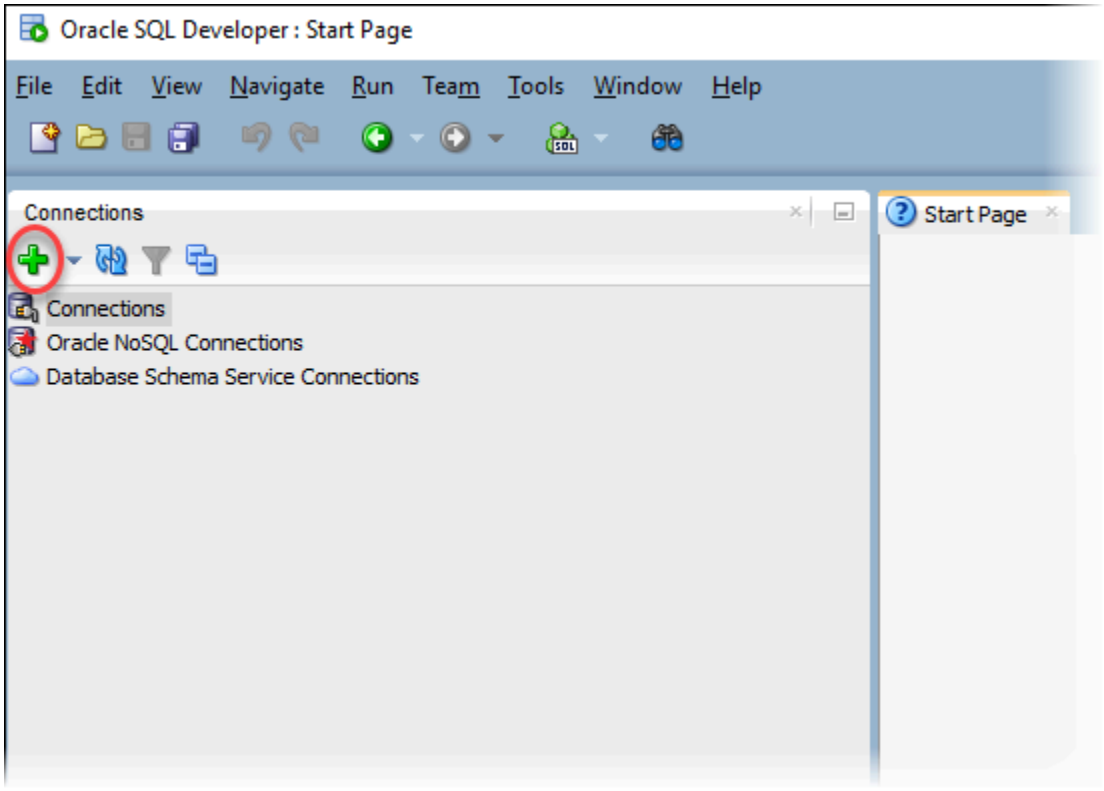
Connexion à votre instance de base de données à l'aide d'Oracle SQL Developer

Dans cette procédure, vous vous connectez à votre instance de base de données à l'aide d'Oracle SQL Developer. Pour télécharger une version autonome de cet utilitaire, consultez la page des téléchargements pour les développeur Oracle SQL.

Pour vous connecter à votre instance de base de données, vous avez besoin de son nom DNS et de son numéro de port. Pour plus d'informations sur la recherche du nom DNS et du numéro de port d'une instance de base de données, consultez Recherche du point de terminaison de votre instance de base de données RDS for Oracle.

Pour vous connecter à une instance de base de données à l'aide de SQL Developer

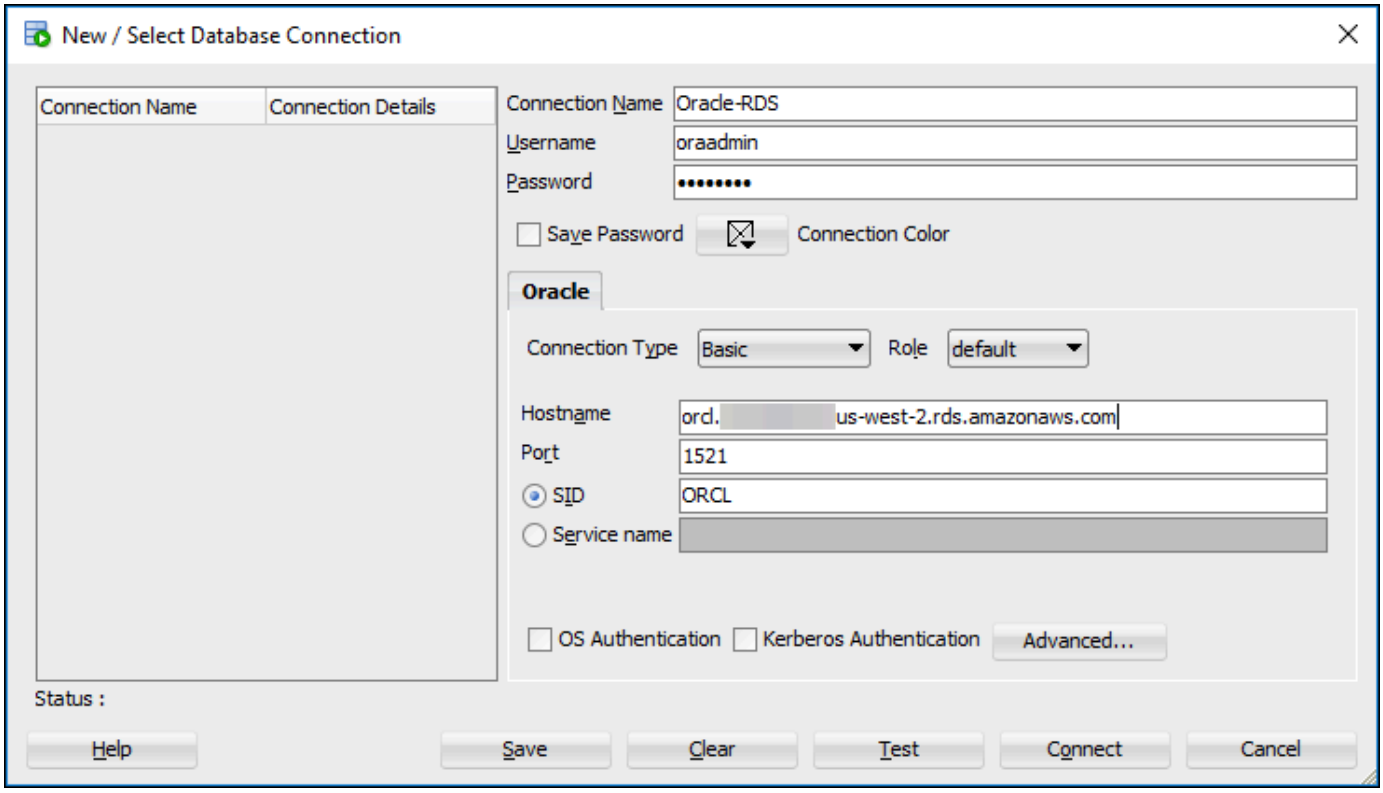
1. Démarrez Oracle SQL Developer.
2. Sous l'onglet Connexions, sélectionnez l'icône ajouter (+).



3. Dans la boîte de dialogue Nouvelle connexion de base de données/Sélectionner une connexion de base de données, entrez les informations relatives à votre instance de base de données :

   - Pour Nom de connexion, saisissez un nom qui décrit la connexion, tel que `Oracle-RDS`.
   - Pour Nom d'utilisateur, saisissez le nom de l'administrateur de base de données pour l'instance de base de données.
   - Pour Mot de passe, saisissez le mot de passe de l'administrateur de base de données.
   - Pour Nom d'hôte, saisissez le nom DNS de l'instance de base de données.
   - Pour Port, saisissez le numéro de port.
   - Pour SID, saisissez le nom de la base de données. Le nom de la base de données se trouve dans l'onglet Configuration de la page de détails de votre base de données.

   La boîte de dialogue dûment remplie doit se présenter comme suit :

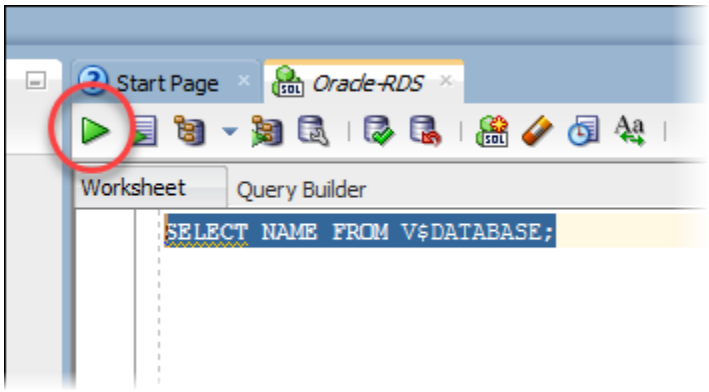


4. Choisissez Connexion.
5. Vous pouvez maintenant commencer à créer vos propres bases de données et à exécuter des requêtes sur votre instance de base de données et vos bases de données comme d'habitude. Pour exécuter une requête de test sur votre instance de base de données, procédez comme suit :

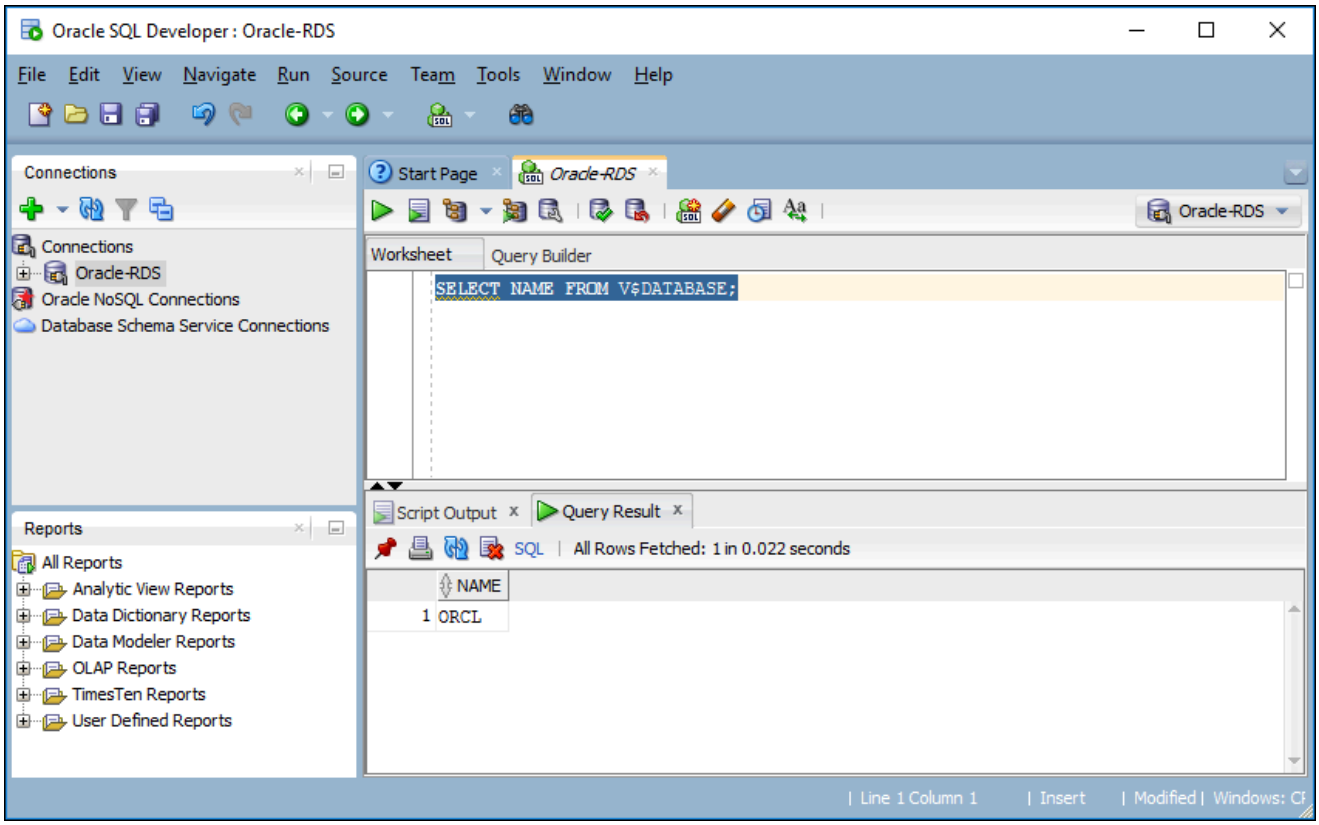
   a. Dans l'onglet Feuille de calcul de votre connexion, saisissez la requête SQL suivante.

   ```
   SELECT NAME FROM V$DATABASE;
   ```

   b. Choisissez l'icône exécuter pour exécuter la requête.



SQL Developer renvoie le nom de la base de données.



Connexion à votre instance de base de données à l'aide de SQL*Plus

Vous pouvez utiliser un utilitaire comme SQL*Plus pour vous connecter à une instance de base de données Amazon RDS exécutant Oracle. Pour télécharger Oracle Instant Client, qui inclut une version autonome de SQL*Plus, consultez la page des téléchargements d'Oracle Instant Client.

Pour vous connecter à votre instance de base de données, vous avez besoin de son nom DNS et de son numéro de port. Pour plus d'informations sur la recherche du nom DNS et du numéro de port d'une instance de base de données, consultez Recherche du point de terminaison de votre instance de base de données RDS for Oracle.

Example Pour se connecter à une instance de base de données Oracle à l'aide de SQL*Plus

Dans les exemples suivants, remplacez le nom d'utilisateur de l'administrateur de votre instance de base de données. Remplacez également le nom DNS de votre instance de base de données, puis incluez le numéro de port et le SID Oracle. La valeur SID est le nom de la base de données de l'instance de base de données que vous avez spécifié lors de la création de l'instance de base de données, pas le nom de l'instance de base de données.

Pour LinuxmacOS, ou Unix :

```
sqlplus 'user_name@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dns_name)(PORT=port))
(CONNECT_DATA=(SID=database_name)))'
```

Dans Windows :

```
sqlplus user_name@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=dns_name)(PORT=port))
(CONNECT_DATA=(SID=database_name)))
```

Vous devez visualiser des résultats similaires à ce qui suit.

```
SQL*Plus: Release 12.1.0.2.0 Production on Mon Aug 21 09:42:20 2017
```

Une fois que vous avez saisi le mot de passe de l'utilisateur, l'invite SQL apparaît.

```
SQL>
```

> **Note**
>
> La chaîne de connexion de format court (EZ Connect), comme `sqlplus USER/PASSWORD@longer-than-63-chars-rds-endpoint-here:1521/database-identifier`, peut comporter une limite de caractères maximale. Nous vous recommandons de ne pas l'utiliser pour vous connecter.

Considérations relatives aux groupes de sécurité

Pour vous connecter à votre instance de base de données, celle-ci doit être associée à un groupe de sécurité contenant les adresses IP et la configuration réseau nécessaires. Votre instance de base de données peut utiliser le groupe de sécurité par défaut. Si vous assignez un groupe de sécurité non configuré par défaut lors de la création de l'instance de base de données, le pare-feu empêche les connexions. Pour obtenir des informations sur la création d'un groupe de sécurité, consultez Contrôle d'accès par groupe de sécurité.

Une fois le groupe de sécurité créé, vous devez modifier votre instance de base de données pour l'associer au groupe de sécurité. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Vous pouvez améliorer la sécurité en utilisant SSL pour chiffrer les connexions à votre instance de base de données. Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).

Considérations relatives à l'architecture des processus

Le serveur traite les connexions utilisateur à une instance de base de données Oracle. Par défaut, l'instance de base de données Oracle utilise des processus serveur dédiés. Avec des processus serveur dédiés, chaque processus serveur ne traite qu'un seul processus utilisateur. Le cas échéant, vous pouvez configurer des processus serveur partagés. Avec des processus serveur partagés, chaque processus serveur peut traiter plusieurs processus utilisateur.

Vous pouvez envisager d'utiliser des processus serveur partagés quand un nombre élevé de sessions utilisateur consomment trop de mémoire sur le serveur. Vous pouvez également envisager d'utiliser des processus serveur partagés quand les sessions se connectent et déconnectent très souvent, ce qui entraîne un problème de performance. L'utilisation de processus serveur partagés présente également des inconvénients. Par exemple, elles peuvent peser sur les ressources d'UC et elles sont plus compliquées à configurer et administrer.

Pour plus d'informations sur les processus serveur dédiés et partagés, consultez About Dedicated and Shared Server Processes (Informations relatives aux processus de serveur dédiés et partagés) dans la documentation Oracle. Pour plus d'informations sur la configuration des processus serveur partagés sur une instance de base de données RDS for Oracle, veuillez consulter Comment dois-je configurer des instances de base de données Amazon RDS for Oracle Database afin qu'elles fonctionnent avec des serveurs partagés ? dans le Centre de connaissances.

Résolution des problèmes de connexion à votre instance de base de données Oracle

Les problèmes suivants peuvent survenir lors d'une tentative de connexion à votre instance de base de données Oracle.

Problème	Suggestions de dépannage
Impossible de se connecter à votre instance de base de données.	Pour une instance de base de données récemment créée, l'état de l'instance de base de données est creating (création) jusqu'à ce qu'elle soit prête à l'emploi. Lorsque l'état devient available (disponible), vous pouvez vous connecter à l'instance de base de données. En fonction de la classe d'instance de base de données

Problème	Suggestions de dépannage
	et de la quantité de stockage, la mise à disposition de la nouvelle instance de base de données peut prendre jusqu'à 20 minutes.
Impossible de se connecter à votre instance de base de données.	Si vous ne pouvez pas envoyer ou recevoir de communications sur le port que vous avez spécifié lors de la création de l'instance de base de données, vous ne pourrez pas vous connecter à l'instance de base de données. Vérifiez auprès de votre administrateur réseau si l'utilisation du port que vous avez spécifié pour votre instance de base de données autorise les communications entrantes et sortantes.
Impossible de se connecter à votre instance de base de données.	Les règles d'accès appliquées par votre pare-feu local et les adresses IP que vous avez autorisées à accéder à votre instance de base de données dans le groupe de sécurité de l'instance de base de données peuvent ne pas correspondre. Le problème est probablement lié aux règles de sortie ou d'entrée sur votre pare-feu. Vous pouvez ajouter ou modifier une règle entrante dans le groupe de sécurité. Pour Source, choisissez Mon IP. Cela autorise à accéder à l'instance de base de données à partir de l'adresse IP détectée dans votre navigateur. Pour plus d'informations, consultez Amazon VPC et Amazon RDS. Pour plus d'informations sur les groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité. Pour passer en revue le processus de configuration de règles de votre groupe de sécurité, consultez Tutoriel : Création d'un VPC à utiliser avec un) IPv4 .

Problème	Suggestions de dépannage
Connexion échouée, car l'hôte ou l'objet cible n'existe pas – Oracle, Erreur : ORA-12545	Vérifiez que vous avez spécifié le nom du serveur et le numéro de port correctement. Pour Nom du serveur, saisissez le nom DNS de la console. Pour plus d'informations sur la recherche du nom DNS et du numéro de port d'une instance de base de données, consultez Recherche du point de terminaison de votre instance de base de données RDS for Oracle.
Invalid username/ password; logon denied – Oracle, Error: ORA-01017	Vous avez pu atteindre l'instance de base de données, mais la connexion a été refusée. Cela est souvent dû au fait que le nom d'utilisateur ou le mot de passe fournis sont incorrects. Vérifiez le nom d'utilisateur et le mot de passe, puis réessayez.
TNS:listener does not currently know of SID given in connect descripto r - Oracle, ERROR: ORA-12505 (Actuellement, TNS:listener ne connaît pas le SID attribué dans le descripteur de connexion - Oracle, ERREUR: ORA-12505)	Assurez-vous que le SID correct est saisi. Le SID est le même que le nom de votre base de données. Recherchez le nom de la base de données dans l'onglet Configuration de la page Databases (Bases de données) de votre instance. Vous pouvez également trouver le nom de la base de données à l'aide de l'AWS CLI : `aws rds describe-db-instances --query 'DBInstances[*].[DBInstanceIdentifier,DBName]' --output text`

Pour de plus amples informations sur les problèmes de connexion, veuillez consulter Impossible de se connecter à l'instance de base de données Amazon RDS.

Modification des propriétés de connexion à l'aide des paramètres sqlnet.ora

Le fichier sqlnet.ora inclut des paramètres qui configurent les fonctions Oracle Net sur les serveurs et les clients de bases de données Oracle. L'utilisation de ces paramètres dans le fichier sqlnet.ora vous permet de modifier les propriétés des connexions à l'intérieur et à l'extérieur de la base de données.

Pour plus d'informations sur les raisons pouvant justifier la définition des paramètres sqlnet.ora, consultez Configuring Profile Parameters (Configurations de paramètres de profil) dans la documentation Oracle.

Définition des paramètres sqlnet.ora

Les groupes de paramètres Amazon RDS for Oracle incluent un sous-ensemble de paramètres sqlnet.ora. Vous les définissez de la même manière que les autres paramètres Oracle. C'est le préfixe `sqlnetora.` qui identifie les paramètres sqlnet.ora. Par exemple, dans un groupe de paramètres Oracle dans Amazon RDS, le paramètre sqlnet.ora `default_sdu_size` est `sqlnetora.default_sdu_size`.

Pour plus d'informations sur la gestion des groupes de paramètres et la définition des valeurs de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Paramètres sqlnet.ora pris en charge

Amazon RDS prend en charge les paramètres sqlnet.ora ci-après. Les modifications apportées aux paramètres sqlnet.ora dynamiques sont appliquées immédiatement.

Paramètre	Valeurs valides	Statique/ Dynamiq	Description
`sqlnetora.default_sdu_size`	`512` sur `20971`	Répartiti on dynamiq	Taille, en octets, de l'unité de données de session (SDU). L'unité de données de session (SDU) est la quantité de données placées dans une mémoire tampon et envoyées dans le réseau en une seule fois.
`sqlnetora.diag_adr_enabled`	`ON`, `OFF`	Répartiti on dynamiq	Valeur qui active ou désactive le traçage Automatic Diagnostic Repository (ADR). `ON` indique que le traçage de fichier ADR est utilisé.

Paramètre	Valeurs valides	Statique/ Dynamiq	Description
			OFF indique que le traçage de fichier non-ADR est utilisé.
sqlnetora.recv_buf_size	8192 sur 268435	Répartition dynamiq	Limite de l'espace tampon pour les opérations de réception de sessions, prise en charge par les protocoles TCP/IP, TCP/IP avec SSL et SDP.
sqlnetora.send_buf_size	8192 sur 268435	Répartition dynamiq	Limite de l'espace tampon pour les opérations d'envoi de sessions, prise en charge par les protocoles TCP/IP, TCP/IP avec SSL et SDP.
sqlnetora.sqlnet.allowed_logon_version_client	8, 10, 11, 12	Répartition dynamiq	Version de protocole d'authentification minimum autorisée pour les clients, et les serveurs agissant en tant que clients, pour établir une connexion aux instances de bases de données Oracle.
sqlnetora.sqlnet.allowed_logon_version_server	8, 9, 10, 11, 12, 12a	Répartition dynamiq	Version de protocole d'authentification minimum autorisée pour établir une connexion aux instances de bases de données Oracle.
sqlnetora.sqlnet.expire_time	0 sur 1440	Répartition dynamiq	Intervalle, en minutes, pour envoyer une instruction visant à vérifier que les connexions client-serveur sont actives.

Paramètre	Valeurs valides	Statique/ Dynamiq	Description
sqlnetora.sqlnet.inbound_connect_timeout	0 ou 10 à 7200	Répartition dynamiq	Délai, en secondes, imparti à un client pour se connecter au serveur de bases de données et pour fournir les informations d'authentification nécessaires.
sqlnetora.sqlnet.outbound_connect_timeout	0 ou 10 à 7200	Répartition dynamiq	Délai, en secondes, imparti à un client pour établir une connexion Oracle Net avec une instance de base de données.
sqlnetora.sqlnet.recv_timeout	0 ou 10 à 7200	Répartition dynamiq	Délai, en secondes, imparti à un serveur de bases de données pour attendre les données client après l'établissement d'une connexion.
sqlnetora.sqlnet.send_timeout	0 ou 10 à 7200	Répartition dynamiq	Délai, en secondes, imparti à un serveur de bases de données pour terminer une opération d'envoi aux clients après l'établissement d'une connexion.
sqlnetora.tcp.connect_timeout	0 ou 10 à 7200	Répartition dynamiq	Délai, en secondes, imparti à un client pour établir une connexion TCP avec un serveur de bases de données.

Paramètre	Valeurs valides	Statique/ Dynamiq	Description
`sqlnetora.trace_level_server`	`0`, `4`, `10`, `16`, `OFF`, `USER`, `ADMIN`, `SUPPOF`	Répartiti on dynamiq	Pour le traçage non-ADR, active le traçage du serveur à un niveau défini ou le désactive.

La valeur par défaut de chaque paramètre sqlnet.ora pris en charge est la valeur par défaut de la base de données Oracle pour la version.

Affichage des paramètres sqlnet.ora

Vous pouvez consulter les paramètres de sqlnet.ora et leurs paramètres à l'aide du AWS Management Console AWS CLI, du ou d'un client SQL.

Affichage des paramètres sqlnet.ora à l'aide de la console

Pour plus d'informations sur l'affichage des paramètres d'un groupe de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Dans les groupes de paramètres Oracle, c'est le préfixe `sqlnetora.` qui identifie les paramètres sqlnet.ora.

Affichage des paramètres sqlnet.ora à l'aide de l AWS CLI

Pour afficher les paramètres sqlnet.ora configurés dans un groupe de paramètres Oracle, utilisez la commande describe-db-parameters. AWS CLI

Pour afficher tous les paramètres sqlnet.ora d'une instance de base de données Oracle, appelez la commande download-db-log-file-portion. AWS CLI Spécifiez l'identifiant de l'instance de base de données, le nom du fichier journal et le type de sortie.

Example

Le code suivant répertorie tous les paramètres sqlnet.ora pour `mydbinstance`.

Pour LinuxmacOS, ou Unix :

```
aws rds download-db-log-file-portion \
    --db-instance-identifier mydbinstance \
    --log-file-name trace/sqlnet-parameters \
    --output text
```

Dans Windows :

```
aws rds download-db-log-file-portion ^
    --db-instance-identifier mydbinstance ^
    --log-file-name trace/sqlnet-parameters ^
    --output text
```

Affichage des paramètres sqlnet.ora à l'aide d'un client SQL

Une fois que vous vous êtes connecté à l'instance de base de données Oracle dans un client SQL, la requête suivante répertorie les paramètres sqlnet.ora.

```
SELECT * FROM TABLE
   (rdsadmin.rds_file_util.read_text_file(
        p_directory => 'BDUMP',
        p_filename  => 'sqlnet-parameters'));
```

Pour plus d'informations sur la connexion à une instance de base de données Oracle dans un client SQL, consultez Connexion à votre instance de base de données Oracle.

Sécurisation des connexions d'instance de base de données Oracle

Amazon RDS for Oracle prend en charge les connexions chiffrées SSL/TLS ainsi que l'option Oracle Native Network Encryption (NNE) pour chiffrer les connexions entre votre application et votre instance de base de données Oracle. Pour plus d'informations sur l'option Oracle Native Network Encryption, consultez Oracle NNE (Native Network Encryption).

Rubriques

- Utilisation de SSL avec une instance de base de données RDS for Oracle
- Mise à jour des applications pour se connecter aux instances de bases de données Oracle à l'aide des nouveaux certificats SSL/TLS
- Utilisation d'un chiffrement NNE (Native Network Encryption) avec une instance de base de données RDS for Oracle
- Configuration de l'authentification Kerberos pour Amazon RDS for Oracle
- Configuration de l'accès UTL_HTTP à l'aide de certificats et d'un portefeuille Oracle

Utilisation de SSL avec une instance de base de données RDS for Oracle

Secure Sockets Layer (SSL) est un protocole de norme industrielle utilisé pour sécuriser les connexions réseau entre client et serveur. Après la version 3.0 de SSL, le nom du protocole est devenu Transport Layer Security (TLS), mais nous y faisons souvent référence en tant que SSL. Amazon RDS prend en charge le chiffrement SSL pour les instances de base de données Oracle. En utilisant SSL, vous pouvez chiffrer une connexion entre votre application client et votre instance de base de données Oracle. Le support SSL est disponible dans toutes les Régions AWS pour Oracle.

Pour activer le chiffrement SSL (Secure Sockets Layer) pour une instance de base de données Oracle, ajoutez l'option Oracle SSL au groupe d'options associé avec l'instance de base de données. Amazon RDS utilise un deuxième port, comme l'exige Oracle, pour les connexions SSL. Cette approche permet aussi bien aux communications en texte clair que celles à chiffrement SSL de se produire simultanément entre une instance de base de données et un client Oracle. Par exemple, vous pouvez utiliser le port avec une communication en texte clair pour communiquer avec d'autres ressources à l'intérieur d'un VPC, tout en utilisant le port avec une communication à chiffrement SSL pour communiquer avec des ressources extérieures au VPC.

Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).

> **Note**
>
> Vous ne pouvez pas utiliser le chiffrement SSL et le chiffrement Oracle NNE sur la même instance de base de données. Avant d'utiliser le chiffrement SSL, vous devez désactiver tout autre chiffrement de connexion.

Mise à jour des applications pour se connecter aux instances de bases de données Oracle à l'aide des nouveaux certificats SSL/TLS

Le 13 janvier 2023, Amazon RDS a publié de nouveaux certificats d'autorité de certification (CA) pour la connexion à vos instances de base de données RDS à l'aide du protocole Secure Socket Layer ou Transport Layer Security (SSL/TLS). Vous trouverez ci-après des informations sur la mise à jour de vos applications afin d'utiliser les nouveaux certificats.

Cette rubrique peut vous aider à déterminer si des applications clientes utilisent un protocole SSL/TLS pour se connecter à vos instances de bases de données.

> **Important**
>
> Lorsque vous modifiez le certificat d'une instance de base de données Amazon RDS for Oracle, seul le processus d'écoute de base de données est redémarré. L'instance de base de données n'est pas redémarrée. Les connexions de base de données existantes ne sont pas affectées, mais les nouvelles connexions rencontrent des erreurs pendant une brève période au cours du redémarrage du processus d'écoute.
> Nous vous recommandons de redémarrer votre base de données Oracle pour éviter toute erreur de connexion.

> **Note**
>
> Pour les applications clientes qui utilisent un protocole SSL/TLS pour se connecter à vos instances de bases de données, vous devez mettre à jour les magasins d'approbations de vos applications clientes qui incluent les nouveaux certificats de l'autorité de certification.

Une fois que vous avez mis à jour les certificats de l'autorité de certification dans les magasins d'approbations des applications clientes, vous pouvez soumettre les certificats de vos instances de

bases de données à une rotation. Nous vous recommandons vivement de tester ces procédures dans un environnement de développement ou intermédiaire avant de les implémenter dans vos environnements de production.

Pour de plus amples informations sur la rotation de certificats, veuillez consulter Rotation de votre certificat SSL/TLS. Pour en savoir plus sur le téléchargement de certificats, consultez . Pour de plus amples informations sur l'utilisation des protocoles SSL/TLS avec les instances de bases de données Oracle, veuillez consulter Oracle Secure Sockets Layer (SSL).

Rubriques

- Déterminer si les applications se connectent via SSL
- Mise à jour du magasin d'approbations de votre application
- Exemple de code Java pour l'établissement de connexions SSL

Déterminer si les applications se connectent via SSL

Si votre instance de base de données Oracle utilise un groupe d'options avec l'option `SSL` ajoutée, vous pouvez utiliser SSL. Procédez à la vérification en suivant les instructions dans Liste des options et des paramètres d'options pour un groupe d'options. Pour plus d'informations sur l'option `SSL`, veuillez consulter Oracle Secure Sockets Layer (SSL).

Vérifiez le journal d'écouteur afin de déterminer s'il existe des connexions SSL. Voici un exemple de sortie dans un journal d'écouteur.

```
date time * (CONNECT_DATA=(CID=(PROGRAM=program)
(HOST=host)(USER=user))(SID=sid)) *
(ADDRESS=(PROTOCOL=tcps)(HOST=host)(PORT=port)) * establish * ORCL * 0
```

Lorsque `PROTOCOL` possède la valeur `tcps` pour une entrée, il affiche une connexion SSL. Cependant, quand `HOST` est `127.0.0.1`, vous pouvez ignorer l'entrée. Les connexions provenant de `127.0.0.1` sont un agent de gestion locale sur l'instance de base de données. Il ne s'agit pas de connexions SSL externes. Par conséquent, vous avez des applications se connectant à l'aide d'un protocole SSL si vous voyez des entrées de journal d'écouteur pour lesquelles `PROTOCOL` est `tcps` et `HOST` n'est pas `127.0.0.1`.

Pour consulter le journal du récepteur, vous pouvez le publier sur Amazon CloudWatch Logs. Pour de plus amples informations, veuillez consulter Publication de journaux Oracle sur Amazon CloudWatch Logs.

Mise à jour du magasin d'approbations de votre application

Vous pouvez mettre à jour le magasin d'approbations pour les applications qui utilisent SQL*Plus ou JDBC dans le cadre de connexions SSL/TLS.

Mise à jour du magasin d'approbations des applications pour SQL*Plus

Vous pouvez mettre à jour le magasin d'approbations pour les applications qui utilisent SQL*Plus dans le cadre de connexions SSL/TLS.

> **Note**
>
> Lors de la mise à jour du magasin d'approbations, vous pouvez conserver les certificats plus anciens en complément de l'ajout des nouveaux certificats.

Pour mettre à jour le magasin d'approbations pour les applications SQL*Plus

1. Téléchargez le nouveau certificat racine qui fonctionne pour toutes les AWS régions et placez le fichier dans le `ssl_wallet` répertoire.

   Pour plus d'informations sur le téléchargement du certificat racine, consultez .
2. Exécutez la commande suivante afin de mettre à jour le portefeuille Oracle.

   ```
   prompt>orapki wallet add -wallet $ORACLE_HOME/ssl_wallet -trusted_cert -cert
         $ORACLE_HOME/ssl_wallet/ssl-cert.pem -auto_login_only
   ```

   Remplacez le nom du fichier par celui du fichier téléchargé.
3. Exécutez la commande suivante pour vérifier que le portefeuille a été installé avec succès.

   ```
   prompt>orapki wallet display -wallet $ORACLE_HOME/ssl_wallet
   ```

   La sortie doit contenir les lignes suivantes :

   ```
   Trusted Certificates:
   Subject: CN=Amazon RDS Root 2019 CA,OU=Amazon RDS,O=Amazon Web Services\,
    Inc.,L=Seattle,ST=Washington,C=US
   ```

Mise à jour du magasin d'approbations de votre application pour JDBC

Vous pouvez mettre à jour le magasin d'approbations pour les applications qui utilisent JDBC dans le cadre de connexions SSL/TLS.

Pour plus d'informations sur le téléchargement du certificat racine, consultez .

Pour obtenir des exemples de scripts qui importent des certificats, consultez Exemple de script pour importer les certificats dans votre magasin d'approbations.

Exemple de code Java pour l'établissement de connexions SSL

L'exemple de code suivant montre comment configurer la connexion SSL à l'aide de JDBC.

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.Properties;

public class OracleSslConnectionTest {
    private static final String DB_SERVER_NAME = "<dns-name-provided-by-amazon-rds>";
    private static final Integer SSL_PORT = "<ssl-option-port-configured-in-option-
group>";
    private static final String DB_SID = "<oracle-sid>";
    private static final String DB_USER = "<user name>";
    private static final String DB_PASSWORD = "<password>";
    // This key store has only the prod root ca.
    private static final String KEY_STORE_FILE_PATH = "<file-path-to-keystore>";
    private static final String KEY_STORE_PASS = "<keystore-password>";

    public static void main(String[] args) throws SQLException {
        final Properties properties = new Properties();
        final String connectionString = String.format(
                "jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=%s)(PORT=
%d))(CONNECT_DATA=(SID=%s)))",
                DB_SERVER_NAME, SSL_PORT, DB_SID);
        properties.put("user", DB_USER);
        properties.put("password", DB_PASSWORD);
        properties.put("oracle.jdbc.J2EE13Compliant", "true");
        properties.put("javax.net.ssl.trustStore", KEY_STORE_FILE_PATH);
        properties.put("javax.net.ssl.trustStoreType", "JKS");
        properties.put("javax.net.ssl.trustStorePassword", KEY_STORE_PASS);
        final Connection connection = DriverManager.getConnection(connectionString,
 properties);
```

```
        // If no exception, that means handshake has passed, and an SSL connection can
 be opened
    }
}
```

> **Important**
>
> Une fois que vous avez déterminé que vos connexions à la base de données utilisent le protocole SSL/TLS et que vous avez mis à jour le magasin de confiance des applications, vous pouvez mettre à jour votre base de données pour utiliser les rds-ca-rsa certificats 2048-g1. Pour obtenir des instructions, veuillez consulter l'étape 3 dans Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données.

Utilisation d'un chiffrement NNE (Native Network Encryption) avec une instance de base de données RDS for Oracle

Oracle Database propose deux méthodes de chiffrement des données sur le réseau : le chiffrement NNE (Native Network Encryption) et le protocole TLS (Transport Layer Security). NNE est une fonctionnalité de sécurité exclusive d'Oracle, tandis que le protocole TLS est une norme industrielle. RDS for Oracle prend en charge NNE pour toutes les éditions d'Oracle Database.

Le chiffrement NNE présente les avantages suivants par rapport au protocole TLS :

- Vous pouvez contrôler NNE sur le client et le serveur à l'aide des paramètres de l'option NNE :
  - `SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS` et `SQLNET.ALLOW_WEAK_CRYPTO`
  - `SQLNET.CRYPTO_CHECKSUM_CLIENT` et `SQLNET.CRYPTO_CHECKSUM_SERVER`
  - `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT` et `SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER`
  - `SQLNET.ENCRYPTION_CLIENT` et `SQLNET.ENCRYPTION_SERVER`
  - `SQLNET.ENCRYPTION_TYPES_CLIENT` et `SQLNET.ENCRYPTION_TYPES_SERVER`
- Dans la plupart des cas, vous n'avez pas besoin de configurer votre client ou votre serveur. En revanche, le protocole TLS vous oblige à configurer à la fois le client et le serveur.
- Aucun certificat n'est requis. Dans le protocole TLS, le serveur nécessite un certificat (qui finit par expirer) et le client nécessite un certificat racine approuvé pour l'autorité de certification qui a émis le certificat du serveur.

Pour activer le chiffrement NNE pour une instance de base de données Oracle, ajoutez l'option Oracle NNE au groupe d'options associé à l'instance de base de données. Pour plus d'informations, consultez Oracle NNE (Native Network Encryption).

> **Note**
>
> Vous ne pouvez pas utiliser à la fois NNE et TLS sur la même instance de base de données.

Configuration de l'authentification Kerberos pour Amazon RDS for Oracle

Vous pouvez désormais utiliser l'authentification Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de base de données Amazon RDS for Oracle. Dans cette configuration, votre instance de base de données fonctionne avec AWS Directory Service for Microsoft Active Directory, également appelé AWS Managed Microsoft AD. Lorsque les utilisateurs s'authentifient auprès d'une instance de base de données RDS for Oracle jointe au domaine d'approbation, les demandes d'authentification sont transmises à l'annuaire de domaine que vous créez avec AWS Directory Service.

Vous pouvez gagner du temps et de l'argent en conservant toutes les informations d'identification dans le même annuaire. Vous avez un endroit centralisé de stockage et de gestion des informations d'identification pour plusieurs instances de base de données. Un annuaire peut également améliorer votre profil de sécurité global.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions de RDS for Oracle avec authentification Kerberos, consultez Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS.

> **Note**
>
> L'authentification Kerberos n'est pas prise en charge pour les classes d'instances de base de données qui sont obsolètes pour les instances de base de données RDS for Oracle. Pour plus d'informations, consultez RDS pour les classes d'instance de base de données Oracle.

Rubriques

- Configuration de Kerberos pour les instances de base de données Oracle
- Gestion d'une instance de base de données dans un domaine
- Connexion à Oracle avec l'authentification Kerberos

Configuration de Kerberos pour les instances de base de données Oracle

Utilisé AWS Directory Service for Microsoft Active Directory, également appelé AWS Managed Microsoft AD, pour configurer l'authentification Kerberos pour une instance de base de données Oracle. Pour configurer l'authentification Kerberos, procédez comme suit :

- Étape 1 : créer un répertoire à l'aide du AWS Managed Microsoft AD
- Étape 2 : Créer une approbation
- Étape 3 : configurer les autorisations IAM pour Amazon RDS
- Étape 4 : Créer et configurer des utilisateurs
- Étape 5 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données
- Étape 6 : Créer ou modifier une instance de base de données Oracle
- Étape 7 : Créer les connexions Oracle d'authentification Kerberos
- Étape 8 : Configurer un client Oracle

> **Note**
>
> Au cours de la configuration, RDS crée un utilisateur de base de données Oracle nommé *`managed_service_user`* @ *`example.com`* avec le `CREATE SESSION` privilège « Où *`example.com`* est votre nom de domaine ». Cet utilisateur correspond à l'utilisateur créé par Directory Service dans votre annuaire Active Directory géré. Régulièrement, RDS utilise les informations d'identification fournies par Directory Service pour se connecter à votre base de données Oracle. Par la suite, RDS détruit immédiatement les tickets mis en cache.

Étape 1 : créer un répertoire à l'aide du AWS Managed Microsoft AD

AWS Directory Service crée un Active Directory entièrement géré dans le AWS cloud. Lorsque vous créez un AWS Managed Microsoft AD annuaire, il AWS Directory Service crée deux contrôleurs de

domaine et des serveurs DNS (Domain Name System) en votre nom. Les serveurs de répertoire sont créés dans des sous-réseaux différents d'un VPC. Cette redondance permet de s'assurer que votre annuaire reste accessible, y compris en cas de défaillance.

Lorsque vous créez un AWS Managed Microsoft AD répertoire, il AWS Directory Service exécute les tâches suivantes en votre nom :

- Configuration d'un annuaire Active Directory dans le VPC.
- Création d'un compte d'administrateur d'annuaire avec le nom d'utilisateur Admin et le mot de passe spécifié. Ce compte est utilisé pour gérer votre annuaire.

  > **Note**
  >
  > Assurez-vous d'enregistrer ce mot de passe. AWS Directory Service ne le stocke pas. Vous pouvez le réinitialiser, mais vous ne pouvez pas le récupérer.

- Création d'un groupe de sécurité pour les contrôleurs de l'annuaire.

Lorsque vous lancez un AWS Managed Microsoft AD, AWS crée une unité organisationnelle (UO) qui contient tous les objets de votre répertoire. Cette unité d'organisation, qui porte le nom NetBIOS que vous avez saisi lorsque vous avez créé votre annuaire, est située dans la racine du domaine. La racine du domaine est détenue et gérée par AWS.

Le compte administrateur créé avec votre AWS Managed Microsoft AD annuaire dispose d'autorisations pour les activités administratives les plus courantes de votre unité d'organisation :

- Création, mise à jour et suppression des utilisateurs
- Ajouter des ressources à votre domaine, comme des serveurs de fichiers ou d'impression, puis attribuer des autorisations pour ces ressources aux utilisateurs dans votre unité d'organisation
- Créez des conteneurs OUs et des conteneurs supplémentaires
- Déléguer des autorités
- Restaurer des objets supprimés de la corbeille Active Directory
- Exécuter les PowerShell modules Windows AD et DNS sur le service Web Active Directory

Le compte Admin dispose également de droits pour exécuter les activités suivantes au niveau du domaine :

- Gérer les configurations DNS (ajouter, supprimer ou mettre à jour des enregistrements, des zones et des redirecteurs)
- Afficher les journaux d'évènements DNS
- Afficher les journaux d'évènements de sécurité

Pour créer le répertoire, utilisez l'API AWS Management Console AWS CLI, le ou l' AWS Directory Service API. Veillez à ouvrir les ports sortants appropriés sur le groupe de sécurité afin que l'annuaire puisse communiquer avec l'instance de base de données Oracle.

Pour créer un répertoire avec AWS Managed Microsoft AD

1. Connectez-vous à la AWS Directory Service console AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/directoryservicev2/.
2. Dans le panneau de navigation, choisissez Directories (Répertoires), puis Set up Directory (Configurer un répertoire).
3. Choisissez AWS Managed Microsoft AD. AWS Managed Microsoft AD est la seule option que vous pouvez actuellement utiliser avec Amazon RDS.
4. Entrez les informations suivantes :

   Nom de DNS de l'annuaire

   Nom complet de l'annuaire, par exemple **`corp.example.com`**.

   Nom NetBIOS de l'annuaire

   Nom court de l'annuaire, par exemple **`CORP`**.

   Description de l'annuaire

   (Facultatif) Une description de l'annuaire.

   Mot de passe administrateur

   Mot de passe de l'administrateur de l'annuaire. Le processus de création d'un annuaire crée un compte d'administrateur avec le nom d'utilisateur Admin et ce mot de passe.

   Le mot de passe de l'administrateur de l'annuaire ne peut pas contenir le terme « admin ». Le mot de passe est sensible à la casse et doit comporter entre 8 et 64 caractères. Il doit également contenir au moins un caractère de trois des quatre catégories suivantes :

   - Lettres minuscules (a–z)

- Lettres majuscules (A–Z)
- Chiffres (0–9)
- Caractères non alphanumériques (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)

Confirmer le mot de passe

Saisissez à nouveau le mot de passe de l'administrateur.

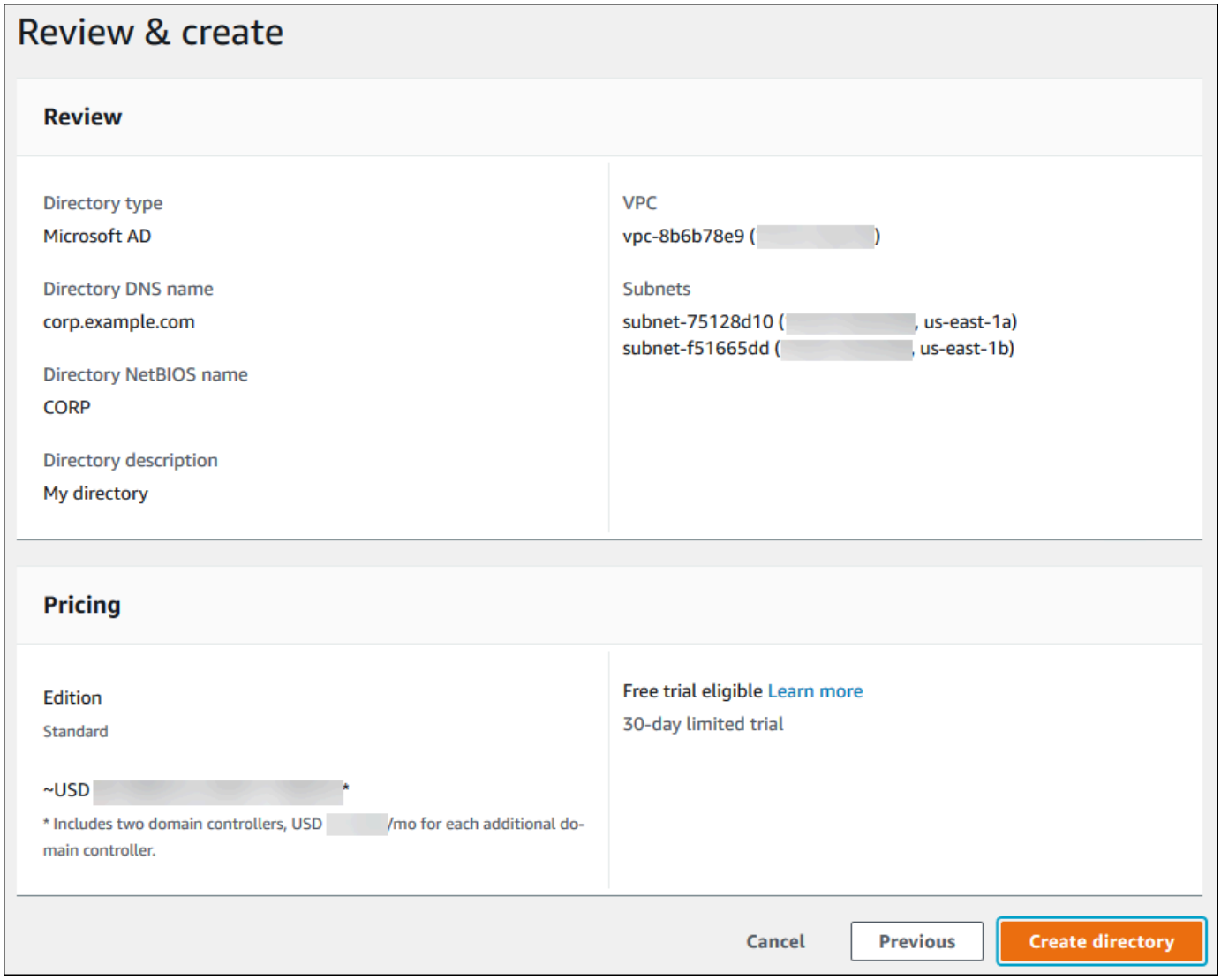
5. Choisissez Suivant.
6. Entrez les informations suivantes dans la section Networking (Réseaux), puis choisissez Suivant (Next) :

   VPC

   VPC de l'annuaire. Créez l'instance de base de données Oracle dans ce même VPC.

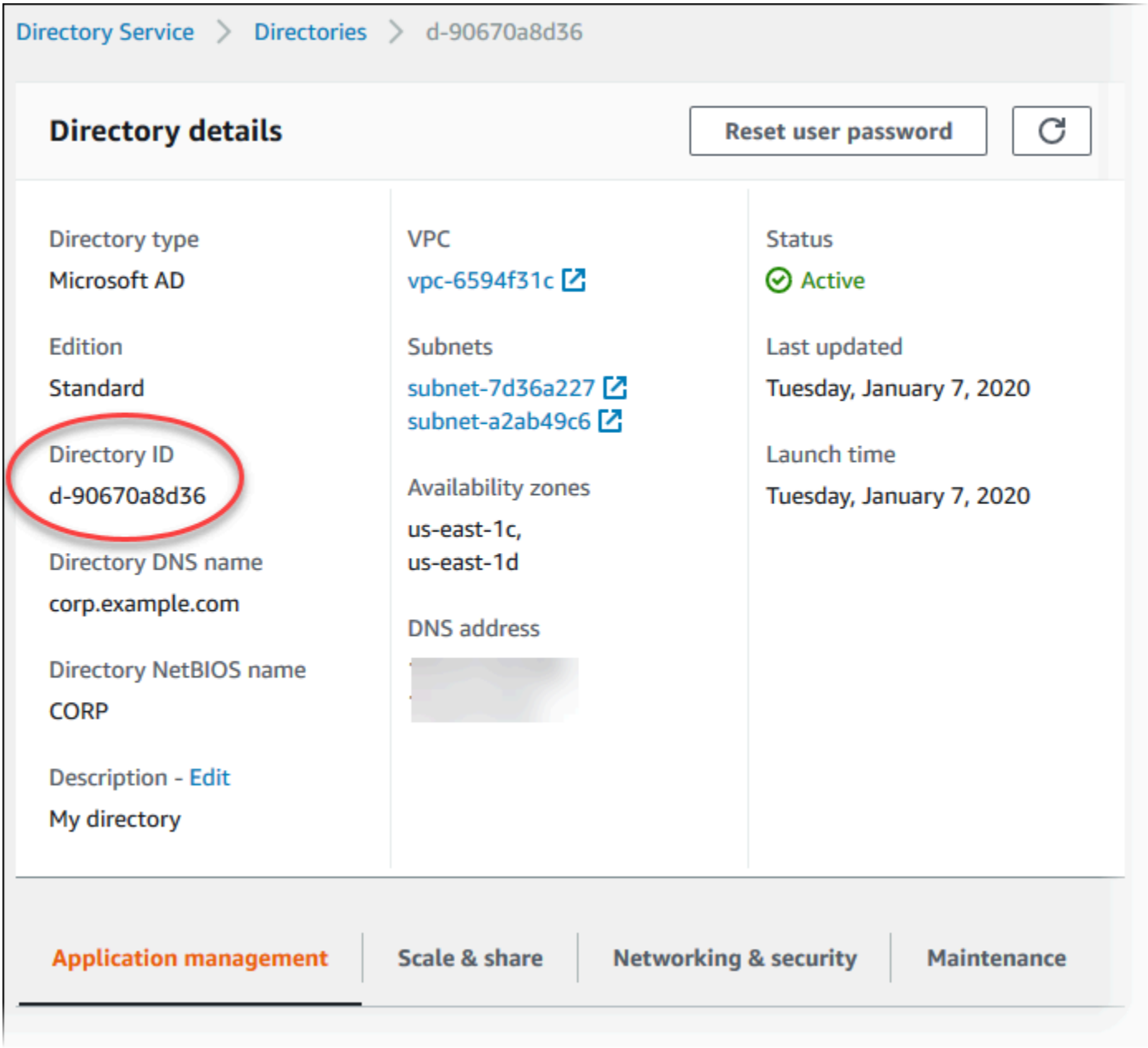
   Sous-réseaux

   Sous-réseaux pour les serveurs d'annuaires. Les deux sous-réseaux doivent être dans des zones de disponibilité différentes.
7. Vérifiez les informations concernant l'annuaire et effectuez les modifications nécessaires. Lorsque les informations sont correctes, choisissez Create directory (Créer le répertoire).



La création de l'annuaire prend plusieurs minutes. Lorsqu'il est créé, la valeur du champ Status (Statut) devient Active (Actif).

Pour consulter les informations relatives à votre annuaire, choisissez le nom de l'annuaire dans la liste. Notez la valeur ID de l'annuaire, car vous avez besoin de cette valeur lorsque vous créez ou modifiez votre instance de base de données Oracle.



Étape 2 : Créer une approbation

Si vous prévoyez d'utiliser AWS Managed Microsoft AD uniquement, passez à[Étape 3 : configurer les autorisations IAM pour Amazon RDS](#).

Pour activer l'authentification Kerberos à l'aide de votre Active Directory autogéré, vous devez créer une relation d'approbation forestière entre votre Active Directory autogéré et celui AWS Managed Microsoft AD créé à l'étape précédente. La confiance peut être unidirectionnelle, c'est-à-dire qu'elle AWS Managed Microsoft AD fait confiance à Active Directory autogéré. L'approbation peut également être bidirectionnelle. Dans ce cas, les deux Active Directory s'approuvent mutuellement. Pour plus

d'informations sur la configuration des approbations forestières à l'aide AWS Directory Service de la section Quand créer une relation de confiance dans le Guide d'AWS Directory Service administration.

Étape 3 : configurer les autorisations IAM pour Amazon RDS

Pour appeler AWS Directory Service pour vous, Amazon RDS a besoin d'un rôle IAM qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess` Ce rôle permet à Amazon RDS d'appeler l' AWS Directory Service.

> **Note**
>
> Pour que le rôle autorise l'accès, le point de terminaison AWS Security Token Service (AWS STS) doit être activé correctement Région AWS pour votre Compte AWS. AWS STS les points de terminaison sont actifs par défaut dans tous les cas Régions AWS, et vous pouvez les utiliser sans autre action. Pour plus d'informations, consultez la section Activation et désactivation AWS STS dans et Région AWS dans le guide de l'utilisateur IAM.

Création d'un rôle IAM

Lorsque vous créez une instance de base de données à l'aide de AWS Management Console, et que l'utilisateur de la console a l'`iam:CreateRole`autorisation, la console `rds-directoryservice-kerberos-access-role` se crée automatiquement. Sinon, vous devez créer le rôle IAM manuellement. Lorsque vous créez un rôle IAM manuellement`Directory Service`, choisissez et associez la politique AWS gérée `AmazonRDSDirectoryServiceAccess` à celui-ci.

Pour plus d'informations sur la création de rôles IAM pour un service, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'utilisateur IAM.

> **Note**
>
> Le rôle IAM utilisé pour l'authentification Windows pour RDS for Microsoft SQL Server ne peut pas être utilisé pour RDS for Oracle.

Création manuelle d'une politique de confiance IAM

Vous pouvez également créer des politiques de ressources avec les autorisations obligatoires au lieu d'utiliser la politique IAM gérée `AmazonRDSDirectoryServiceAccess`. Spécifiez `directoryservice.rds.amazonaws.com` et `rds.amazonaws.com` comme principaux.

Afin de limiter les autorisations octroyées par Amazon RDS à un autre service pour une ressource spécifique, nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans les politiques de ressources. Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet d'une ressource Amazon RDS. Pour de plus amples informations, veuillez consulter Prévention des problèmes d'adjoint confus entre services.

L'exemple suivant montre comment utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` pour dans Amazon RDS afin d'éviter le problème de l'adjoint confus.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.amazonaws.com",
          "rds.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:rds:us-east-1:123456789012:db:mydbinstance"
        },
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

Pour les régions optionnelles, vous devez également inclure un principal de service pour cette région sous la forme de`directoryservice.rds.region_name.amazonaws.com`. Par exemple, dans la région Afrique (Le Cap), appliquez la politique de confiance suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.amazonaws.com",
          "directoryservice.rds.af-south-1.amazonaws.com",
          "rds.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "ArnLike": {
          "aws:SourceArn": "arn:aws:rds:af-south-1:123456789012:db:mydbinstance"
        },
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

Le rôle doit également avoir la politique IAM suivante.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
```

```
      "Action": [
        "ds:DescribeDirectories",
        "ds:AuthorizeApplication",
        "ds:UnauthorizeApplication",
        "ds:GetAuthorizedApplicationDetails"
      ],
    "Effect": "Allow",
    "Resource": "*"
    }
  ]
}
```

Étape 4 : Créer et configurer des utilisateurs

Vous pouvez créer des utilisateurs à l'aide de l'outil Active Directory Users and Computers, qui fait partie des outils Active Directory Domain Services et Active Directory Lightweight Directory Services. Dans ce cas, les utilisateurs sont des personnes ou des entités qui ont accès à votre annuaire.

Pour créer des utilisateurs dans un AWS Directory Service annuaire, vous devez être connecté à une EC2 instance Amazon basée sur Windows qui est membre de l' AWS Directory Service annuaire. Parallèlement, vous devez être connecté en tant qu'utilisateur disposant de privilèges pour créer des utilisateurs. Pour de plus amples informations sur la création d'utilisateurs dans votre annuaire Microsoft Active Directory, veuillez consulter Gérer les utilisateurs et les groupes dans AWS Managed Microsoft AD dans le Guide d'administration AWS Directory Service .

Étape 5 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données

Si vous avez l'intention de rechercher l'annuaire et l'instance de base de données dans le même VPC, ignorez cette étape et passez à Étape 6 : Créer ou modifier une instance de base de données Oracle.

Si vous prévoyez de localiser le répertoire et l'instance de base de données dans différents AWS comptes ou VPCs de configurer le trafic inter-VPC à l'aide du peering VPC ou de Transit Gateway.AWS La procédure suivante active le trafic entre les utilisateurs de VPCs l'appairage VPC. Suivez les instructions de Qu'est-ce que l'appairage de VPC ? dans le Guide de l'appairage Amazon Virtual Private Cloud.

Pour activer le trafic entre VPC à l'aide de l'appairage de VPC

1. Configurez les règles de routage de VPC appropriées afin de veiller à ce que le trafic réseau puisse être acheminé dans les deux sens.

2. Assurez-vous que le groupe de sécurité de l'instance de base de données puisse recevoir le trafic entrant depuis le groupe de sécurité de cet annuaire. Pour plus d'informations, consultez Meilleures pratiques pour AWS Managed Microsoft AD dans le Guide d'administration AWS Directory Service .
3. Assurez-vous qu'il n'existe aucune règle de liste de contrôle d'accès (ACL) pour bloquer le trafic.

Si le répertoire appartient à un autre AWS compte, vous devez le partager.

Pour partager le répertoire entre AWS comptes

1. Commencez à partager le répertoire avec le AWS compte dans lequel l'instance de base de données sera créée en suivant les instructions du Tutoriel : Partage de votre AWS Managed Microsoft AD répertoire pour une connexion de EC2 domaine fluide dans le Guide d'AWS Directory Service administration.
2. Connectez-vous à la AWS Directory Service console à l'aide du compte de l'instance de base de données et assurez-vous que le domaine possède le `SHARED` statut requis avant de continuer.
3. Lorsque vous êtes connecté à la AWS Directory Service console à l'aide du compte de l'instance de base de données, notez la valeur de l'ID du répertoire. Vous utilisez cet ID pour joindre l'instance de base de données au domaine.

Étape 6 : Créer ou modifier une instance de base de données Oracle

Créez ou modifiez une instance de base de données Oracle en vue de son utilisation avec votre annuaire. Vous pouvez utiliser la console, la CLI ou l'API RDS pour associer une instance de base de données à un annuaire. Vous pouvez effectuer cette opération de différentes manières :

- Créez une nouvelle instance de base de données Oracle à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération Create DBInstance RDS API.

  Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données Oracle existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstanceModify RDS API.

  Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.

- Restaurez une instance de base de données Oracle à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération DBInstance Restore DBSnapshot From RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.

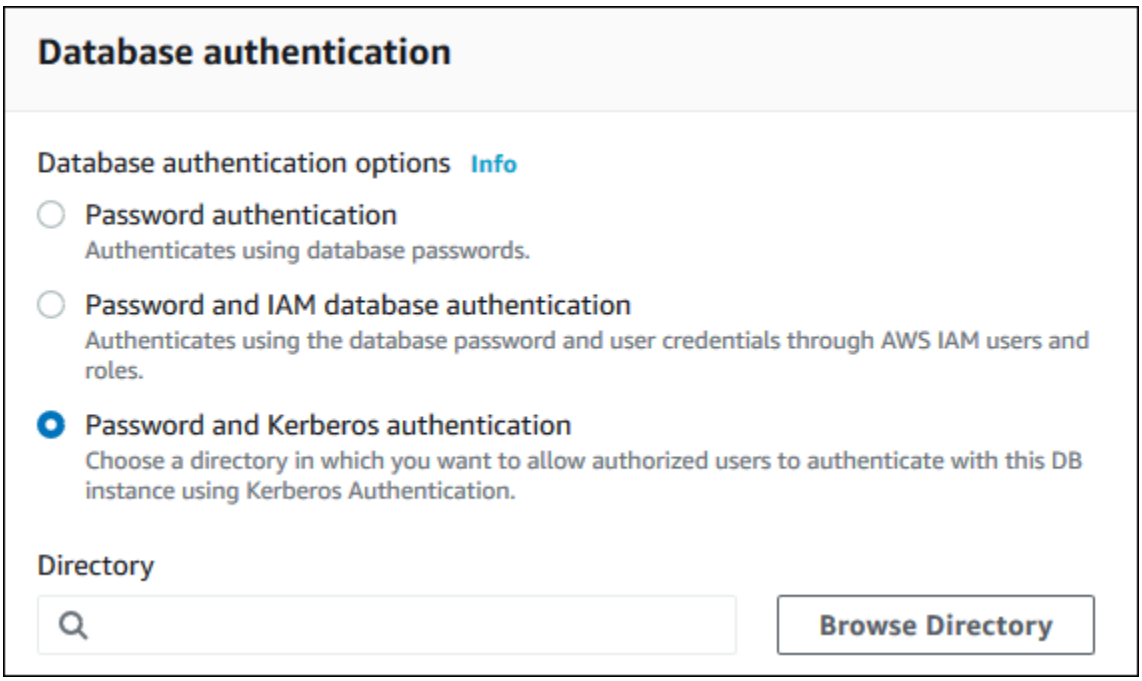
- Restaurez une instance de base de données Oracle à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération Restore DBInstance ToPointInTime RDS API.

  Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

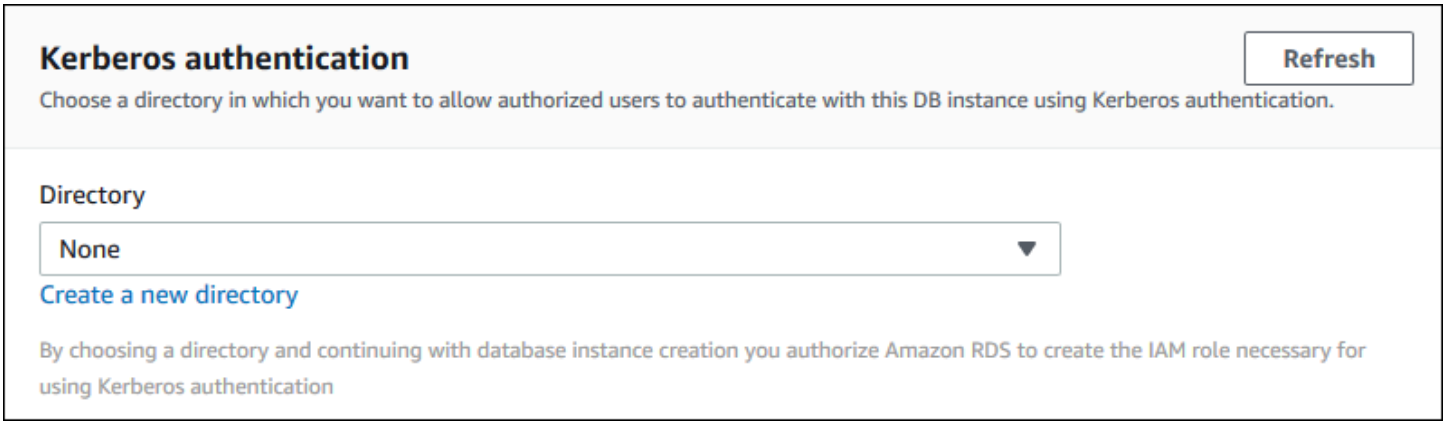
L'authentification Kerberos est uniquement prise en charge pour les instances de base de données Oracle dans un VPC. L'instance de base de données peut être dans le même VPC que l'annuaire ou dans un VPC différent. Lors de la création ou de la modification de l'instance de base de données, procédez comme suit :

- Fournissez l'identifiant du domaine (identifiant `d-*`) qui a été généré lors de la création de votre annuaire.
- Fournissez le nom du rôle IAM que vous avez créé.
- Veillez à ce que le groupe de sécurité de l'instance de base de données puisse recevoir le trafic entrant depuis le groupe de sécurité de l'annuaire et envoyer le trafic sortant vers l'annuaire.

Lorsque vous utilisez la console pour créer une instance de base de données, choisissez Mot de passe et authentification Kerberos dans la section Authentification de base de données. Choisissez Browse Directory (Parcourir les répertoires), puis sélectionnez le répertoire, ou choisissez Create a new directory (Créer un nouveau répertoire).



Lorsque vous utilisez la console pour modifier ou restaurer une instance de base de données, choisissez le répertoire dans la section Kerberos authentication (Authentification Kerberos) ou choisissez Create a new directory (Créer un nouveau répertoire).



Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de base de données puisse utiliser le répertoire que vous avez créé :

- Pour le paramètre `--domain`, vous devez indiquer l'identifiant du domaine (identifiant « d-* ») généré lors de la création de l'annuaire.
- Pour le paramètre `--domain-iam-role-name`, utilisez le rôle que vous avez créé qui utilise la politique IAM gérée `AmazonRDSDirectoryServiceAccess`.

Par exemple, la commande de CLI suivante modifie une instance de base de données de façon à utiliser un annuaire.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --domain d-ID \
    --domain-iam-role-name role-name
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --domain d-ID ^
    --domain-iam-role-name role-name
```

Important

Si vous modifiez une instance de base de données de façon à activer l'authentification Kerberos, redémarrez l'instance de base de données après avoir effectué la modification.

Note

*MANAGED_SERVICE_USER*est un compte de service dont le nom est généré aléatoirement par Directory Service for RDS. Lors de la configuration de l'authentification Kerberos, RDS pour Oracle crée un utilisateur portant le même nom et lui attribue le privilège `CREATE SESSION`. L'utilisateur de la base de données Oracle est identifié de manière externe comme *MANAGED_SERVICE_USER@EXAMPLE.COM* suit : où se *EXAMPLE.COM* trouve le nom de votre domaine. Régulièrement, RDS utilise les informations d'identification fournies par Directory Service pour se connecter à votre base de données Oracle. Par la suite, RDS détruit immédiatement les tickets mis en cache.

Étape 7 : Créer les connexions Oracle d'authentification Kerberos

Utilisez les informations d'identification de l'utilisateur principal Amazon RDS for vous connecter à l'instance de base de données Oracle comme vous le faites pour n'importe quelle instance de base

de données. L'instance de base de données est jointe au AWS Managed Microsoft AD domaine. Ainsi, vous pouvez configurer des connexions et des utilisateurs Oracle à partir des utilisateurs Microsoft Active Directory de votre domaine. Pour gérer les autorisations de base de données, vous pouvez octroyer et annuler les autorisations Oracle standard pour ces connexions.

Pour autoriser un utilisateur Microsoft Active Directory à s'authentifier avec Oracle

1. Connectez l'instance de base de données Oracle à l'aide de vos informations d'identification de l'utilisateur principal Amazon RDS.
2. Créez un utilisateur authentifié en externe dans la base de données Oracle.

   Dans l'exemple suivant, remplacez *KRBUSER@CORP.EXAMPLE.COM* par le nom d'utilisateur et le nom de domaine.

   ```
   CREATE USER "KRBUSER@CORP.EXAMPLE.COM" IDENTIFIED EXTERNALLY;
   GRANT CREATE SESSION TO "KRBUSER@CORP.EXAMPLE.COM";
   ```

   Les utilisateurs (personnes et applications) de votre domaine peuvent désormais se connecter à l'instance de base de données Oracle à partir d'un ordinateur client joint au domaine à l'aide de l'authentification Kerberos.

Étape 8 : Configurer un client Oracle

Pour configurer un client Oracle, vous devez vous conformer aux exigences suivantes :

- Créez un fichier de configuration nommé krb5.conf (Linux) ou krb5.ini (Windows) pour pointer vers le domaine. Configurez le client Oracle pour qu'il utilise ce fichier de configuration.
- Vérifiez que le trafic peut circuler entre l'hôte du client et AWS Directory Service via le port DNS 53 via TCP/UDP, les ports Kerberos (88 et 464 pour les ports gérés AWS Directory Service) via TCP et le port LDAP 389 sur TCP.
- Vérifiez que le trafic peut circuler entre l'hôte du client et l'instance de base de données via le port de la base de données.

Vous trouverez ci-dessous un exemple de contenu pour AWS Managed Microsoft AD.

```
[libdefaults]
 default_realm = EXAMPLE.COM
[realms]
```

```
 EXAMPLE.COM = {
  kdc = example.com
  admin_server = example.com
 }
[domain_realm]
 .example.com = CORP.EXAMPLE.COM
 example.com = CORP.EXAMPLE.COM
```

Vous trouverez ci-dessous un exemple de contenu pour Microsoft AD sur site. Dans votre fichier krb5.conf ou krb5.ini, remplacez par *on-prem-ad-server-name* le nom de votre serveur AD local.

```
[libdefaults]
 default_realm = ONPREM.COM
[realms]
 AWSAD.COM = {
  kdc = awsad.com
  admin_server = awsad.com
 }
 ONPREM.COM = {
  kdc = on-prem-ad-server-name
  admin_server = on-prem-ad-server-name
 }
[domain_realm]
 .awsad.com = AWSAD.COM
 awsad.com= AWSAD.COM
 .onprem.com = ONPREM.COM
 onprem.com= ONPREM.COM
```

> **Note**
>
> Après avoir configuré votre fichier krb5.ini ou krb5.conf, nous vous recommandons de redémarrer le serveur.

Vous trouverez ci-après un exemple de contenu sqlnet.ora pour une configuration SQL*Plus :

```
SQLNET.AUTHENTICATION_SERVICES=(KERBEROS5PRE,KERBEROS5)
SQLNET.KERBEROS5_CONF=path_to_krb5.conf_file
```

Pour obtenir un exemple de configuration SQL Developer, veuillez consulter Document 1609359.1 sur le support Oracle.

Gestion d'une instance de base de données dans un domaine

Vous pouvez utiliser la console, la CLI ou l'API RDS pour gérer votre instance de base de données et sa relation avec votre Microsoft Active Directory. Par exemple, vous pouvez associer un Microsoft Active Directory de façon à activer l'authentification Kerberos. Vous pouvez également annuler l'association d'un Microsoft Active Directory pour désactiver l'authentification Kerberos. Vous pouvez également transférer une instance de base de données vers une autre afin qu'elle soit authentifiée en externe par un Microsoft Active Directory.

Par exemple, la CLI vous permet d'effectuer les actions suivantes :

- Pour retenter l'activation de l'authentification Kerberos en cas d'échec d'appartenance, utilisez la commande de CLI modify-db-instance et spécifiez l'ID d'annuaire d'appartenance actuelle pour l'option `--domain`.
- Pour désactiver l'authentification Kerberos sur une instance de base de données, utilisez la commande de CLI modify-db-instance et spécifiez `none` pour l'option `--domain`.
- Pour transférer une instance de base de données d'un domaine vers un autre, utilisez la commande de CLI modify-db-instance et spécifiez l'identifiant du nouveau domaine pour l'option `--domain`.

Affichage du statut de l'appartenance au domaine

Après la création ou la modification de votre instance de base de données, cette dernière devient un membre du domaine. Vous pouvez consulter le statut de l'appartenance au domaine pour l'instance de base de données dans la console ou en exécutant la commande de CLI describe-db-instances. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- `kerberos-enabled` – L'instance de base de données a l'authentification Kerberos activée.
- `enabling-kerberos` – AWS est le processus d'activation de l'authentification Kerberos sur cette instance de base de données.
- `pending-enable-kerberos` – L'activation de l'authentification Kerberos est en attente sur cette instance de base de données.
- `pending-maintenance-enable-kerberos` – AWS tentera d'activer l'authentification Kerberos sur cette instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `pending-disable-kerberos` – La désactivation de l'authentification Kerberos est en attente sur cette instance de base de données.

- `pending-maintenance-disable-kerberos` – AWS tentera de désactiver l'authentification Kerberos sur cette instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `enable-kerberos-failed` – Un problème de configuration a empêché AWS d'activer l'authentification Kerberos sur l'instance de base de données. Corrigez le problème de configuration avant de réémettre la commande de modification de l'instance de base de données.
- `disabling-kerberos` – AWS est en train de désactiver l'authentification Kerberos sur cette instance de base de données.

Une demande d'activation de l'authentification Kerberos peut échouer à cause d'un problème de connectivité réseau ou d'un rôle IAM incorrect. Si la tentative d'activation de l'authentification Kerberos échoue lorsque vous créez ou modifiez une instance de base de données, vérifiez d'abord que vous utilisez le rôle IAM correct. Ensuite, modifiez l'instance de base de données pour rejoindre le domaine.

> **Note**
>
> Seule l'authentification Kerberos avec Amazon RDS for Oracle envoie le trafic aux serveurs DNS du domaine. Toutes les autres demandes DNS sont traitées comme un accès réseau sortant sur vos instances de bases de données exécutant Oracle. Pour de plus amples informations sur l'accès réseau sortant avec Amazon RDS for Oracle, veuillez consulter Configuration d'un DNS serveur personnalisé.

Clés Kerberos à rotation forcée

Une clé secrète est partagée entre AWS Managed Microsoft AD et une instance de base de données Amazon RDS for Oracle. Cette clé subit une rotation automatique tous les 45 jours. Vous pouvez utiliser la procédure Amazon RDS suivante pour forcer la rotation de cette clé.

```
SELECT rdsadmin.rdsadmin_kerberos_auth_tasks.rotate_kerberos_keytab AS TASK_ID FROM
 DUAL;
```

> **Note**
>
> Dans une configuration de réplica en lecture, cette procédure est uniquement disponible sur l'instance de base de données source et non sur le réplica en lecture.

L'instruction SELECT renvoie l'ID de la tâche dans un type de données VARCHAR2. Vous pouvez consulter le statut d'une tâche en cours dans un fichier bdump. Les fichiers bdump se trouvent dans le répertoire /rdsdbdata/log/trace. Chaque nom de fichier bdump a le format suivant.

```
dbtask-task-id.log
```

Vous pouvez afficher le résultat en affichant le fichier de sortie de la tâche.

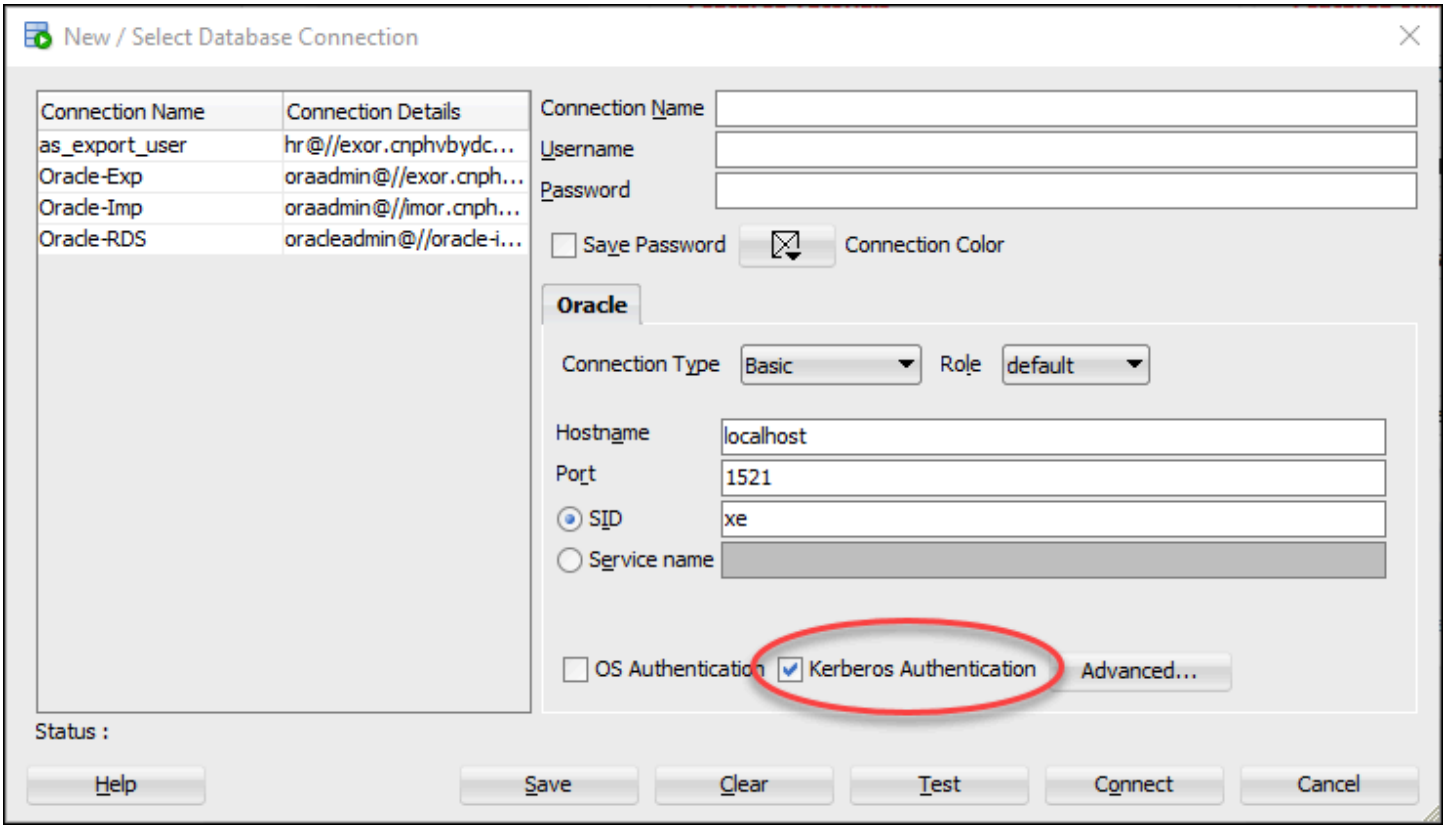
```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-task-
id.log'));
```

Remplacez *task-id* par l'ID de tâche renvoyé par la procédure.

> **Note**
>
> Les tâches sont exécutées de manière asynchrone.

Connexion à Oracle avec l'authentification Kerberos

Cette section suppose que vous avez configuré votre client Oracle de la façon décrite dans Étape 8 : Configurer un client Oracle. Pour vous connecter à la base de données Oracle avec l'authentification Kerberos, connectez-vous à l'aide du type d'authentification Kerberos. Par exemple, après avoir lancé Oracle SQL Developer, choisissez Kerberos Authentication comme type d'authentification, comme indiqué dans l'exemple suivant.



Pour vous connecter à Oracle avec l'authentification Kerberos avec SQL*Plus :

1. A partir d'une invite de commande, exécutez la commande suivante :

   ```
   kinit username
   ```

   Remplacez *username* par le nom d'utilisateur puis, à l'invite, entrez le mot de passe stocké dans le Microsoft Active Directory pour l'utilisateur.
2. Ouvrez SQL*Plus et connectez-vous à l'aide du nom et du numéro de port DNS pour l'instance de base de données Oracle.

   Pour de plus amples informations sur la connexion à l'instance de base de données Oracle dans SQL*Plus, veuillez consulter Connexion à votre instance de base de données à l'aide de SQL*Plus.

> **Tip**
>
> Si vous utilisez un cache Windows natif, vous pouvez également définir le `SQLNET.KERBEROS5_CC_NAME` paramètre sur `OSMSFT://` ou `MSLSA` dans le fichier

sqlnet.ora pour utiliser les informations d'identification stockées dans Microsoft Active Directory.

Configuration de l'accès UTL_HTTP à l'aide de certificats et d'un portefeuille Oracle

Amazon RDS prend en charge l'accès réseau sortant sur votre RDS pour les instances de base de données Oracle. Pour connecter votre instance de base de données au réseau, vous pouvez utiliser les packages PL/SQL suivants :

`UTL_HTTP`

Ce package effectue des appels HTTP depuis SQL et PL/SQL. Vous pouvez l'utiliser pour accéder à des données sur Internet via HTTP. Pour plus d'informations, consultez UTL_HTTP dans la documentation Oracle.

`UTL_TCP`

Ce package fournit la fonction d'accès côté client TCP/IP dans PL/SQL. Ce package est utile pour les applications PL/SQL qui utilisent les protocoles Internet et les e-mails. Pour plus d'informations, consultez UTL_TCP dans la documentation Oracle.

`UTL_SMTP`

Ce package fournit des interfaces aux commandes SMTP qui permettent à un client d'envoyer des e-mails à un serveur SMTP. Pour plus d'informations, consultez UTL_SMTP dans la documentation Oracle.

En effectuant les tâches suivantes, vous pouvez configurer `UTL_HTTP.REQUEST` pour qu'il fonctionne avec des sites web nécessitant des certificats d'authentification client pendant la liaison SSL. Vous pouvez également configurer l'authentification par mot de passe pour permettre à `UTL_HTTP` d'accéder aux sites web en modifiant les commandes de génération de portefeuilles Oracle et la procédure `DBMS_NETWORK_ACL_ADMIN.APPEND_WALLET_ACE`. Pour plus d'informations, veuillez consulter DBMS_NETWORK_ACL_ADMIN dans la documentation de la base de données Oracle.

> **Note**
>
> Vous pouvez adapter les tâches suivantes pour UTL_SMTP afin d'envoyer des e-mails via SSL/TLS (y compris Amazon Simple Email Service).

Rubriques

- Considérations relatives à la configuration de l'accès UTL_HTTP
- Étape 1 : Obtenir le certificat racine d'un site web
- Étape 2 : Créer un portefeuille Oracle
- Étape 3 : Télécharger votre portefeuille Oracle sur votre instance RDS for Oracle
- Étape 4 : Accorder des autorisations utilisateur pour le portefeuille Oracle
- Étape 5 : Configurer l'accès à un site web à partir de votre instance de base de données
- Étape 6 : Tester les connexions de votre instance de base de données à un site web

Considérations relatives à la configuration de l'accès UTL_HTTP

Avant de configurer l'accès, tenez compte des points suivants :

- Vous pouvez utiliser le protocole SMTP avec l'option UTL_MAIL. Pour plus d'informations, consultez Oracle UTL _ MAIL.
- Le nom DNS (Domain Name Server) de l'hôte distant doit avoir l'une des caractéristiques suivantes :
  - être publiquement résolu ;
  - être le point de terminaison d'une instance de base de données Amazon RDS ;
  - pouvant être résolu via un serveur DNS personnalisé. Pour plus d'informations, consultez Configuration d'un DNS serveur personnalisé.
  - être le nom DNS privé d'une instance Amazon EC2 dans le même VPC ou un VPC appairé. Dans ce cas, assurez-vous que le nom peut être résolu via un serveur DNS personnalisé. De même, pour utiliser le DNS fourni par Amazon, vous pouvez activer l'attribut `enableDnsSupport` dans les paramètres VPC et activer la prise en charge de la résolution DNS pour la connexion d'appairage de VPC. Pour plus d'informations, consultez Prise en charge du DNS dans votre VPC et Modification de votre connexion d'appairage de VPC.

- Pour vous connecter en toute sécurité à des ressources SSL/TLS distantes, nous vous recommandons de créer et de charger des portefeuilles Oracle personnalisés. En utilisant l'intégration de Amazon S3 à la fonction Amazon RDS for Oracle, vous pouvez télécharger un portefeuille depuis Amazon S3 vers des instances de base de données Oracle. Pour plus d'informations sur l'intégration de Amazon S3 pour Oracle, consultez Intégration Amazon S3.
- Vous pouvez établir des liens de base de données entre des instances de base de données Oracle via un point de terminaison SSL/TLS si l'option Oracle SSL est configurée pour chaque instance. Aucune autre configuration n'est requise. Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).

Étape 1 : Obtenir le certificat racine d'un site web

Pour que l'instance de base de données RDS pour Oracle établisse des connexions sécurisées à un site Web, ajoutez le certificat CA racine. Amazon RDS utilise le certificat racine pour signer le certificat de site web au portefeuille Oracle.

Vous pouvez obtenir le certificat racine de différentes manières. Par exemple, vous pouvez effectuer les opérations suivantes :

1. Utilisez un serveur web pour visiter le site web sécurisé par le certificat.
2. Téléchargez le certificat racine utilisé pour la signature.

Pour les services AWS, les certificats racines résident généralement dans le Référentiel des services d'approbation Amazon.

Étape 2 : Créer un portefeuille Oracle

Créez un portefeuille Oracle contenant à la fois les certificats du serveur web et les certificats d'authentification client. L'instance Oracle RDS utilise le certificat du serveur web pour établir une connexion sécurisée au site web. Le site web a besoin du certificat client pour authentifier l'utilisateur de la base de données Oracle.

Vous pouvez configurer des connexions sécurisées sans utiliser de certificats clients pour l'authentification. Dans ce cas, vous pouvez ignorer les étapes du keystore Java dans la procédure suivante.

Pour créer un portefeuille Oracle

1. Placez les certificats racine et client dans un seul répertoire, puis passez dans ce répertoire.
2. Convertissez le certificat client .p12 en keystore Java.

   > **Note**
   >
   > Si vous n'utilisez pas de certificats client pour l'authentification, vous pouvez ignorer cette étape.

   L'exemple suivant convertit le certificat client nommé *client_certificate.p12* vers le keystore Java nommé *client_keystore.jks*. Le keystore est alors inclus dans le portefeuille Oracle. Le mot de passe du keystore est *P12PASSWORD*.

   ```
   orapki wallet pkcs12_to_jks -wallet ./client_certificate.p12 -
   jksKeyStoreLoc ./client_keystore.jks -jksKeyStorepwd P12PASSWORD
   ```

3. Créez un répertoire pour votre portefeuille Oracle qui est différent du répertoire de certificat.

   L'exemple suivant crée le répertoire `/tmp/wallet`.

   ```
   mkdir -p /tmp/wallet
   ```

4. Créez un portefeuille Oracle dans votre répertoire de portefeuille.

   L'exemple suivant définit le mot de passe du portefeuille Oracle sur *P12PASSWORD*, qui est le même mot de passe utilisé par le keystore Java lors d'une étape précédente. L'utilisation du même mot de passe est pratique, mais pas nécessaire. Le paramètre `-auto_login` active la fonction de connexion automatique, de sorte que vous n'avez pas besoin de spécifier un mot de passe chaque fois que vous souhaitez y accéder.

   > **Note**
   >
   > Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

```
orapki wallet create -wallet /tmp/wallet -pwd P12PASSWORD -auto_login
```

5. Ajoutez le keystore Java à votre portefeuille Oracle.

> **Note**
>
> Si vous n'utilisez pas de certificats client pour l'authentification, vous pouvez ignorer cette étape.

L'exemple suivant ajoute le keystore *client_keystore.jks* au portefeuille Oracle nommé */tmp/wallet*. Dans cet exemple, vous spécifiez le même mot de passe pour le keystore Java et pour le portefeuille Oracle.

```
orapki wallet jks_to_pkcs12 -wallet /tmp/wallet -pwd P12PASSWORD -
keystore ./client_keystore.jks -jkspwd P12PASSWORD
```

6. Ajoutez le certificat racine de votre site web cible au portefeuille Oracle.

L'exemple suivant ajoute un certificat nommé *Root_CA.cer*.

```
orapki wallet add -wallet /tmp/wallet -trusted_cert -cert ./Root_CA.cer -
pwd P12PASSWORD
```

7. Ajoutez les certificats intermédiaires.

L'exemple suivant ajoute un certificat nommé *Intermediate.cer*. Répétez cette étape autant de fois que nécessaire pour charger tous les certificats intermédiaires.

```
orapki wallet add -wallet /tmp/wallet -trusted_cert -cert ./Intermediate.cer -
pwd P12PASSWORD
```

8. Vérifiez que votre nouveau portefeuille Oracle possède les certificats requis.

```
orapki wallet display -wallet /tmp/wallet -pwd P12PASSWORD
```

Étape 3 : Télécharger votre portefeuille Oracle sur votre instance RDS for Oracle

Au cours de cette étape, vous chargez votre portefeuille Oracle sur Amazon S3, puis vous téléchargez le portefeuille depuis Amazon S3 vers votre instance RDS for Oracle.

Pour télécharger votre portefeuille Oracle sur votre instance de base de données RDS for Oracle

1. Respectez les conditions préalables de l'intégration de Amazon S3 à Oracle et ajoutez l'option `S3_INTEGRATION` à votre instance de base de données Oracle. Assurez-vous que le rôle IAM pour l'option a accès au compartiment Amazon S3 que vous utilisez.

   Pour plus d'informations, consultez Intégration Amazon S3.
2. Connectez-vous à votre instance de base de données en tant qu'utilisateur principal, puis créez un répertoire Oracle qui contiendra le portefeuille Oracle.

   L'exemple suivant crée un répertoire Oracle nommé *WALLET_DIR*.

   ```
   EXEC rdsadmin.rdsadmin_util.create_directory('WALLET_DIR');
   ```

   Pour plus d'informations, consultez Création et suppression de répertoires dans l'espace de stockage de données principal.
3. Chargez le portefeuille Oracle dans votre compartiment Amazon S3.

   Vous pouvez utiliser n'importe quelle technique de téléchargement prise en charge.
4. Si vous téléchargez à nouveau un portefeuille Oracle, supprimez le portefeuille existant. Sinon, passez à l'étape suivante.

   L'exemple suivant supprime le portefeuille existant, nommé *cwallet.sso*.

   ```
   EXEC UTL_FILE.FREMOVE ('WALLET_DIR','cwallet.sso');
   ```
5. Téléchargez le portefeuille depuis votre compartiment Amazon S3 vers l'instance de base de données Oracle.

   L'exemple suivant télécharge le portefeuille nommé *cwallet.sso* à partir du compartiment Amazon S3 nommé *my_s3_bucket* vers le répertoire d'instance de base de données nommé *WALLET_DIR*.

   ```
   SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
   ```

```
      p_bucket_name    =>  'my_s3_bucket',
      p_s3_prefix      =>  'cwallet.sso',
      p_directory_name =>  'WALLET_DIR')
   AS TASK_ID FROM DUAL;
```

6. (Facultatif) Téléchargez un portefeuille Oracle protégé par un mot de passe.

   Téléchargez ce portefeuille uniquement si vous souhaitez demander un mot de passe pour chaque utilisation du portefeuille. L'exemple suivant télécharge le portefeuille protégé par un mot de passe nommé *ewallet.p12*.

   ```
   SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
         p_bucket_name    =>  'my_s3_bucket',
         p_s3_prefix      =>  'ewallet.p12',
         p_directory_name =>  'WALLET_DIR')
      AS TASK_ID FROM DUAL;
   ```

7. Vérifiez l'état de votre tâche de base de données.

   Remplacez l'ID de tâche renvoyé par les étapes précédentes pour *dbtask-1234567890123-4567.log* dans l'exemple suivant.

   ```
   SELECT TEXT FROM
    TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-1234567890123-4567.log'));
   ```

8. Vérifiez le contenu du répertoire que vous utilisez pour stocker le portefeuille Oracle.

   ```
   SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir(p_directory => 'WALLET_DIR'));
   ```

   Pour plus d'informations, consultez Établissement de la liste des fichiers situés dans un répertoire d'instance de base de données.

Étape 4 : Accorder des autorisations utilisateur pour le portefeuille Oracle

Vous pouvez créer un nouvel utilisateur de base de données ou configurer un utilisateur existant. Dans les deux cas, vous devez configurer l'utilisateur pour qu'il accède au portefeuille Oracle pour des connexions sécurisées et une authentification client à l'aide de certificats.

Pour accorder des autorisations utilisateur pour le portefeuille Oracle

1. Connectez-vous à votre instance de base de données RDS for Oracle en tant qu'utilisateur principal.
2. Si vous ne souhaitez pas configurer un utilisateur de base de données existant, créez un nouvel utilisateur. Sinon, passez à l'étape suivante.

   L'exemple suivant crée un utilisateur de base de données nommé *my-user*.

   ```
   CREATE USER my-user IDENTIFIED BY my-user-pwd;
   GRANT CONNECT TO my-user;
   ```

3. Accordez l'autorisation à l'utilisateur de votre base de données pour le répertoire contenant votre portefeuille Oracle.

   L'exemple suivant accorde l'accès en lecture à l'utilisateur *my-user* au répertoire *WALLET-DIR*.

   ```
   GRANT READ ON DIRECTORY WALLET_DIR TO my-user;
   ```

4. Accordez l'autorisation à votre utilisateur de base de données d'utiliser le package UTL_HTTP.

   Le programme PL/SQL suivant accorde l'accès UTL_HTTP à l'utilisateur*my-user*.

   ```
   BEGIN
     rdsadmin.rdsadmin_util.grant_sys_object('UTL_HTTP', UPPER('my-user'));
     END;
   /
   ```

5. Accordez l'autorisation à votre utilisateur de base de données d'utiliser le package UTL_FILE.

   Le programme PL/SQL suivant accorde l'accès UTL_FILE à l'utilisateur*my-user*.

   ```
   BEGIN
     rdsadmin.rdsadmin_util.grant_sys_object('UTL_FILE', UPPER('my-user'));
     END;
   /
   ```

Étape 5 : Configurer l'accès à un site web à partir de votre instance de base de données

Dans cette étape, vous configurez votre utilisateur de base de données Oracle afin qu'il puisse se connecter à votre site web cible à l'aide de UTL_HTTP, de votre portefeuille Oracle chargé et du certificat client. Pour de plus amples informations, veuillez consulter Configuring Access Control to an Oracle Wallet dans la documentation de la base de données Oracle.

Pour configurer l'accès à un site web depuis votre instance de base de données RDS for Oracle

1. Connectez-vous à votre instance de base de données RDS for Oracle en tant qu'utilisateur principal.
2. Créez une entrée Host Access Control Entry (ACE) pour votre utilisateur et le site web cible sur un port sécurisé.

   L'exemple suivant configure *my-user* pour qu'il accède à *secret.encrypted-website.com* sur le port sécurisé 443.

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host       => 'secret.encrypted-website.com',
    lower_port => 443,
    upper_port => 443,
    ace        => xs$ace_type(privilege_list => xs$name_list('http'),
                              principal_name => 'my-user',
                              principal_type => xs_acl.ptype_db));
                           -- If the program unit results in PLS-00201, set
                           -- the principal_type parameter to 2 as follows:
                           -- principal_type => 2));
END;
/
```

> **Important**
>
> L'unité de programme précédente peut entraîner l'erreur suivante : `PLS-00201: identifier 'XS_ACL' must be declared` Si cette erreur est renvoyée, remplacez la ligne qui affecte une valeur à `principal_type` par la ligne suivante, puis réexécutez l'unité de programme :

```
principal_type => 2));
```

Pour plus d'informations sur les constantes du package PL/SQLXS_ACL, consultez le guide de l'administrateur et du développeur de Real Application Security dans la documentation de la base de données Oracle.

Pour de plus amples informations, veuillez consulter Configuring Access Control for External Network Services dans la documentation de la base de données Oracle.

3. (Facultatif) Créez une ACE pour votre utilisateur et votre site web cible sur le port standard.

   Vous devrez peut-être utiliser le port standard si certaines pages web sont diffusées à partir du port du serveur web standard (80) au lieu du port sécurisé (443).

```
BEGIN
  DBMS_NETWORK_ACL_ADMIN.APPEND_HOST_ACE(
    host       => 'secret.encrypted-website.com',
    lower_port => 80,
    upper_port => 80,
    ace        => xs$ace_type(privilege_list => xs$name_list('http'),
                              principal_name => 'my-user',
                              principal_type => xs_acl.ptype_db));
                           -- If the program unit results in PLS-00201, set
                           -- the principal_type parameter to 2 as follows:
                           -- principal_type => 2));
END;
/
```

4. Vérifiez que les entrées de contrôle d'accès existent.

```
SET LINESIZE 150
COLUMN HOST FORMAT A40
COLUMN ACL FORMAT A50

SELECT HOST, LOWER_PORT, UPPER_PORT, ACL
  FROM DBA_NETWORK_ACLS
ORDER BY HOST;
```

5. Accordez l'autorisation à votre utilisateur de base de données d'utiliser le package UTL_HTTP.

Le programme PL/SQL suivant accorde l'accès UTL_HTTP à l'utilisateur*my-user*.

```
BEGIN
  rdsadmin.rdsadmin_util.grant_sys_object('UTL_HTTP', UPPER('my-user'));
  END;
/
```

6. Vérifiez que les listes de contrôle d'accès associées existent.

```
SET LINESIZE 150
COLUMN ACL FORMAT A50
COLUMN PRINCIPAL FORMAT A20
COLUMN PRIVILEGE FORMAT A10

SELECT ACL, PRINCIPAL, PRIVILEGE, IS_GRANT,
       TO_CHAR(START_DATE, 'DD-MON-YYYY') AS START_DATE,
       TO_CHAR(END_DATE, 'DD-MON-YYYY') AS END_DATE
  FROM DBA_NETWORK_ACL_PRIVILEGES
ORDER BY ACL, PRINCIPAL, PRIVILEGE;
```

7. Accordez l'autorisation à votre utilisateur de base de données d'utiliser des certificats pour l'authentification client et votre portefeuille Oracle pour les connexions.

> **Note**
>
> Si vous n'utilisez pas de certificats client pour l'authentification, vous pouvez ignorer cette étape.

```
DECLARE
  l_wallet_path all_directories.directory_path%type;
BEGIN
  SELECT DIRECTORY_PATH
    INTO l_wallet_path
    FROM ALL_DIRECTORIES
   WHERE UPPER(DIRECTORY_NAME)='WALLET_DIR';
  DBMS_NETWORK_ACL_ADMIN.APPEND_WALLET_ACE(
    wallet_path => 'file:/' || l_wallet_path,
    ace         =>  xs$ace_type(privilege_list => xs
$name_list('use_client_certificates'),
                                principal_name => 'my-user',
```

```
                                    principal_type => xs_acl.ptype_db));
END;
/
```

Étape 6 : Tester les connexions de votre instance de base de données à un site web

Dans cette étape, vous configurez votre utilisateur de base de données afin qu'il puisse se connecter au site web cible à l'aide de UTL_HTTP, de votre portefeuille Oracle chargé et du certificat client.

Pour configurer l'accès à un site web depuis votre instance de base de données RDS for Oracle

1. Connectez-vous à votre instance de base de données RDS for Oracle en tant qu'utilisateur de base de données avec les autorisation UTL_HTTP.
2. Vérifiez qu'une connexion à votre site web cible peut résoudre l'adresse de l'hôte.

   L'exemple suivant montre l'adresse de l'hôte depuis *secret.encrypted-website.com*.

   ```
   SELECT UTL_INADDR.GET_HOST_ADDRESS(host => 'secret.encrypted-website.com')
     FROM DUAL;
   ```
3. Testez une connexion échouée.

   La requête suivante échoue car UTL_HTTP nécessite l'emplacement du portefeuille Oracle avec les certificats.

   ```
   SELECT UTL_HTTP.REQUEST('secret.encrypted-website.com') FROM DUAL;
   ```
4. Testez l'accès au site web en utilisant UTL_HTTP.SET_WALLET et en sélectionnant depuis DUAL.

   ```
   DECLARE
     l_wallet_path all_directories.directory_path%type;
   BEGIN
     SELECT DIRECTORY_PATH
       INTO l_wallet_path
       FROM ALL_DIRECTORIES
      WHERE UPPER(DIRECTORY_NAME)='WALLET_DIR';
     UTL_HTTP.SET_WALLET('file:/' || l_wallet_path);
   END;
   /
   ```

```
SELECT UTL_HTTP.REQUEST('secret.encrypted-website.com') FROM DUAL;
```

5. (Facultatif) Testez l'accès au site web en stockant votre requête dans une variable et en utilisant `EXECUTE IMMEDIATE`.

```
DECLARE
  l_wallet_path all_directories.directory_path%type;
  v_webpage_sql VARCHAR2(1000);
  v_results     VARCHAR2(32767);
BEGIN
  SELECT DIRECTORY_PATH
    INTO l_wallet_path
    FROM ALL_DIRECTORIES
   WHERE UPPER(DIRECTORY_NAME)='WALLET_DIR';
  v_webpage_sql := 'SELECT UTL_HTTP.REQUEST(''secret.encrypted-website.com'', '''',
 ''file:/' ||l_wallet_path||''') FROM DUAL';
  DBMS_OUTPUT.PUT_LINE(v_webpage_sql);
  EXECUTE IMMEDIATE v_webpage_sql INTO v_results;
  DBMS_OUTPUT.PUT_LINE(v_results);
END;
/
```

6. (Facultatif) Recherchez l'emplacement du système de fichiers de votre répertoire de portefeuille Oracle.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir(p_directory => 'WALLET_DIR'));
```

Utilisez la sortie de la commande précédente pour effectuer une requête HTTP. Par exemple, si le répertoire est *rdsdbdata/userdirs/01*, exécutez la requête suivante.

```
SELECT UTL_HTTP.REQUEST('https://secret.encrypted-website.com/', '',
 'file://rdsdbdata/userdirs/01')
FROM   DUAL;
```

Utilisation des CDB dans RDS for Oracle

Dans l'architecture multilocataire d'Oracle, une base de données de conteneurs (CDB) peut inclure des bases de données enfichables (PDB) créées par le client. Pour plus d'informations sur les CDB, consultez Introduction à l'architecture multilocataire (langue française non garantie) dans la documentation Oracle Database.

Rubriques

- Présentation de RDS pour Oracle CDBs
- Configuration d'une CDB RDS for Oracle
- Sauvegarde et restauration d'une CDB
- Conversion d'une base de données non-CDB RDS for Oracle en CDB
- Conversion de la configuration à locataire unique en configuration à locataires multiples
- Ajout d'une base de données locataire RDS for Oracle à votre instance de CDB
- Modification d'une base de données locataire RDS for Oracle
- Suppression d'une base de données locataire RDS for Oracle de votre CDB
- Affichage des détails de la base de données locataire
- Mise à niveau de votre CDB

Présentation de RDS pour Oracle CDBs

Vous pouvez créer une instance de base de données RDS for Oracle en tant que base de données de conteneurs (CDB) lorsque vous exécutez Oracle Database 19c ou version ultérieure. À partir d'Oracle Database 21c, toutes les bases de données le sont CDBs. Une CDB se distingue d'une non-CDB car elle peut contenir des bases de données enfichables (PDBs), appelées bases de données mutualisées dans RDS pour Oracle. Une PDB est une collection portable de schémas et d'objets qui apparaît à une application en tant que base de données distincte.

Vous créez votre base de données de locataires initiale (PDB) lorsque vous créez votre instance CDB. Dans RDS for Oracle, votre application cliente interagit avec un PDB plutôt qu'avec le CDB. Votre expérience avec une PDB est essentiellement identique à votre expérience avec une base de données non-CDB.

Rubriques

- Configuration à locataires multiples de l'architecture CDB

- Configuration à locataire unique de l'architecture CDB
- Options de création et de conversion pour CDBs
- Comptes d'utilisateur et privilèges dans une CDB
- Familles de groupes de paramètres dans une CDB
- Limites de RDS pour Oracle CDBs

Configuration à locataires multiples de l'architecture CDB

RDS for Oracle prend en charge la configuration multilocataire de l'architecture mutualisée Oracle, également appelée architecture CDB. Dans cette configuration, votre instance RDS pour Oracle CDB peut contenir de 1 à 30 bases de données mutualisées, selon l'édition de la base de données et les licences d'option requises. Dans la base de données Oracle, une base de données mutualisée est une PDB. Votre instance de base de données doit utiliser la version 19.0.0.0.ru-2022-01.rur-2022.r1 ou versions ultérieures de la base de données Oracle.

> **Note**
>
> La configuration Amazon RDS est appelée « multi-tenant » plutôt que « multitenant » car il s'agit d'une fonctionnalité d'Amazon RDS, et pas seulement du moteur de base de données Oracle. De même, le terme « locataire » du RDS fait référence à tout locataire d'une configuration RDS, et pas seulement à Oracle. PDBs Dans la documentation RDS, le terme sans trait d'union « Oracle multitenant » fait exclusivement référence à l'architecture CDB de la base de données Oracle, qui est compatible avec les déploiements sur site et RDS.

Vous pouvez configurer les paramètres suivants :

- Nom de la base de données locataire
- Nom d'utilisateur principal de la base de données locataire
- Mot de passe principal de la base de données du locataire (éventuellement intégré à Secrets Manager)
- Jeu de caractères de base de données locataire
- Jeu de caractères national de base de données locataire

Le jeu de caractères de base de données locataire peut être différent du jeu de caractères de CDB. Il en va de même pour le jeu de caractères national. Après avoir créé votre base de données de locataires initiale, vous pouvez créer, modifier ou supprimer des bases de données de locataires à l'aide de RDS. APIs Le nom de la CDB est défini par défaut sur `RDSCDB` et ne peut pas être modifié. Pour plus d'informations, consultez Paramètres des instances de base de données et Modification d'une base de données locataire RDS for Oracle.

Configuration à locataire unique de l'architecture CDB

RDS for Oracle prend en charge la configuration existante de l'architecture multilocataire Oracle appelée configuration à locataire unique. Dans cette configuration, une instance de CDB RDS pour Oracle CDB ne peut contenir qu'un seul locataire (PDB). Vous ne pourrez pas en créer d'autres PDBs ultérieurement.

Options de création et de conversion pour CDBs

Oracle Database 21c prend uniquement en charge CDBs, tandis qu'Oracle Database 19c prend en charge les deux CDBs et non. CDBs Toutes les instances de CDB RDS for Oracle prennent en charge à la fois les configurations à locataires multiples et à locataire unique.

Options de création, de conversion et de mise à niveau pour l'architecture de base de données Oracle

Le tableau suivant présente les différentes options d'architecture pour la création et la mise à niveau des bases de données RDS for Oracle.

Version	Options de création de base de données	Options de conversion d'architecture	Cibles de mise à niveau de version majeure
Oracle Database 21c	Architecture CDB uniquement	N/A	N/A
Oracle Database 19c	Architecture CDB ou non CDB	Architecture non CDB vers CDB (RU d'avril 2021 ou supérieure)	Base de données Oracle 21c CDB

Comme indiqué dans le tableau précédent, vous ne pouvez pas directement effectuer la mise à niveau d'une base de données non CDB vers une base de données CDB dans une nouvelle version majeure de base de données. En revanche, vous pouvez convertir une base de données non-CDB Oracle Database 19c en CDB Oracle Database 19c, puis mettre à niveau la CDB Oracle Database 19c en une CDB Oracle Database 21c. Pour de plus amples informations, veuillez consulter Conversion d'une base de données non-CDB RDS for Oracle en CDB.

Options de conversion pour les configurations d'architecture CDB

Le tableau suivant présente les différentes options de conversion de la configuration d'architecture d'une instance de base de données RDS for Oracle.

Architecture et configuration actuelles	Conversion vers la configuration à locataire unique de l'architecture CDB	Conversion vers la configuration à locataires multiples de l'architecture CDB	Conversion vers l'architecture non CDB
Non CDB	Pris en charge	Pris en charge*	N/A
CDB utilisant la configuration à locataire unique	N/A	Pris en charge	Non pris en charge
CDB utilisant la configuration à locataires multiples	Non pris en charge	N/A	Non pris en charge

* Vous ne pouvez pas convertir une base de données non CDB en configuration à locataires multiples au cours d'une seule opération. Lorsque vous convertissez une base de données non CDB en CDB, la CDB est dans la configuration à locataire unique. Vous pouvez ensuite convertir la configuration à locataire unique en configuration à locataires multiples dans le cadre d'une opération distincte.

Comptes d'utilisateur et privilèges dans une CDB

Dans l'architecture multilocataire d'Oracle, tous les comptes d'utilisateur sont des utilisateurs courants ou des utilisateurs locaux. Un utilisateur courant de base de données de conteneur (CDB)

est un utilisateur de base de données dont l'identité unique et le mot de passe sont connus à la racine de la CDB et dans chaque base de données enfichable (PDB) existante et future. En revanche, un utilisateur local n'existe que dans une seule PDB.

L'utilisateur principal RDS est un compte d'utilisateur local dans la PDB, que vous nommez lorsque vous créez votre instance de base de données. Si vous créez de nouveaux comptes d'utilisateur, ces utilisateurs seront également des utilisateurs locaux résidant dans la PDB. Vous ne pouvez utiliser aucun compte utilisateur pour créer un nouveau PDB PDBs ou modifier l'état de celui-ci.

L'utilisateur `rdsadmin` est un compte d'utilisateur courant. Vous pouvez exécuter des packages RDS for Oracle qui existent dans ce compte, mais vous ne pouvez pas vous connecter en tant que `rdsadmin`. Pour plus d'informations, reportez-vous à la section À propos des utilisateurs courants et des utilisateurs locaux dans la documentation Oracle.

Pour les utilisateurs principaux dans les configurations à locataires multiples et à locataire unique, vous pouvez utiliser des informations d'identification autogérées ou gérées par. AWS Secrets Manager Dans la configuration à locataire unique, vous utilisez des commandes CLI au niveau de l'instance, par exemple pour les mots `create-db-instance` de passe principaux gérés. Dans la configuration multi-tenant, vous utilisez des commandes de base de données mutualisées, par exemple `create-tenant-database` pour les mots de passe principaux gérés. Pour plus d'informations sur l'intégration de Secrets Manager, consultezGestion du mot de passe utilisateur principal pour une base de données cliente RDS pour Oracle avec Secrets Manager.

Familles de groupes de paramètres dans une CDB

CDBs ont leurs propres familles de groupes de paramètres et leurs propres valeurs de paramètres par défaut. Les familles de groupes de paramètres de CDB sont les suivantes :

- oracle-ee-cdb-21
- oracle-se2-cdb-21
- oracle-ee-cdb-19
- oracle-se2-cdb-19

Limites de RDS pour Oracle CDBs

RDS for Oracle prend en charge un sous-ensemble de fonctionnalités disponibles dans une CDB sur site.

Limitations de CDB

Les limites suivantes s'appliquent à RDS pour Oracle au niveau de la CDB :

- Vous ne pouvez pas vous connecter à une CDB. Vous vous connectez toujours à la base de données locataire (PDB) plutôt qu'à la CDB. Spécifiez le point de terminaison pour la base de données enfichable (PDB), tout comme pour une base de données non-CDB. La seule différence est que vous spécifiez pdb_name pour le nom de base de données, où pdb_name est le nom que vous avez choisi pour votre base de données enfichable (PDB).
- Vous ne pouvez pas convertir une CDB dans la configuration à locataires multiples en une CDB dans la configuration à locataire unique. La conversion vers la configuration à locataires multiples est unidirectionnelle et irréversible.
- Vous ne pouvez pas activer la configuration à locataires multiples ni effectuer une conversion vers ce type de configuration si votre instance de base de données utilise une version de la base de données Oracle antérieure à 19.0.0.0.ru-2022-01.rur-2022.r1.
- Vous ne pouvez pas utiliser un RDS pour Oracle CDB avec ORDS 22 ou version ultérieure. Pour contourner le problème, vous pouvez utiliser une version inférieure d'ORDS ou une base de données non-CDB Oracle Database 19c.
- Vous ne pouvez pas utiliser Oracle Data Guard dans une configuration à locataires multiples, mais vous pouvez l'utiliser dans une configuration à locataire unique.
- Vous ne pouvez pas utiliser les flux d'activité de base de données dans une base de données CDB.
- Vous ne pouvez pas activer l'audit depuis l'intérieur`CDB$ROOT`. Vous devez activer l'audit au sein de chaque PDB individuellement.

Limitations de la base de données locataire (PDB)

Les limitations suivantes s'appliquent aux bases de données locataire dans la configuration à locataires multiples RDS for Oracle :

- Vous ne pouvez pas reporter les opérations de la base de données locataire vers la fenêtre de maintenance. Toutes les modifications sont effectuées immédiatement.
- Vous ne pouvez pas ajouter une base de données locataire à une CDB qui utilise la configuration à locataire unique.
- Vous ne pouvez pas ajouter ou modifier plusieurs bases de données locataire au cours d'une seule opération. Vous ne pouvez les ajouter ou les modifier qu'une par une.

- Vous ne pouvez pas modifier une base de données locataire pour qu'elle soit nommée `CDB$ROOT` ou `PDB$SEED`.
- Vous ne pouvez pas supprimer une base de données locataire si elle est le seul locataire de la CDB.
- Tous les types de classes d'instance de base de données ne disposent pas de ressources suffisantes pour prendre PDBs en charge plusieurs dans une instance RDS pour Oracle CDB. L'augmentation du nombre de PDB affecte les performances et la stabilité des classes d'instances plus petites et augmente la durée de la plupart des opérations au niveau de l'instance, par exemple les mises à niveau de base de données.
- Vous ne pouvez pas utiliser plusieurs Comptes AWS pour créer PDBs dans le même CDB. PDBsdoivent appartenir au même compte que l'instance de base de données sur laquelle PDBs ils sont hébergés.
- Tous les PDBs composants d'un CDB utilisent le même point de terminaison et le même écouteur de base de données.
- Les opérations suivantes ne sont pas prises en charge au niveau de la PDB mais le sont au niveau de la CDB :
  - Sauvegarde et restauration
  - Mises à niveau de base de données
  - Opérations de maintenance
- Les fonctionnalités suivantes ne sont pas prises en charge au niveau de la PDB mais le sont au niveau de la CDB :
  - Groupes d'options (les options sont installées PDBs sur toutes les instances de votre instance CDB)
  - Groupes de paramètres (tous les paramètres sont dérivés du groupe de paramètres associé à votre instance de CDB)
- Les opérations au niveau de la PDB qui sont prises en charge dans l'architecture CDB sur site mais qui ne le sont pas dans une CDB RDS for Oracle sont les suivantes :

> **Note**
>
> La liste suivante n'est pas exhaustive.

  - Demande PDBs

- Proxy PDBs
- Démarrage et arrêt d'une PDB
- Débranchement et branchement PDBs

  Pour faire entrer des données dans votre CDB ou les en faire sortir, utilisez les mêmes techniques que pour une base de données non-CDB. Pour en savoir plus sur la migration de données, consultez Importation de données dans Oracle sur Amazon RDS.
- Configuration des options au niveau de la PDB

  La PDB hérite des paramètres d'options du groupe d'options de la CDB. Pour plus d'informations sur la configuration des options, consultez Groupes de paramètres pour (Amazon RDS). Pour connaître les bonnes pratiques, consultez Utilisation des groupes de paramètres DB.
- Configuration des paramètres dans une PDB

  La PDB hérite des paramètres de la CDB. Pour plus d'informations sur la configuration des options, consultez Ajout d'options aux instances de base de données Oracle.
- Configuration de différents auditeurs PDBs dans le même CDB
- Fonctionnalités Oracle Flashback

Configuration d'une CDB RDS for Oracle

La configuration d'une CDB est similaire à la configuration d'une base de données non-CDB.

Rubriques

- Création d'une instance de CDB RDS for Oracle
- Connexion à une PDB dans votre CDB RDS for Oracle

Création d'une instance de CDB RDS for Oracle

Dans RDS pour Oracle, la création d'une instance CDB est presque identique à la création d'une instance non CDB. La différence réside dans le fait que vous devez choisir l'architecture multilocataire Oracle lors de la création de votre instance de base de données et devez également choisir une configuration d'architecture : multilocataire ou à locataire unique. Si vous créez des balises lorsque vous créez une CDB dans la configuration multilocataire, RDS les propage vers la base de données locataire initiale. Pour créer un CDB, utilisez l'API AWS Management Console AWS CLI, la ou l'API RDS.

console

Pour créer une instance de CDB

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez l'instance Région AWS dans laquelle vous souhaitez créer l'instance CDB.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).
5. Dans Choose a database creation method (Choisir une méthode de création de base de données), sélectionnez Standard Create (Création standard).
6. Sous Engine options (Options de moteur), choisissez Oracle.
7. Pour Type de gestion de base de données, choisissez Amazon RDS.
8. Pour Paramètres d'architecture, choisissez Architecture à locataires multiples Oracle.
9. Pour Configuration de l'architecture, effectuez l'une des opérations suivantes :
   - Choisissez Configuration à locataires multiples et passez à l'étape suivante.
   - Choisissez Configuration à locataire unique et passez à l'étape 11.
10. (Configuration à locataires multiples) Pour Paramètres de base de données locataire, apportez les modifications suivantes :
    - Pour Nom de la base de données locataire, entrez le nom de votre PDB initiale. Le nom de la PDB doit être différent du nom de CDB, qui est par défaut `RDSCDB`.
    - Pour Nom d'utilisateur principal de la base de données locataire, entrez le nom d'utilisateur principal de votre PDB. Vous ne pouvez pas utiliser le nom d'utilisateur principal de la base de données locataire pour vous connecter à la CDB elle-même.
    - Pour la gestion des informations d'identification, choisissez l'une des options de gestion des informations d'identification suivantes :
      - Géré dans AWS Secrets Manager

        Le mot de passe géré est destiné à la base de données locataire initiale plutôt qu'à l'instance. Dans Sélectionnez la clé de chiffrement, choisissez soit une clé KMS créée par Secrets Manager, soit une clé que vous avez créée.

> **Note**
>
> C'est la technique la plus sûre AWS Secrets Manager que nous recommandons pour gérer les informations d'identification. Des frais supplémentaires seront facturés. Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.

- Autogéré

  Pour spécifier un mot de passe, désactivez la case à cocher Générer automatiquement un mot de passe si elle est sélectionnée. Entrez le même mot de passe dans Mot de passe principal et Confirmez le mot de passe principal.

- Pour Jeu de caractères de base de données locataire, choisissez un jeu de caractères pour la PDB. Vous pouvez choisir un jeu de caractères de base de données locataire différent du jeu de caractères de CDB.

  Le jeu de caractères PDB par défaut est AL32UTF8. Si vous choisissez un jeu de caractères PDB autre que celui par défaut, la création de CDB peut être plus lente.

> **Note**
>
> Vous ne pouvez pas spécifier plusieurs bases de données mutualisées lors de l'opération de création. Le CDB possède un PDB lorsqu'il est créé. Vous pouvez l'ajouter PDBs à un CDB existant dans le cadre d'une opération séparée.

11. (Configuration à locataire unique) Choisissez les paramètres souhaités en fonction des options répertoriées dans Paramètres des instances de base de données :

    - Dans la section Settings (Paramètres), ouvrez Credential Settings (Paramètres des informations d'identification). Ensuite, procédez comme suit :

      i. Pour Identifiant principal, entrez le nom d'un utilisateur local dans votre PDB. Vous ne pouvez pas utiliser l'identifiant principal pour vous connecter à la racine de la CDB.

      ii. Pour la gestion des informations d'identification, choisissez l'une des options de gestion des informations d'identification suivantes :

        - Géré dans AWS Secrets Manager

Dans Sélectionnez la clé de chiffrement, choisissez soit une clé KMS créée par Secrets Manager, soit une clé que vous avez créée.

> **Note**
>
> C'est la technique la plus sûre AWS Secrets Manager que nous recommandons pour gérer les informations d'identification. Des frais supplémentaires seront facturés. Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.

- Autogéré

  Pour spécifier un mot de passe, désactivez la case à cocher Générer automatiquement un mot de passe si elle est sélectionnée. Entrez le même mot de passe dans Mot de passe principal et Confirmez le mot de passe principal.

12. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour plus d'informations sur chaque paramètre, consultez la section Paramètres pour les instances de base de données.
13. Choisissez Créer une base de données.

AWS CLI

Pour créer un CDB dans la configuration multi-locataires, utilisez la create-db-instancecommande avec les paramètres suivants :

- `--db-instance-identifier`
- `--db-instance-class`
- `--engine { oracle-ee-cdb | oracle-se2-cdb }`
- `--master-username`
- `--master-user-password` ou `--manage-master-user-password`
- `--multi-tenant` (pour la configuration à locataire unique, ne spécifiez pas `multi-tenant` ou spécifiez `--no-multi-tenant`)
- `--allocated-storage`
- `--backup-retention-period`

Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

L'exemple suivant crée une instance de base de données RDS pour Oracle nommée *my-cdb-inst* dans la configuration multi-locataires. Si vous spécifiez `--no-multi-tenant` ou que vous ne spécifiez pas `--multi-tenant`, la configuration de CDB par défaut est à locataire unique. Le moteur est `oracle-ee-cdb` : une commande qui spécifie `oracle-ee` et `--multi-tenant` échoue avec une erreur. La base de données des locataires initiale est nommée*mypdb*.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-db-instance \
    --engine oracle-ee-cdb \
    --db-instance-identifier my-cdb-inst \
    --multi-tenant \
    --db-name mypdb \
    --allocated-storage 250 \
    --db-instance-class db.t3.large \
    --master-username pdb_admin \
    --manage-master-user-password \
    --backup-retention-period 3
```

Dans Windows:

```
aws rds create-db-instance ^
    --engine oracle-ee-cdb ^
    --db-instance-identifier my-cdb-inst ^
    --multi-tenant ^
    --db-name mypdb ^
    --allocated-storage 250 ^
    --db-instance-class db.t3.large ^
    --master-username pdb_admin ^
    --manage-master-user-password \ ^
    --backup-retention-period 3
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit : Le nom de la base de données, le jeu de caractères, le jeu de caractères national, l'utilisateur principal et le secret de l'utilisateur principal ne sont pas inclus dans la sortie. Vous pouvez afficher ces informations à l'aide de la commande CLI `describe-tenant-databases`.


```
{
    "DBInstance": {
        "DBInstanceIdentifier": "my-cdb-inst",
        "DBInstanceClass": "db.t3.large",
        "MultiTenant": true,
        "Engine": "oracle-ee-cdb",
        "DBResourceId": "db-ABCDEFGJIJKLMNOPQRSTUVWXYZ",
        "DBInstanceStatus": "creating",
        "AllocatedStorage": 250,
        "PreferredBackupWindow": "04:59-05:29",
        "BackupRetentionPeriod": 3,
        "DBSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-0a1bcd2e",
                "Status": "active"
            }
        ],
        "DBParameterGroups": [
            {
                "DBParameterGroupName": "default.oracle-ee-cdb-19",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "DBSubnetGroup": {
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "VpcId": "vpc-1234567a",
            "SubnetGroupStatus": "Complete",
            ...
```


API RDS

Pour créer une instance de base de données à l'aide de l'API Amazon RDS, appelez l'DBInstanceopération Create.

Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Connexion à une PDB dans votre CDB RDS for Oracle

Vous pouvez utiliser un utilitaire tel que SQL*Plus pour vous connecter à une PDB. Pour télécharger Oracle Instant Client, qui inclut une version autonome de SQL*Plus, consultez la page des téléchargements d'Oracle Instant Client.

Pour connecter SQL*Plus à votre PDB, vous avez besoin des informations suivantes :

- Nom de la PDB
- Nom d'utilisateur et mot de passe de la base de données
- Point de terminaison pour votre instance de base de données
- Numéro de port

Pour obtenir des informations sur la recherche des informations précédentes, consultez Recherche du point de terminaison de votre instance de base de données RDS for Oracle.

Example Pour vous connecter à votre PDB à l'aide de SQL*Plus

Dans les exemples suivants, remplacez votre utilisateur principal par*master_user_name*. Remplacez également le point de terminaison par votre instance de base de données, puis incluez le numéro de port et le SID Oracle. La valeur SID est le nom de la PDB que vous avez spécifié lors de la création de votre instance de base de données, et non pas l'identifiant de l'instance de base de données.

Dans Linux, macOS, ou Unix:

```
sqlplus 'master_user_name@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=endpoint)
(PORT=port))(CONNECT_DATA=(SID=pdb_name)))'
```

Dans Windows:

```
sqlplus master_user_name@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=endpoint)
(PORT=port))(CONNECT_DATA=(SID=pdb_name)))
```

Vous devez visualiser des résultats similaires à ce qui suit.

```
SQL*Plus: Release 19.0.0.0.0 Production on Mon Aug 21 09:42:20 2021
```

Une fois que vous avez saisi le mot de passe de l'utilisateur, l'invite SQL apparaît.

```
SQL>
```

> **Note**
>
> La chaîne de connexion de format court (Easy connect ou EZCONNECT), comme `sqlplus username/password@LONGER-THAN-63-CHARS-RDS-ENDPOINT-HERE:1521/database-identifier`, peut comporter une limite de caractères maximale et ne doit pas être utilisée pour se connecter.

Sauvegarde et restauration d'une CDB

Vous pouvez sauvegarder et restaurer votre CDB à l'aide d'instantanés de base de données RDS ou de Recovery Manager (RMAN).

Sauvegarde et restauration d'une CDB à l'aide d'instantanés de base de données

Les instantanés de base de données fonctionnent de la même manière dans les architectures CDB et non CDB. Les principales différences sont les suivantes :

- Lorsque vous restaurez un instantané de base de données d'une CDB, vous ne pouvez pas renommer la CDB. Le nom de la CDB est `RDSCDB` et ne peut pas être modifié.
- Lorsque vous restaurez un instantané de base de données d'une CDB, vous ne pouvez pas renommer les CDB. Vous pouvez modifier le nom de la PDB à l'aide de la commande modify-tenant-database.
- Pour rechercher des bases de données locataire dans un instantané, utilisez la commande de l'interface de ligne de commande describe-db-snapshot-tenant-databases.
- Vous ne pouvez pas interagir directement avec les bases de données locataire dans un instantané de CDB qui utilise la configuration de l'architecture à locataires multiples. Si vous restaurez l'instantané de base de données, vous restaurez toutes ses bases de données locataire.
- RDS for Oracle copie implicitement les balises d'une base de données locataire vers la base de données locataire située dans un instantané de base de données. Lorsque vous restaurez une base de données locataire, les balises apparaissent dans la base de données restaurée.

- Si vous restaurez un instantané de base de données et que vous spécifiez de nouvelles balises à l'aide du paramètre `--tags`, les nouvelles balises remplacent toutes les balises existantes.
- Si vous prenez un instantané de base de données d'une instance de CDB comportant des balises et que vous spécifiez `--copy-tags-to-snapshot`, RDS for Oracle copie les balises des bases de données locataire vers les bases de données locataire situées dans l'instantané.

Pour plus d'informations, consultez Considérations relatives à Oracle Database.

Sauvegarde et restauration d'une CDB à l'aide de RMAN

Pour découvrir comment sauvegarder et restaurer une CDB ou une base de données locataire individuelle à l'aide de RMAN, consultez Exécution de RMAN tâches courantes pour les instances de base de données Oracle.

Conversion d'une base de données non-CDB RDS for Oracle en CDB

Vous pouvez modifier l'architecture d'une base de données Oracle de l'architecture non-CDB à l'architecture Oracle multitenant, également appelée architecture CDB, à l'aide de la commande. `modify-db-instance` Dans la plupart des cas, cette technique est préférable à la création d'un nouveau CDB et à l'importation de données. L'opération de conversion entraîne des temps d'arrêt.

Lorsque vous mettez à niveau la version de votre moteur de base de données, vous ne pouvez pas modifier l'architecture de base de données lors de la même opération. Par conséquent, pour mettre à niveau une base de données non CDB Oracle Database 19c vers une CDB Oracle Database 21c, vous devez d'abord convertir la base de données non CDB en CDB au cours d'une étape, puis mettre à niveau la CDB 19c vers une CDB 21c au cours d'une étape distincte.

L'opération de conversion de la base de données non-CDB présente les exigences suivantes :

- Vous devez spécifier `oracle-ee-cdb` ou `oracle-se2-cdb` pour le type de moteur de base de données. Ce sont les seules valeurs prises en charge.
- Votre moteur de base de données doit utiliser Oracle Database 19c avec une mise à jour de version (RU) d'avril 2021 ou ultérieure.

Les restrictions suivantes s'appliquent à l'opération :

- Vous ne pouvez pas convertir une CDB en base de données non-CDB. Vous pouvez uniquement convertir une base de données non-CDB en CDB.

- Vous ne pouvez pas convertir une base de données non CDB en configuration multilocataire au cours d'un seul appel `modify-db-instance`. Une fois que vous avez converti une base de données non CDB en CDB, votre CDB est dans la configuration à locataire unique. Pour convertir la configuration à locataire unique en configuration multilocataire, réexécutez `modify-db-instance`. Pour de plus amples informations, veuillez consulter Conversion de la configuration à locataire unique en configuration à locataires multiples.
- Vous ne pouvez pas convertir une base de données principale ou de réplica sur laquelle Oracle Data Guard est activé. Pour convertir une base de données non CDB contenant des réplicas en lecture, supprimez d'abord toutes les réplicas en lecture.
- Vous ne pouvez pas mettre à niveau la version du moteur de base de données et convertir une base de données non-CDB en CDB dans la même opération.

Avant de convertir votre fichier non-CDB, tenez compte des points suivants :

- Les considérations relatives aux groupes d'options et de paramètres sont les mêmes que pour la mise à niveau du moteur de base de données. Pour de plus amples informations, veuillez consulter Considérations relatives aux mises à niveau des bases de données.
- Vous pouvez convertir des instances non CDB existantes qui utilisent des mots de passe principaux gérés en instances à locataire unique en une seule opération. Les instances à locataire unique héritent des mots de passe gérés.
- Si l'`OEMAGENT`option est installée sur votre instance de base de données, il est recommandé de supprimer cette option avant de convertir votre instance non CDB. Une fois que votre fichier non-CDB est converti en CDB, réinstallez l'option. Pour de plus amples informations, veuillez consulter Oracle Management Agent pour Enterprise Manager Cloud Control.
- Au cours du processus de conversion, RDS rétablit la taille du journal de restauration en ligne à 128 Mo par défaut.

console

Pour convertir une base de données non-CDB en CDB

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS où réside votre instance de base de données.

3. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données non-CDB que vous voulez convertir en instance de CDB.
4. Sélectionnez Modifier.
5. Pour Paramètres d'architecture, sélectionnez Architecture à locataires multiples Oracle. Après la conversion, votre CDB sera dans la configuration à locataire unique.
6. (Facultatif) Pour Groupe de paramètres de base de données, choisissez un nouveau groupe de paramètres pour votre instance de CDB. Les mêmes considérations relatives aux groupes de paramètres s'appliquent lors de la conversion d'une instance de base de données que lors de la mise à niveau d'une instance de base de données. Pour de plus amples informations, veuillez consulter Considérations relatives au groupe de paramètres.
7. (Facultatif) Pour Groupe d'options, choisissez un nouveau groupe d'options pour votre instance de CDB. Les mêmes considérations relatives aux groupes d'options s'appliquent lors de la conversion d'une instance de base de données que lors de la mise à niveau d'une instance de base de données. Pour de plus amples informations, veuillez consulter Considérations relatives au groupe d'options.
8. (Facultatif) Pour la gestion des informations d'identification, choisissez Géré dans AWS Secrets Manager ou Autogéré. Pour de plus amples informations, veuillez consulter Gestion du mot de passe d'utilisateur principal pour une instance de base de données avec Secrets Manager.
9. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez le résumé des modifications.
10. (Facultatif) Choisissez Appliquer immédiatement pour appliquer les modifications immédiatement. La sélection de cette option peut entraîner des temps d'arrêt dans certains cas. Pour plus d'informations, consultez Utilisation du paramètre de modification du calendrier.
11. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modifier l'instance de base de données.

    Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

AWS CLI

Pour convertir la base de données non CDB de votre instance de base de données en CDB dans la configuration à locataire unique, définissez `oracle-ee-cdb` ou `oracle-se2-cdb` dans `--engine` la commande. AWS CLI modify-db-instance Pour de plus amples informations, veuillez consulter Paramètres des instances de base de données.

L'exemple suivant convertit l'instance de base de données nommée *my-non-cdb* et spécifie un groupe d'options et un groupe de paramètres personnalisés. La commande permet également de gérer les mots de passe avec Secrets Manager.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier my-non-cdb \
    --engine oracle-ee-cdb \
    --option-group-name custom-option-group \
    --db-parameter-group-name custom-parameter-group \
    --manage-master-user-password
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier my-non-cdb ^
    --engine oracle-ee-cdb ^
    --option-group-name custom-option-group ^
    --db-parameter-group-name custom-parameter-group ^
    --manage-master-user-password
```

API RDS

Pour convertir un objet non CDB en CDB, spécifiez `Engine` dans l'API RDS l'opération Modifier. DBInstance

Conversion de la configuration à locataire unique en configuration à locataires multiples

Vous pouvez remplacer la configuration à locataire unique par la configuration à locataires multiples pour l'architecture d'une CDB RDS for Oracle. Avant et après la conversion, votre CDB contient une base de données à locataire unique (PDB). Les balises de l'instance de base de données se propagent vers la base de données locataire initiale créée lors de la conversion.

Avant de commencer, assurez-vous que votre politique IAM est autorisée à créer une base de données mutualisée. Au cours de la conversion, RDS for Oracle migre les métadonnées suivantes vers la nouvelle base de données locataire :

- L'identifiant principal
- Le mot de passe principal géré (si le CDB source s'intègre à Secrets Manager)
- Le nom de la base de données
- Le jeu de caractères
- Le jeu de caractères national

Avant la conversion, vous pouvez consulter les informations précédentes à l'aide de la `describe-db-instances` commande. Après la conversion, vous les affichez à l'aide de la commande `describe-tenant-database`.

La conversion du statut de locataire unique en locataire multiple présente les limites suivantes :

- Vous ne pouvez pas revenir ultérieurement à la configuration à locataire unique après avoir effectué la conversion à la configuration à locataires multiples. La conversion est irréversible.
- Vous ne pouvez pas convertir une base de données principale ou de réplica sur laquelle Oracle Data Guard est activé.
- Vous ne pouvez pas mettre à niveau la version du moteur de base de données et effectuer une conversion vers une configuration à locataires multiples au cours de la même opération.
- Vous ne pouvez pas activer ou désactiver les mots de passe d'utilisateur principal gérés pendant la conversion.

console

Pour convertir une CDB utilisant la configuration à locataire unique en configuration à locataires multiples

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS où réside votre instance de base de données.
3. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données non-CDB que vous voulez convertir en instance de CDB.
4. Sélectionnez Modifier.
5. Pour Paramètres d'architecture, sélectionnez Architecture à locataires multiples Oracle.
6. Pour Configuration de l'architecture, sélectionnez Configuration à locataires multiples.

7. (Facultatif) Pour Groupe de paramètres de base de données, choisissez un nouveau groupe de paramètres pour votre instance de CDB. Les mêmes considérations relatives aux groupes de paramètres s'appliquent lors de la conversion d'une instance de base de données que lors de la mise à niveau d'une instance de base de données.
8. (Facultatif) Pour Groupe d'options, choisissez un nouveau groupe d'options pour votre instance de CDB. Les mêmes considérations relatives aux groupes d'options s'appliquent lors de la conversion d'une instance de base de données que lors de la mise à niveau d'une instance de base de données.
9. Lorsque tous les changements vous conviennent, choisissez Continuer et vérifiez le résumé des modifications.
10. Choisissez Apply immediately (Appliquer immédiatement). Cette option est requise lorsque vous basculez vers une configuration à locataires multiples. Notes que cette option peut entraîner des temps d'arrêt dans certains cas.
11. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modifier l'instance de base de données.

    Vous pouvez également sélectionner Retour pour revoir vos modifications ou Annuler pour les annuler.

AWS CLI

Pour convertir un CDB utilisant la configuration mono-locataire en configuration multi-tenant, spécifiez-le `--multi-tenant` dans la commande. AWS CLI modify-db-instance

L'exemple suivant convertit l'instance de base de données nommée `my-st-cdb` de la configuration à locataire unique en configuration à locataires multiples. L'option `--apply-immediately` est obligatoire.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance --region us-east-1\
    --db-instance-identifier my-st-cdb \
    --multi-tenant \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance --region us-east-1 ^
    --db-instance-identifier my-st-cdb ^
    --multi-tenant ^
    --apply-immediately
```

Le résultat se présente comme suit.

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "my-st-cdb",
        "DBInstanceClass": "db.r5.large",
        "MultiTenant": false,
        "Engine": "oracle-ee-cdb",
        "DBResourceId": "db-AB1CDE2FGHIJK34LMNOPRLXTXU",
        "DBInstanceStatus": "modifying",
        "MasterUsername": "admin",
        "DBName": "ORCL",
        ...
        "EngineVersion": "19.0.0.0.ru-2022-01.rur-2022-01.r1",
        "AutoMinorVersionUpgrade": true,
        "ReadReplicaDBInstanceIdentifiers": [],
        "LicenseModel": "bring-your-own-license",
        "OptionGroupMemberships": [
            {
                "OptionGroupName": "default:oracle-ee-cdb-19",
                "Status": "in-sync"
            }
        ],
        ...
        "PendingModifiedValues": {
            "MultiTenant": "true"
        }
    }
}
```

Ajout d'une base de données locataire RDS for Oracle à votre instance de CDB

Dans la configuration multi-locataires RDS for Oracle, une base de données locataire est une PDB. Pour ajouter une base de données locataire, vérifiez que vous respectez les conditions prérequises suivantes :

- La configuration multi-locataires est activée sur votre CDB. Pour de plus amples informations, veuillez consulter Configuration à locataires multiples de l'architecture CDB.
- Vous disposez des autorisations IAM nécessaires pour créer la base de données locataire.

Vous pouvez ajouter une base de données mutualisée à l'aide de l'API AWS Management Console, de AWS CLI, ou de l'API RDS. Vous ne pouvez pas ajouter plusieurs bases de données locataire au cours d'une seule opération : vous devez les ajouter une par une. Si la conservation des sauvegardes est activée sur la CDB, Amazon RDS sauvegarde l'instance de base de données avant et après l'ajout d'une nouvelle base de données locataire.

console

Pour ajouter une base de données locataire à votre instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS dans laquelle vous voulez créer la base de données locataire.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez l'instance de CDB à laquelle vous souhaitez ajouter une base de données locataire. Votre instance de base de données doit utiliser la configuration multilocataire de l'architecture CDB.
5. Choisissez Actions, puis Ajouter une base de données locataire.
6. Pour Paramètres de base de données locataire, procédez comme suit :
   - Pour Nom de la base de données locataire, entrez le nom de votre nouvelle PDB.
   - Pour Nom d'utilisateur principal de la base de données locataire, entrez le nom de l'utilisateur principal de votre PDB.
   - Choisissez l'une des options de gestion des informations d'identification suivantes :
     - Géré dans AWS Secrets Manager

       Dans Sélectionnez la clé de chiffrement, choisissez soit une clé KMS créée par Secrets Manager, soit une clé que vous avez créée.

> **Note**
>
> C'est la technique la plus sûre AWS Secrets Manager que nous recommandons pour gérer les informations d'identification. Des frais supplémentaires seront facturés. Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.

- Autogéré

  Pour spécifier un mot de passe, désactivez la case à cocher Générer automatiquement un mot de passe si elle est sélectionnée. Entrez le même mot de passe dans Mot de passe principal et Confirmez le mot de passe principal.

- Sous Configuration supplémentaire, entrez le nom de votre PDB pour Nom de base de données initial. Vous ne pouvez pas nommer la CDB, qui porte le nom par défaut `RDSCDB`.

- Pour Jeu de caractères de base de données locataire, choisissez un jeu de caractères pour la PDB. L'argument par défaut est AL32UTF8. Vous pouvez choisir un jeu de caractères de PDB différent du jeu de caractères de CDB.
- Pour Jeu de caractères national de base de données locataire, choisissez un jeu de caractères national pour la PDB. L'argument par défaut est AL32UTF8. Le jeu de caractères national spécifie l'encodage uniquement pour les colonnes qui utilisent le type de données `NCHAR` (`NCHAR`, `NVARCHAR2` et `NCLOB`) et n'affecte pas les métadonnées de la base de données.

Pour plus d'informations sur les paramètres précédents, consultez Paramètres des instances de base de données.

7. Choisissez Ajouter un locataire.

AWS CLI

Pour ajouter une base de données mutualisée à votre CDB à l'aide du AWS CLI, utilisez la commande create-tenant-databaseavec les paramètres obligatoires suivants :

- `--db-instance-identifier`
- `--tenant-db-name`
- `--master-username`

- `--master-user-password`

L'exemple suivant crée une base de données de locataires nommée *mypdb2* dans l'instance RDS pour Oracle CDB nommée. *my-cdb-inst* Le jeu de caractères de PDB est `UTF-16`.

Example

Dans Linux, macOS, ou Unix:

```
aws rds create-tenant-database --region us-east-1 \
    --db-instance-identifier my-cdb-inst \
    --tenant-db-name mypdb2 \
    --master-username mypdb2-admin \
    --master-user-password mypdb2-pwd \
    --character-set-name UTF-16
```

Dans Windows:

```
aws rds create-tenant-database --region us-east-1 ^
    --db-instance-identifier my-cdb-inst ^
    --tenant-db-name mypdb2 ^
    --master-username mypdb2-admin ^
    --master-user-password mypdb2-pwd ^
    --character-set-name UTF-16
```

La sortie ressemble à ce qui suit.

```
...}
    "TenantDatabase" :
         {
            "DbiResourceId" : "db-abc123",
            "TenantDatabaseResourceId" : "tdb-bac567",
            "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-
inst:mypdb2",
            "DBInstanceIdentifier" : "my-cdb-inst",
            "TenantDBName" : "mypdb2",
            "Status" : "creating",
            "MasterUsername" : "mypdb2",
            "CharacterSetName" : "UTF-16",
            ...
        }
```

```
}...
```

Modification d'une base de données locataire RDS for Oracle

Vous pouvez uniquement modifier le nom de la PDB et le mot de passe utilisateur principal d'une base de données locataire dans votre CDB. Prenez note des exigences et limitations suivantes :

- Pour modifier les paramètres d'une base de données locataire dans votre instance de base de données, la base de données locataire doit exister.
- Vous ne pouvez pas modifier plusieurs bases de données locataire au cours d'une seule opération. Vous ne pouvez modifier qu'une seule base de données locataire à la fois.
- Vous ne pouvez pas remplacer le nom d'une base de données locataire par `CDB$ROOT` ou `PDB$SEED`.

Vous pouvez modifier des PDB à l'aide de la AWS Management Console, de l'AWS CLI ou de l'API RDS.

Console

Pour modifier le nom de PDB ou le mot de passe principal d'une base de données locataire

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS dans laquelle vous voulez créer la base de données locataire.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez la base de données locataire dont vous souhaitez modifier le nom de base de données ou le mot de passe utilisateur principal.
5. Sélectionnez Modifier.
6. Pour Paramètres de base de données locataire, effectuez l'une des opérations suivantes :
   - Pour Nom de la base de données locataire, entrez le nouveau nom de votre nouvelle PDB.
   - Pour Mot de passe principal de base de données locataire, entrez un nouveau mot de passe.
7. Choisissez Modifier le locataire.

AWS CLI

Pour modifier une base de données mutualisée à l'aide deAWS CLI, appelez la modify-tenant-databasecommande avec les paramètres suivants :

- `--db-instance-identifier value`
- `--tenant-db-name value`
- `[--new-tenant-db-name value]`
- `[--master-user-password value]`

L'exemple suivant renomme la base de données locataire `pdb1` en `pdb-hr` sur l'instance de base de données `my-cdb-inst`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-tenant-database --region us-east-1 \
    --db-instance-identifier my-cdb-inst \
    --tenant-db-name pdb1 \
    --new-tenant-db-name pdb-hr
```

Dans Windows :

```
aws rds modify-tenant-database --region us-east-1 ^
    --db-instance-identifier my-cdb-inst ^
    --tenant-db-name pdb1 ^
    --new-tenant-db-name pdb-hr
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
{
    "TenantDatabase" : {
        "DbiResourceId" : "db-abc123",
        "TenantDatabaseResourceId" : "tdb-bac567",
        "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-inst:pdb1",
        "DBInstanceIdentifier" : "my-cdb-inst",
        "TenantDBName" : "pdb1",
        "Status" : "modifying",
        "MasterUsername" : "tenant-admin-user"
        "Port" : "6555",
```

```
        "CharacterSetName" : "UTF-16",
        "MaxAllocatedStorage" : "1000",
        "ParameterGroups": [
            {
                "ParameterGroupName": "pdb1-params",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "OptionGroupMemberships": [
            {
                "OptionGroupName": "pdb1-options",
                "Status": "in-sync"
            }
        ],
        "PendingModifiedValues": {
            "TenantDBName": "pdb-hr"
        }
    }
}
```

Suppression d'une base de données locataire RDS for Oracle de votre CDB

Vous pouvez supprimer une base de données locataire (PDB) à l'aide de la AWS Management Console, de l'AWS CLI ou de l'API RDS. Tenez compte des conditions préalables requises et des limitations suivantes :

- La base de données locataire et l'instance de base de données doivent exister.
- Pour que la suppression réussisse, l'une des situations suivantes doit exister :
  - La base de données locataire et l'instance de base de données sont disponibles.

    > **Note**
    >
    > Vous pouvez prendre un instantané final, mais uniquement si la base de données locataire et l'instance de base de données étaient disponibles avant l'émission de la commande `delete-tenant-database`.

  - La base de données locataire est en cours de création.
  - L'instance de base de données modifie la base de données locataire.
- Vous ne pouvez pas supprimer plusieurs bases de données locataire au cours d'une seule opération.

- Vous ne pouvez pas supprimer une base de données locataire si elle est le seul locataire de la CDB.

Console

Pour supprimer une base de données locataire

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis choisissez la base de données locataire que vous souhaitez supprimer.
3. Pour Actions, choisissez Supprimer.
4. Pour créer un instantané de base de données final pour l'instance de base de données, choisissez Create final snapshot? (Créer un instantané final ?).
5. Si vous avez choisi de créer un instantané final, entrez le paramètre Final snapshot name (Nom de l'instantané final).
6. Saisissez **`delete me`** dans la zone.
7. Sélectionnez Delete.

AWS CLI

Pour supprimer une base de données mutualisée à l'aide deAWS CLI, appelez la delete-tenant-databasecommande avec les paramètres suivants :

- `--db-instance-identifier` *`value`*
- `--tenant-db-name` *`value`*
- `[--skip-final-snapshot | --no-skip-final-snapshot]`
- `[--final-snapshot-identifier` *`value`*`]`

L'exemple suivant supprime la base de données mutualisée nommée *`pdb-test`* de la base de données nommée. *`my-cdb-inst`* Par défaut, l'opération crée un instantané final.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds delete-tenant-database --region us-east-1 \
    --db-instance-identifier my-cdb-inst \
    --tenant-db-name pdb-test \
    --final-snapshot-identifier final-snap-pdb-test
```

Dans Windows :

```
aws rds delete-tenant-database --region us-east-1 ^
    --db-instance-identifier my-cdb-inst ^
    --tenant-db-name pdb-test ^
    --final-snapshot-identifier final-snap-pdb-test
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
{
    "TenantDatabase" : {
        "DbiResourceId" : "db-abc123",
        "TenantDatabaseResourceId" : "tdb-bac456",
        "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-inst:pdb-
test",
        "DBInstanceIdentifier" : "my-cdb-inst",
        "TenantDBName" : "pdb-test",
        "Status" : "deleting",
        "MasterUsername" : "pdb-test-admin"
        "Port" : "6555",
        "CharacterSetName" : "UTF-16",
        "MaxAllocatedStorage" : "1000",
        "ParameterGroups": [
            {
                "ParameterGroupName": "tenant-1-params",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "OptionGroupMemberships": [
            {
                "OptionGroupName": "tenant-1-options",
                "Status": "in-sync"
            }
        ]
    }
}
```

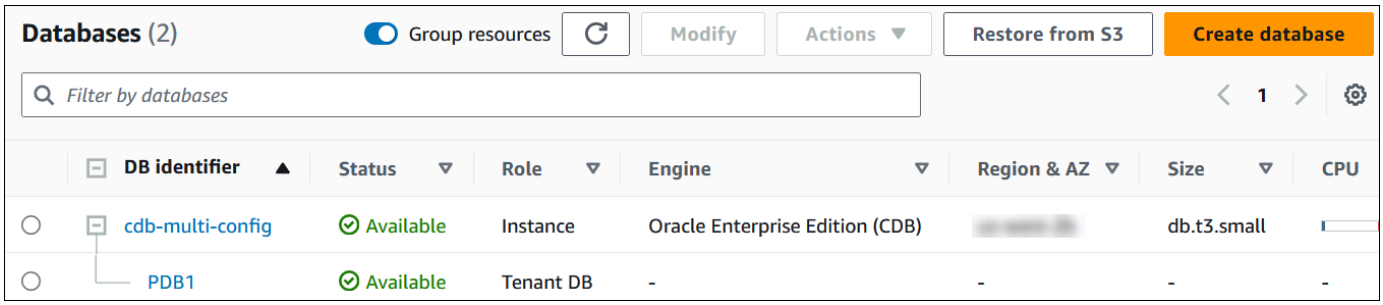
Affichage des détails de la base de données locataire

Vous pouvez afficher les détails d'une base de données locataire de la même manière que pour une base de données non CDB ou CDB.

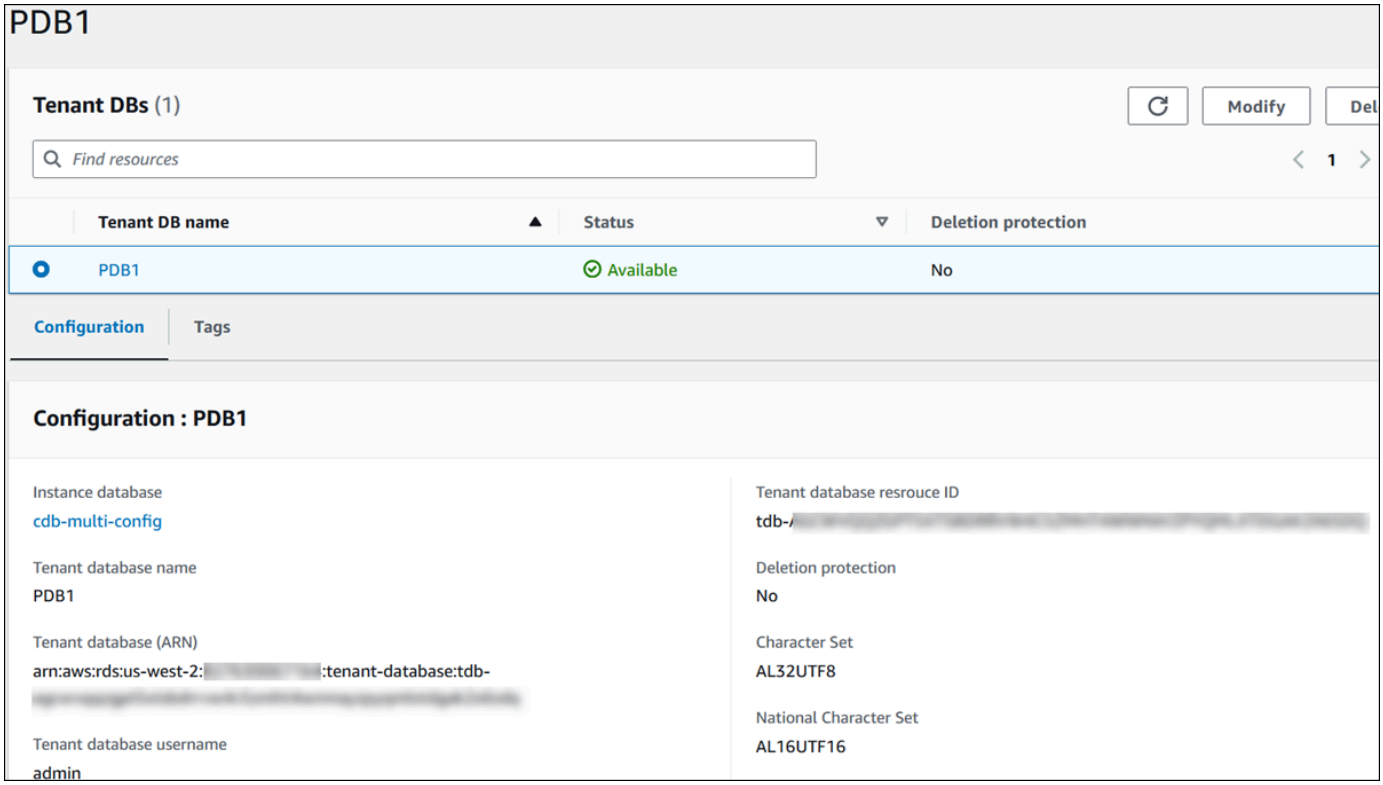
Console

Pour afficher les détails d'une base de données locataire

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le coin supérieur droit de la console Amazon RDS, choisissez la Région AWS où réside votre instance de base de données.
3. Dans le panneau de navigation, choisissez Databases (Bases de données).



   Dans l'image précédente, la base de données locataire unique (PDB) apparaît en tant qu'enfant de l'instance de base de données.
4. Choisissez le nom d'une base de données locataire.



AWS CLI

Pour voir les détails de vos PDB, utilisez la AWS CLI commande describe-tenant-databases.

L'exemple suivant décrit toutes les bases de données locataire de la région spécifiée.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-tenant-databases --region us-east-1
```

Dans Windows :

```
aws rds describe-tenant-databases --region us-east-1
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
    "TenantDatabases" : [
        {
           "DBInstanceIdentifier" : "my-cdb-inst",
           "TenantDBName" : "pdb-test",
```

```
            "Status" : "available",
            "MasterUsername" : "pdb-test-admin",
            "DbiResourceId" : "db-abc123",
            "TenantDatabaseResourceId" : "tdb-bac456",
            "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-
inst:pdb-test",
            "CharacterSetName": "AL32UTF8",
            "NcharCharacterSetName": "AL16UTF16",
            "DeletionProtection": false,
            "PendingModifiedValues": {
                 "MasterUserPassword": "****"
            },
            "TagList": []
         },
         {

            "DBInstanceIdentifier" : "my-cdb-inst2",
            "TenantDBName" : "pdb-dev",
            "Status" : "modifying",
            "MasterUsername" : "masterrdsuser"
            "DbiResourceId" : "db-xyz789",
            "TenantDatabaseResourceId" : "tdb-ghp890",
            "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-
inst2:pdb-dev",
            "CharacterSetName": "AL32UTF8",
            "NcharCharacterSetName": "AL16UTF16",
            "DeletionProtection": false,
            "PendingModifiedValues": {
                 "MasterUserPassword": "****"
            },
            "TagList": []
         },
         ... other truncated data
```

L'exemple suivant décrit les bases de données locataire sur l'instance de base de données `my-cdb-inst` dans la région spécifiée.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-tenant-databases --region us-east-1 \
    --db-instance-identifier my-cdb-inst
```

Dans Windows :

```
aws rds describe-tenant-databases --region us-east-1 ^
    --db-instance-identifier my-cdb-inst
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
{
    "TenantDatabase": {
        "TenantDatabaseCreateTime": "2023-10-19T23:55:30.046Z",
        "DBInstanceIdentifier": "my-cdb-inst",
        "TenantDBName": "pdb-hr",
        "Status": "creating",
        "MasterUsername": "tenant-admin-user",
        "DbiResourceId": "db-abc123",
        "TenantDatabaseResourceId": "tdb-bac567",
        "TenantDatabaseARN": "arn:aws:rds:us-west-2:579508833180:pdb-hr:tdb-
abcdefghi1jklmno2p3qrst4uvw5xy6zabc7defghi8jklmn90op",
        "CharacterSetName": "AL32UTF8",
        "NcharCharacterSetName": "AL16UTF16",
        "DeletionProtection": false,
        "PendingModifiedValues": {
            "MasterUserPassword": "****"
        },
        "TagList": [
            {
                "Key": "TEST",
                "Value": "testValue"
            }
        ]
    }
}
```

L'exemple suivant décrit la base de données locataire pdb1 sur l'instance de base de données `my-cdb-inst` dans la région USA Est (Virginie du Nord).

Example

Pour LinuxmacOS, ou Unix :

```
aws rds describe-tenant-databases --region us-east-1 \
--db-instance-identifier my-cdb-inst \
```

```
--tenant-db-name pdb1
```

Dans Windows :

```
aws rds describe-tenant-databases --region us-east-1 ^
--db-instance-identifier my-cdb-inst ^
--tenant-db-name pdb1
```

Le résultat produit lors de l'exécution de cette commande est semblable à ce qui suit :

```
{
    "TenantDatabases" : [
        {
            "DbiResourceId" : "db-abc123",
            "TenantDatabaseResourceId" : "tdb-bac567",
            "TenantDatabaseArn" : "arn:aws:rds:us-east-1:123456789012:db:my-cdb-
inst:pdb1"
            "DBInstanceIdentifier" : "my-cdb-inst",
            "TenantDBName" : "pdb1",
            "Status" : "ACTIVE",
            "MasterUsername" : "masterawsuser"
            "Port" : "1234",
            "CharacterSetName": "UTF-8",
            "ParameterGroups": [
                {
                    "ParameterGroupName": "tenant-custom-pg",
                    "ParameterApplyStatus": "in-sync"
                }
            ],
            {
            "OptionGroupMemberships": [
                {
                    "OptionGroupName": "tenant-custom-og",
                    "Status": "in-sync"
                }
            ]
         }
    ]
}
```

Mise à niveau de votre CDB

Vous pouvez mettre à niveau une instance CDB vers une version différente de Oracle Database. Par exemple, vous pouvez mettre à niveau une CDB Oracle Database 19c en CDB Oracle Database 21c. Vous ne pouvez pas modifier l'architecture de base de données au cours d'une mise à niveau. Ainsi, vous ne pouvez pas mettre à niveau une base de données non-CDB en CDB ni mettre à niveau une CDB en base de données non-CDB.

La procédure de mise à niveau d'une CDB vers une CDB est la même que pour la mise à niveau d'une base de données non-CDB vers une base de données non-CDB. Pour de plus amples informations, veuillez consulter Mise à niveau du moteur de base de données RDS for Oracle.

Administration de votre instance de base de données RDS for Oracle

Voici les tâches de gestion courantes que vous effectuez avec une instance de base de données RDS for Oracle. Certaines tâches sont les mêmes pour toutes les instances de base de données RDS. D'autres tâches sont spécifiques à RDS for Oracle.

Les tâches suivantes sont communes à toutes les bases de données RDS, mais Oracle Database a des considérations spéciales. Par exemple, vous vous connectez à une base de données Oracle à l'aide des clients Oracle SQL*Plus et SQL Developer.

Type de tâche	Documentation
Classes d'instance, stockage et PIOPS Si vous créez une instance de production, découvrez comment fonctionnent les classes d'instance, les types de stockage et les IOPS provisionnées dans Amazon RDS.	RDS pour les classes d'instance de base de données Oracle Types de stockage Amazon RDS
Déploiements multi-AZ Une instance de base de données de production doit utiliser des déploiements multi-AZ. Les déploiements Multi-AZ améliorent la disponibilité, la durabilité des donnés et la tolérance aux pannes pour les instances de bases de données.	Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS
Amazon VPC Si votre compte AWS dispose d'un cloud privé virtuel (VPC) par défaut, votre instance de base de données est automatiq uement créée dans le VPC par défaut. Si votre compte n'a pas de VPC par défaut et que vous voulez que l'instance de base de données soit dans un VPC, créez le VPC et les groupes de sous-réseaux avant de créer l'instance.	Utilisation d'un(e) instance de base de données dans un VPC
Groupes de sécurité	Contrôle d'accès par groupe de sécurité

Type de tâche	Documentation
Par défaut, les instances de base de données utilisent un pare-feu qui empêche l'accès. Veillez à créer un groupe de sécurité avec les adresses IP et la configuration réseau voulues pour accéder à l'instance de base de données.	
Groupes de paramètres Si votre instance de base de données doit nécessiter des paramètres de base de données spécifiques, créez un groupe de paramètres avant de créer l'instance de base de données.	Groupes de paramètres pour (Amazon RDS)
Groupes d'options Si votre instance de base de données nécessite des options spécifiques, créez un groupe d'options avant de créer l'instance de base de données.	Ajout d'options aux instances de base de données Oracle
Connexion à votre instance de base de données Après avoir créé un groupe de sécurité et l'avoir associé à une instance de base de données, vous pouvez vous connecter à l'instance de base de données en utilisant une application cliente SQL standard quelconque telle qu'Oracle SQL*Plus.	Connexion à votre instance de base de données Oracle
Sauvegarde et restauration Vous pouvez configurer votre instance de base de données pour que les sauvegardes soient exécutées automatiquement ou que les instantanés soient créés manuellement, puis que les instances soient restaurées à partir des sauvegardes ou des instantanés.	Sauvegarde, restauration et exportation de données
Surveillance Vous pouvez surveiller une instance de base de données Oracle à l'aide des métriques, des événements et de la surveillance améliorée d' CloudWatch Amazon RDS.	Afficher les métriques dans la RDS console Amazon Consulter les RDS événements Amazon

Type de tâche	Documentation
Les fichiers journaux Vous pouvez accéder aux fichiers journaux de votre instance de base de données Oracle.	Surveillance des fichiers journaux RDSAmazon

Vous trouverez ci-après une description d'implémentations spécifiques d'Amazon RDS de tâches courantes d'administrateur de base de données pour RDS Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données. RDS restreint également l'accès à certaines procédures système et tables qui requièrent des privilèges avancés. Dans de nombreuses tâches, vous exécutez le package `rdsadmin`, un outil spécifique d'Amazon RDS qui vous permet d'administrer votre base de données.

Les tâches DBA courantes pour les instances de base de données exécutant Oracle sont les suivantes :

- Tâches système

Déconnexion d'une session	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.disconnect` Méthode Oracle : `alter system disconnect session`
Terminer une session	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.kill` Méthode Oracle : `alter system kill session`
Annulation d'un SQL relevé au cours d'une session	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.cancel` Méthode Oracle : `alter system cancel sql`
Activation et désactivation de sessions restreintes	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.restricted_session` Méthode Oracle : `alter system enable restricted session`
Vidage du pool partagé	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.flush_shared_pool`

	Méthode Oracle : `alter system flush shared_pool`
Vidage du cache de tampon	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.flush_buffer_cache` Méthode Oracle : `alter system flush buffer_cache`
Octroi SELECT de EXECUTE privilèges à SYS des objets	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.grant_sys_object` Méthode Oracle : `grant`
Révocation SELECT de EXECUTE privilèges sur des objets SYS	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.revoke_sys_object` Méthode Oracle : `revoke`
Gestion des vues RDS_X$ pour les instances de base de données Oracle	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.create_sys_x$_view` Méthode Oracle : `CREATE VIEW`
Attribution de privilèges à des utilisateurs non-maîtres	Méthode Amazon RDS : `grant`
Création de fonctions personnalisées pour vérifier les mots de passe	Méthode Amazon RDS : `rdsadmin.rdsadmin_password_verify.create_verify_function` Méthode Amazon RDS : `rdsadmin.rdsadmin_password_verify.create_passthrough_verify_fcn`
Configuration d'un DNS serveur personnalisé	—

Liste des événements de diagnostic système autorisés	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.list_allowed_system_events` Méthode Oracle : —
Activation des événements de diagnostic système	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.set_allowed_system_events` Méthode Oracle : `ALTER SYSTEM SET EVENTS 'set_event_clause'`
Liste des événements de diagnostic système activés	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.list_set_system_events` Méthode Oracle : `ALTER SESSION SET EVENTS 'IMMEDIATE EVENTDUMP(SYSTEM)'`
Désactivation des événements de diagnostic système	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.unset_system_event` Méthode Oracle : `ALTER SYSTEM SET EVENTS 'unset_event_clause'`

- Tâches de base de données

Modification du nom global d'une base de données	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.rename_global_name` Méthode Oracle : `alter database rename`
Création et dimensionnement des espaces de table	Méthode Amazon RDS : `create tablespace` Méthode Oracle : `alter database`

Définition de l'espace de table par défaut	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.alter_default_tablespace` Méthode Oracle : `alter database default tablespace`
Définition de l'espace de table temporaire par défaut	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.alter_default_temp_tablespace` Méthode Oracle : `alter database default temporary tablespace`
Création d'un espace de table temporaire sur le stockage d'instances	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.create_inst_store_tmp_tblspace` Méthode Oracle : `create temporary tablespace`
Création d'un point de contrôle de base de données	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.checkpoint` Méthode Oracle : `alter system checkpoint`
Définition d'une récupération distribuée	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.enable_distr_recovery` Méthode Oracle : `alter system enable distributed recovery`
Définition du fuseau horaire de la base de données	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.alter_db_time_zone` Méthode Oracle : `alter database set time_zone`
Utilisation de tables externes Oracle	—
Génération de rapports de performance avec Automatic Workload Repository (AWR)	Méthode Amazon RDS : procédures `rdsadmin.rdsadmin_diagnostic_util` Méthode Oracle : package `dbms_workload_repository`

Ajustement des liens de base de données à utiliser avec des instances de base de données dans un VPC	—
Définition de l'édition par défaut d'une instance de base de données	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.alter_default_edition` Méthode Oracle : `alter database default edition`
Activation de l'audit pourSYS.AUDtable $	Méthode Amazon RDS : `rdsadmin.rdsadmin_master_util.audit_all_sys_aud_table` Méthode Oracle : `audit`
Désactivation de l'audit pour. SYS AUDtable $	Méthode Amazon RDS : `rdsadmin.rdsadmin_master_util.noaudit_all_sys_aud_table` Méthode Oracle : `noaudit`
Nettoyage de builds d'index en ligne interrompues	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_repair.online_index_clean` Méthode Oracle : `dbms_repair.online_index_clean`
Ignorer les blocs corrompus	Méthode Amazon RDS : plusieurs procédures `rdsadmin.rdsadmin_dbms_repair` Méthode Oracle : package `dbms_repair`
Redimensionnement des espaces de table, des fichiers de données et des fichiers temporaires	Méthode Amazon RDS : procédures `rdsadmin.rdsadmin_util.resize_temp_tablespace` , `rdsadmin.rdsadmin_util.resize_tempfile` ou `rdsadmin.rdsadmin_util.autoextend_tempfile` Procédure `rdsadmin.rdsadmin_util.resize_datafile` ou `rdsadmin.rdsadmin_util.autoextend_datafile` Méthode Oracle : —

Purge de la corbeille	Méthode Amazon RDS : `EXEC rdsadmin.rdsadmin_util.purge_dba_recyclebin` Méthode Oracle : `purge dba_recyclebin`
Définition des valeurs affichées par défaut pour une édition complète	Méthode Amazon RDS : `EXEC rdsadmin.rdsadmin_util.dbms_redact_upd_full_rdct_val` Méthode Oracle : `exec dbms_redact.UPDATE_FULL_REDACTION_VALUES`

- Tâches de journal

Configuration du mode FORCE LOGGING	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.force_logging` Méthode Oracle : `alter database force logging`
Configuration d'une journalisation supplémentaire	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.alter_supplemental_logging` Méthode Oracle : `alter database add supplemental log`
Changement de fichiers journaux en ligne	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.switch_logfile` Méthode Oracle : `alter system switch logfile`

Ajout de journaux redo en ligne	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.add_logfile`
Suppression de journaux redo en ligne	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.drop_logfile`
Redimensionnement de journaux redo en ligne	—
Conservation des journaux redo archivés	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.set_configuration`
Téléchargement des journaux de reprise archivés à partir d'Amazon S3	Méthode Amazon RDS : `rdsadmin.rdsadmin_archive_log_download.download_log_with_seqnum` Méthode Amazon RDS : `rdsadmin.rdsadmin_archive_log_download.download_logs_in_seqnum_range`
Accès aux journaux de reprise en ligne et archivés	Méthode Amazon RDS : `rdsadmin.rdsadmin_master_util.create_archivelog_dir` Méthode Amazon RDS : `rdsadmin.rdsadmin_master_util.create_onlinelog_dir`

- RMANtâches

Validation des fichiers de base de données dans RDS Oracle	Méthode Amazon RDS : `rdsadmin_rman_util . procedure` Méthode Oracle : `RMAN VALIDATE`
Activation et désactivation du suivi des modifications de bloc	Méthode Amazon RDS : `rdsadmin_rman_util . procedure` Méthode Oracle : `ALTER DATABASE`
Recoupement des journaux redo archivés	Méthode Amazon RDS : `rdsadmin_rman_util .crosscheck_archivelog` Méthode Oracle : `RMAN BACKUP`
Sauvegarde des fichiers de journalisation archivés	Méthode Amazon RDS : `rdsadmin_rman_util . procedure` Méthode Oracle : `RMAN BACKUP`
Réalisation d'une sauvegarde complète de base de données	Méthode Amazon RDS : `rdsadmin_rman_util .backup_database_full` Méthode Oracle : `RMAN BACKUP`

Réalisation d'une sauvegarde incrémentielle de base de données	Méthode Amazon RDS : `rdsadmin_rman_util.backup_database_incremental` Méthode Oracle : `RMAN BACKUP`
Sauvegarde d'un espace de table	Méthode Amazon RDS : `rdsadmin_rman_util.backup_database_tablespace` Méthode Oracle : `RMAN BACKUP`

- Tâches Oracle Scheduler

Modification de DBMS _ SCHEDULER jobs	Méthode Amazon RDS : `dbms_scheduler.set_attribute` Méthode Oracle : `dbms_scheduler.set_attribute`
Modification des fenêtres AutoTask de maintenance	Méthode Amazon RDS : `dbms_scheduler.set_attribute` Méthode Oracle : `dbms_scheduler.set_attribute`

Définition du fuseau horaire pour les tâches d'Oracle Scheduler	Méthode Amazon RDS : `dbms_scheduler.set_scheduler_attribute` Méthode Oracle : `dbms_scheduler.set_scheduler_attribute`
Désactiver les tâches Oracle Scheduler détenues par SYS	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_scheduler.disable` Méthode Oracle : `dbms_scheduler.disable`
Activation des tâches Oracle Scheduler détenues par SYS	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_scheduler.enable` Méthode Oracle : `dbms_scheduler.enable`
Modification de l'intervalle de répétition d'Oracle Scheduler pour les tâches de type CALENDAR	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_scheduler.set_attribute` Méthode Oracle : `dbms_scheduler.set_attribute`

Modification de l'intervalle de répétition d'Oracle Scheduler pour les tâches de type NAMED	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_scheduler.set_attribute` Méthode Oracle : `dbms_scheduler.set_attribute`
Désactivation de la validation automatique pour la création de travaux Oracle Scheduler	Méthode Amazon RDS : `rdsadmin.rdsadmin_dbms_scheduler.set_no_commit_flag` Méthode Oracle : `dbms_isched.set_no_commit_flag`

- Diagnostic des problèmes

Répertorier les incidents	Méthode Amazon RDS : `rdsadmin.rdsadmin_adrci_util.list_adrci_incidents` Méthode Oracle : commande ADRCI `show incident`
Répertorier les problèmes	Méthode Amazon RDS : `rdsadmin.rdsadmin_adrci_util.list_adrci_problem` Méthode Oracle : commande ADRCI `show problem`

Création de packages d'incidents	Méthode Amazon RDS : `rdsadmin.rdsadmin_adrci_util.create_adrci_package` Méthode Oracle : commande ADRCI `ips create package`
Affichage des fichiers de trace	Méthode Amazon RDS : `rdsadmin.rdsadmin_adrci_util.show_adrci_tracefile` Méthode Oracle : commande ADRCI `show tracefile`

- Autres tâches

Création et suppression de répertoires dans l'espace de stockage de données principal	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.create_directory` Méthode Oracle : `CREATE DIRECTORY` Méthode Amazon RDS : `rdsadmin.rdsadmin_util.drop_directory` Méthode Oracle : `DROP DIRECTORY`

Établissement de la liste des fichiers situés dans un répertoire d'instance de base de données	Méthode Amazon RDS : `rdsadmin.rds_file_util.listdir` Méthode Oracle : —
Lecture de fichiers dans un répertoire d'instance de base de données	Méthode Amazon RDS : `rdsadmin.rds_file_util.read_text_file` Méthode Oracle : —
Accès aux fichiers Opatch	Méthode Amazon RDS : `rdsadmin.rds_file_util.read_text_file` ou `rdsadmin.tracefile_listing` Méthode Oracle : `opatch`
Définition des paramètres des tâches de conseiller	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.advisor_task_set_parameter` Méthode Oracle : Plusieurs procédures de package stockées
Désactivation de AUTO_STATS_ADVISOR_TASK	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.advisor_task_drop` Méthode Oracle : —

Réactivation de AUTO_STATS_ADVISOR_TASK	Méthode Amazon RDS : `rdsadmin.rdsadmin_util.dbms_stats_init` Méthode Oracle : —

Vous pouvez également utiliser les procédures Amazon RDS for l'intégration de Amazon S3 avec Oracle et pour l'exécution de tâches de base de données OEM Management Agent. Pour plus d'informations, consultez Intégration Amazon S3 et Exécution de tâches de base de données avec l'option Management Agent.

Exécution des tâches système courantes pour les instances de bases de données Oracle

Vous trouverez ci-dessous comment effectuer certaines DBA tâches courantes liées au système sur vos RDS instances de base de données Amazon exécutant Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données et restreint l'accès à certaines procédures et tables du système qui nécessitent des privilèges avancés.

Rubriques

- Déconnexion d'une session
- Terminer une session
- Annulation d'un SQL relevé au cours d'une session
- Activation et désactivation de sessions restreintes
- Vidage du pool partagé
- Octroi SELECT de EXECUTE privilèges à SYS des objets
- Révocation SELECT de EXECUTE privilèges sur des objets SYS
- Gestion des vues RDS_X$ pour les instances de base de données Oracle
- Attribution de privilèges à des utilisateurs non-maîtres
- Création de fonctions personnalisées pour vérifier les mots de passe
- Configuration d'un DNS serveur personnalisé

- Activation et désactivation des événements de diagnostic système

Déconnexion d'une session

Pour déconnecter la session en cours en mettant fin au processus du serveur dédié, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.disconnect`. La procédure `disconnect` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`sid`	nombre	—	Oui	Identifiant de session.
`serial`	nombre	—	Oui	Numéro de série de la session.
`method`	varchar	'IMMEDIATE'	Non	Les valeurs valides sont `'IMMEDIATE'` ou `'POST_TRANSACTION'` .

L'exemple suivant déconnecte une session.

```
begin
    rdsadmin.rdsadmin_util.disconnect(
        sid    => sid,
        serial => serial_number);
end;
/
```

Pour obtenir l'identifiant et le numéro de série de la session, interrogez la vue `V$SESSION`. L'exemple suivant obtient toutes les sessions pour l'utilisateur `AWSUSER`.

```
SELECT SID, SERIAL#, STATUS FROM V$SESSION WHERE USERNAME = 'AWSUSER';
```

La base de données doit être ouverte pour pouvoir utiliser cette méthode. Pour plus d'informations sur la déconnexion d'une session, consultez ALTERSYSTEMla documentation Oracle.

Terminer une session

Pour mettre fin à une session, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.kill`. La procédure `kill` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`sid`	nombre	—	Oui	Identifiant de session.
`serial`	nombre	—	Oui	Numéro de série de la session.
`method`	varchar	null	Non	Les valeurs valides sont `'IMMEDIATE'` ou `'PROCESS'` . Si vous spécifiez `IMMEDIATE` , cela a le même effet que d'exécuter l'instruction suivante : `ALTER SYSTEM KILL SESSION 'sid,serial#' IMMEDIATE` Si vous spécifiez `PROCESS`, vous résiliez les processus associés à une séance. Spécifiez `PROCESS` uniquement si la résiliation de la session à l'aide de `IMMEDIATE` n'a pas abouti.

Pour obtenir l'identifiant et le numéro de série de la session, interrogez la vue `V$SESSION`. L'exemple suivant obtient toutes les sessions de l'utilisateur *`AWSUSER`*.

```
SELECT SID, SERIAL#, STATUS FROM V$SESSION WHERE USERNAME = 'AWSUSER';
```

L'exemple suivant met fin à une session.

```
BEGIN
    rdsadmin.rdsadmin_util.kill(
        sid    => sid,
        serial => serial_number,
        method => 'IMMEDIATE');
END;
/
```

L'exemple suivant résilie les processus associés à une séance.

```
BEGIN
    rdsadmin.rdsadmin_util.kill(
        sid    => sid,
        serial => serial_number,
        method => 'PROCESS');
END;
/
```

Annulation d'un SQL relevé au cours d'une session

Pour annuler un SQL relevé au cours d'une session, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.cancel`.

> **Note**
>
> Cette procédure est prise en charge pour Oracle Database 19c (19.0.0) et toutes les versions majeures et mineures supérieures RDS d'Oracle.

La procédure `cancel` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`sid`	nombre	—	Oui	Identifiant de session.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`serial`	nombre	—	Oui	Numéro de série de la session.
`sql_id`	varchar2	null	Non	SQLIdentifiant de la SQL déclaration.

L'exemple suivant annule une SQL instruction au cours d'une session.

```
begin
    rdsadmin.rdsadmin_util.cancel(
        sid    => sid,
        serial => serial_number,
        sql_id => sql_id);
end;
/
```

Pour obtenir l'identifiant de session, le numéro de série de session et l'SQLidentifiant d'une SQL instruction, interrogez la `V$SESSION` vue. L'exemple suivant permet d'obtenir toutes les sessions et tous SQL les identifiants de l'utilisateur`AWSUSER`.

```
select SID, SERIAL#, SQL_ID, STATUS from V$SESSION where USERNAME = 'AWSUSER';
```

Activation et désactivation de sessions restreintes

Pour activer et désactiver les sessions restreintes, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.restricted_session`. La procédure `restricted_session` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Oui	Description
`p_enable`	booléen	true	Non	Définissez ce paramètre sur `true` pour activer les sessions restreintes

Nom du paramètre	Type de données	Par défaut	Oui	Description
				ou sur `false` pour les désactiver.

L'exemple suivant montre comment activer et désactiver les sessions restreintes.

```
/* Verify that the database is currently unrestricted. */

SELECT LOGINS FROM V$INSTANCE;

LOGINS
-------
ALLOWED

/* Enable restricted sessions */

EXEC rdsadmin.rdsadmin_util.restricted_session(p_enable => true);


/* Verify that the database is now restricted. */

SELECT LOGINS FROM V$INSTANCE;

LOGINS
----------
RESTRICTED


/* Disable restricted sessions */

EXEC rdsadmin.rdsadmin_util.restricted_session(p_enable => false);


/* Verify that the database is now unrestricted again. */

SELECT LOGINS FROM V$INSTANCE;

LOGINS
-------
```

```
ALLOWED
```

Vidage du pool partagé

Pour vider le pool partagé, utilisez la RDS procédure `rdsadmin.rdsadmin_util.flush_shared_pool` Amazon. La procédure `flush_shared_pool` ne comporte aucun paramètre.

L'exemple suivant vide le pool partagé.

```
EXEC rdsadmin.rdsadmin_util.flush_shared_pool;
```

Vidage du cache de tampon

Pour vider le cache de la mémoire tampon, utilisez la RDS procédure `rdsadmin.rdsadmin_util.flush_buffer_cache` Amazon. La procédure `flush_buffer_cache` ne comporte aucun paramètre.

L'exemple suivant vide le cache des tampons.

```
EXEC rdsadmin.rdsadmin_util.flush_buffer_cache;
```

Vider le cache Smart Flash de la base de données

Pour vider le cache Smart Flash de la base de données, utilisez la RDS procédure `rdsadmin.rdsadmin_util.flush_flash_cache` Amazon. La procédure `flush_flash_cache` ne comporte aucun paramètre. L'exemple suivant vide le cache Smart Flash de la base de données.

```
EXEC rdsadmin.rdsadmin_util.flush_flash_cache;
```

Pour plus d'informations sur l'utilisation du cache Smart Flash de base RDS de données avec Oracle, consultezStockage de données temporaires dans un stockage d'instances RDS for Oracle.

Octroi SELECT de EXECUTE privilèges à SYS des objets

Généralement, vous transférez les privilèges en utilisant des rôles, qui peuvent contenir de nombreux objets. Pour accorder des privilèges à un seul objet, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.grant_sys_object`. La procédure accorde uniquement les privilèges qui ont déjà été accordés à l'utilisateur principal par le biais d'un rôle ou d'une attribution directe.

La procédure `grant_sys_object` possède les paramètres suivants.

> **Important**
>
> Pour toutes les valeurs de paramètre, utilisez les majuscules sauf si vous avez créé l'utilisateur avec un identifiant sensible à la casse. Par exemple, si vous exécutez `CREATE USER myuser` ou `CREATE USER MYUSER`, le dictionnaire de données stocke `MYUSER`. Toutefois, si vous utilisez des guillemets doubles dans `CREATE USER "MyUser"`, le dictionnaire de données stocke `MyUser`.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_obj_name`	varchar2	—	Oui	Nom de l'objet pour lequel des privilèges seront accordés. L'objet peut être un répertoire, une fonction, un package, une procédure, une séquence, une table ou une vue. Les noms d'objet doivent être orthographiés correctement lorsqu'ils apparaissent dans `DBA_OBJECTS`. La plupart des objets système sont définis en majuscules, si bien que nous vous recommandons d'essayer cela en premier lieu.
`p_grantee`	varchar2	—	Oui	Nom de l'objet auquel des privilèges seront accordés. L'objet peut

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				être un schéma ou un rôle.
p_privilege	varchar2	null	Oui	—
p_grant_option	booléen	false	Non	Définissez ce paramètre sur `true` pour l'utiliser avec l'option d'attribution.

L'exemple suivant accorde des privilèges select sur un objet nommé `V_$SESSION` à un utilisateur nommé `USER1`.

```
begin
    rdsadmin.rdsadmin_util.grant_sys_object(
        p_obj_name  => 'V_$SESSION',
        p_grantee   => 'USER1',
        p_privilege => 'SELECT');
end;
/
```

L'exemple suivant accorde des privilèges select sur un objet nommé `V_$SESSION` à un utilisateur nommé `USER1` avec l'option d'attribution.

```
begin
    rdsadmin.rdsadmin_util.grant_sys_object(
        p_obj_name     => 'V_$SESSION',
        p_grantee      => 'USER1',
        p_privilege    => 'SELECT',
        p_grant_option => true);
end;
/
```

Pour pouvoir attribuer des privilèges sur un objet, votre compte doit avoir ces privilèges directement attribués avec l'option appropriée ou via un rôle accordé avec `with admin option`. Dans le cas le plus courant, vous souhaiterez peut-être accorder une autorisation `SELECT` sur une DBA vue qui a été accordée au `SELECT_CATALOG_ROLE` rôle. Si ce rôle n'est pas déjà directement attribué à votre

utilisateur avec `with admin option`, vous ne pouvez pas transférer le privilège. Si vous avez le DBA privilège, vous pouvez attribuer le rôle directement à un autre utilisateur.

L'exemple suivant accorde les rôles `SELECT_CATALOG_ROLE` et `EXECUTE_CATALOG_ROLE` à `USER1`. Puisque le `with admin option` est utilisé, il `USER1` peut désormais accorder l'accès aux SYS objets qui ont été accordés à`SELECT_CATALOG_ROLE`.

```
GRANT SELECT_CATALOG_ROLE TO USER1 WITH ADMIN OPTION;
GRANT EXECUTE_CATALOG_ROLE to USER1 WITH ADMIN OPTION;
```

Les objets déjà attribués à `PUBLIC` n'ont pas besoin d'être réattribués. Si vous utilisez la procédure `grant_sys_object` pour accorder de nouveau l'accès, l'appel de procédure réussit.

Révocation SELECT de EXECUTE privilèges sur des objets SYS

Pour révoquer des privilèges sur un seul objet, utilisez la RDS procédure `rdsadmin.rdsadmin_util.revoke_sys_object` Amazon. La procédure révoque uniquement les privilèges qui ont déjà été accordés au compte principal par le biais d'un rôle ou d'une attribution directe.

La procédure `revoke_sys_object` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_obj_name`	varchar2	—	Oui	Nom de l'objet pour lequel des privilèges seront révoqués. L'objet peut être un répertoire, une fonction, un package, une procédure, une séquence, une table ou une vue. Les noms d'objet doivent être orthographiés correctement lorsqu'ils apparaissent dans `DBA_OBJECTS` . La plupart des objets système sont

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				définis en majuscules, c'est pourquoi nous vous recommandons d'essayer cette méthode en premier lieu.
p_revokee	varchar2	—	Oui	Nom de l'objet pour lequel des privilèges seront révoqués. L'objet peut être un schéma ou un rôle.
p_privilege	varchar2	null	Oui	—

L'exemple suivant révoque des privilèges select sur un objet nommé V_$SESSION à un utilisateur nommé USER1.

```
begin
    rdsadmin.rdsadmin_util.revoke_sys_object(
        p_obj_name  => 'V_$SESSION',
        p_revokee   => 'USER1',
        p_privilege => 'SELECT');
end;
/
```

Gestion des vues RDS_X$ pour les instances de base de données Oracle

Vous devrez peut-être accéder à des tables SYS.X$ fixes, qui ne sont accessibles que parSYS. Pour créer des SYS.RDS_X$ vues sur X$ les tables éligibles, utilisez les procédures du rdsadmin.rdsadmin_util package. Votre utilisateur principal se voit automatiquement octroyer le privilège SELECT … WITH GRANT OPTION d'accès aux RDS_X$ vues.

Les rdsadmin.rdsadmin_util procédures sont disponibles dans les cas suivants :

- Instances de base de données existantes qui n'ont jamais été mises à niveau et qui utilisent les versions suivantes :

- `21.0.0.0.ru-2023-10.rur-2023-10.r1`Versions 21c et supérieures
- `19.0.0.0.ru-2023-10.rur-2023-10.r1`Versions 19c et supérieures

- Toute nouvelle instance de base de données que vous créez
- Toute instance de base de données existante que vous avez mise à niveau

> **Important**
>
> En interne, le `rdsadmin.rdsadmin_util` package crée des vues sur `X$` les tables. Les `X$` tables sont des objets système internes qui ne sont pas décrits dans la documentation de la base de données Oracle. Nous vous recommandons de tester des vues spécifiques dans votre base de données hors production et de ne créer des vues dans votre base de données de production que sous la supervision d'Oracle Support.

Liste des tables fixes X$ pouvant être utilisées dans les vues RDS_X$

Pour répertorier les tables X$ pouvant être utilisées dans les `RDS_X$` vues, utilisez la procédure RDS. `rdsadmin.rdsadmin_util.list_allowed_sys_x$_views` Cette procédure n'accepte aucun paramètre. Les déclarations suivantes répertorient tous les `X$` tableaux éligibles (échantillon de sortie inclus).

```
SQL> SET SERVEROUTPUT ON
SQL> SELECT * FROM TABLE(rdsadmin.rdsadmin_util.list_allowed_sys_x$_views);

'X$BH'
'X$K2GTE'
'X$KCBWBPD'
'X$KCBWDS'
'X$KGLLK'
'X$KGLOB'
'X$KGLPN'
'X$KSLHOT'
'X$KSMSP'
'X$KSPPCV'
'X$KSPPI'
'X$KSPPSV'
'X$KSQEQ'
'X$KSQRS'
'X$KTUXE'
```

```
'X$KQRFP'
```

La liste des X$ tables éligibles peut changer au fil du temps. Pour vous assurer que votre liste de tables X$ fixes éligibles est à jour, réexécutez-la `list_allowed_sys_x$_views` régulièrement.

Création de vues SYS.RDS_X$

Pour créer une RDS_X$ vue sur une X$ table éligible, utilisez la procédure `rdsadmin.rdsadmin_util.create_sys_x$_view` RDS. Vous ne pouvez créer des vues que pour les tables répertoriées dans la sortie de`rdsadmin.rdsadmin_util.list_allowed_sys_x$_views`. La procédure `create_sys_x$_view` accepte les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_x$_tbl`	varchar2	Null	Oui	Nom de X$ table valide. La valeur doit être l'une des X$ tables rapportées par`list_allowed_sys_x$_views` .
`p_force_creation`	Booléen	`FALSE`	Non	Une valeur indiquant s'il faut forcer la création d'une RDS_X$ vue qui existe déjà pour une X$ table. Par défaut, RDS ne crée pas de vue si elle existe déjà. Pour forcer la création, définissez ce paramètre sur`TRUE`.

L'exemple suivant crée la `SYS.RDS_X$KGLOB` vue sur la table`X$KGLOB`. Le format du nom de la vue estRDS_*X$tablename*.

```
SQL> SET SERVEROUTPUT ON
SQL> EXEC rdsadmin.rdsadmin_util.create_sys_x$_view('X$KGLOB');
```

```
PL/SQL procedure successfully completed.
```

La requête de dictionnaire de données suivante répertorie la vue SYS.RDS_X$KGLOB et indique son état. Votre utilisateur principal est automatiquement autorisé à accéder `SELECT ... WITH GRANT OPTION` à cette vue.

```
SQL> SET SERVEROUTPUT ON
SQL> COL OWNER FORMAT A30
SQL> COL OBJECT_NAME FORMAT A30
SQL> COL STATUS FORMAT A30
SQL> SET LINESIZE 200
SQL> SELECT OWNER, OBJECT_NAME, STATUS
FROM DBA_OBJECTS
WHERE OWNER = 'SYS' AND OBJECT_NAME = 'RDS_X$KGLOB';

OWNER                          OBJECT_NAME                    STATUS
------------------------------ ------------------------------
 -------------------------------
SYS                            RDS_X$KGLOB                    VALID
```

> **Important**
>
> X$il n'est pas garanti que les tables resteront les mêmes avant et après un surclassement. RDS for Oracle supprime et recrée les RDS_X$ vues sur les X$ tables lors d'une mise à niveau du moteur. Il accorde ensuite le `SELECT ... WITH GRANT OPTION` privilège à l'utilisateur principal. Après une mise à niveau, accordez des privilèges aux utilisateurs de la base de données selon les besoins sur les RDS_X$ vues correspondantes.

Répertorier les vues SYS.RDS_X$

Pour répertorier les RDS_X$ vues existantes, utilisez la procédure `rdsadmin.rdsadmin_util.list_created_sys_x$_views` RDS. La procédure répertorie uniquement les vues créées par la procédure`create_sys_x$_view`. L'exemple suivant répertorie les X$ tables associées aux RDS_X$ vues correspondantes (exemple de sortie inclus).

```
SQL> SET SERVEROUTPUT ON
SQL> COL XD_TBL_NAME FORMAT A30
SQL> COL STATUS FORMAT A30
SQL> SET LINESIZE 200
```

```
SQL> SELECT * FROM TABLE(rdsadmin.rdsadmin_util.list_created_sys_x$_views);

XD_TBL_NAME                    STATUS
------------------------------ ------------------------------
X$BH                           VALID
X$K2GTE                        VALID
X$KCBWBPD                      VALID

3 rows selected.
```

Supprimer les vues RDS_X$

Pour supprimer une SYS.RDS_X$ vue, utilisez la procédure `rdsadmin.rdsadmin_util.drop_sys_x$_view` RDS. Vous ne pouvez supprimer que les vues répertoriées dans la sortie de`rdsadmin.rdsadmin_util.list_allowed_sys_x$_views`. La procédure `drop_sys_x$_view` accepte les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_x$_tbl`	varchar2	Null	Oui	Nom de table X$ fixe valide. La valeur doit être l'une des tables X$ fixes rapportées par`list_created_sys_x$_views` .

L'exemple suivant supprime la RDS_X$KGLOB vue créée sur le tableauX$KGLOB.

```
SQL> SET SERVEROUTPUT ON
SQL> EXEC rdsadmin.rdsadmin_util.drop_sys_x$_view('X$KGLOB');

PL/SQL procedure successfully completed.
```

L'exemple suivant montre que la vue SYS.RDS_X$KGLOB a été supprimée (exemple de sortie inclus).

```
SQL> SET SERVEROUTPUT ON
SQL> COL OWNER FORMAT A30
SQL> COL OBJECT_NAME FORMAT A30
```

```
SQL> COL STATUS FORMAT A30
SQL> SET LINESIZE 200
SQL> SELECT OWNER, OBJECT_NAME, STATUS
FROM DBA_OBJECTS
WHERE OWNER = 'SYS' AND OBJECT_NAME = 'RDS_X$KGLOB';

no rows selected
```

Attribution de privilèges à des utilisateurs non-maîtres

Vous pouvez accorder des privilèges select pour de nombreux objets dans le schéma SYS grâce au rôle SELECT_CATALOG_ROLE. Le rôle SELECT_CATALOG_ROLE accorde aux utilisateurs des privilèges SELECT sur les vues du dictionnaire de données. L'exemple suivant accorde le rôle SELECT_CATALOG_ROLE à un utilisateur nommé user1.

```
GRANT SELECT_CATALOG_ROLE TO user1;
```

Vous pouvez accorder des privilèges EXECUTE pour de nombreux objets dans le schéma SYS grâce au rôle EXECUTE_CATALOG_ROLE. Le rôle EXECUTE_CATALOG_ROLE accorde aux utilisateurs des privilèges EXECUTE pour les packages et les procédures du dictionnaire de données. L'exemple suivant accorde le rôle EXECUTE_CATALOG_ROLE à un utilisateur nommé user1.

```
GRANT EXECUTE_CATALOG_ROLE TO user1;
```

L'exemple suivant obtient les autorisations permises par les rôles SELECT_CATALOG_ROLE et EXECUTE_CATALOG_ROLE.

```
  SELECT *
    FROM ROLE_TAB_PRIVS
   WHERE ROLE IN ('SELECT_CATALOG_ROLE','EXECUTE_CATALOG_ROLE')
ORDER BY ROLE, TABLE_NAME ASC;
```

L'exemple suivant crée un utilisateur non maître nommé user1, accorde le privilège CREATE SESSION, puis accorde le privilège SELECT sur une base de données nommée sh.sales.

```
CREATE USER user1 IDENTIFIED BY PASSWORD;
GRANT CREATE SESSION TO user1;
GRANT SELECT ON sh.sales TO user1;
```

Création de fonctions personnalisées pour vérifier les mots de passe

Vous pouvez créer une fonction de vérification de mot de passe personnalisée des manières suivantes :

- Pour utiliser la logique de vérification standard et stocker votre fonction dans le schéma SYS, utilisez la procédure `create_verify_function`.
- Pour utiliser la logique de vérification personnalisée, ou pour éviter de stocker votre fonction dans le schéma SYS, utilisez la procédure `create_passthrough_verify_fcn`.

Procédure create_verify_function

Vous pouvez créer une fonction personnalisée pour vérifier les mots de passe en utilisant la RDS procédure Amazon`rdsadmin.rdsadmin_password_verify.create_verify_function`. La `create_verify_function` procédure est prise en charge pour toutes les versions de RDS for Oracle.

La procédure `create_verify_function` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_verify_function_name`	varchar2	—	Oui	Nom de la fonction personnalisée. Cette fonction est créée pour vous dans le SYS schéma. Elle est affectée à des profils utilisateur.
`p_min_length`	nombre	8	Non	Nombre minimal de caractères requis.
`p_max_length`	nombre	256	Non	Nombre maximal de caractères autorisés.
`p_min_letters`	nombre	1	Non	Nombre minimal de lettres requises.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_min_uppercase`	nombre	0	Non	Nombre minimal de lettres majuscules requises.
`p_min_lowercase`	nombre	0	Non	Nombre minimal de lettres minuscules requises.
`p_min_digits`	nombre	1	Non	Nombre minimal de chiffres requis.
`p_min_special`	nombre	0	Non	Nombre minimal de caractères spéciaux requis.
`p_min_different_chars`	nombre	3	Non	Nombre minimal de caractères différents requis entre l'ancien et le nouveau mot de passe.
`p_disallow_username`	booléen	true	Non	Définissez ce paramètre sur `true` pour interdire le nom d'utilisateur dans le mot de passe.
`p_disallow_reverse`	booléen	true	Non	Définissez ce paramètre sur `true` pour interdire le nom d'utilisateur inversé dans le mot de passe.
`p_disallow_db_name`	booléen	true	Non	Définissez ce paramètre sur `true` pour interdire le nom de la base de données ou du serveur dans le mot de passe.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_disallow_simple_strings	booléen	true	Non	Définissez ce paramètre sur true pour interdire l'utilisation de chaînes simples comme mot de passe.
p_disallow_whitespace	booléen	false	Non	Définissez ce paramètre sur true pour interdire les espaces dans le mot de passe.
p_disallow_at_sign	booléen	false	Non	Définissez ce paramètre sur true pour interdire le caractère @ dans le mot de passe.

Vous pouvez créer plusieurs fonctions de vérification de mot de passe.

Il existe des restrictions sur le nom de votre fonction personnalisée. Votre fonction personnalisée ne peut pas avoir le même nom qu'un objet système existant. Ce nom ne peut pas comporter plus de 30 caractères. De plus, le nom doit inclure l'une des chaînes suivantes : PASSWORD, VERIFY, COMPLEXITY, ENFORCE ou STRENGTH.

L'exemple suivant crée une fonction nommée CUSTOM_PASSWORD_FUNCTION. La fonction exige qu'un mot de passe comporte au moins 12 caractères, 2 majuscules, 1 chiffre et 1 caractère spécial, et interdit le caractère @.

```
begin
    rdsadmin.rdsadmin_password_verify.create_verify_function(
        p_verify_function_name => 'CUSTOM_PASSWORD_FUNCTION',
        p_min_length           => 12,
        p_min_uppercase        => 2,
        p_min_digits           => 1,
        p_min_special          => 1,
        p_disallow_at_sign     => true);
end;
```

```
/
```

Pour voir le texte de votre fonction de vérification, interrogez DBA_SOURCE. L'exemple suivant récupère le texte d'une fonction de mot de passe personnalisé nommée CUSTOM_PASSWORD_FUNCTION.

```
COL TEXT FORMAT a150

  SELECT TEXT
    FROM DBA_SOURCE
   WHERE OWNER = 'SYS'
     AND NAME = 'CUSTOM_PASSWORD_FUNCTION'
ORDER BY LINE;
```

Pour associer votre fonction de vérification à un profil utilisateur, utilisez `alter profile`. L'exemple suivant associe une fonction de vérification au profil utilisateur DEFAULT.

```
ALTER PROFILE DEFAULT LIMIT PASSWORD_VERIFY_FUNCTION CUSTOM_PASSWORD_FUNCTION;
```

Pour savoir quels profils utilisateur sont associés à quelles fonctions de vérification, interrogez DBA_PROFILES. L'exemple suivant obtient les profils qui sont associés à la fonction de vérification personnalisée nommée CUSTOM_PASSWORD_FUNCTION.

```
SELECT * FROM DBA_PROFILES WHERE RESOURCE_NAME = 'PASSWORD' AND LIMIT =
 'CUSTOM_PASSWORD_FUNCTION';


PROFILE                   RESOURCE_NAME                    RESOURCE  LIMIT
------------------------  -------------------------------  --------
 ------------------------
DEFAULT                   PASSWORD_VERIFY_FUNCTION         PASSWORD
 CUSTOM_PASSWORD_FUNCTION
```

L'exemple suivant récupère tous les profils et les fonctions de vérification de mot de passe associées.

```
SELECT * FROM DBA_PROFILES WHERE RESOURCE_NAME = 'PASSWORD_VERIFY_FUNCTION';

PROFILE                   RESOURCE_NAME                    RESOURCE  LIMIT
------------------------  -------------------------------  --------
 ------------------------
```

```
DEFAULT                       PASSWORD_VERIFY_FUNCTION          PASSWORD
 CUSTOM_PASSWORD_FUNCTION
RDSADMIN                      PASSWORD_VERIFY_FUNCTION          PASSWORD  NULL
```

Procédure create_passthrough_verify_fcn

La `create_passthrough_verify_fcn` procédure est prise en charge pour toutes les versions de RDS for Oracle.

Vous pouvez créer une fonction personnalisée pour vérifier les mots de passe à l'aide de la RDS procédure Amazon`rdsadmin.rdsadmin_password_verify.create_passthrough_verify_fcn`. La procédure `create_passthrough_verify_fcn` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_verify_function_name`	varchar2	—	Oui	Nom de la fonction de vérification personnalisée. Il s'agit d'une fonction d'encapsulation créée pour vous dans le SYS schéma, et elle ne contient aucune logique de vérification. Elle est affectée à des profils utilisateur.
`p_target_owner`	varchar2	—	Oui	Propriétaire de schéma de la fonction de vérification personnalisée.
`p_target_function_name`	varchar2	—	Oui	Nom de la fonction personnalisée existante qui contient la logique de vérification. Votre fonction personnalisée doit renvoyer une valeur

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				booléenne. La fonction doit renvoyer la valeur `true` si le mot de passe est valide, ou `false` s'il ne l'est pas.

L'exemple suivant crée une fonction de vérification de mot de passe qui utilise la logique provenant de la fonction nommée `PASSWORD_LOGIC_EXTRA_STRONG`.

```
begin
    rdsadmin.rdsadmin_password_verify.create_passthrough_verify_fcn(
        p_verify_function_name => 'CUSTOM_PASSWORD_FUNCTION',
        p_target_owner         => 'TEST_USER',
        p_target_function_name => 'PASSWORD_LOGIC_EXTRA_STRONG');
end;
/
```

Pour associer la fonction de vérification à un profil utilisateur, utilisez `alter profile`. L'exemple suivant associe la fonction de vérification au profil utilisateur `DEFAULT`.

```
ALTER PROFILE DEFAULT LIMIT PASSWORD_VERIFY_FUNCTION CUSTOM_PASSWORD_FUNCTION;
```

Configuration d'un DNS serveur personnalisé

Amazon RDS prend en charge l'accès réseau sortant sur vos instances de base de données exécutant Oracle. Pour plus d'informations sur l'utilisation de l'accès réseau sortant et les prérequis, consultez Configuration de l'accès UTL_HTTP à l'aide de certificats et d'un portefeuille Oracle.

Amazon RDS Oracle autorise la résolution du Domain Name Service (DNS) à partir d'un DNS serveur personnalisé appartenant au client. Vous ne pouvez résoudre que des noms de domaine entièrement qualifiés à partir de votre RDS instance de base de données Amazon via votre DNS serveur personnalisé.

Une fois que vous avez configuré votre serveur de DNS noms personnalisé, la propagation des modifications sur votre instance de base de données prend jusqu'à 30 minutes. Une fois

les modifications propagées à votre instance de base de données, tout le trafic réseau sortant nécessitant une DNS recherche interroge votre DNS serveur via le port 53.

Pour configurer un DNS serveur personnalisé pour votre instance de base de données Amazon RDS pour Oracle, procédez comme suit :

- Parmi les DHCP options définies associées à votre cloud privé virtuel (VPC), définissez l'`domain-name-servers`option sur l'adresse IP de votre serveur de DNS noms. Pour plus d'informations, consultez la section Ensembles DHCP d'options.

  > **Note**
  >
  > L'`domain-name-servers`option accepte jusqu'à quatre valeurs, mais votre RDS instance de base de données Amazon n'utilise que la première valeur.

- Assurez-vous que votre DNS serveur peut résoudre toutes les requêtes de recherche, y compris les DNS noms publics, les DNS noms EC2 privés Amazon et les noms spécifiques aux clientsDNS. Si le trafic réseau sortant contient des DNS recherches que votre DNS serveur ne peut pas gérer, les DNS fournisseurs en amont appropriés doivent être configurés sur votre DNS serveur.
- Configurez votre DNS serveur pour produire des réponses User Datagram Protocol (UDP) de 512 octets ou moins.
- Configurez votre DNS serveur pour produire des réponses du protocole de contrôle de transmission (TCP) de 1 024 octets ou moins.
- Configurez votre DNS serveur pour autoriser le trafic entrant depuis vos RDS instances de base de données Amazon via le port 53. Si votre DNS serveur se trouve dans un AmazonVPC, VPC il doit disposer d'un groupe de sécurité contenant des règles UDP entrantes autorisant le TCP trafic sur le port 53. Si votre DNS serveur ne se trouve pas sur AmazonVPC, il doit disposer d'une liste d'autorisations de pare-feu appropriée pour autoriser UDP le trafic TCP entrant sur le port 53.

  Pour plus d'informations, consultez les sections Groupes de sécurité pour vous VPC et Ajout et suppression de règles.
- Configurez votre RDS instance VPC de base de données Amazon pour autoriser le trafic sortant via le port 53. Vous VPC devez disposer d'un groupe de sécurité contenant des règles sortantes autorisant le TCP trafic sur le port 53. UDP

  Pour plus d'informations, consultez les sections Groupes de sécurité pour vous VPC et Ajout et suppression de règles.

- Le chemin de routage entre l'RDSinstance de base de données Amazon et le DNS serveur doit être correctement configuré pour autoriser DNS le trafic.
  - Si l'RDSinstance de base de données Amazon et le DNS serveur ne se trouvent pas dans le même emplacementVPC, une connexion d'appairage doit être établie entre eux. Pour plus d'informations, consultez Qu'est-ce que le VPC peering ?

Activation et désactivation des événements de diagnostic système

Pour définir et annuler les événements de diagnostic au niveau de la session, vous pouvez utiliser l'SQLinstruction `ALTER SESSION SET EVENTS` Oracle. Toutefois, pour définir des événements au niveau du système, vous ne pouvez pas utiliser OracleSQL. Pour cela, utilisez les procédures d'événements système du package `rdsadmin.rdsadmin_util`. Les procédures d'événements système sont disponibles dans les versions suivantes du moteur :

- Toutes les versions de Oracle Database 21c
- 19.0.0.0.ru-2020-10.rur-2020-10.r1 et versions ultérieures de Oracle Database 19c

  Pour plus d'informations, consultez la version 19.0.0.0.ru-2020-10.rur-2020-10.r1 dans les notes de mise à jour d'Amazon pour Oracle RDS

> **Important**
>
> En interne, le package `rdsadmin.rdsadmin_util` active les événements à l'aide de l'instruction `ALTER SYSTEM SET EVENTS`. Cette instruction `ALTER SYSTEM` ne figure pas dans la documentation Oracle Database. Certains événements de diagnostic système peuvent générer de grandes quantités d'informations de suivi, provoquer des conflits ou affecter la disponibilité de la base de données. Nous vous recommandons de tester des événements de diagnostic spécifiques dans votre base de données hors production, et de n'activer des événements dans votre base de données de production que sous la direction du support Oracle.

Liste des événements de diagnostic système autorisés

Pour répertorier les événements système que vous pouvez définir, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.list_allowed_system_events`. Cette procédure n'accepte aucun paramètre.

L'exemple suivant répertorie tous les événements système que vous pouvez activer.

```
SET SERVEROUTPUT ON
EXEC rdsadmin.rdsadmin_util.list_allowed_system_events;
```

L'exemple de sortie suivant répertorie les numéros des événements et leur description. Utilisez les RDS procédures Amazon `set_system_event` pour définir ou annuler `unset_system_event` ces événements.

```
604   - error occurred at recursive SQL level
942   - table or view does not exist
1401  - inserted value too large for column
1403  - no data found
1410  - invalid ROWID
1422  - exact fetch returns more than requested number of rows
1426  - numeric overflow
1427  - single-row subquery returns more than one row
1476  - divisor is equal to zero
1483  - invalid length for DATE or NUMBER bind variable
1489  - result of string concatenation is too long
1652  - unable to extend temp segment by  in tablespace
1858  - a non-numeric character was found where a numeric was expected
4031  - unable to allocate  bytes of shared memory ("","","","")
6502  - PL/SQL: numeric or value error
10027 - Specify Deadlock Trace Information to be Dumped
10046 - enable SQL statement timing
10053 - CBO Enable optimizer trace
10173 - Dynamic Sampling time-out error
10442 - enable trace of kst for ORA-01555 diagnostics
12008 - error in materialized view refresh path
12012 - error on auto execute of job
12504 - TNS:listener was not given the SERVICE_NAME in CONNECT_DATA
14400 - inserted partition key does not map to any partition
31693 - Table data object  failed to load/unload and is being skipped due to error:
```

> **Note**
>
> La liste des événements système autorisés peut changer au fil du temps. Pour vous assurer que vous disposez de la liste la plus récente des événements éligibles, utilisez `rdsadmin.rdsadmin_util.list_allowed_system_events`.

Activation des événements de diagnostic système

Pour définir un événement système, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.set_system_event`. Vous ne pouvez activer que les événements répertoriés dans la sortie de `rdsadmin.rdsadmin_util.list_allowed_system_events`. La procédure `set_system_event` accepte les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_event`	nombre	—	Oui	Numéro de l'événement système. La valeur doit correspondre à l'un des numéros d'événement signalés par `list_allowed_system_events` .
`p_level`	nombre	—	Oui	Niveau de l'événement. Pour obtenir les descriptions des différentes valeurs de niveau, consultez la documentation Oracle Database ou contactez le support Oracle.

La procédure `set_system_event` permet de créer et d'exécuter les instructions `ALTER SYSTEM SET EVENTS` requises selon les principes suivants :

- Le type d'événement (`context` ou `errorstack`) est déterminé automatiquement.
- Une instruction du formulaire `ALTER SYSTEM SET EVENTS '`*`event`*` LEVEL `*`event_level`*`'` active les événements de contexte. Cette notation équivaut à `ALTER SYSTEM SET EVENTS '`*`event`*` TRACE NAME CONTEXT FOREVER, LEVEL `*`event_level`*`'`.

- Une instruction du formulaire `ALTER SYSTEM SET EVENTS 'event ERRORSTACK (event_level)'` active les événements de pile d'erreurs. Cette notation équivaut à `ALTER SYSTEM SET EVENTS 'event TRACE NAME ERRORSTACK LEVEL event_level'`.

L'exemple suivant active l'événement 942 au niveau 3, et l'événement 10442 au niveau 10. Un exemple de sortie est inclus.

```
SQL> SET SERVEROUTPUT ON
SQL> EXEC rdsadmin.rdsadmin_util.set_system_event(942,3);
Setting system event 942 with: alter system set events '942 errorstack (3)'

PL/SQL procedure successfully completed.

SQL> EXEC rdsadmin.rdsadmin_util.set_system_event(10442,10);
Setting system event 10442 with: alter system set events '10442 level 10'

PL/SQL procedure successfully completed.
```

Liste des événements de diagnostic système activés

Pour répertorier les événements système actuellement définis, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.list_set_system_events`. Cette procédure signale uniquement les événements activés au niveau système par `set_system_event`.

L'exemple suivant répertorie les événements système actifs.

```
SET SERVEROUTPUT ON
EXEC rdsadmin.rdsadmin_util.list_set_system_events;
```

L'exemple de sortie suivant contient la liste des événements, le type d'événement, le niveau auquel les événements sont activés et l'heure à laquelle ils ont été activés.

```
942 errorstack (3) - set at 2020-11-03 11:42:27
10442 level 10 - set at 2020-11-03 11:42:41

PL/SQL procedure successfully completed.
```

Désactivation des événements de diagnostic système

Pour annuler un événement système, utilisez la RDS procédure `rdsadmin.rdsadmin_util.unset_system_event` Amazon. Vous ne pouvez désactiver que les événements répertoriés dans la sortie de `rdsadmin.rdsadmin_util.list_allowed_system_events`. La procédure `unset_system_event` accepte les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_event`	nombre	—	Oui	Numéro de l'événement système. La valeur doit correspondre à l'un des numéros d'événement signalés par `list_allowed_system_events` .

L'exemple suivant désactive les événements 942 et 10442. Un exemple de sortie est inclus.

```
SQL> SET SERVEROUTPUT ON
SQL> EXEC rdsadmin.rdsadmin_util.unset_system_event(942);
Unsetting system event 942 with: alter system set events '942 off'

PL/SQL procedure successfully completed.

SQL> EXEC rdsadmin.rdsadmin_util.unset_system_event(10442);
Unsetting system event 10442 with: alter system set events '10442 off'

PL/SQL procedure successfully completed.
```

Exécution des tâches de base de données courantes pour les instances de base de données Oracle

Vous trouverez ci-dessous comment effectuer certaines DBA tâches courantes liées aux bases de données sur vos RDS instances de base de données Amazon exécutant Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de

données. Amazon restreint RDS également l'accès à certaines procédures et tables du système qui nécessitent des privilèges avancés.

Rubriques

- Modification du nom global d'une base de données
- Utilisation des tablespaces
- Utilisation de fichiers temporaires
- Création d'un point de contrôle de base de données
- Définition d'une récupération distribuée
- Définition du fuseau horaire de la base de données
- Utilisation de tables externes Oracle
- Génération de rapports de performance avec Automatic Workload Repository (AWR)
- Ajustement des liens de base de données à utiliser avec des instances de base de données dans un VPC
- Définition de l'édition par défaut d'une instance de base de données
- Activation de l'audit pourSYS. AUDtable $
- Désactivation de l'audit pour. SYS AUDtable $
- Nettoyage de builds d'index en ligne interrompues
- Ignorer les blocs corrompus
- Redimensionnement des espaces de table, des fichiers de données et des fichiers temporaires
- Purge de la corbeille
- Définition des valeurs affichées par défaut pour une édition complète

Modification du nom global d'une base de données

Pour modifier le nom global d'une base de données, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.rename_global_name`. La procédure `rename_global_name` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_new_global_name	varchar2	—	Oui	Nouveau nom global de la base de données.

La base de données doit être ouverte pour que la modification du nom puisse se produire. Pour plus d'informations sur la modification du nom global d'une base de données, consultez ALTERDATABASEla documentation Oracle.

L'exemple suivant remplace le nom global de la base de données par new_global_name.

```
EXEC rdsadmin.rdsadmin_util.rename_global_name(p_new_global_name => 'new_global_name');
```

Utilisation des tablespaces

Vous pouvez utiliser des tablespaces avec RDS for Oracle, qui est une unité de stockage logique qui stocke les données de la base de données.

Rubriques

- Création et dimensionnement des espaces de table
- Définition de l'espace de table par défaut
- Définition de l'espace de table temporaire par défaut
- Création d'un espace de table temporaire sur le stockage d'instances

Création et dimensionnement des espaces de table

Amazon prend RDS uniquement en charge Oracle Managed Files (OMF) pour les fichiers de données, les fichiers journaux et les fichiers de contrôle. Lorsque vous créez des fichiers de données et des fichiers journaux, vous ne pouvez pas spécifier les noms de fichiers physiques.

Par défaut, si vous ne spécifiez pas de taille de fichier de données, les espaces de table sont créés avec AUTOEXTEND ON par défaut et sans taille maximum. Dans l'exemple suivant, le tablespace *users1* est autoextensible.

```
CREATE TABLESPACE users1;
```

A cause de ces paramètres par défaut, les espaces de table peuvent se développer pour utiliser l'ensemble du stockage alloué. Nous vous recommandons de spécifier une taille maximum appropriée sur les espaces de table permanents et temporaires, et de surveiller attentivement l'utilisation de l'espace.

L'exemple suivant crée un tablespace nommé *users2* avec une taille initiale de 1 gigaoctet. Puisque la taille du fichier de données est spécifiée, mais pas `AUTOEXTEND ON`, l'espace de tables n'est pas auto-extensible.

```
CREATE TABLESPACE users2 DATAFILE SIZE 1G;
```

L'exemple suivant crée un tablespace nommé *users3* avec une taille initiale de 1 gigaoctet, l'extension automatique activée et une taille maximale de 10 gigaoctets.

```
CREATE TABLESPACE users3 DATAFILE SIZE 1G AUTOEXTEND ON MAXSIZE 10G;
```

L'exemple suivant crée un tablespace temporaire nommé *temp01*.

```
CREATE TEMPORARY TABLESPACE temp01;
```

Vous pouvez redimensionner un espace de table bigfile en utilisant `ALTER TABLESPACE`. Vous pouvez spécifier la taille en kilo-octets (Ko), mégaoctets (Mo), gigaoctets (Go) ou téraoctets (To). L'exemple suivant redimensionne un tablespace de grande taille nommé *users_bf* jusqu'à 200 Mo.

```
ALTER TABLESPACE users_bf RESIZE 200M;
```

L'exemple suivant ajoute un fichier de données supplémentaire à un espace de table de type petit fichier nommé *users_sf*.

```
ALTER TABLESPACE users_sf ADD DATAFILE SIZE 100000M AUTOEXTEND ON NEXT 250m
 MAXSIZE UNLIMITED;
```

Définition de l'espace de table par défaut

Pour définir le tablespace par défaut, utilisez la procédure AmazonRDS. `rdsadmin.rdsadmin_util.alter_default_tablespace` La procédure `alter_default_tablespace` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`tablespace_name`	varchar	—	Oui	Nom de l'espace de table par défaut.

L'exemple suivant définit le tablespace par défaut sur *`users2`*:

```
EXEC rdsadmin.rdsadmin_util.alter_default_tablespace(tablespace_name => 'users2');
```

Définition de l'espace de table temporaire par défaut

Pour définir le tablespace temporaire par défaut, utilisez la procédure AmazonRDS. `rdsadmin.rdsadmin_util.alter_default_temp_tablespace` La procédure `alter_default_temp_tablespace` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`tablespace_name`	varchar	—	Oui	Nom de l'espace de table temporaire par défaut.

L'exemple suivant définit le tablespace temporaire par défaut sur *`temp01`*.

```
EXEC rdsadmin.rdsadmin_util.alter_default_temp_tablespace(tablespace_name => 'temp01');
```

Création d'un espace de table temporaire sur le stockage d'instances

Pour créer un tablespace temporaire sur le magasin d'instance, utilisez la procédure AmazonRDS. `rdsadmin.rdsadmin_util.create_inst_store_tmp_tblspace` La procédure `create_inst_store_tmp_tblspace` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_tablespace_name	varchar	—	Oui	Nom de l'espace de table temporaire.

L'exemple suivant crée le tablespace temporaire *temp01* dans le magasin d'instances.

```
EXEC rdsadmin.rdsadmin_util.create_inst_store_tmp_tblspace(p_tablespace_name =>
 'temp01');
```

> **Important**
>
> Lorsque vous exécutez `rdsadmin_util.create_inst_store_tmp_tblspace`, l'espace de table temporaire nouvellement créé n'est pas automatiquement défini comme l'espace de table temporaire par défaut. Pour le définir comme valeur par défaut, consultez Définition de l'espace de table temporaire par défaut.

Pour de plus amples informations, veuillez consulter Stockage de données temporaires dans un stockage d'instances RDS for Oracle.

Utilisation de fichiers temporaires

Ajout d'un fichier temporaire au stockage d'instances sur un réplica en lecture

Lorsque vous créez un espace de table temporaire sur une instance de base de données principale, le réplica en lecture ne crée pas de fichiers temporaires. Supposons qu'un espace de table temporaire vide existe sur votre réplica en lecture pour l'une des raisons suivantes :

- Vous avez déposé un fichier temporaire de l'espace de table sur votre réplica en lecture. Pour de plus amples informations, veuillez consulter Dépôt de fichiers temporaires sur un réplica en lecture.
- Vous avez créé un nouvel espace de table temporaire sur l'instance de base de données principale. Dans ce cas, RDS for Oracle synchronise les métadonnées avec le réplica lu.

Vous pouvez ajouter un fichier temporaire à l'espace de table temporaire vide et stocker le fichier temporaire dans le stockage d'instances. Pour créer un fichier temporaire dans le magasin d'instance,

utilisez la RDS procédure `rdsadmin.rdsadmin_util.add_inst_store_tempfile` Amazon. Vous pouvez utiliser cette procédure uniquement sur un réplica en lecture. La procédure possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_tablespace_name`	varchar	—	Oui	Nom de l'espace de table temporaire sur votre réplica en lecture.

Dans l'exemple suivant, le tablespace temporaire vide *`temp01`* existe sur votre réplique de lecture. Exécutez la commande suivante pour créer un fichier temporaire pour cet espace de table et le stocker dans le stockage d'instances.

```
EXEC rdsadmin.rdsadmin_util.add_inst_store_tempfile(p_tablespace_name => 'temp01');
```

Pour de plus amples informations, veuillez consulter Stockage de données temporaires dans un stockage d'instances RDS for Oracle.

Dépôt de fichiers temporaires sur un réplica en lecture

Vous ne pouvez pas créer un espace de table temporaire existant sur un réplica en lecture. Vous pouvez modifier le stockage des fichiers temporaires sur une réplique en lecture depuis Amazon EBS vers le magasin d'instance, ou depuis le magasin d'instance vers AmazonEBS. Pour atteindre ces objectifs, procédez comme suit :

1. Déposez les fichiers temporaires actuels dans l'espace de table temporaire du réplica en lecture.
2. Créez de nouveaux fichiers temporaires sur différents stockages.

Pour supprimer les fichiers temporaires, utilisez la RDS procédure `rdsadmin.rdsadmin_util.drop_replica_tempfiles` Amazon. Vous pouvez utiliser cette procédure uniquement sur des réplicas en lecture. La procédure `drop_replica_tempfiles` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_tablespace_name	varchar	—	Oui	Nom de l'espace de table temporaire sur votre réplica en lecture.

Supposons qu'un tablespace temporaire nommé *temp01* se trouve dans le magasin d'instances de votre réplique de lecture. Déposez tous les fichiers temporaires dans cet espace de table en exécutant la commande suivante.

```
EXEC rdsadmin.rdsadmin_util.drop_replica_tempfiles(p_tablespace_name => 'temp01');
```

Pour de plus amples informations, veuillez consulter Stockage de données temporaires dans un stockage d'instances RDS for Oracle.

Création d'un point de contrôle de base de données

Pour vérifier la base de données, utilisez la RDS procédure Amazon `rdsadmin.rdsadmin_util.checkpoint`. La procédure `checkpoint` ne comporte aucun paramètre.

L'exemple suivant crée un point de contrôle sur la base de données.

```
EXEC rdsadmin.rdsadmin_util.checkpoint;
```

Définition d'une récupération distribuée

Pour configurer la restauration distribuée, utilisez les RDS procédures Amazon `rdsadmin.rdsadmin_util.enable_distr_recovery` et`disable_distr_recovery`. Ces procédures ne comportent aucun paramètre.

L'exemple suivant active la récupération distribuée.

```
EXEC rdsadmin.rdsadmin_util.enable_distr_recovery;
```

L'exemple suivant désactive la récupération distribuée.

```
EXEC rdsadmin.rdsadmin_util.disable_distr_recovery;
```

Définition du fuseau horaire de la base de données

Vous pouvez définir le fuseau horaire de votre base de données Amazon RDS Oracle de la manière suivante :

- L'option `Timezone`

  L'option `Timezone` modifie le fuseau horaire au niveau de l'hôte et impacte toutes les valeurs et colonnes date, telles que `SYSDATE`. Pour de plus amples informations, veuillez consulter Fuseau horaire Oracle.

- La RDS procédure Amazon `rdsadmin.rdsadmin_util.alter_db_time_zone`

  La procédure `alter_db_time_zone` modifie le fuseau horaire uniquement pour certains types de données, et ne change pas `SYSDATE`. Il existe des restrictions supplémentaires sur la définition du fuseau horaire, répertoriées dans la documentation Oracle.

> **Note**
>
> Vous pouvez également définir le fuseau horaire par défaut pour Oracle Scheduler. Pour plus d'informations, consultez Définition du fuseau horaire pour les tâches d'Oracle Scheduler.

La procédure `alter_db_time_zone` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_new_tz`	varchar2	—	Oui	Le nouveau fuseau horaire sous forme de région nommée ou de décalage absolu par rapport au temps universel coordonné (UTC). Les décalages

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				valides s'étendent de -12h00 à +14h00.

L'exemple suivant modifie le fuseau horaire en UTC plus de trois heures.

```
EXEC rdsadmin.rdsadmin_util.alter_db_time_zone(p_new_tz => '+3:00');
```

L'exemple suivant définit le fuseau horaire de la région Africa/Algiers.

```
EXEC rdsadmin.rdsadmin_util.alter_db_time_zone(p_new_tz => 'Africa/Algiers');
```

Après avoir modifié le fuseau horaire grâce à la procédure `alter_db_time_zone`, redémarrez l'instance de base de données pour que la modification prenne effet. Pour plus d'informations, consultez Redémarrage d'une instance de base de données. Pour de plus amples informations sur la mise à niveau des fuseaux horaires, veuillez consulter Considérations relatives au fuseau horaire

Utilisation de tables externes Oracle

Les tables externes Oracle sont des tables contenant des données ne figurant pas dans la base de données. À la place, les données se trouvent dans des fichiers externes auxquels la base de données peut accéder. L'utilisation de tables externes vous permet d'accéder aux données sans les charger dans la base de données. Pour de plus amples informations sur les tables externes, veuillez consulter Managing External Tables dans la documentation Oracle.

Amazon vous permet RDS de stocker des fichiers de table externes dans des objets de répertoire. Vous pouvez créer un objet de répertoire ou utiliser un objet prédéfini dans la base de données Oracle, tel que le DIR répertoire DATA PUMP _ _. Pour plus d'informations sur la création d'objets de répertoire, consultez Création et suppression de répertoires dans l'espace de stockage de données principal. Vous pouvez interroger la DIRECTORIES vue ALL _ pour répertorier les objets de répertoire de votre instance de base de données Amazon RDS Oracle.

> **Note**
>
> Les objets de répertoire pointent vers l'espace de stockage de données principal (EBSvolume Amazon) utilisé par votre instance. L'espace utilisé—ainsi que les fichiers de données, journaux redo, d'audit, de suivi et autres—sont déduits du stockage alloué.

Vous pouvez déplacer un fichier de données externe d'une base de données Oracle à une autre en utilisant le TRANSFER package DBMSFILE_ _ ou le FILE package UTL_. Le fichier de données externes est déplacé d'un répertoire de la base de données source vers le répertoire spécifié sur la base de données de destination. Pour plus d'informations sur l'utilisation de DBMS FILE _ _TRANSFER, consultezImportation à l'aide d'Oracle Data Pump.

Après avoir déplacé le fichier de données externe, celui-ci peut vous permettre de créer une table externe. L'exemple suivant crée une table externe qui utilise le `emp_xt_file1.txt` fichier du DIR1 répertoire USER _.

```
CREATE TABLE emp_xt (
  emp_id      NUMBER,
  first_name  VARCHAR2(50),
  last_name   VARCHAR2(50),
  user_name   VARCHAR2(20)
)
ORGANIZATION EXTERNAL (
  TYPE ORACLE_LOADER
  DEFAULT DIRECTORY USER_DIR1
  ACCESS PARAMETERS (
    RECORDS DELIMITED BY NEWLINE
    FIELDS TERMINATED BY ','
    MISSING FIELD VALUES ARE NULL
    (emp_id,first_name,last_name,user_name)
  )
  LOCATION ('emp_xt_file1.txt')
)
PARALLEL
REJECT LIMIT UNLIMITED;
```

Supposons que vous souhaitiez déplacer les données d'une instance de base de données Amazon RDS Oracle vers un fichier de données externe. Dans ce as, vous pouvez remplir le fichier de données externe en créant une table externe et en sélectionnant les données de la table de la base

de données. Par exemple, l'SQLinstruction suivante crée la table `orders_xt` externe en interrogeant la `orders` table dans la base de données.

```
CREATE TABLE orders_xt
  ORGANIZATION EXTERNAL
   (
     TYPE ORACLE_DATAPUMP
     DEFAULT DIRECTORY DATA_PUMP_DIR
     LOCATION ('orders_xt.dmp')
   )
   AS SELECT * FROM orders;
```

Dans cet exemple, les données sont renseignées dans le `orders_xt.dmp` fichier du DIR répertoire DATA PUMP _ _.

Génération de rapports de performance avec Automatic Workload Repository (AWR)

Pour recueillir des données de performance et générer des rapports, Oracle recommande Automatic Workload Repository (AWR). AWRnécessite Oracle Database Enterprise Edition et une licence pour les packs de diagnostic et de réglage. Pour l'activerAWR, définissez le paramètre `CONTROL_MANAGEMENT_PACK_ACCESS` d'initialisation sur `DIAGNOSTIC` ou`DIAGNOSTIC+TUNING`.

Utilisation de AWR rapports dans RDS

Pour générer des AWR rapports, vous pouvez exécuter des scripts tels que`awrrpt.sql`. Ces scripts sont installés sur le serveur hôte de base de données. Sur AmazonRDS, vous n'avez pas d'accès direct à l'hôte. Vous pouvez toutefois obtenir des copies de SQL scripts à partir d'une autre installation d'Oracle Database.

Vous pouvez également l'utiliser AWR en exécutant des procédures dans le SQL package `SYS.DBMS_WORKLOAD_REPOSITORY` PL/. Vous pouvez utiliser ce package pour gérer les lignes de base et les instantanés, ainsi que pour les ASH affichages et les rapports. AWR Par exemple, pour générer un AWR rapport au format texte, exécutez la `DBMS_WORKLOAD_REPOSITORY.AWR_REPORT_TEXT` procédure. Cependant, vous ne pouvez pas accéder à ces AWR rapports depuis le AWS Management Console.

Lorsque vous travaillez avecAWR, nous vous recommandons d'utiliser les `rdsadmin.rdsadmin_diagnostic_util` procédures. Vous pouvez utiliser ces procédures pour générer les éléments suivants :

- AWRrapports

- Rapports sur l'historique des sessions actives (ASH)
- Rapports du moniteur de diagnostic automatique de base de données (ADDM)
- Oracle Data Pump Exporte des fichiers de vidage de AWR données

Les procédures `rdsadmin_diagnostic_util` enregistrent les rapports dans le système de fichiers de l'instance de base de données. Vous pouvez accéder à ces rapports à partir de la console. Vous pouvez également accéder aux rapports à l'aide des procédures `rdsadmin.rds_file_util`. Vous pouvez accéder aux rapports copiés dans Amazon S3 à l'aide de l'option S3 Integration. Pour plus d'informations, consultez Lecture de fichiers dans un répertoire d'instance de base de données et Intégration Amazon S3.

Vous pouvez utiliser les `rdsadmin_diagnostic_util` procédures des versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Toutes les versions de Oracle Database 21c
- 19.0.0.0.ru-2020-04.rur-2020-04.r1 et versions ultérieures de Oracle Database 19c

Pour consulter un blog expliquant comment utiliser les rapports de diagnostic dans un scénario de réplication, consultez Generate AWR reports for Amazon RDS for Oracle read replicas.

Paramètres communs pour le package d'utilitaires de diagnostic

Vous utilisez généralement les paramètres suivants lors de la gestion AWR et de ADDM l'utilisation du `rdsadmin_diagnostic_util` package.

Paramètre	Type de donnée	Par défaut	Obliga re	Description
`begin_snap_id`	`NUMBEI`	—	Oui	ID de l'instantané de début.
`end_snap_id`	`NUMBEI`	—	Oui	ID de l'instantané de fin.
`dump_directory`	`VARCHA`	`BDUMI`	Non	Répertoire dans lequel le rapport ou le fichier d'exportation sera écrit. Si vous spécifiez un répertoire autre

Paramètre	Type de donnée	Par défaut	Obliga re	Description
				que le répertoire par défaut, l'utilisateur qui exécute les procédures `rdsadmin_diagnostic_util` doit disposer des autorisations en écriture pour le répertoire.
`p_tag`	VARCHA	—	Non	Chaîne pouvant être utilisée pour distinguer les sauvegardes afin d'indiquer leur but ou leur utilisation, telles que `incremental` ou `daily`. Vous pouvez spécifier jusqu'à 30 caractères. Les caractères valides sont `a-z`, `A-Z`, `0-9`, un trait de soulignement (_), un tiret (-), et un point (.). L'identif ication n'est pas sensible à la casse. RMANstocke toujours les étiquettes en majuscules, quelle que soit la majuscule utilisée lors de leur saisie. Les identifications n'ont pas besoin d'être uniques, de sorte que plusieurs sauvegardes peuvent avoir la même. Si vous ne spécifiez pas de balise, RMAN assigne automatiquement une balise par défaut en utilisant le formatTAG*YYYYMMDDTHHMMSS* , où *YYYY* c'est l'année, *MM* c'est le mois, *DD* c'est le jour, *HH* est l'heure (au format 24 heures), *MM* est le procès-verbal, et *SS* ce sont les secondes. La date et l'heure indiquent le RMAN début de la sauvegarde. Par exemple, une sauvegarde avec l'identification par défaut `TAG201909 27T214517` indique une sauvegarde qui a commencé le 27 septembre 2019 à 21:45:17. Le `p_tag` paramètre est pris en charge pour les versions suivantes RDS du moteur de base de données Oracle : • Oracle Database 21c (21.0.0) • Oracle Database 19c (19.0.0), avec 19.0.0.0.ru-2021-1 0.rur-2021-10.r1 et versions ultérieures

Paramètre	Type de donnée	Par défaut	Obliga re	Description
report_type	VARCHA	HTML	Non	Format du rapport. Les valeurs valides sont TEXT et HTML.
dbid	NUMBEF	—	Non	Identifiant de base de données valide (DBID) affiché dans la DBA_HIST_DATABASE_INSTANCE vue pour Oracle. Si ce paramètre n'est pas spécifié, RDS utilise le courant DBID affiché dans la V$DATABASE.DBID vue.

Vous utilisez généralement les paramètres suivants lors de la gestion ASH avec le package rdsadmin_diagnostic_util.

Paramètre	Type de données	Par défaut	Obligat re	Description
begin_time	DATE	—	Oui	Heure de début de l'ASHanalyse.
end_time	DATE	—	Oui	Heure de fin de l'ASHanalyse.
slot_width	NUMBER	0	Non	Durée des créneaux (en secondes) utilisés dans la section « Activité principale » du ASH rapport. Si ce paramètre n'est pas spécifié, l'intervalle de temps entre begin_time et end_time n'utilise pas plus de 10 emplacements.
sid	NUMBER	Null	Non	ID de session.
sql_id	VARCHAR2	Null	Non	La SQL pièce d'identité.
wait_class	VARCHAR2	Null	Non	Nom de la classe d'attente.
service_hash	NUMBER	Null	Non	Hachage du nom de service.

Paramètre	Type de données	Par défaut	Obligatoire	Description
module_name	VARCHAR2	Null	Non	Nom du module.
action_name	VARCHAR2	Null	Non	Nom de l'action.
client_id	VARCHAR2	Null	Non	ID spécifique à l'application de la session de base de données.
plsql_entry	VARCHAR2	Null	Non	Le point d'SQLentrée PL/.

Génération d'un AWR rapport

Pour générer un AWR rapport, utilisez la `rdsadmin.rdsadmin_diagnostic_util.awr_report` procédure suivante.

L'exemple suivant génère un AWR rapport pour la plage de clichés 101 à 106. Le fichier texte en sortie est nommé `awrrpt_101_106.txt`. Vous pouvez accéder à ce rapport à partir d AWS Management Console.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.awr_report(101,106,'TEXT');
```

L'exemple suivant génère un HTML rapport pour la plage de clichés 63 à 65. Le HTML fichier de sortie est nomméawrrpt_63_65.html. La procédure écrit le rapport dans un répertoire de base de données autre que le répertoire par défaut et nommé AWR_RPT_DUMP.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.awr_report(63,65,'HTML','AWR_RPT_DUMP');
```

Extraction de AWR données dans un fichier de vidage

Pour extraire AWR des données dans un fichier de vidage, `rdsadmin.rdsadmin_diagnostic_util.awr_extract` procédez comme suit.

L'exemple suivant extrait la plage d'instantanés comprise entre 101 et 106. Le fichier de vidage en sortie est nommé `awrextract_101_106.dmp`. Vous pouvez accéder à ce fichier via la console.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.awr_extract(101,106);
```

L'exemple suivant extrait la plage d'instantanés comprise entre 63 et 65. Le fichier de vidage en sortie est nommé `awrextract_63_65.dmp`. Le fichier est stocké dans un répertoire de base de données autre que le répertoire par défaut et nommé `AWR_RPT_DUMP`.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.awr_extract(63,65,'AWR_RPT_DUMP');
```

Génération d'un ADDM rapport

Pour générer un ADDM rapport, utilisez la `rdsadmin.rdsadmin_diagnostic_util.addm_report` procédure suivante.

L'exemple suivant génère un ADDM rapport pour la plage de clichés 101 à 106. Le fichier texte en sortie est nommé `addmrpt_101_106.txt`. Vous pouvez accéder au rapport via la console.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.addm_report(101,106);
```

L'exemple suivant génère un ADDM rapport pour la plage de clichés 63 à 65. Le fichier texte en sortie est nommé `addmrpt_63_65.txt`. Le fichier est stocké dans un répertoire de base de données autre que le répertoire par défaut et nommé `ADDM_RPT_DUMP`.

```
EXEC rdsadmin.rdsadmin_diagnostic_util.addm_report(63,65,'ADDM_RPT_DUMP');
```

Génération d'un ASH rapport

Pour générer un ASH rapport, utilisez la `rdsadmin.rdsadmin_diagnostic_util.ash_report` procédure suivante.

L'exemple suivant génère un ASH rapport qui inclut les données d'il y a 14 minutes jusqu'à l'heure actuelle. Le nom du fichier en sortie utilise le format `ashrpt`*`begin_time`*`end_time`.`txt`, où *`begin_time`* et *`end_time`* utilisent le format `YYYYMMDDHH24MISS`. Vous pouvez accéder au fichier via la console.

```
BEGIN
    rdsadmin.rdsadmin_diagnostic_util.ash_report(
        begin_time     =>     SYSDATE-14/1440,
        end_time       =>     SYSDATE,
        report_type    =>     'TEXT');
```

```
END;
/
```

L'exemple suivant génère un ASH rapport qui inclut les données du 18 novembre 2019 à 18 h 07 au 18 novembre 2019 à 18 h 15. Le nom du HTML rapport de sortie est `ashrpt_20190918180700_20190918181500.html`. Le rapport est stocké dans un répertoire de base de données autre que le répertoire par défaut et nommé `AWR_RPT_DUMP`.

```
BEGIN
    rdsadmin.rdsadmin_diagnostic_util.ash_report(
        begin_time     =>    TO_DATE('2019-09-18 18:07:00', 'YYYY-MM-DD HH24:MI:SS'),
        end_time       =>    TO_DATE('2019-09-18 18:15:00', 'YYYY-MM-DD HH24:MI:SS'),
        report_type    =>    'html',
        dump_directory =>    'AWR_RPT_DUMP');
END;
/
```

Accès aux AWR rapports depuis la console ou CLI

Pour accéder aux AWR rapports ou exporter des fichiers de vidage, vous pouvez utiliser le AWS Management Console ou AWS CLI. Pour de plus amples informations, veuillez consulter Téléchargement d'un fichier journal de base de données.

Ajustement des liens de base de données à utiliser avec des instances de base de données dans un VPC

Pour utiliser les liens de base de données Oracle avec des RDS instances de base de données Amazon situées dans le même cloud privé virtuel (VPC) ou pairVPCs, les deux instances de base de données doivent disposer d'un itinéraire valide entre elles. Vérifiez l'itinéraire valide entre les instances de base de données à l'aide de vos tables de VPC routage et de votre liste de contrôle d'accès réseau (ACL).

Le groupe de sécurité de chaque instance de base de données doit autoriser le trafic entrant dans l'autre instance de base de données et le trafic sortant de cette instance. Les règles entrantes et sortantes peuvent faire référence à des groupes de sécurité provenant du même groupe VPC ou d'un groupe pair. VPC Pour plus d'informations, consultez la section Mise à jour de vos groupes de sécurité pour faire référence à des groupes de VPC sécurité homologues.

Si vous avez configuré un DNS serveur personnalisé à l'aide des ensembles d'DHCPoptions de votre DNS serveur personnaliséVPC, celui-ci doit être en mesure de résoudre le nom de la cible du lien

de base de données. Pour de plus amples informations, veuillez consulter Configuration d'un DNS serveur personnalisé.

Pour plus d'informations sur l'utilisation des liens de base de données avec Oracle Data Pump, consultez Importation à l'aide d'Oracle Data Pump.

Définition de l'édition par défaut d'une instance de base de données

Vous pouvez redéfinir les objets de base de données dans un environnement privé appelé une édition. Vous pouvez utiliser la redéfinition basée sur l'édition pour mettre à niveau les objets de base de données d'une application avec un temps d'arrêt minimal.

Vous pouvez définir l'édition par défaut d'une instance de base de données Amazon RDS Oracle à l'aide de la RDS procédure Amazon`rdsadmin.rdsadmin_util.alter_default_edition`.

L'exemple suivant définit l'édition par défaut de l'instance de base de données Amazon RDS Oracle sur`RELEASE_V1`.

```
EXEC rdsadmin.rdsadmin_util.alter_default_edition('RELEASE_V1');
```

L'exemple suivant rétablit l'édition par défaut de l'instance de base de données Amazon RDS Oracle sur celle par défaut d'Oracle.

```
EXEC rdsadmin.rdsadmin_util.alter_default_edition('ORA$BASE');
```

Pour de plus amples informations concernant la redéfinition basée sur l'édition d'Oracle, veuillez consulter About Editions and Edition-Based Redefinition dans la documentation Oracle.

Activation de l'audit pour`SYS.AUD`table $

Pour activer l'audit sur la table de suivi des audits de la base de données`SYS.AUD$`, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_master_util.audit_all_sys_aud_table`. La seule propriété d'audit prise en charge est `ALL`. Vous ne pouvez pas auditer ou ne pas auditer des instructions ou des opérations individuelles.

L'activation de l'audit est prise en charge pour les instances de base de données Oracle qui exécutent les versions suivantes :

- Oracle Database 21c (21.0.0)

- Oracle Database 19c (19.0.0)

La procédure `audit_all_sys_aud_table` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_by_access`	booléen	true	Non	Définissez ce paramètre sur `true` pour auditer `BY ACCESS`. Définissez ce paramètre sur `false` pour auditer `BY SESSION`.

La requête suivante retourne la configuration d'audit actuelle de `SYS.AUD$` pour une base de données.

```
SELECT * FROM DBA_OBJ_AUDIT_OPTS WHERE OWNER='SYS' AND OBJECT_NAME='AUD$';
```

Les commandes suivantes activent l'audit de `ALL` sur `SYS.AUD$ BY ACCESS`.

```
EXEC rdsadmin.rdsadmin_master_util.audit_all_sys_aud_table;

EXEC rdsadmin.rdsadmin_master_util.audit_all_sys_aud_table(p_by_access => true);
```

La commande suivante active l'audit de `ALL` sur `SYS.AUD$ BY SESSION`.

```
EXEC rdsadmin.rdsadmin_master_util.audit_all_sys_aud_table(p_by_access => false);
```

Pour plus d'informations, voir AUDIT(audit traditionnel) dans la documentation Oracle.

Désactivation de l'audit pour. SYS AUDtable $

Pour désactiver l'audit sur la table de suivi des audits de la base de donnéesSYS.AUD$, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_master_util.noaudit_all_sys_aud_table`. Cette procédure ne prend aucun paramètre.

La requête suivante retourne la configuration d'audit actuelle pour SYS.AUD$, pour une base de données :

```
SELECT * FROM DBA_OBJ_AUDIT_OPTS WHERE OWNER='SYS' AND OBJECT_NAME='AUD$';
```

La commande suivante désactive l'audit de ALL sur SYS.AUD$.

```
EXEC rdsadmin.rdsadmin_master_util.noaudit_all_sys_aud_table;
```

Pour plus d'informations, voir NOAUDIT(audit traditionnel) dans la documentation Oracle.

Nettoyage de builds d'index en ligne interrompues

Pour corriger les erreurs de création d'index en ligne, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_dbms_repair.online_index_clean`.

La procédure `online_index_clean` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`object_id`	binary_integer	`ALL_INDEX_ID`	Non	ID d'objet de l'index. Généralement, vous pouvez utiliser l'ID d'objet indiqué dans le texte d'erreur ORA -08104.
`wait_for_lock`	binary_integer	`rdsadmin.rdsadmin_dbms_repair.lock_wait`	Non	Spécifiez `rdsadmin.rdsadmin_dbms_repair.lock_wait` , la valeur par défaut pour tenter de verrouiller l'objet sous-jacent et réessayer jusqu'à ce qu'une limite interne soit atteinte si le verrouillage échoue.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				Spécifiez `rdsadmin.rdsadmin_dbms_repair.lock_nowait` pour essayer d'obtenir un verrouillage sur l'objet sous-jacent, sans réessayer si le verouillage échoue.

L'exemple suivant nettoie une build d'index en ligne ayant échoué.

```
declare
  is_clean boolean;
begin
  is_clean := rdsadmin.rdsadmin_dbms_repair.online_index_clean(
    object_id     => 1234567890,
    wait_for_lock => rdsadmin.rdsadmin_dbms_repair.lock_nowait
  );
end;
/
```

Pour plus d'informations, voir la CLEANfonction ONLINE INDEX _ _ dans la documentation Oracle.

Ignorer les blocs corrompus

Pour ignorer les blocs corrompus pendant les analyses d'index et de table, utilisez le package `rdsadmin.rdsadmin_dbms_repair`.

Les procédures suivantes encapsulent la fonctionnalité de la procédure `sys.dbms_repair.admin_table` et ne prennent aucun paramètre :

- `rdsadmin.rdsadmin_dbms_repair.create_repair_table`
- `rdsadmin.rdsadmin_dbms_repair.create_orphan_keys_table`
- `rdsadmin.rdsadmin_dbms_repair.drop_repair_table`

- `rdsadmin.rdsadmin_dbms_repair.drop_orphan_keys_table`
- `rdsadmin.rdsadmin_dbms_repair.purge_repair_table`
- `rdsadmin.rdsadmin_dbms_repair.purge_orphan_keys_table`

Les procédures suivantes prennent les mêmes paramètres que leurs homologues du package `DBMS_REPAIR` pour les bases de données Oracle :

- `rdsadmin.rdsadmin_dbms_repair.check_object`
- `rdsadmin.rdsadmin_dbms_repair.dump_orphan_keys`
- `rdsadmin.rdsadmin_dbms_repair.fix_corrupt_blocks`
- `rdsadmin.rdsadmin_dbms_repair.rebuild_freelists`
- `rdsadmin.rdsadmin_dbms_repair.segment_fix_status`
- `rdsadmin.rdsadmin_dbms_repair.skip_corrupt_blocks`

Pour plus d'informations sur la gestion de la corruption de base de données, consultez DBMS_REPAIR dans la documentation Oracle.

Example Réponse aux blocs corrompus

Cet exemple présente le flux de travail de base pour répondre aux blocs corrompus. Vos étapes dépendront de l'emplacement et de la nature de votre corruption de bloc.

> **Important**
>
> Avant de tenter de réparer des blocs corrompus, consultez attentivement la REPAIR documentation DBMS_.

Pour ignorer les blocs corrompus pendant les analyses d'index et de table

1. Exécutez les procédures suivantes pour créer des tables de réparation si elles n'existent pas déjà.

```
EXEC rdsadmin.rdsadmin_dbms_repair.create_repair_table;
EXEC rdsadmin.rdsadmin_dbms_repair.create_orphan_keys_table;
```

2. Exécutez les procédures suivantes pour vérifier s'il existe des enregistrements et les purger si nécessaire.

```
SELECT COUNT(*) FROM SYS.REPAIR_TABLE;
SELECT COUNT(*) FROM SYS.ORPHAN_KEY_TABLE;
SELECT COUNT(*) FROM SYS.DBA_REPAIR_TABLE;
SELECT COUNT(*) FROM SYS.DBA_ORPHAN_KEY_TABLE;

EXEC rdsadmin.rdsadmin_dbms_repair.purge_repair_table;
EXEC rdsadmin.rdsadmin_dbms_repair.purge_orphan_keys_table;
```

3. Exécutez la procédure suivante pour rechercher les blocs corrompus.

```
SET SERVEROUTPUT ON
DECLARE v_num_corrupt INT;
BEGIN
  v_num_corrupt := 0;
  rdsadmin.rdsadmin_dbms_repair.check_object (
    schema_name => '&corruptionOwner',
    object_name => '&corruptionTable',
    corrupt_count =>  v_num_corrupt
  );
  dbms_output.put_line('number corrupt: '||to_char(v_num_corrupt));
END;
/

COL CORRUPT_DESCRIPTION FORMAT a30
COL REPAIR_DESCRIPTION FORMAT a30

SELECT OBJECT_NAME, BLOCK_ID, CORRUPT_TYPE, MARKED_CORRUPT,
       CORRUPT_DESCRIPTION, REPAIR_DESCRIPTION
FROM   SYS.REPAIR_TABLE;

SELECT SKIP_CORRUPT
FROM   DBA_TABLES
WHERE  OWNER = '&corruptionOwner'
AND    TABLE_NAME = '&corruptionTable';
```

4. Utilisez la procédure `skip_corrupt_blocks` pour activer ou désactiver l'ignorance de corruption pour les tables affectées. Selon la situation, vous devrez peut-être également extraire des données dans une nouvelle table, puis supprimer la table contenant le bloc corrompu.

   Exécutez la procédure suivante pour permettre d'ignorer la corruption pour les tables affectées.

```
begin
  rdsadmin.rdsadmin_dbms_repair.skip_corrupt_blocks (
    schema_name => '&corruptionOwner',
    object_name => '&corruptionTable',
    object_type => rdsadmin.rdsadmin_dbms_repair.table_object,
    flags => rdsadmin.rdsadmin_dbms_repair.skip_flag);
end;
/
select skip_corrupt from dba_tables where owner = '&corruptionOwner' and table_name
 = '&corruptionTable';
```

Exécutez la procédure suivante pour ne pas ignorer la corruption.

```
begin
  rdsadmin.rdsadmin_dbms_repair.skip_corrupt_blocks (
    schema_name => '&corruptionOwner',
    object_name => '&corruptionTable',
    object_type => rdsadmin.rdsadmin_dbms_repair.table_object,
    flags => rdsadmin.rdsadmin_dbms_repair.noskip_flag);
end;
/

select skip_corrupt from dba_tables where owner = '&corruptionOwner' and table_name
 = '&corruptionTable';
```

5. Une fois tous les travaux de réparation terminés, exécutez les procédures suivantes pour supprimer les tables de réparation.

```
EXEC rdsadmin.rdsadmin_dbms_repair.drop_repair_table;
EXEC rdsadmin.rdsadmin_dbms_repair.drop_orphan_keys_table;
```

Redimensionnement des espaces de table, des fichiers de données et des fichiers temporaires

Par défaut, les espaces de table Oracle sont créés avec l'option « auto extend » activée et sans aucune restriction de taille maximum. À cause de ces paramètres par défaut, les espaces de table peuvent parfois trop se développer. Nous vous recommandons de spécifier une taille maximum appropriée sur les espaces de table permanents et temporaires, et de surveiller attentivement l'utilisation de l'espace.

Redimensionnement des espaces de table permanents

Pour redimensionner un tablespace permanent dans une instance de base de données RDS pour Oracle, utilisez l'une des procédures Amazon suivantes : RDS

- `rdsadmin.rdsadmin_util.resize_datafile`
- `rdsadmin.rdsadmin_util.autoextend_datafile`

La procédure `resize_datafile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_data_file_id`	nombre	—	Oui	L'identifiant du fichier de données à redimensionner.
`p_size`	varchar2	—	Oui	La taille du fichier de données. Spécifiez la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go).

La procédure `autoextend_datafile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_data_file_id`	nombre	—	Oui	L'identifiant du fichier de données à redimensionner.
`p_autoextend_state`	varchar2	—	Oui	L'état de la fonction d'auto-extension. Spécifiez `ON` pour étendre automatiquement le

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				fichier de données et OFF pour désactiver l'extensi on automatique.
p_next	varchar2	—	Non	La taille de la prochaine incrémentation du fichier de données. Spécifiez la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go).
p_maxsize	varchar2	—	Non	L'espace disque maximal autorisé pour l'extension automatique. Spécifiez la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go). Vous pouvez spécifier UNLIMITED pour supprimer la limite de taille de fichier.

L'exemple suivant redimensionne le fichier de données 4 à 500 Mo.

```
EXEC rdsadmin.rdsadmin_util.resize_datafile(4,'500M');
```

L'exemple suivant désactive l'option d'auto-extension pour le fichier de données 4. Il active également l'extension automatique pour le fichier de données 5, avec une incrémentation de 128 Mo et aucune taille maximum.

```
EXEC rdsadmin.rdsadmin_util.autoextend_datafile(4,'OFF');
EXEC rdsadmin.rdsadmin_util.autoextend_datafile(5,'ON','128M','UNLIMITED');
```

Redimensionnement des espaces de table temporaires

Pour redimensionner un espace disque logique temporaire dans une instance de base RDS de données Oracle, y compris une réplique en lecture, utilisez l'une des procédures Amazon suivantes : RDS

- `rdsadmin.rdsadmin_util.resize_temp_tablespace`
- `rdsadmin.rdsadmin_util.resize_tempfile`
- `rdsadmin.rdsadmin_util.autoextend_tempfile`

La procédure `resize_temp_tablespace` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_temp_tablespace_name`	varchar2	—	Oui	Nom de l'espace de table temporaire à redimensionner.
`p_size`	varchar2	—	Oui	La taille de l'espace de table. Spécifiez la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go).

La procédure `resize_tempfile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_temp_file_id`	nombre	—	Oui	L'identifiant du fichier temporaire à redimensionner.
`p_size`	varchar2	—	Oui	La taille du fichier temporaire. Spécifiez

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go).

La procédure `autoextend_tempfile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_temp_file_id`	nombre	—	Oui	L'identifiant du fichier temporaire à redimensionner.
`p_autoextend_state`	varchar2	—	Oui	L'état de la fonction d'auto-extension. Spécifiez `ON` pour étendre automatiquement le fichier temporaire et `OFF` pour désactiver l'extension automatique.
`p_next`	varchar2	—	Non	La taille de la prochaine incrémentation du fichier temporaire. Spécifiez la taille en octets (par défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go).
`p_maxsize`	varchar2	—	Non	L'espace disque maximal autorisé pour l'extension automatique. Spécifiez la taille en octets (par

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				défaut), kilooctets (Ko), mégaoctets (Mo) ou gigaoctets (Go). Vous pouvez spécifier `UNLIMITED` pour supprimer la limite de taille de fichier.

Les exemples suivants redimensionnent un espace de table temporaire nommé `TEMP` pour qu'il fasse 4 Go.

```
EXEC rdsadmin.rdsadmin_util.resize_temp_tablespace('TEMP','4G');
```

```
EXEC rdsadmin.rdsadmin_util.resize_temp_tablespace('TEMP','4096000000');
```

L'exemple suivant redimensionne un espace de table temporaire basé sur le fichier temporaire avec l'identifiant de fichier 1 pour qu'il fasse 2 Mo.

```
EXEC rdsadmin.rdsadmin_util.resize_tempfile(1,'2M');
```

L'exemple suivant désactive l'option d'auto-extension pour le fichier temporaire 1. Il définit également la taille maximale d'extension automatique du fichier temporaire 2 à 10 Go, avec une incrémentation de 100 Mo.

```
EXEC rdsadmin.rdsadmin_util.autoextend_tempfile(1,'OFF');
EXEC rdsadmin.rdsadmin_util.autoextend_tempfile(2,'ON','100M','10G');
```

Pour plus d'informations sur les réplicas en lecture pour les instances de base de données Oracle, consultez Utilisation de réplicas en lecture pour Amazon RDS for Oracle.

Purge de la corbeille

Lorsque vous supprimez une table, votre base de données Oracle ne supprime pas immédiatement son espace de stockage. La base de données renomme la table et la place, ainsi que les objets

associés, dans une corbeille. La purge de la corbeille supprime ces éléments et libère leur espace de stockage.

Pour purger l'intégralité de la corbeille, suivez la RDS procédure Amazon`rdsadmin.rdsadmin_util.purge_dba_recyclebin`. Toutefois, cette procédure ne peut pas purger la corbeille des objets `SYS` et `RDSADMIN`. Si vous devez purger ces objets, contactez AWS Support.

L'exemple suivant purge l'ensemble de la corbeille.

```
EXEC rdsadmin.rdsadmin_util.purge_dba_recyclebin;
```

Définition des valeurs affichées par défaut pour une édition complète

Pour modifier les valeurs affichées par défaut pour une rédaction complète sur votre instance Amazon RDS Oracle, utilisez la RDS procédure `rdsadmin.rdsadmin_util.dbms_redact_upd_full_rdct_val` Amazon. Notez que vous créez une politique de rédaction avec le SQL package `DBMS_REDACT` PL/, comme expliqué dans la documentation de la base de données Oracle. La procédure `dbms_redact_upd_full_rdct_val` spécifie les caractères à afficher pour les différents types de données affectés par une politique existante.

La procédure `dbms_redact_upd_full_rdct_val` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_number_val`	nombre	Null	Non	Modifie la valeur par défaut des colonnes de type de données `NUMBER`.
`p_binfloat_val`	binary_float	Null	Non	Modifie la valeur par défaut des colonnes de type de données `BINARY_FLOAT` .
`p_bindouble_val`	binary_double	Null	Non	Modifie la valeur par défaut des

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				colonnes de type de données BINARY_DOUBLE .
p_char_val	char	Null	Non	Modifie la valeur par défaut des colonnes de type de données CHAR.
p_varchar_val	varchar2	Null	Non	Modifie la valeur par défaut des colonnes de type de données VARCHAR2.
p_nchar_val	nchar	Null	Non	Modifie la valeur par défaut des colonnes de type de données NCHAR.
p_nvarchar_val	nvarchar2	Null	Non	Modifie la valeur par défaut des colonnes de type de données NVARCHAR2 .
p_date_val	date	Null	Non	Modifie la valeur par défaut des colonnes de type de données DATE.
p_ts_val	timestamp	Null	Non	Modifie la valeur par défaut des colonnes de type de données TIMESTAMP .

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_tswtz_val	timestamp with time zone	Null	Non	Modifie la valeur par défaut des colonnes de type de données TIMESTAMP WITH TIME ZONE.
p_blob_val	blob	Null	Non	Modifie la valeur par défaut des colonnes de type de données BLOB.
p_clob_val	clob	Null	Non	Modifie la valeur par défaut des colonnes de type de données CLOB.
p_nclob_val	nclob	Null	Non	Modifie la valeur par défaut des colonnes de type de données NCLOB.

L'exemple suivant remplace la valeur expurgée par défaut par * pour le type de données CHAR :

```
EXEC rdsadmin.rdsadmin_util.dbms_redact_upd_full_rdct_val(p_char_val => '*');
```

L'exemple suivant modifie les valeurs expurgées par défaut pour les types de données NUMBER, DATE et CHAR :

```
BEGIN
rdsadmin.rdsadmin_util.dbms_redact_upd_full_rdct_val(
    p_number_val=>1,
    p_date_val=>to_date('1900-01-01','YYYY-MM-DD'),
    p_varchar_val=>'X');
END;
/
```

Après avoir modifié les valeurs par défaut pour l'édition complète avec la procédure dbms_redact_upd_full_rdct_val, redémarrez votre instance de base de données

pour que la modification prenne effet. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Exécution des tâches courantes liées au journal pour les instances de base de données Oracle

Vous trouverez ci-dessous des informations sur la façon d'effectuer certaines tâches DBA courantes liées à la journalisation sur vos instances de base de données Amazon RDS exécutant Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données et limite l'accès à certaines tables et procédures système qui requièrent des privilèges avancés.

Pour plus d'informations, consultez Fichiers journaux RDS de base de données Amazon pour Oracle.

Rubriques

- Configuration du mode FORCE LOGGING
- Configuration d'une journalisation supplémentaire
- Changement de fichiers journaux en ligne
- Ajout de journaux redo en ligne
- Suppression de journaux redo en ligne
- Redimensionnement de journaux redo en ligne
- Conservation des journaux redo archivés
- Accès aux journaux de reprise en ligne et archivés
- Téléchargement des journaux de reprise archivés à partir d'Amazon S3

Configuration du mode FORCE LOGGING

En mode FORCE LOGGING, Oracle enregistre toutes les modifications apportées à la base de données, à l'exception de celles apportées aux espaces de table temporaires et aux segments temporaires (`NOLOGGING` des clauses sont ignorées). Pour de plus amples informations, veuillez consulter Specifying FORCE LOGGING Mode dans la documentation Oracle.

Pour définir le mode FORCE LOGGING, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.force_logging`. La procédure `force_logging` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Oui	Description
p_enable	booléen	true	Non	Définissez ce paramètre sur true pour mettre la base de données en mode FORCE LOGGING ou sur false pour sortir la base de données de ce mode.

L'exemple suivant met la base de données en mode FORCE LOGGING.

```
EXEC rdsadmin.rdsadmin_util.force_logging(p_enable => true);
```

Configuration d'une journalisation supplémentaire

Si vous activez la journalisation supplémentaire, LogMiner dispose des informations nécessaires pour prendre en charge les lignes chaînées et les tables en cluster. Pour de plus amples informations, veuillez consulter journalisation supplémentaire dans la documentation Oracle.

Oracle Database n'active pas la journalisation supplémentaire par défaut. Pour activer et désactiver la journalisation supplémentaire, utilisez la procédure Amazon RDS rdsadmin.rdsadmin_util.alter_supplemental_logging. Pour plus d'informations sur la façon dont Amazon RDS gère la conservation des journaux redo archivés pour les instances de base de données Oracle, consultez Conservation des journaux redo archivés.

La procédure alter_supplemental_logging possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_action	varchar2	—	Oui	'ADD' pour ajouter la journalisation supplémentaire, 'DROP' pour la supprimer.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_type	varchar2	null	Non	Type de journalisation supplémentaire. Les valeurs valides sont 'ALL', 'FOREIGN KEY', 'PRIMARY KEY', 'UNIQUE' et PROCEDURAL .

L'exemple suivant active la journalisation supplémentaire.

```
begin
    rdsadmin.rdsadmin_util.alter_supplemental_logging(
        p_action => 'ADD');
end;
/
```

L'exemple suivant active la journalisation supplémentaire pour toutes les colonnes de taille maximale et de longueur fixe.

```
begin
    rdsadmin.rdsadmin_util.alter_supplemental_logging(
        p_action => 'ADD',
        p_type   => 'ALL');
end;
/
```

L'exemple suivant active la journalisation supplémentaire pour les colonnes de clés primaires.

```
begin
    rdsadmin.rdsadmin_util.alter_supplemental_logging(
        p_action => 'ADD',
        p_type   => 'PRIMARY KEY');
end;
/
```

Changement de fichiers journaux en ligne

Pour changer des fichiers journaux, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.switch_logfile`. La procédure `switch_logfile` ne comporte aucun paramètre.

L'exemple suivant change des fichiers journaux.

```
EXEC rdsadmin.rdsadmin_util.switch_logfile;
```

Ajout de journaux redo en ligne

Une instance de base de données Amazon RDS exécutant Oracle démarre avec quatre journaux redo en ligne de 128 Mo chacun. Pour ajouter des journaux redo supplémentaires, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.add_logfile`.

La procédure `add_logfile` possède les paramètres suivants.

> **Note**
>
> Les paramètres s'excluent mutuellement.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`bytes`	positives	null	Non	Taille du fichier journal en octets. Utilisez ce paramètre uniquement si la taille du journal est inférieure à 2147483648 octets (2 GiB). Dans le cas contraire, RDS émet une erreur. Pour les tailles de log supérieures à cette valeur d'octet,

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				utilisez plutôt le `p_size` paramètre.
`p_size`	varchar2	—	Oui	Taille du fichier journal en kilo-octets (K), mégaoctets (M) ou gigaoctets (G).

La commande suivante ajoute un fichier journal de 100 Mo.

```
EXEC rdsadmin.rdsadmin_util.add_logfile(p_size => '100M');
```

Suppression de journaux redo en ligne

Pour supprimer des journaux redo, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.drop_logfile`. La procédure `drop_logfile` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`grp`	positives	—	Oui	Numéro de groupe du journal.

L'exemple suivant supprime le journal doté du numéro de groupe 3.

```
EXEC rdsadmin.rdsadmin_util.drop_logfile(grp => 3);
```

Vous pouvez uniquement supprimer des journaux dont le statut est inutilisé ou inactif. L'exemple suivant permet d'obtenir les statuts des journaux.

```
SELECT GROUP#, STATUS FROM V$LOG;

GROUP#     STATUS
---------- ----------------
1          CURRENT
```

```
2          INACTIVE
3          INACTIVE
4          UNUSED
```

Redimensionnement de journaux redo en ligne

Une RDS instance de base de données Amazon exécutant Oracle commence par quatre journaux de restauration en ligne de 128 Mo chacun. L'exemple suivant montre comment vous pouvez utiliser RDS les procédures Amazon pour redimensionner vos journaux de 128 Mo chacun à 512 Mo chacun.

```
/* Query V$LOG to see the logs.          */
/* You start with 4 logs of 128 MB each. */

SELECT GROUP#, BYTES, STATUS FROM V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
1          134217728  INACTIVE
2          134217728  CURRENT
3          134217728  INACTIVE
4          134217728  INACTIVE


/* Add four new logs that are each 512 MB */

EXEC rdsadmin.rdsadmin_util.add_logfile(bytes => 536870912);
EXEC rdsadmin.rdsadmin_util.add_logfile(bytes => 536870912);
EXEC rdsadmin.rdsadmin_util.add_logfile(bytes => 536870912);
EXEC rdsadmin.rdsadmin_util.add_logfile(bytes => 536870912);


/* Query V$LOG to see the logs. */
/* Now there are 8 logs.        */

SELECT GROUP#, BYTES, STATUS FROM V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
1          134217728  INACTIVE
2          134217728  CURRENT
3          134217728  INACTIVE
4          134217728  INACTIVE
```

```
5          536870912  UNUSED
6          536870912  UNUSED
7          536870912  UNUSED
8          536870912  UNUSED


/* Drop each inactive log using the group number. */

EXEC rdsadmin.rdsadmin_util.drop_logfile(grp => 1);
EXEC rdsadmin.rdsadmin_util.drop_logfile(grp => 3);
EXEC rdsadmin.rdsadmin_util.drop_logfile(grp => 4);


/* Query V$LOG to see the logs. */
/* Now there are 5 logs.        */

select GROUP#, BYTES, STATUS from V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
2          134217728  CURRENT
5          536870912  UNUSED
6          536870912  UNUSED
7          536870912  UNUSED
8          536870912  UNUSED


/* Switch logs so that group 2 is no longer current. */

EXEC rdsadmin.rdsadmin_util.switch_logfile;


/* Query V$LOG to see the logs.        */
/* Now one of the new logs is current. */

SQL>SELECT GROUP#, BYTES, STATUS FROM V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
2          134217728  ACTIVE
5          536870912  CURRENT
6          536870912  UNUSED
7          536870912  UNUSED
8          536870912  UNUSED
```

```
/* If the status of log 2 is still "ACTIVE", issue a checkpoint to clear it to
 "INACTIVE".  */

EXEC rdsadmin.rdsadmin_util.checkpoint;


/* Query V$LOG to see the logs.            */
/* Now the final original log is inactive. */

select GROUP#, BYTES, STATUS from V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
2          134217728  INACTIVE
5          536870912  CURRENT
6          536870912  UNUSED
7          536870912  UNUSED
8          536870912  UNUSED


# Drop the final inactive log.

EXEC rdsadmin.rdsadmin_util.drop_logfile(grp => 2);


/* Query V$LOG to see the logs.    */
/* Now there are four 512 MB logs. */

SELECT GROUP#, BYTES, STATUS FROM V$LOG;

GROUP#     BYTES      STATUS
---------- ---------- ----------------
5          536870912  CURRENT
6          536870912  UNUSED
7          536870912  UNUSED
8          536870912  UNUSED
```

Conservation des journaux redo archivés

Vous pouvez conserver les journaux de restauration archivés localement sur votre instance de base de données pour les utiliser avec des produits tels qu'Oracle LogMiner (`DBMS_LOGMNR`). Une fois

que vous avez conservé les journaux redo, vous pouvez les utiliser LogMiner pour analyser les journaux. Pour plus d'informations, consultez la section Utilisation LogMiner pour analyser les fichiers de journalisation dans la documentation Oracle.

Pour conserver les journaux redo archivés, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.set_configuration`. Si vous utilisez cette procédure sur une instance principale dans Oracle Data Guard, RDS modifie le paramètre de conservation du journal d'archivage sur l'instance principale et sur les répliques en lecture ouverte, mais pas sur les répliques montées. RDS conserve les derniers journaux de restauration des archives sur les répliques montées pendant une courte période. RDS supprime automatiquement les anciens journaux téléchargés sur les répliques montées.

La procédure `set_configuration` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`name`	varchar	—	Oui	Nom de la configuration à mettre à jour. Pour modifier les heures de conservation du journal redo archivé, définissez le nom sur `archivelog retention hours`.
`value`	varchar	—	Oui	Valeur pour la configuration. Définissez comme valeur le nombre d'heures pendant lesquelles les journaux sont conservés.

L'exemple suivant conserve les journaux redo pendant 24 heures.

```
begin
    rdsadmin.rdsadmin_util.set_configuration(
        name  => 'archivelog retention hours',
        value => '24');
end;
```

```
/
commit;
```

> **Note**
>
> La validation est obligatoire pour que la modification prenne effet.

Pour voir combien de temps les journaux redo archivés sont conservés pour votre instance de base de données, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.show_configuration`.

L'exemple suivant affiche la durée de conservation des journaux.

```
set serveroutput on
EXEC rdsadmin.rdsadmin_util.show_configuration;
```

La sortie affiche le paramètre actuel pour `archivelog retention hours`. La sortie suivante montre que les journaux redo archivés sont conservés pendant 48 heures.

```
NAME:archivelog retention hours
VALUE:48
DESCRIPTION:ArchiveLog expiration specifies the duration in hours before archive/redo
 log files are automatically deleted.
```

Étant donné que les journaux redo archivés sont conservés sur votre instance de base de données, vérifiez que votre instance de base de données dispose d'un stockage alloué suffisant pour les journaux conservés. Pour déterminer la quantité d'espace que votre instance de base de données a utilisée au cours des X dernières heures, vous pouvez exécuter la requête suivante en remplaçant X par le nombre d'heures.

```
SELECT SUM(BLOCKS * BLOCK_SIZE) bytes
  FROM V$ARCHIVED_LOG
 WHERE FIRST_TIME >= SYSDATE-(X/24) AND DEST_ID=1;
```

RDS for Oracle ne génère des journaux de reprise archivés que si la période de rétention des sauvegardes de votre instance de base de données est supérieure à zéro. Par défaut, la période de rétention des sauvegardes est supérieure à zéro.

Lorsque la période de rétention des journaux archivés expire, RDS for Oracle supprime les journaux de reprise archivés de votre instance de base de données. Pour prendre en charge la restauration de votre instance de base de données à un moment donné, Amazon RDS conserve les journaux de reprise archivés en dehors de votre instance de base de données pendant la période de rétention des sauvegardes. Pour modifier la période de rétention des sauvegardes pour votre instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

> **Note**
>
> Dans certains cas, vous pouvez utiliser JDBC sur Linux pour télécharger les journaux redo archivés et connaître des temps de latence élevés et des réinitalisations de connexion. Dans ces cas, les problèmes peuvent être causés par le paramétrage du générateur de nombres aléatoires sur votre client Java. Nous vous recommandons de définir vos pilotes JDBC pour l'utilisation d'un générateur de nombres aléatoires sans blocage.

Accès aux journaux de reprise en ligne et archivés

Vous souhaiterez peut-être accéder à vos fichiers de journalisation en ligne et archivés pour le minage à l'aide d'outils externes tels que GoldenGate Attunity, Informatica, etc. Pour accéder à ces fichiers, procédez comme suit :

1. Créez des objets de répertoire qui donnent un accès en lecture seule aux chemins d'accès de fichiers physiques.

   Utilisation de `rdsadmin.rdsadmin_master_util.create_archivelog_dir` et `rdsadmin.rdsadmin_master_util.create_onlinelog_dir`.
2. Lisez les fichiers à l'aide de PL/SQL.

   Vous pouvez lire les fichiers en utilisant PL/SQL. Pour de plus amples informations sur la lecture de fichiers à partir d'objets de répertoire, veuillez consulter Établissement de la liste des fichiers situés dans un répertoire d'instance de base de données et Lecture de fichiers dans un répertoire d'instance de base de données.

L'accès aux journaux des transactions est pris en charge pour les versions suivantes :

- Oracle Database 21c
- Oracle Database 19c

Le code suivant crée des répertoires qui fournissent un accès en lecture seule à vos fichiers de journalisation Redo en ligne et archivés :

> **Important**
>
> Ce code retire également le privilège `DROP ANY DIRECTORY`.

```
EXEC rdsadmin.rdsadmin_master_util.create_archivelog_dir;
EXEC rdsadmin.rdsadmin_master_util.create_onlinelog_dir;
```

Le code suivant supprime les répertoires pour vos fichiers journaux redo en ligne et archivés.

```
EXEC rdsadmin.rdsadmin_master_util.drop_archivelog_dir;
EXEC rdsadmin.rdsadmin_master_util.drop_onlinelog_dir;
```

Le code suivant accorde et révoque le privilège `DROP ANY DIRECTORY`.

```
EXEC rdsadmin.rdsadmin_master_util.revoke_drop_any_directory;
EXEC rdsadmin.rdsadmin_master_util.grant_drop_any_directory;
```

Téléchargement des journaux de reprise archivés à partir d'Amazon S3

Vous pouvez télécharger les journaux de reprise archivés sur votre instance de base de données à l'aide du package `rdsadmin.rdsadmin_archive_log_download`. Si les journaux de reprise archivés ne sont plus sur votre instance de base de données, vous pouvez les télécharger à nouveau à partir d'Amazon S3. Ensuite, vous pouvez les exploiter ou les utiliser pour récupérer ou répliquer votre base de données.

> **Note**
>
> Vous ne pouvez pas télécharger des Journaux de reprise archivés sur des instances de réplica en lecture.

Téléchargement des journaux de reprise archivés : étapes de base

La disponibilité de vos journaux de reprise archivés dépend des politiques de rétention suivantes :

- Politique de conservation des sauvegardes : les journaux liés à cette politique sont disponibles dans Amazon S3. Les journaux étrangers à cette politique sont supprimés.
- Politique de conservation des journaux archivés : les journaux liés à cette politique sont disponibles sur votre instance de base de données. Les journaux étrangers à cette politique sont supprimés.

Si les journaux ne figurent pas sur votre instance mais sont protégés par votre période de rétention des sauvegardes, utilisez `rdsadmin.rdsadmin_archive_log_download` pour les télécharger à nouveau. RDSfor Oracle enregistre les journaux dans le `/rdsdbdata/log/arch` répertoire de votre instance de base de données.

Pour télécharger des journaux de reprise archivés à partir d'Amazon S3

1. Configurez votre période de conservation pour vous assurer que les journaux redo archivés que vous avez téléchargés sont conservés pendant la durée où vous en avez besoin. Veillez à valider (`COMMIT`) votre changement.

   RDSconserve vos journaux téléchargés conformément à la politique de conservation des journaux archivés, à compter du moment où les journaux ont été téléchargés. Pour découvrir comment définir la politique de rétention, consultez Conservation des journaux redo archivés.
2. Attendez jusqu'à 5 minutes pour que la modification de la politique de rétention des journaux archivés prenne effet.
3. Téléchargez les journaux de reprise archivés à partir d'Amazon S3 à l'aide de `rdsadmin.rdsadmin_archive_log_download`.

   Pour plus d'informations, consultez Téléchargement d'un journal de reprise archivé unique et Téléchargement d'une série de journaux de reprise archivés.

   > **Note**
   >
   > RDSvérifie automatiquement l'espace de stockage disponible avant le téléchargement. Si les journaux demandés consomment un pourcentage élevé d'espace, vous recevez une alerte.

4. Vérifiez que les journaux ont bien été téléchargés à partir d'Amazon S3.

   Vous pouvez consulter l'état de votre tâche de téléchargement dans un fichier bdump. Les fichiers bdump ont le nom des chemin d'accès `/rdsdbdata/log/trace/dbtask-`*`task-id`*`.log`. A l'étape de téléchargement précédente, vous avez exécuté une instruction `SELECT` qui

renvoie l'ID de tâche dans un type de données VARCHAR2. Pour plus d'informations, consultez des exemples similaires dans Surveillance du statut d'un transfert de fichiers.

Téléchargement d'un journal de reprise archivé unique

Pour télécharger un journal de reprise archivé unique dans le répertoire /rdsdbdata/log/arch, utilisez rdsadmin.rdsadmin_archive_log_download.download_log_with_seqnum. Cette procédure utilise le paramétrage suivant.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
seqnum	nombre	—	Oui	Numéro de séquence du journal de reprise archivé.

L'exemple suivant télécharge le journal avec le numéro de séquence 20.

```
SELECT rdsadmin.rdsadmin_archive_log_download.download_log_with_seqnum(seqnum => 20)
       AS TASK_ID
FROM   DUAL;
```

Téléchargement d'une série de journaux de reprise archivés

Pour télécharger une série de journaux de reprise archivés dans le répertoire /rdsdbdata/log/arch, utilisez download_logs_in_seqnum_range. Votre téléchargement est limité à 300 journaux par requête. La procédure download_logs_in_seqnum_range possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
start_seq	nombre	—	Oui	Numéro de séquence initial de la série.
end_seq	nombre	—	Oui	Numéro de séquence final de la série.

L'exemple suivant télécharge les journaux portant les numéros de séquence 50 à 100.

```
SELECT rdsadmin.rdsadmin_archive_log_download.download_logs_in_seqnum_range(start_seq
 => 50, end_seq => 100)
       AS TASK_ID
FROM   DUAL;
```

Exécution de RMAN tâches courantes pour les instances de base de données Oracle

Dans la section suivante, vous découvrirez comment effectuer des DBA tâches Oracle Recovery Manager (RMAN) sur vos RDS instances de base de données Amazon exécutant Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui requièrent des privilèges avancés.

Utilisez le RDS package Amazon `rdsadmin.rdsadmin_rman_util` pour effectuer des RMAN sauvegardes de votre base de données Amazon RDS pour Oracle sur disque. Le package `rdsadmin.rdsadmin_rman_util` prend en charge les sauvegardes de fichiers de base de données complètes et incrémentielles, les sauvegardes d'espace de table et les sauvegardes des journaux redo archivés.

Une fois la RMAN sauvegarde terminée, vous pouvez copier les fichiers de sauvegarde depuis l'hôte de l'instance de base de données Amazon RDS pour Oracle. Vous pouvez le faire pour effectuer une restauration sur un site non RDS hôte ou pour le stockage à long terme des sauvegardes. Par exemple, vous pouvez copier les fichiers de sauvegarde dans un compartiment Amazon S3. Pour de plus amples informations, reportez-vous à l'utilisation d'Intégration Amazon S3.

Les fichiers de sauvegarde pour les RMAN sauvegardes restent sur l'hôte de l'instance Amazon RDS DB jusqu'à ce que vous les supprimiez manuellement. Vous pouvez utiliser la procédure Oracle `UTL_FILE.FREMOVE` pour supprimer les fichiers d'un répertoire. Pour plus d'informations, consultez FREMOVEla procédure dans la documentation de la base de données Oracle.

Vous ne pouvez pas utiliser le RDS pour RMAN restaurer des instances de base de données Oracle. Toutefois, vous pouvez l'utiliser RMAN pour restaurer une sauvegarde sur une EC2 instance sur site ou Amazon. Pour plus d'informations, consultez l'article de blog Restaurer une instance Amazon RDS pour Oracle vers une instance autogérée.

> **Note**
>
> Pour la sauvegarde et la restauration sur une autre instance de base de données Amazon RDS pour Oracle, vous pouvez continuer à utiliser les fonctionnalités de RDS sauvegarde et de restauration d'Amazon. Pour de plus amples informations, veuillez consulter Sauvegarde, restauration et exportation de données.

Rubriques

- Conditions préalables pour les sauvegardes RMAN
- Paramètres communs pour les RMAN procédures
- Validation des fichiers de base de données dans RDS Oracle
- Activation et désactivation du suivi des modifications de bloc
- Recoupement des journaux redo archivés
- Sauvegarde des fichiers de journalisation archivés
- Réalisation d'une sauvegarde complète de base de données
- Réalisation d'une sauvegarde complète d'une base de données locataire
- Réalisation d'une sauvegarde incrémentielle de base de données
- Réalisation d'une sauvegarde incrémentielle d'une base de données locataire
- Sauvegarde d'un espace de table
- Sauvegarde d'un fichier de contrôle
- Exécution de la restauration de blocs multimédias

Conditions préalables pour les sauvegardes RMAN

Avant de sauvegarder votre base de données à l'aide du package `rdsadmin.rdsadmin_rman_util`, assurez-vous que vous répondez aux prérequis suivants :

- Assurez-vous que votre base de données RDS for Oracle est en `ARCHIVELOG` mode. Pour activer ce mode, définissez la période de conservation des sauvegardes sur une valeur différente de zéro.
- Lorsque vous sauvegardez les journaux redo archivés ou effectuez une sauvegarde complète ou incrémentielle incluant des journaux redo archivés, et lorsque vous effectuez la sauvegarde de la base de données, veillez à ce que la conservation des journaux redo soit définie sur une valeur

non nulle. Les journaux redo archivés sont nécessaires pour assurer la cohérence des fichiers de base de données pendant la restauration. Pour de plus amples informations, veuillez consulter Conservation des journaux redo archivés.

- Assurez-vous que votre instance de base de données dispose de suffisamment d'espace disponible pour stocker les sauvegardes. Lorsque vous sauvegardez votre base de données, vous spécifiez un objet de répertoire Oracle en tant que paramètre dans l'appel de procédure. RMANplace les fichiers dans le répertoire spécifié. Vous pouvez utiliser les répertoires par défaut, tels que `DATA_PUMP_DIR`, ou créer un répertoire. Pour de plus amples informations, veuillez consulter Création et suppression de répertoires dans l'espace de stockage de données principal.

  Vous pouvez surveiller l'espace libre actuel dans une instance Oracle RDS à l'aide de la CloudWatch métrique`FreeStorageSpace`. Nous recommandons que votre espace libre dépasse la taille actuelle de la base de données, mais il RMAN sauvegarde uniquement les blocs formatés et supporte la compression.

Paramètres communs pour les RMAN procédures

Vous pouvez utiliser les procédures du RDS package Amazon `rdsadmin.rdsadmin_rman_util` pour effectuer des tâches avecRMAN. Plusieurs paramètres sont communs aux procédures figurant dans le package. Le package possède les paramètres communs suivants.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
`p_directory_name`	varchar	Nom de répertoire de base de données valide.	—	Oui	Nom du répertoire devant contenir les fichiers de sauvegarde.
`p_label`	varchar	`a-z`, `A-Z`, `0-9`, `'_'`, `'-'`, `'.'`	—	Non	Chaîne unique incluse dans les noms de fichiers de sauvegarde.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
					Note La limite est de 30 caractères.
`p_owner`	varchar	Propriétaire valide du répertoire spécifié dans `p_directory_name` .	—	Oui	Propriétaire du répertoire devant contenir les fichiers de sauvegarde.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
p_tag	varchar	a-z, A-Z, 0-9, '_', '-', '.'	NULL	Non	Chaîne pouvant être utilisée pour distinguer les sauvegardes afin d'indiquer leur but ou leur utilisation, telles que les sauvegardes journalière, hebdomadaire, ou de niveau progressif. La limite est de 30 caractères. L'identification n'est pas sensible à la casse. Les balises sont toujours enregistrées en majuscules, quelle que soit la casse utilisée lors de leur saisie. Les identifications n'ont pas besoin d'être uniques, de sorte que plusieurs sauvegardes peuvent avoir la même. Si vous ne spécifiez pas de balise, RMAN assigne automatiquement une balise par défaut en utilisant le formatTAG*YYYYMMDDTHHMMSS* , où *YYYY* c'est l'année, *MM* c'est le mois, *DD* c'est le jour *HH* est l'heure (au format 24 heures), *MM* est le procès-verbal, et *SS* ce sont les secondes. La date et l'heure font référence au moment où la sauvegarde a été RMAN lancée. Par exemple, une sauvegarde peut se voir attribuer une balise TAG20190927T214517 , pour une sauvegarde démarrée le 27 septembre 2019 à 21:45:17.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
					Le `p_tag` paramètre est pris en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle : • Oracle Database 21c (21.0.0) • Oracle Database 19c (19.0.0), avec 19.0.0.0.ru-2021-10.rur-2021-10.r1 ou versions ultérieures
`p_compress`	booléer	`TRUE`, `FALSE`	`FALSE`	Non	Spécifiez `TRUE` pour activer la compression des BASIC sauvegardes. Spécifiez `FALSE` pour désactiver la compression des BASIC sauvegardes.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
p_include_archive_logs	booléen	TRUE, FALSE	FALSE	Non	Spécifiez TRUE pour inclure les journaux redo archivés dans la sauvegarde. Spécifiez FALSE pour exclure les journaux redo archivés de la sauvegarde. Si vous incluez les journaux redo archivés dans la sauvegarde, définissez la conservation sur une heure ou plus à l'aide de la procédure rdsadmin.rdsadmin_util.set_configuration . De plus, appelez la procédure rdsadmin.rdsadmin_rman_util.crosscheck_archivelog immédiatement avant d'exécuter la sauvegarde. Dans le cas contraire, la sauvegarde risque d'échouer en raison de l'absence de fichiers de journalisation archivés qui ont été supprimés par les procédures de RDS gestion d'Amazon.
p_include_controlfile	booléen	TRUE, FALSE	FALSE	Non	Spécifiez TRUE pour inclure le fichier de contrôle dans la sauvegarde. Spécifiez FALSE pour exclure le fichier de contrôle de la sauvegarde.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
`p_optimize`	booléen	`TRUE`, `FALSE`	`TRUE`	Non	Spécifiez `TRUE` pour activer l'optimisation de la sauvegarde, si des journaux redo archivés sont inclus, afin de réduire la taille de la sauvegarde. Spécifiez `FALSE` pour désactiver l'optimisation de la sauvegarde.
`p_parallel`	nombre	Entier valide compris entre 1 et 254 pour Oracle Database Enterprise Edition (EE) 1 pour d'autres éditions d'Oracle Database	1	Non	Nombre de canaux.

Nom du paramètre	Type de donnée	Valeurs valides	Par défaut	Obligatoire	Description
p_rman_to_dbms_output	booléen	TRUE, FALSE	FALSE	Non	LorsqueTRUE, la RMAN sortie est envoyée au DBMS_OUTPUT package en plus d'un fichier dans le BDUMP répertoire. Dans SQL *Plus, utilisez SET SERVEROUTPUT ON pour voir le résultat. LorsqueFALSE, la RMAN sortie n'est envoyée qu'à un fichier du BDUMP répertoire.
p_section_size_mb	nombre	Entier valide	NULL	Non	Taille de la section en mégaoctets (Mo). Valide en parallèle en divisant chaque fichier dans la taille de section spécifiée . Lorsque la valeur est NULL, le paramètre est ignoré.
p_validation_type	varchar	'PHYSICAL' , 'PHYSICAL+LOGICAL'	'PHYS'	Non	Niveau de détection de la corruption. Spécifiez 'PHYSICAL' pour rechercher de la corruption physique. Par exemple, la corruption physique peut être un bloc dont l'en-tête et le pied de page ne correspondent pas. Spécifiez 'PHYSICAL+LOGICAL' pour rechercher les incohérences logiques en plus de la corruption physique. Un bloc corrompu est un exemple de corruption logique.

Validation des fichiers de base de données dans RDS Oracle

Vous pouvez utiliser le RDS package Amazon `rdsadmin.rdsadmin_rman_util` pour valider les fichiers de base de données Amazon RDS pour Oracle, tels que les fichiers de données, les tablespaces, les fichiers de contrôle et les fichiers de paramètres du serveur ()SPFILEs.

Pour plus d'informations sur RMAN la validation, voir Validation des fichiers de base de données et des sauvegardes et VALIDATEdans la documentation Oracle.

Rubriques

- Validation d'une base de données
- Validation d'une base de données locataire
- Validation d'un espace de table
- Validation d'un fichier de contrôle
- Validation d'un SPFILE
- Validation d'un fichier de données Oracle

Validation d'une base de données

Pour valider tous les fichiers pertinents utilisés par une base de données Oracle dans RDS for Oracle, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.validate_database`.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_validation_type`
- `p_parallel`
- `p_section_size_mb`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

L'exemple suivant valide la base de données à l'aide des valeurs par défaut des paramètres.

```
EXEC rdsadmin.rdsadmin_rman_util.validate_database;
```

L'exemple suivant valide la base de données à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.validate_database(
        p_validation_type     => 'PHYSICAL+LOGICAL',
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_rman_to_dbms_output => FALSE);
END;
/
```

Lorsque le `p_rman_to_dbms_output` paramètre est défini sur`FALSE`, la RMAN sortie est écrite dans un fichier du `BDUMP` répertoire.

Pour afficher les fichiers dans le répertoire `BDUMP`, exécutez l'instruction `SELECT` suivante.

```
SELECT * FROM table(rdsadmin.rds_file_util.listdir('BDUMP')) order by mtime;
```

Pour afficher le contenu d'un fichier dans le répertoire `BDUMP`, exécutez l'instruction `SELECT` suivante.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','rds-rman-
validate-nnn.txt'));
```

Remplacez le nom du fichier par celui du fichier que vous souhaitez afficher.

Validation d'une base de données locataire

Pour valider les fichiers de données de la base de données client dans une base de données conteneur (CDB), utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.validate_tenant`.

Cette procédure s'applique uniquement à la base de données des locataires actuelle et utilise les paramètres courants suivants pour les RMAN tâches :

- `p_validation_type`
- `p_parallel`
- `p_section_size_mb`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures. Cette procédure est prise en charge pour les versions suivantes du moteur de base de données :

- Base de données Oracle 21c (21.0.0) CDB
- Base de données Oracle 19c (19.0.0) CDB

L'exemple suivant valide la base de données locataire actuelle à l'aide des valeurs par défaut pour les paramètres.

```
EXEC rdsadmin.rdsadmin_rman_util.validate_tenant;
```

L'exemple suivant valide la base de données locataire actuelle à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.validate_tenant(
        p_validation_type     => 'PHYSICAL+LOGICAL',
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_rman_to_dbms_output => FALSE);
END;
/
```

Lorsque le `p_rman_to_dbms_output` paramètre est défini sur`FALSE`, la RMAN sortie est écrite dans un fichier du `BDUMP` répertoire.

Pour afficher les fichiers dans le répertoire `BDUMP`, exécutez l'instruction `SELECT` suivante.

```
SELECT * FROM table(rdsadmin.rds_file_util.listdir('BDUMP')) order by mtime;
```

Pour afficher le contenu d'un fichier dans le répertoire `BDUMP`, exécutez l'instruction `SELECT` suivante.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','rds-rman-
validate-nnn.txt'));
```

Remplacez le nom du fichier par celui du fichier que vous souhaitez afficher.

Validation d'un espace de table

Pour valider les fichiers associés à un tablespace, utilisez la procédure AmazonRDS. `rdsadmin.rdsadmin_rman_util.validate_tablespace`

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_validation_type`
- `p_parallel`
- `p_section_size_mb`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_tablespace_name`	varchar2	Nom d'espace de table valide	—	Oui	Nom de l'espace de table.

Validation d'un fichier de contrôle

Pour valider uniquement le fichier de contrôle utilisé par une instance de base de données Amazon RDS Oracle, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.validate_current_controlfile`.

Cette procédure utilise le paramètre commun suivant pour les RMAN tâches :

- `p_validation_type`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Validation d'un SPFILE

Pour valider uniquement le fichier de paramètres du serveur (SPFILE) utilisé par une instance de base de données Amazon RDS Oracle, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.validate_spfile`.

Cette procédure utilise le paramètre commun suivant pour les RMAN tâches :

- `p_validation_type`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Validation d'un fichier de données Oracle

Pour valider un fichier de données, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.validate_datafile`.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_validation_type`
- `p_parallel`
- `p_section_size_mb`
- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également les paramètres supplémentaires suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_datafile`	varchar2	Numéro d'ID de	—	Oui	Numéro d'ID de fichier de données (issu de v

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
		fichier de données valide ou nom de fichier de données valide comprenant le chemin complet			`$datafile.file#`) ou nom de fichier de données complet comprenant le chemin (issu de `v$datafile.name`).
`p_from_block`	nombre	Entier valide	`NULL`	Non	Numéro du bloc par lequel la validation commence à l'intérieur du fichier de données. Lorsqu'il est `NULL`, 1 est utilisé.
`p_to_block`	nombre	Entier valide	`NULL`	Non	Numéro du bloc par lequel la validation finit à l'intérieur du fichier de données. Lorsqu'il est `NULL`, le bloc le plus grand du fichier de données est utilisé.

Activation et désactivation du suivi des modifications de bloc

Le suivi des modifications de bloc enregistre les blocs dans un fichier de suivi. Cette technique permet d'améliorer les performances des sauvegardes RMAN incrémentielles. Pour plus d'informations, consultez Utilisation du suivi des modifications de bloc pour améliorer les performances des sauvegardes incrémentielles dans la documentation Oracle Database.

RMANles fonctionnalités ne sont pas prises en charge dans une réplique en lecture. Toutefois, dans le cadre de votre stratégie de haute disponibilité, vous pouvez choisir d'activer le suivi des blocs dans un réplica en lecture seule à l'aide de la procédure `rdsadmin.rdsadmin_rman_util.enable_block_change_tracking`. Si vous promouvez ce réplica en lecture seule en instance de base de données source, le suivi des modifications de bloc est activé pour la nouvelle instance source. Ainsi, votre instance peut bénéficier de sauvegardes incrémentielles rapides.

Les procédures de suivi des modifications de bloc sont prises en charge dans la version Enterprise Edition uniquement pour les versions suivantes du moteur de base de données :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

> **Note**
>
> Dans un locataire uniqueCDB, les opérations suivantes fonctionnent, mais aucun mécanisme visible par le client ne peut détecter l'état actuel des opérations. Voir aussi Limites de RDS pour Oracle CDBs.

Pour activer le suivi des modifications de bloc pour une instance de base de données, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.enable_block_change_tracking`. Pour désactiver le suivi des modifications de bloc, utilisez `disable_block_change_tracking`. Ces procédures ne prennent aucun paramètre.

Pour déterminer si le suivi des modifications de bloc est activé pour votre instance de base de données, exécutez la requête suivante.

```
SELECT STATUS, FILENAME FROM V$BLOCK_CHANGE_TRACKING;
```

L'exemple suivant active le suivi des modifications de bloc pour une instance de base de données.

```
EXEC rdsadmin.rdsadmin_rman_util.enable_block_change_tracking;
```

L'exemple suivant désactive le suivi des modifications de bloc pour une instance de base de données.

```
EXEC rdsadmin.rdsadmin_rman_util.disable_block_change_tracking;
```

Recoupement des journaux redo archivés

Vous pouvez vérifier les journaux de redo archivés à l'aide de la procédure AmazonRDS. `rdsadmin.rdsadmin_rman_util.crosscheck_archivelog`

Vous pouvez utiliser cette procédure pour recouper les journaux redo archivés inscrits dans le fichier de contrôle et supprimer éventuellement les enregistrements de journaux ayant expiré. Lorsqu'il RMAN effectue une sauvegarde, il crée un enregistrement dans le fichier de contrôle. Au fil du temps, ces enregistrements augmentent la taille du fichier de contrôle. Nous vous recommandons de supprimer périodiquement les enregistrements expirés.

> **Note**
>
> RDSLes sauvegardes Amazon standard ne sont pas utilisées RMAN et ne créent donc pas d'enregistrements dans le fichier de contrôle.

Cette procédure utilise le paramètre commun `p_rman_to_dbms_output` pour les RMAN tâches.

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_delete_expired`	booléen	`TRUE`, `FALSE`	`TRUE`	Non	Lorsque la valeur est `TRUE`, supprimez les enregistrements de journaux redo archivés expirés du fichier de contrôle. Lorsque la valeur est `FALSE`, conservez

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
					les enregistrements de journaux redo archivés expirés dans le fichier de contrôle.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

L'exemple suivant marque les enregistrements de journaux redo archivés dans le fichier de contrôle comme ayant expiré, mais ne les supprime pas.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.crosscheck_archivelog(
        p_delete_expired      => FALSE,
        p_rman_to_dbms_output => FALSE);
END;
/
```

L'exemple suivant supprime les enregistrements de journaux redo archivés expirés du fichier de contrôle.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.crosscheck_archivelog(
        p_delete_expired      => TRUE,
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde des fichiers de journalisation archivés

Vous pouvez utiliser le RDS package Amazon `rdsadmin.rdsadmin_rman_util` pour sauvegarder les journaux redo archivés pour une instance de base de données Amazon RDS Oracle.

Les procédures de sauvegarde des journaux redo archivés sont prises en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

Rubriques

- Sauvegarde de tous les journaux redo archivés
- Sauvegarde d'un journal redo archivé à partir d'une plage de dates
- Sauvegarde d'un journal de restauration archivé à partir d'une plage SCN
- Sauvegarde d'un journal redo archivé à partir d'une plage de numéros de séquence

Sauvegarde de tous les journaux redo archivés

Pour sauvegarder tous les redo logs archivés pour une instance de base de données Amazon RDS Oracle, utilisez la RDS procédure `rdsadmin.rdsadmin_rman_util.backup_archivelog_all` Amazon.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

L'exemple suivant sauvegarde tous les journaux redo archivés pour l'instance de base de données.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_archivelog_all(
```

```
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_parallel            => 4,
        p_tag                 => 'MY_LOG_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde d'un journal redo archivé à partir d'une plage de dates

Pour sauvegarder des journaux de journalisation archivés spécifiques pour une instance de base de données Amazon RDS Oracle en spécifiant une plage de dates, utilisez la RDS procédure `rdsadmin.rdsadmin_rman_util.backup_archivelog_date` Amazon. La plage de dates indique quels journaux redo archivés sauvegarder.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également les paramètres supplémentaires suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_from_date`	date	Date comprise entre `start_da`	—	Oui	Date de début des sauvegardes des journaux archivés.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
		e et next_date d'un journal redo archivé qui existe sur le disque. Cette valeur doit être inférieure ou égale à la valeur spécifiée pour p_to_date .			

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_to_date	date	Date comprise entre start_date et next_date d'un journal redo archivé qui existe sur le disque. Cette valeur doit être supérieure ou égale à la valeur spécifiée pour p_from_date .	—	Oui	Date de fin des sauvegardes des journaux archivés.

L'exemple suivant sauvegarde les journaux redo archivés dans la plage de dates pour l'instance de base de données.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_archivelog_date(
        p_owner                => 'SYS',
        p_directory_name       => 'MYDIRECTORY',
        p_from_date            => '03/01/2019 00:00:00',
```

```
        p_to_date             => '03/02/2019 00:00:00',
        p_parallel            => 4,
        p_tag                 => 'MY_LOG_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde d'un journal de restauration archivé à partir d'une plage SCN

Pour sauvegarder des journaux de journalisation archivés spécifiques pour une instance de base de données Amazon RDS Oracle en spécifiant une plage de numéros de modification du système (SCN), utilisez la RDS procédure `rdsadmin.rdsadmin_rman_util.backup_archivelog_scn` Amazon. La SCN plage indique les journaux de restauration archivés à sauvegarder.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également les paramètres supplémentaires suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_from_scn`	nombre	Il s'SCNagit d'un journal de	—	Oui	Le point de départ SCN des sauvegardes des journaux archivés.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
		restauration archivé qui existe sur le disque. Cette valeur doit être inférieure ou égale à la valeur spécifiée pour `p_to_scn`.			

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_to_scn	nombre	Il s'SCNagit d'un journal de restauration archivé qui existe sur le disque. Cette valeur doit être supérieure ou égale à la valeur spécifiée pour p_from_scn .	—	Oui	Fin SCN des sauvegardes des journaux archivés.

L'exemple suivant sauvegarde les journaux de restauration archivés dans la SCN plage correspondant à l'instance de base de données.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_archivelog_scn(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_from_scn            => 1533835,
        p_to_scn              => 1892447,
        p_parallel            => 4,
```

```
        p_tag                 => 'MY_LOG_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde d'un journal redo archivé à partir d'une plage de numéros de séquence

Pour sauvegarder des journaux redo archivés spécifiques pour une instance de base de données Amazon RDS Oracle en spécifiant une plage de numéros de séquence, utilisez la RDS procédure `rdsadmin.rdsadmin_rman_util.backup_archivelog_sequence` Amazon. La plage de numéros de séquence indique quels journaux redo archivés sauvegarder.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également les paramètres supplémentaires suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_from_sequence`	nombre	Numéro de séquence d'un journal redo	—	Oui	Numéro de séquence de début des sauvegardes des journaux archivés.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
		archivé qui existe sur le disque. Cette valeur doit être inférieure ou égale à la valeur spécifiée pour `p_to_sequence` .			

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_to_sequence	nombre	Numéro de séquence d'un journal redo archivé qui existe sur le disque. Cette valeur doit être supérieure ou égale à la valeur spécifiée pour p_from_sequence .	—	Oui	Numéro de séquence de fin des sauvegardes des journaux archivés.

L'exemple suivant sauvegarde les journaux redo archivés dans la plage de numéros de séquence pour l'instance de base de données.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_archivelog_sequence(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_from_sequence       => 11160,
        p_to_sequence         => 11160,
        p_parallel            => 4,
        p_tag                 => 'MY_LOG_BACKUP',
        p_rman_to_dbms_output => FALSE);
```

```
END;
/
```

Réalisation d'une sauvegarde complète de base de données

Vous pouvez effectuer une sauvegarde de tous les blocs de fichiers de données inclus dans la sauvegarde à l'aide de la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.backup_database_full`.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_section_size_mb`
- `p_include_archive_logs`
- `p_optimize`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

L'exemple suivant effectue une sauvegarde complète de l'instance de base de données à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
```

```
    rdsadmin.rdsadmin_rman_util.backup_database_full(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_tag                 => 'FULL_DB_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Réalisation d'une sauvegarde complète d'une base de données locataire

Vous pouvez effectuer une sauvegarde de tous les blocs de données inclus dans une base de données mutualisée dans une base de données conteneur (CDB). Utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.backup_tenant_full`. Cette procédure s'applique uniquement à la sauvegarde de base de données en cours et utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_section_size_mb`
- `p_include_archive_logs`
- `p_optimize`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

La `rdsadmin_rman_util.backup_tenant_full` procédure est prise en charge pour les versions suivantes RDS du moteur de base de données Oracle :

- Base de données Oracle 21c (21.0.0) CDB
- Base de données Oracle 19c (19.0.0) CDB

L'exemple suivant effectue une sauvegarde complète de la base de données locataire actuelle à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_tenant_full(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_tag                 => 'FULL_TENANT_DB_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Réalisation d'une sauvegarde incrémentielle de base de données

Vous pouvez effectuer une sauvegarde incrémentielle de votre instance de base de données à l'aide de la RDS procédure `rdsadmin.rdsadmin_rman_util.backup_database_incremental` Amazon.

Pour de plus amples informations sur les sauvegardes incrémentielles, veuillez consulter Incremental Backups dans la documentation Oracle.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_section_size_mb`
- `p_include_archive_logs`
- `p_include_controlfile`
- `p_optimize`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

Cette procédure utilise également le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_level	nombre	0, 1	0	Non	Spécifiez 0 pour activer une sauvegarde incrémentielle complète. Spécifiez 1 pour activer une sauvegarde incrémentielle non cumulative.

L'exemple suivant effectue une sauvegarde incrémentielle de l'instance de base de données à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_database_incremental(
        p_owner                => 'SYS',
        p_directory_name       => 'MYDIRECTORY',
        p_level                => 1,
        p_parallel             => 4,
        p_section_size_mb      => 10,
        p_tag                  => 'MY_INCREMENTAL_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Réalisation d'une sauvegarde incrémentielle d'une base de données locataire

Vous pouvez effectuer une sauvegarde incrémentielle de la base de données des locataires actuelle dans votreCDB. Utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.backup_tenant_incremental`.

Pour de plus amples informations sur les sauvegardes incrémentielles, consultez Incremental Backups dans la documentation Oracle Database.

Cette procédure s'applique uniquement à la base de données des locataires actuelle et utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_parallel`
- `p_section_size_mb`
- `p_include_archive_logs`
- `p_include_controlfile`
- `p_optimize`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Base de données Oracle 21c (21.0.0) CDB
- Base de données Oracle 19c (19.0.0) CDB

Cette procédure utilise également le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_level	nombre	0, 1	0	Non	Spécifiez 0 pour activer une sauvegarde incrémentielle complète. Spécifiez 1 pour activer une sauvegarde incrémentielle non cumulative.

L'exemple suivant effectue une sauvegarde incrémentielle de la base de données locataire actuelle à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_tenant_incremental(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_level               => 1,
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_tag                 => 'MY_INCREMENTAL_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde d'un espace de table

Vous pouvez sauvegarder un tablespace à l'aide de la procédure AmazonRDS. `rdsadmin.rdsadmin_rman_util.backup_tablespace`

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`

- p_parallel
- p_section_size_mb
- p_include_archive_logs
- p_include_controlfile
- p_optimize
- p_compress
- p_rman_to_dbms_output
- p_tag

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise également le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_tablespace_name	varchar2	Nom d'espace de table valide.	—	Oui	Nom de l'espace de table à sauvegarder.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

L'exemple suivant effectue une sauvegarde d'espace de table à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_tablespace(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
```

```
        p_tablespace_name     => 'MYTABLESPACE',
        p_parallel            => 4,
        p_section_size_mb     => 10,
        p_tag                 => 'MYTABLESPACE_BACKUP',
        p_rman_to_dbms_output => FALSE);
END;
/
```

Sauvegarde d'un fichier de contrôle

Vous pouvez sauvegarder un fichier de contrôle à l'aide de la RDS procédure Amazon`rdsadmin.rdsadmin_rman_util.backup_current_controlfile`.

Cette procédure utilise les paramètres courants suivants pour les RMAN tâches :

- `p_owner`
- `p_directory_name`
- `p_label`
- `p_compress`
- `p_rman_to_dbms_output`
- `p_tag`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

L'exemple suivant sauvegarde un fichier de contrôle à l'aide des valeurs spécifiées pour les paramètres.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.backup_current_controlfile(
        p_owner               => 'SYS',
        p_directory_name      => 'MYDIRECTORY',
        p_tag                 => 'CONTROL_FILE_BACKUP',
```

```
        p_rman_to_dbms_output => FALSE);
END;
/
```

Exécution de la restauration de blocs multimédias

Vous pouvez récupérer des blocs de données individuels, ce que l'on appelle récupération multimédia par blocs, à l'aide RDS des procédures Amazon`rdsadmin.rdsadmin_rman_util.recover_datafile_block`. Vous pouvez utiliser cette procédure de surcharge pour récupérer un bloc de données individuel ou une série de blocs de données.

Cette procédure utilise le paramètre commun suivant pour les RMAN tâches :

- `p_rman_to_dbms_output`

Pour de plus amples informations, veuillez consulter Paramètres communs pour les RMAN procédures.

Cette procédure utilise les paramètres supplémentaires suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`p_datafile`	`NUMBER`	Numéro d'identification de fichier de données valide.	—	Oui	Le fichier de données contenant les blocs corrompus. Spécifiez le fichier de données de l'une des manières suivantes : • Le numéro d'identification du fichier de données, qui se trouve dans `V$DATAFILE.FILE#` • Le nom complet du fichier de données,

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
					y compris le chemin, situé dans `V$DATAFILE.NAME`
`p_block`	`NUMBER`	Un entier valide.	—	Oui	Numéro d'un bloc individuel à récupérer. Les paramètres suivants s'excluent mutuellement : • `p_block` • `p_from_block` et `p_to_block`
`p_from_block`	`NUMBER`	Un entier valide.	—	Oui	Le premier numéro de bloc d'une série de blocs à récupérer. Les paramètres suivants s'excluent mutuellement : • `p_block` • `p_from_block` et `p_to_block`

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
p_to_block	NUMBER	Un entier valide.	—	Oui	Le dernier numéro de bloc d'une série de blocs à récupérer. Les paramètres suivants s'excluent mutuellement : • p_block • p_from_block et p_to_block

Cette procédure est prise en charge pour les versions suivantes du moteur de base de données Amazon RDS pour Oracle :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

L'exemple suivant permet de récupérer le bloc 100 dans le fichier de données 5.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.recover_datafile_block(
        p_datafile            => 5,
        p_block               => 100,
        p_rman_to_dbms_output => TRUE);
END;
/
```

L'exemple suivant permet de récupérer les blocs 100 à 150 dans le fichier de données 5.

```
BEGIN
    rdsadmin.rdsadmin_rman_util.recover_datafile_block(
        p_datafile            => 5,
        p_from_block          => 100,
        p_to_block            => 150,
```

```
        p_rman_to_dbms_output => TRUE);
END;
/
```

Exécution des tâches de planification courantes pour les instances de base de données Oracle

Certaines tâches Oracle Scheduler détenues par SYS peuvent interférer avec les opérations de base de données normales. Dans de tels cas, Oracle Support vous recommande de modifier le calendrier. Si vous devez activer ou désactiver des SYS tâches, testez le fonctionnement sur des tâches planifiées dans un environnement de test avant de l'implémenter dans un environnement de production. Pour effectuer des tâches pour les tâches Oracle Scheduler détenues parSYS, utilisez le package AmazonRDS. `rdsadmin.rdsadmin_dbms_scheduler`

Les `rdsadmin.rdsadmin_dbms_scheduler` procédures sont prises en charge pour les versions du moteur de base de données Amazon RDS pour Oracle indiquées dans le tableau suivant. Lorsque vous utilisez ce package, vous pouvez spécifier les SYS tâches répertoriées dans le tableau.

Version de base de données	Tâches activées par défaut	Tâches désactivées par défaut
Oracle Database 19c	BSLN_MAINTAIN_STATS_JOB CLEANUP_NON_EXIST_OBJ CLEANUP_ONLINE_IND_BUILD CLEANUP_ONLINE_PMO CLEANUP_TAB_IOT_PMO CLEANUP_TRANSIENT_PKG CLEANUP_TRANSIENT_TYPE DRA_REEVALUATE_OPEN_FAILU RES FILE_SIZE_UPD ORA$AUTOTASK_CLEAN PMO_DEFERRED_GIDX_MAINT_JO B PURGE_LOG RSE$CLEAN_RECOVERABLE_SC RIPT SM$CLEAN_AUTO_SPLIT_MERGE	FGR$AUTOPURGE_JOB FILE_WATCHER HM_CREATE_OFFLINE_DICTIONARY LOAD_OPATCH_INVENTORY ORA$PREPLUGIN_BACKUP_JOB XMLDB_NFS_CLEANUP_JOB

Version de base de données	Tâches activées par défaut	Tâches désactivées par défaut
Oracle Database 21c	BSLN_MAINTAIN_STATS_JOB CLEANUP_NON_EXIST_OBJ CLEANUP_ONLINE_IND_BUILD CLEANUP_ONLINE_PMO CLEANUP_TAB_IOT_PMO CLEANUP_TRANSIENT_PKG CLEANUP_TRANSIENT_TYPE DRA_REEVALUATE_OPEN_FAILURES FILE_SIZE_UPD ORA$AUTOTASK_CLEAN PMO_DEFERRED_GIDX_MAINT_JOB PURGE_LOG	FGR$AUTOPURGE_JOB FILE_WATCHER HM_CREATE_OFFLINE_DICTIONARY LOAD_OPATCH_INVENTORY ORA$PREPLUGIN_BACKUP_JOB ORA$_ATSK_AUTOSTS XMLDB_NFS_CLEANUP_JOB

Paramètres communs pour les procédures d'Oracle Scheduler

Pour effectuer des tâches avec Oracle Scheduler, utilisez les procédures du package AmazonRDS. `rdsadmin.rdsadmin_dbms_scheduler` Plusieurs paramètres sont communs aux procédures figurant dans le package. Le package possède les paramètres communs suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`name`	varchar2	Les procédures répertoriées dans le tableau de Exécution des tâches	—	Oui	Nom du travail à modifier.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
		de planification courantes pour les instances de base de données Oracle			
`attribute`	varchar2	`'REPEAT_INTERVAL'` `_NAME'`	–	Oui	Attribut à modifier. Pour modifier l'intervalle de répétition du travail, spécifiez `'REPEAT_INTERVAL'` . Pour modifier le nom de planification du travail, spécifiez `'SCHEDULE_NAME'` .
`value`	varchar2	Intervalle ou nom de planification valide, selon l'attribut utilisé.	–	Oui	Nouvelle valeur de l'attribut.

Modification de DBMS _ SCHEDULER jobs

Utilisez la procédure Oracle `dbms_scheduler.set_attribute` pour modifier certains composants d'Oracle Scheduler. Pour plus d'informations, consultez les ATTRIBUTEprocédures DBMSSET_ SCHEDULER et _ dans la documentation Oracle.

Lorsque vous travaillez avec des instances Amazon RDS DB, ajoutez le nom du schéma `SYS` au nom de l'objet. L'exemple suivant définit l'attribut du plan de la ressource pour l'objet monday window.

```
BEGIN
    DBMS_SCHEDULER.SET_ATTRIBUTE(
        name      => 'SYS.MONDAY_WINDOW',
        attribute => 'RESOURCE_PLAN',
        value     => 'resource_plan_1');
END;
/
```

Modification des fenêtres AutoTask de maintenance

Les instances Amazon RDS pour Oracle sont créées avec les paramètres par défaut pour les fenêtres de maintenance. Les tâches de maintenance automatisées, telles que la collecte de statistiques de l'optimiseur, s'exécutent lors de ces fenêtres. Par défaut, les fenêtres de maintenance activent le gestionnaire de ressources Oracle Database.

Pour modifier la fenêtre, utilisez le package `DBMS_SCHEDULER`. Vous devrez peut-être modifier les paramètres de la fenêtre de maintenance pour les raisons suivantes :

- Vous voulez que les tâches de maintenance s'exécutent à un moment différent, avec des paramètres différents, ou pas du tout. Par exemple, vous souhaitez modifier la durée de la fenêtre ou modifier l'heure et l'intervalle de répétition.
- Vous souhaitez éviter les répercussions sur les performances de l'activation du gestionnaire de ressources pendant la maintenance. Par exemple, si le plan de maintenance par défaut est spécifié et si la fenêtre de maintenance s'ouvre alors que la base de données est en cours de chargement, des événements d'attente tels que `resmgr:cpu quantum` peuvent apparaître. Cet événement d'attente est lié au gestionnaire de ressources de base de données. Vous avez les options suivantes :
  - Vérifiez que les fenêtres de maintenance sont actives pendant les heures creuses pour votre instance de base de données.

- Désactivez le plan de maintenance par défaut en réglant l'attribut `resource_plan` sur une chaîne vide.
- Définissez le paramètre `resource_manager_plan` de votre groupe de paramètres sur `FORCE:`. Si votre instance utilise Enterprise Edition, ce paramètre empêche l'activation des plans du gestionnaire de ressources de base de données.

Pour modifier les paramètres de votre fenêtre de maintenance

1. Connectez-vous à votre base de données à l'aide d'un SQL client Oracle.
2. Interrogez la configuration actuelle pour une fenêtre de planificateur.

   L'exemple suivant interroge la configuration pour `MONDAY_WINDOW`.

   ```
   SELECT ENABLED, RESOURCE_PLAN, DURATION, REPEAT_INTERVAL
   FROM   DBA_SCHEDULER_WINDOWS
   WHERE  WINDOW_NAME='MONDAY_WINDOW';
   ```

   La sortie suivante indique que la fenêtre utilise les valeurs par défaut.

   ```
   ENABLED         RESOURCE_PLAN                  DURATION         REPEAT_INTERVAL
   --------------- ------------------------------ ----------------
    ------------------------------
   TRUE            DEFAULT_MAINTENANCE_PLAN       +000 04:00:00
    freq=daily;byday=MON;byhour=22
                                                                    ;byminute=0;
    bysecond=0
   ```

3. Modifiez la fenêtre à l'aide du package `DBMS_SCHEDULER`.

   L'exemple suivant définit le plan de ressources sur null, afin que le gestionnaire de ressources ne s'exécute pas pendant la fenêtre de maintenance.

   ```
   BEGIN
     -- disable the window to make changes
     DBMS_SCHEDULER.DISABLE(name=>'"SYS"."MONDAY_WINDOW"',force=>TRUE);

     -- specify the empty string to use no plan
     DBMS_SCHEDULER.SET_ATTRIBUTE(name=>'"SYS"."MONDAY_WINDOW"',
    attribute=>'RESOURCE_PLAN', value=>'');
   ```

```
  -- re-enable the window
  DBMS_SCHEDULER.ENABLE(name=>'"SYS"."MONDAY_WINDOW"');
END;
/
```

L'exemple suivant définit la durée maximale de la fenêtre sur 2 heures.

```
BEGIN
  DBMS_SCHEDULER.DISABLE(name=>'"SYS"."MONDAY_WINDOW"',force=>TRUE);
  DBMS_SCHEDULER.SET_ATTRIBUTE(name=>'"SYS"."MONDAY_WINDOW"',
 attribute=>'DURATION', value=>'0 2:00:00');
  DBMS_SCHEDULER.ENABLE(name=>'"SYS"."MONDAY_WINDOW"');
END;
/
```

L'exemple suivant définit l'intervalle de répétition sur tous les lundis, à 10 heures.

```
BEGIN
  DBMS_SCHEDULER.DISABLE(name=>'"SYS"."MONDAY_WINDOW"',force=>TRUE);
  DBMS_SCHEDULER.SET_ATTRIBUTE(name=>'"SYS"."MONDAY_WINDOW"',
 attribute=>'REPEAT_INTERVAL',
 value=>'freq=daily;byday=MON;byhour=10;byminute=0;bysecond=0');
  DBMS_SCHEDULER.ENABLE(name=>'"SYS"."MONDAY_WINDOW"');
END;
/
```

Définition du fuseau horaire pour les tâches d'Oracle Scheduler

Pour modifier le fuseau horaire d'Oracle Scheduler, vous pouvez utiliser la procédure Oracle `dbms_scheduler.set_scheduler_attribute`. Pour plus d'informations sur le `dbms_scheduler` package, consultez DBMS_ SCHEDULER et SET_ SCHEDULER _ ATTRIBUTE dans la documentation Oracle.

Pour modifier le paramètre de fuseau horaire actuel

1. Connectez-vous à la base de données à l'aide d'un client tel que SQL Developer. Pour de plus amples informations, veuillez consulter Connexion à votre instance de base de données à l'aide d'Oracle SQL Developer.

2. Définissez le fuseau horaire par défaut comme suit, en remplaçant votre fuseau horaire par *time_zone_name*.

```
BEGIN
  DBMS_SCHEDULER.SET_SCHEDULER_ATTRIBUTE(
    attribute => 'default_timezone',
    value => 'time_zone_name'
  );
END;
/
```

Dans l'exemple suivant, vous changez le fuseau horaire pour Asia/Shanghai.

Commencez par interroger le fuseau horaire actuel, comme indiqué ci-dessous.

```
SELECT VALUE FROM DBA_SCHEDULER_GLOBAL_ATTRIBUTE WHERE
 ATTRIBUTE_NAME='DEFAULT_TIMEZONE';
```

La sortie indique que le fuseau horaire actuel estETC/UTC.

```
VALUE
-------
Etc/UTC
```

Ensuite, vous définissez le fuseau horaire sur Asia/Shanghai.

```
BEGIN
  DBMS_SCHEDULER.SET_SCHEDULER_ATTRIBUTE(
    attribute => 'default_timezone',
    value => 'Asia/Shanghai'
  );
END;
/
```

Pour plus d'informations sur la modification du fuseau horaire système, consultez Fuseau horaire Oracle.

Désactiver les tâches Oracle Scheduler détenues par SYS

Pour désactiver une tâche Oracle Scheduler appartenant à l'SYSutilisateur, suivez la `rdsadmin.rdsadmin_dbms_scheduler.disable` procédure ci-dessous.

Cette procédure utilise le paramètre commun `name` pour les tâches Oracle Scheduler. Pour plus d'informations, consultez Paramètres communs pour les procédures d'Oracle Scheduler.

L'exemple suivant désactive le travail Oracle Scheduler `SYS.CLEANUP_ONLINE_IND_BUILD`.

```
BEGIN
   rdsadmin.rdsadmin_dbms_scheduler.disable('SYS.CLEANUP_ONLINE_IND_BUILD');
END;
/
```

Activation des tâches Oracle Scheduler détenues par SYS

Pour activer une tâche Oracle Scheduler appartenant àSYS, suivez la `rdsadmin.rdsadmin_dbms_scheduler.enable` procédure ci-dessous.

Cette procédure utilise le paramètre commun `name` pour les tâches Oracle Scheduler. Pour plus d'informations, consultez Paramètres communs pour les procédures d'Oracle Scheduler.

L'exemple suivant active le travail Oracle Scheduler `SYS.CLEANUP_ONLINE_IND_BUILD`.

```
BEGIN
   rdsadmin.rdsadmin_dbms_scheduler.enable('SYS.CLEANUP_ONLINE_IND_BUILD');
END;
/
```

Modification de l'intervalle de répétition d'Oracle Scheduler pour les tâches de type CALENDAR

Pour modifier l'intervalle de répétition afin de modifier une tâche Oracle Scheduler SYS appartenant à Oracle Scheduler, utilisez `CALENDAR` la procédure suivante. `rdsadmin.rdsadmin_dbms_scheduler.disable`

Cette procédure utilise les paramètres communs suivants pour les tâches Oracle Scheduler :

- `name`

- attribute
- value

Pour plus d'informations, consultez Paramètres communs pour les procédures d'Oracle Scheduler.

L'exemple suivant modifier l'intervalle de répétition du travail Oracle Scheduler SYS.CLEANUP_ONLINE_IND_BUILD.

```
BEGIN
     rdsadmin.rdsadmin_dbms_scheduler.set_attribute(
          name      => 'SYS.CLEANUP_ONLINE_IND_BUILD',
          attribute => 'repeat_interval',
          value     => 'freq=daily;byday=FRI,SAT;byhour=20;byminute=0;bysecond=0');
END;
/
```

Modification de l'intervalle de répétition d'Oracle Scheduler pour les tâches de type NAMED

Certains travaux Oracle Scheduler utilisent non pas un intervalle, mais un nom de planification. Pour ce type de travaux, vous devez créer une planification nommée dans le schéma d'utilisateur principal. Pour cela, utilisez la procédure Oracle standard sys.dbms_scheduler.create_schedule. De même, utilisez rdsadmin.rdsadmin_dbms_scheduler.set_attribute procedure pour affecter la nouvelle planification nommée à la tâche.

Cette procédure utilise le paramètre commun suivant pour les tâches Oracle Scheduler :

- name
- attribute
- value

Pour plus d'informations, consultez Paramètres communs pour les procédures d'Oracle Scheduler.

L'exemple suivant modifier l'intervalle de répétition du travail Oracle Scheduler SYS.BSLN_MAINTAIN_STATS_JOB.

```
BEGIN
```

```
     DBMS_SCHEDULER.CREATE_SCHEDULE (
          schedule_name   => 'rds_master_user.new_schedule',
          start_date      => SYSTIMESTAMP,
          repeat_interval =>
 'freq=daily;byday=MON,TUE,WED,THU,FRI;byhour=0;byminute=0;bysecond=0',
          end_date        => NULL,
          comments        => 'Repeats daily forever');
END;
/

BEGIN
     rdsadmin.rdsadmin_dbms_scheduler.set_attribute (
          name      => 'SYS.BSLN_MAINTAIN_STATS_JOB',
          attribute => 'schedule_name',
          value     => 'rds_master_user.new_schedule');
END;
/
```

Désactivation de la validation automatique pour la création de travaux Oracle Scheduler

Quand `DBMS_SCHEDULER.CREATE_JOB` crée des travaux Oracle Scheduler, il les crée immédiatement et valide les modifications. Vous devrez peut-être intégrer la création de travaux Oracle Scheduler dans la transaction utilisateur pour effectuer les opérations suivantes :

- Annuler le travail Oracle Scheduler lorsque la transaction utilisateur est annulée.
- Créer la tâche Oracle Scheduler lorsque la transaction utilisateur principale est validée.

Vous pouvez utiliser la procédure `rdsadmin.rdsadmin_dbms_scheduler.set_no_commit_flag` pour activer ce comportement. Cette procédure ne prend aucun paramètre. Vous pouvez utiliser cette procédure dans les versions suivantes RDS d'Oracle :

- 21.0.0.0.ru-2022-07.rur-2022-07.r1 et versions ultérieures
- 19.0.0.0.ru-2022-07.rur-2022-07.r1 et versions ultérieures

L'exemple suivant désactive la validation automatique pour Oracle Scheduler, crée un travail Oracle Scheduler, puis annule la transaction. Comme la validation automatique est désactivée, la base de données annule également la création du travail Oracle Scheduler.

```
BEGIN
  rdsadmin.rdsadmin_dbms_scheduler.set_no_commit_flag;
  DBMS_SCHEDULER.CREATE_JOB(job_name   => 'EMPTY_JOB',
                            job_type   => 'PLSQL_BLOCK',
                            job_action => 'begin null; end;',
                            auto_drop  => false);
  ROLLBACK;
END;
/

PL/SQL procedure successfully completed.

SELECT * FROM DBA_SCHEDULER_JOBS WHERE JOB_NAME='EMPTY_JOB';

no rows selected
```

Diagnostic des problèmes liés aux instances RDS de base de données Oracle

Oracle Database inclut une infrastructure de diagnostic des pannes que vous pouvez utiliser pour analyser les problèmes de base de données. Dans la terminologie Oracle, un problème est une erreur critique, par exemple, un bogue de code ou une corruption de données. Un incident est la survenue d'un problème. Si la même erreur se produit trois fois, l'infrastructure affiche trois incidents de ce problème. Pour de plus amples informations, veuillez consulter Diagnostic et résolution de problèmes dans la documentation Oracle Database.

L'utilitaire Automatic Diagnostic Repository Command Interpreter (ADRCI) est un outil de ligne de commande Oracle que vous utilisez pour gérer les données de diagnostic. Par exemple, vous pouvez utiliser cet outil pour analyser les problèmes et regrouper les données de diagnostic. Un package d'incidents inclut les données de diagnostic d'un incident ou de tous les incidents qui se rapportent à un problème spécifique. Vous pouvez charger un package d'incidents, qui est implémenté en tant que fichier .zip, vers le support Oracle.

Pour offrir une expérience de service géré, Amazon RDS ne fournit pas d'accès shell àADRCI. Pour effectuer des tâches de diagnostic RDS pour votre instance de base de données for Oracle, utilisez le RDS package Amazon`rdsadmin.rdsadmin_adrci_util`.

Grâce aux fonctions incluses dans `rdsadmin_adrci_util`, vous pouvez répertorier et regrouper les problèmes et les incidents, et également afficher les fichiers de trace. Toutes les fonctions renvoient un ID de tâche. Cet identifiant fait partie du nom du fichier journal qui contient la ADRCI

sortie, comme dans`dbtask-task_id.log`. Le fichier journal se trouve dans le BDUMP répertoire. Vous pouvez télécharger le fichier journal en suivant la procédure décrite dansTéléchargement d'un fichier journal de base de données.

Paramètres courants pour les procédures de diagnostic

Pour effectuer des tâches de diagnostic, utilisez les fonctions RDS du package Amazon`rdsadmin.rdsadmin_adrci_util`. Le package possède les paramètres communs suivants.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`incident_id`	nombre	Un ID d'incident valide ou null	Null	Non	Si la valeur est null, la fonction affiche tous les incidents. Si la valeur n'est pas null et représente un ID d'incident valide, la fonction affiche l'incident spécifié.
`problem_id`	nombre	Un ID de problème valide ou null	Null	Non	Si la valeur est null, la fonction affiche tous les problèmes. Si la valeur n'est pas null et représente un ID de problème valide, la fonction affiche le problème spécifié.
`last`	nombre	Un entier valide supérieur à 0 ou null	Null	Non	Si la valeur est null, la fonction affiche au maximum 50 éléments. Si la valeur n'est pas null, la fonction affiche le nombre spécifié.

Répertorier les incidents

Pour répertorier les incidents de diagnostic pour Oracle, utilisez la RDS fonction Amazon`rdsadmin.rdsadmin_adrci_util.list_adrci_incidents`. Vous pouvez répertorier les incidents en mode basique ou détaillé. Par défaut, la fonction répertorie les 50 incidents les plus récents.

Cette fonction utilise les paramètres communs suivants :

- `incident_id`
- `problem_id`
- `last`

Si vous spécifiez `incident_id` et `problem_id`, alors `incident_id` remplace `problem_id`. Pour de plus amples informations, veuillez consulter Paramètres courants pour les procédures de diagnostic.

Cette fonction utilise le paramètre supplémentaire suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`detail`	booléen	`TRUE` ou `FALSE`	`FALSE`	Non	Si `TRUE`, la fonction répertorie les incidents en mode détail. Si `FALSE`, la fonction répertorie les incidents en mode basique.

Pour répertorier tous les incidents, effectuez une requête à la fonction `rdsadmin.rdsadmin_adrci_util.list_adrci_incidents` sans aucun argument. La requête renvoie l'ID de tâche.

```
SQL> SELECT rdsadmin.rdsadmin_adrci_util.list_adrci_incidents AS task_id FROM DUAL;
```

```
TASK_ID
------------------
1590786706158-3126
```

Vous pouvez également appeler la `rdsadmin.rdsadmin_adrci_util.list_adrci_incidents` fonction sans aucun argument et stocker le résultat dans une variable SQL client. Vous pouvez utiliser la variable dans d'autres instructions.

```
SQL> VAR task_id VARCHAR2(80);
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.list_adrci_incidents;

PL/SQL procedure successfully completed.
```

Pour lire le fichier journal, appelez la RDS procédure Amazon`rdsadmin.rds_file_util.read_text_file`. Indiquez l'ID de tâche dans le nom du fichier. La sortie suivante montre trois incidents : 53523, 53522 et 53521.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:11:46.193 UTC [INFO ] Listing ADRCI incidents.
2020-05-29 21:11:46.256 UTC [INFO ]
ADR Home = /rdsdbdata/log/diag/rdbms/orcl_a/ORCL:
*************************************************************************
INCIDENT_ID PROBLEM_KEY                                                 CREATE_TIME
----------- ------------------------------------------------------------
 ----------------------------------------
53523       ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_003 2020-05-29
 20:15:20.928000 +00:00
53522       ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_002 2020-05-29
 20:15:15.247000 +00:00
53521       ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_001 2020-05-29
 20:15:06.047000 +00:00
3 rows fetched


2020-05-29 21:11:46.256 UTC [INFO ] The ADRCI incidents were successfully listed.
2020-05-29 21:11:46.256 UTC [INFO ] The task finished successfully.
```

```
14 rows selected.
```

Pour répertorier un incident particulier, spécifiez son ID à l'aide du paramètre `incident_id`. Dans l'exemple suivant, vous interroger le fichier journal pour l'incident 53523 uniquement.

```
SQL> EXEC :task_id :=
 rdsadmin.rdsadmin_adrci_util.list_adrci_incidents(incident_id=>53523);

PL/SQL procedure successfully completed.

SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:15:25.358 UTC [INFO ] Listing ADRCI incidents.
2020-05-29 21:15:25.426 UTC [INFO ]
ADR Home = /rdsdbdata/log/diag/rdbms/orcl_a/ORCL:
*************************************************************************
INCIDENT_ID          PROBLEM_KEY
 CREATE_TIME
-------------------- ----------------------------------------------------------
 ---------------------------------
53523                ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_003
 2020-05-29 20:15:20.928000 +00:00
1 rows fetched


2020-05-29 21:15:25.427 UTC [INFO ] The ADRCI incidents were successfully listed.
2020-05-29 21:15:25.427 UTC [INFO ] The task finished successfully.

12 rows selected.
```

Répertorier les problèmes

Pour répertorier les problèmes de diagnostic relatifs à Oracle, utilisez la RDS fonction Amazon`rdsadmin.rdsadmin_adrci_util.list_adrci_problems`.

Par défaut, la fonction répertorie les 50 problèmes les plus récents.

Cette fonction utilise les paramètres courants `problem_id` et `last`. Pour de plus amples informations, veuillez consulter Paramètres courants pour les procédures de diagnostic.

Pour obtenir l'ID de tâche pour tous les problèmes, appelez la `rdsadmin.rdsadmin_adrci_util.list_adrci_problems` fonction sans aucun argument et stockez le résultat dans une variable SQL client.

```
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.list_adrci_problems;

PL/SQL procedure successfully completed.
```

Pour lire le fichier journal, appelez la fonction `rdsadmin.rds_file_util.read_text_file`, en fournissant l'ID de tâche dans le nom du fichier. Dans la sortie suivante, le fichier journal présente trois problèmes : 1, 2 et 3.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:18:50.764 UTC [INFO ] Listing ADRCI problems.
2020-05-29 21:18:50.829 UTC [INFO ]
ADR Home = /rdsdbdata/log/diag/rdbms/orcl_a/ORCL:
*************************************************************************
PROBLEM_ID   PROBLEM_KEY                                                 LAST_INCIDENT
       LASTINC_TIME
---------- ------------------------------------------------------------ -------------
 -------------------------------
2          ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_003 53523
 2020-05-29 20:15:20.928000 +00:00
3          ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_002 53522
 2020-05-29 20:15:15.247000 +00:00
1          ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_001 53521
 2020-05-29 20:15:06.047000 +00:00
3 rows fetched


2020-05-29 21:18:50.829 UTC [INFO ] The ADRCI problems were successfully listed.
2020-05-29 21:18:50.829 UTC [INFO ] The task finished successfully.

14 rows selected.
```

Dans l'exemple suivant, vous répertoriez uniquement le problème 3.

```
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.list_adrci_problems(problem_id=>3);
```

```
PL/SQL procedure successfully completed.
```

Pour lire le fichier journal du problème 3, appelez `rdsadmin.rds_file_util.read_text_file`. Indiquez l'ID de tâche dans le nom du fichier.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:19:42.533 UTC [INFO ] Listing ADRCI problems.
2020-05-29 21:19:42.599 UTC [INFO ]
ADR Home = /rdsdbdata/log/diag/rdbms/orcl_a/ORCL:
*********************************************************************
PROBLEM_ID PROBLEM_KEY                                                 LAST_INCIDENT
 LASTINC_TIME
---------- ------------------------------------------------------------ -------------
 -------------------------------
3          ORA 700 [EVENT_CREATED_INCIDENT] [942] [SIMULATED_ERROR_002 53522
 2020-05-29 20:15:15.247000 +00:00
1 rows fetched


2020-05-29 21:19:42.599 UTC [INFO ] The ADRCI problems were successfully listed.
2020-05-29 21:19:42.599 UTC [INFO ] The task finished successfully.

12 rows selected.
```

Création de packages d'incidents

Vous pouvez créer des packages d'incidents à l'aide de la RDS fonction Amazon`rdsadmin.rdsadmin_adrci_util.create_adrci_package`. La sortie est un fichier .zip que vous pouvez fournir au support Oracle.

Cette fonction utilise les paramètres communs suivants :

- `problem_id`
- `incident_id`

Assurez-vous de spécifier l'un des paramètres précédents. Si vous spécifiez les deux paramètres, `incident_id` remplace `problem_id`. Pour de plus amples informations, veuillez consulter Paramètres courants pour les procédures de diagnostic.

Pour créer un package pour un incident spécifique, appelez la RDS fonction Amazon `rdsadmin.rdsadmin_adrci_util.create_adrci_package` avec le `incident_id` paramètre. L'exemple suivant crée un package pour l'incident 53523.

```
SQL> EXEC :task_id :=
 rdsadmin.rdsadmin_adrci_util.create_adrci_package(incident_id=>53523);

PL/SQL procedure successfully completed.
```

Pour lire le fichier journal, appelez le fichier `rdsadmin.rds_file_util.read_text_file`. Vous pouvez fournir l'ID de tâche dans le nom du fichier. La sortie indique que vous avez généré le package d'incidents `ORA700EVE_20200529212043_COM_1.zip`.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:20:43.031 UTC [INFO ] The ADRCI package is being created.
2020-05-29 21:20:47.641 UTC [INFO ] Generated package 1 in file /rdsdbdata/log/trace/
ORA700EVE_20200529212043_COM_1.zip, mode complete
2020-05-29 21:20:47.642 UTC [INFO ] The ADRCI package was successfully created.
2020-05-29 21:20:47.642 UTC [INFO ] The task finished successfully.
```

Pour regrouper les données de diagnostic concernant un problème particulier, spécifiez son ID à l'aide du paramètre `problem_id`. Dans l'exemple suivant, vous regrouper les données pour le problème 3 uniquement.

```
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.create_adrci_package(problem_id=>3);

PL/SQL procedure successfully completed.
```

Pour lire la sortie de la tâche, appelez `rdsadmin.rds_file_util.read_text_file`, en fournissant l'ID de la tâche dans le nom du fichier. La sortie indique que vous avez généré le package d'incidents `ORA700EVE_20200529212111_COM_1.zip`.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log'));

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:21:11.050 UTC [INFO ] The ADRCI package is being created.
2020-05-29 21:21:15.646 UTC [INFO ] Generated package 2 in file /rdsdbdata/log/trace/
ORA700EVE_20200529212111_COM_1.zip, mode complete
2020-05-29 21:21:15.646 UTC [INFO ] The ADRCI package was successfully created.
2020-05-29 21:21:15.646 UTC [INFO ] The task finished successfully.
```

Vous pouvez également télécharger le fichier journal. Pour de plus amples informations, veuillez consulter Téléchargement d'un fichier journal de base de données.

Affichage des fichiers de trace

Vous pouvez utiliser la RDS fonction Amazon `rdsadmin.rdsadmin_adrci_util.show_adrci_tracefile` pour répertorier les fichiers de trace dans le répertoire de trace et tous les répertoires d'incidents dans le répertoire d'ADRaccueil actuel. Vous pouvez également afficher le contenu des fichiers de suivi et des fichiers de suivi des incidents.

Cette fonction utilise le paramètre suivant.

Nom du paramètre	Type de données	Valeurs valides	Par défaut	Obligatoire	Description
`filename`	varchar2	Un nom de fichier de trace valide	Null	Non	Si la valeur est null, la fonction affiche tous les fichiers de trace. Si elle n'est pas null, la fonction affiche le fichier spécifié.

Pour afficher le fichier de trace, appelez la RDS fonction Amazon`rdsadmin.rdsadmin_adrci_util.show_adrci_tracefile`.

```
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.show_adrci_tracefile;
```

```
PL/SQL procedure successfully completed.
```

Pour répertorier les noms des fichiers de trace, appelez la RDS procédure `rdsadmin.rds_file_util.read_text_file` Amazon en fournissant l'ID de tâche dans le nom du fichier.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log')) WHERE TEXT LIKE '%/alert_%';

TEXT
--------------------------------------------------------------
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-28
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-27
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-26
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-25
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-24
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-23
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-22
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log.2020-05-21
     diag/rdbms/orcl_a/ORCL/trace/alert_ORCL.log

9 rows selected.
```

Dans l'exemple suivant, vous générez une sortie pour`alert_ORCL.log`.

```
SQL> EXEC :task_id := rdsadmin.rdsadmin_adrci_util.show_adrci_tracefile('diag/rdbms/
orcl_a/ORCL/trace/alert_ORCL.log');

PL/SQL procedure successfully completed.
```

Pour lire le fichier journal, appelez `rdsadmin.rds_file_util.read_text_file`. Indiquez l'ID de tâche dans le nom du fichier. La sortie affiche les 10 premières lignes de alert_ ORCL .log.

```
SQL> SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||:task_id||'.log')) WHERE ROWNUM <= 10;

TEXT
--------------------------------------------------------------------------------
2020-05-29 21:24:02.083 UTC [INFO ] The trace files are being displayed.
2020-05-29 21:24:02.128 UTC [INFO ] Thu May 28 23:59:10 2020
```

```
Thread 1 advanced to log sequence 2048 (LGWR switch)
  Current log# 3 seq# 2048 mem# 0: /rdsdbdata/db/ORCL_A/onlinelog/o1_mf_3_hbl2p8xs_.log
Thu May 28 23:59:10 2020
Archived Log entry 2037 added for thread 1 sequence 2047 ID 0x5d62ce43 dest 1:
Fri May 29 00:04:10 2020
Thread 1 advanced to log sequence 2049 (LGWR switch)
  Current log# 4 seq# 2049 mem# 0: /rdsdbdata/db/ORCL_A/onlinelog/o1_mf_4_hbl2qgmh_.log
Fri May 29 00:04:10 2020

10 rows selected.
```

Vous pouvez également télécharger le fichier journal. Pour de plus amples informations, veuillez consulter Téléchargement d'un fichier journal de base de données.

Exécution des tâches diverses pour les instances de base de données Oracle

Vous trouverez ci-dessous des informations sur la façon d'effectuer diverses tâches DBA sur vos instances de base de données Amazon RDS exécutant Oracle. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell aux instances de base de données et limite l'accès à certaines tables et procédures système qui requièrent des privilèges avancés.

Rubriques

- Création et suppression de répertoires dans l'espace de stockage de données principal
- Établissement de la liste des fichiers situés dans un répertoire d'instance de base de données
- Lecture de fichiers dans un répertoire d'instance de base de données
- Accès aux fichiers Opatch
- Gestion des tâches de conseiller
- Transport des espaces de table

Création et suppression de répertoires dans l'espace de stockage de données principal

Pour créer des répertoires, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.create_directory`. Vous pouvez créer jusqu'à 10 000 répertoires, tous situés dans votre espace principal de stockage des données. Pour supprimer des répertoires, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.drop_directory`.

Les procédures `create_directory` et `drop_directory` ont le paramètre requis suivant.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_directory_name	VARCHAR2	—	Oui	Nom du répertoire.

L'exemple suivant crée un répertoire nommé PRODUCT_DESCRIPTIONS.

```
EXEC rdsadmin.rdsadmin_util.create_directory(p_directory_name =>
 'product_descriptions');
```

Le dictionnaire de données stocke le nom du répertoire en majuscules. Vous pouvez lister les répertoires en interrogeant DBA_DIRECTORIES. Le système choisit le nom du chemin réel de l'hôte automatiquement. L'exemple suivant récupère le chemin du répertoire nommé PRODUCT_DESCRIPTIONS:

```
SELECT DIRECTORY_PATH
  FROM DBA_DIRECTORIES
 WHERE DIRECTORY_NAME='PRODUCT_DESCRIPTIONS';

DIRECTORY_PATH
----------------------------------------
/rdsdbdata/userdirs/01
```

Le nom d'utilisateur maître de l'instance de base de données possède des privilèges de lecture et d'écriture dans le nouveau répertoire et peut accorder l'accès à d'autres utilisateurs. Les privilèges EXECUTE ne sont pas disponibles pour les répertoires sur une instance de base de données. Les répertoires sont créés dans votre espace principal de stockage des données et consomment de l'espace, ainsi que de la bande passante d'I/O.

L'exemple suivant supprime le répertoire nommé PRODUCT_DESCRIPTIONS.

```
EXEC rdsadmin.rdsadmin_util.drop_directory(p_directory_name => 'product_descriptions');
```

> **Note**
>
> Vous pouvez également supprimer un répertoire à l'aide de la commande SQL Oracle DROP DIRECTORY.

La suppression d'un répertoire ne supprime pas son contenu. Étant donné que la procédure `rdsadmin.rdsadmin_util.create_directory` peut réutiliser les noms de chemin, les fichiers figurant dans les répertoires supprimés peuvent apparaître dans un répertoire nouvellement créé. Avant de supprimer un répertoire, nous vous recommandons d'utiliser `UTL_FILE.FREMOVE` pour supprimer les fichiers du répertoire. Pour de plus amples informations, veuillez consulter FREMOVE Procedure dans la documentation Oracle.

Établissement de la liste des fichiers situés dans un répertoire d'instance de base de données

Pour lister les fichiers contenus dans un répertoire, utilisez la procédure Amazon RDS `rdsadmin.rds_file_util.listdir`. Cette procédure n'est pas prise en charge sur un réplica Oracle. La procédure `listdir` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_directory`	varchar2	—	Oui	Nom du répertoire à lister.

L'exemple suivant accorde des privilèges de lecture/écriture sur le répertoire `PRODUCT_DESCRIPTIONS` à l'utilisateur `rdsadmin`, puis répertorie les fichiers dans ce répertoire.

```
GRANT READ,WRITE ON DIRECTORY PRODUCT_DESCRIPTIONS TO rdsadmin;
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir(p_directory =>
 'PRODUCT_DESCRIPTIONS'));
```

Lecture de fichiers dans un répertoire d'instance de base de données

Pour lire un fichier texte, utilisez la procédure Amazon RDS `rdsadmin.rds_file_util.read_text_file`. La procédure `read_text_file` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_directory`	varchar2	—	Oui	Nom du répertoire qui contient le fichier.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_filename	varchar2	—	Oui	Nom du fichier à lire.

L'exemple suivant lit le fichier `rice.txt` dans le répertoire `PRODUCT_DESCRIPTIONS`.

```
declare
  fh sys.utl_file.file_type;
begin
  fh := utl_file.fopen(location=>'PRODUCT_DESCRIPTIONS', filename=>'rice.txt',
 open_mode=>'w');
  utl_file.put(file=>fh, buffer=>'AnyCompany brown rice, 15 lbs');
  utl_file.fclose(file=>fh);
end;
/
```

L'exemple suivant lit le fichier `rice.txt` figurant dans le répertoire `PRODUCT_DESCRIPTIONS`.

```
SELECT * FROM TABLE
    (rdsadmin.rds_file_util.read_text_file(
        p_directory => 'PRODUCT_DESCRIPTIONS',
        p_filename  => 'rice.txt'));
```

Accès aux fichiers Opatch

Opatch est un utilitaire Oracle qui permet l'application et la restauration de correctifs sur le logiciel Oracle. Le mécanisme Oracle qui permet de déterminer les correctifs ayant été appliqués à une base de données est la commande `opatch lsinventory`. Pour ouvrir des demandes de service pour les clients Bring Your Own Licence (BYOL), le support Oracle demande le fichier `lsinventory` et parfois le fichier `lsinventory_detail` généré par Opatch.

Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès shell à Opatch. En lieu et place, le `lsinventory-`*`dbv`*`.txt` dans le répertoire BDUMP contient les informations de correctif relatives à la version actuelle de votre moteur. Lorsque vous effectuez une mise à niveau mineure ou majeure, Amazon RDS met à jour `lsinventory-`*`dbv`*`.txt` dans l'heure suivant l'application du correctif. Pour vérifier les correctifs appliqués, consultez `lsinventory-`*`dbv`*`.txt`. Cette action revient à exécuter la commande `opatch lsinventory`.

> **Note**
>
> Les exemples de cette section supposent que le répertoire BDUMP est nommé `BDUMP`. Sur un réplica en lecture, le nom du répertoire BDUMP est différent. Pour savoir comment obtenir le nom BDUMP en interrogeant `V$DATABASE.DB_UNIQUE_NAME` sur un réplica en lecture, veuillez consulter Liste de fichiers.

Les fichiers d'inventaire utilisent la convention de dénomination Amazon RDS `lsinventory-`*`dbv`*`.txt` et `lsinventory_detail-`*`dbv`*`.txt`, où *`dbv`* est le nom complet de votre version de base de données. Le fichier `lsinventory-`*`dbv`*`.txt` est disponible sur toutes les versions de base de données. Le correspondant `lsinventory_detail-`*`dbv`*`.txt` est disponible sur 19.0.0.0, ru-2020-01.rur-2020-01.r1 ou version ultérieure.

Par exemple, si votre version de base de données est 19.0.0.0.ru-2021-07.rur-2021-07.r1, vos fichiers d'inventaire portent les noms suivants.

```
lsinventory-19.0.0.0.ru-2021-07.rur-2021-07.r1.txt
lsinventory_detail-19.0.0.0.ru-2021-07.rur-2021-07.r1.txt
```

Assurez-vous de télécharger les fichiers qui correspondent à la version actuelle de votre moteur de base de données.

Console

Pour télécharger un fichier d'inventaire à l'aide de la console

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de l'instance de base de données qui contient le fichier journal que vous voulez consulter.
4. Choisissez l'onglet Logs & events (Journaux et événements).
5. Faites défiler jusqu'à la section Journaux.
6. Dans la section Logs (Journaux) recherchez `lsinventory`.
7. Sélectionnez le fichier auquel vous souhaitez accéder, puis choisissez Download (Télécharger).

SQL

Pour lire le fichier `lsinventory-dbv.txt` dans un client SQL, vous pouvez utiliser une instruction `SELECT`. Pour cette technique, utilisez l'une des fonctions `rdsadmin` suivantes : `rdsadmin.rds_file_util.read_text_file` ou `rdsadmin.tracefile_listing`.

Dans l'exemple de requête suivant, remplacez *dbv* par votre version de base de données Oracle. Par exemple, la version de votre base de données peut être 19.0.0.0.ru-2020-04.rur-2020-04.r1.

```
SELECT text
FROM   TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP', 'lsinventory-dbv.txt'));
```

PL/SQL

Pour lire le fichier `lsinventory-dbv.txt` dans un client SQL, vous pouvez écrire un programme PL/SQL. Ce programme utilise `utl_file` pour lire le fichier et `dbms_output` pour l'imprimer. Ce sont des packages fournis par Oracle.

Dans l'exemple de programme suivant, remplacez *dbv* par la version de votre base de données Oracle. Par exemple, la version de votre base de données peut être 19.0.0.0.ru-2020-04.rur-2020-04.r1.

```
SET SERVEROUTPUT ON
DECLARE
  v_file              SYS.UTL_FILE.FILE_TYPE;
  v_line              VARCHAR2(1000);
  v_oracle_home_type  VARCHAR2(1000);
  c_directory         VARCHAR2(30) := 'BDUMP';
  c_output_file       VARCHAR2(30) := 'lsinventory-dbv.txt';
BEGIN
  v_file := SYS.UTL_FILE.FOPEN(c_directory, c_output_file, 'r');
  LOOP
    BEGIN
      SYS.UTL_FILE.GET_LINE(v_file, v_line,1000);
      DBMS_OUTPUT.PUT_LINE(v_line);
    EXCEPTION
      WHEN no_data_found THEN
        EXIT;
    END;
  END LOOP;
END;
```

```
/
```

Ou interrogez `rdsadmin.tracefile_listing` et spoulez la sortie vers un fichier. L'exemple suivant spoule la sortie vers `/tmp/tracefile.txt`.

```
SPOOL /tmp/tracefile.txt
SELECT *
FROM   rdsadmin.tracefile_listing
WHERE  FILENAME LIKE 'lsinventory%';
SPOOL OFF;
```

Gestion des tâches de conseiller

Oracle Database comprend un nombre de conseillers. Chaque conseiller prend en charge des tâches automatisées et manuelles. Vous pouvez utiliser des procédures dans le package `rdsadmin.rdsadmin_util` pour gérer certaines tâches de conseiller.

Les procédures de tâches de conseiller sont disponibles dans les versions suivantes du moteur :

- Oracle Database 21c (21.0.0)
- Version 19.0.0.0.ru-2021-01.rur-2021-01.r1 et versions ultérieures de Oracle Database 19c

  Pour plus d'informations, consultez Version 19.0.0.0.ru-2021-01.rur-2021-01.r1 dans Amazon RDS for Oracle Release Notes (Notes de mise à jour de Amazon RDS for Oracle).

Rubriques

- Définition des paramètres des tâches de conseiller
- Désactivation de AUTO_STATS_ADVISOR_TASK
- Réactivation de AUTO_STATS_ADVISOR_TASK

Définition des paramètres des tâches de conseiller

Pour définir les paramètres de certaines tâches de conseiller, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.advisor_task_set_parameter`. La procédure `advisor_task_set_parameter` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_task_name	varchar2	—	Oui	Nom de la tâche de conseiller dont vous voulez modifier les paramètres. Les valeurs suivantes sont valides : • AUTO_STATS_ADVISOR_TASK • INDIVIDUAL_STATS_ADVISOR_TASK • SYS_AUTO_SPM_EVOLVE_TASK • SYS_AUTO_SQL_TUNING_TASK

p_parameter	varchar2	—	Oui	Nom du paramètre de la tâche. Pour rechercher des paramètres valides d'une tâche de conseiller, exécutez la requête suivante. Remplacer *p_task_name* par une valeur valide de p_task_name :
				`COL PARAMETER_NAME FORMAT a30`
				`COL PARAMETER_VALUE FORMAT a30`
				`SELECT PARAMETER_NAME, PARAMETER_VALUE`
				`FROM DBA_ADVISOR_PARAMETERS`
				`WHERE TASK_NAME='` *p_task_name* `'`
				`AND PARAMETER_VALUE != 'UNUSED'`
				`ORDER BY PARAMETER_NAME;`

p_value	varchar2	—	Oui	Valeur d'un paramètre de la tâche. Pour rechercher des valeurs valides pour des paramètres de la tâche, exécutez la requête suivante. Remplacer *p_task_name* par une valeur valide de p_task_name :
				`COL PARAMETER_NAME FORMAT a30`
				`COL PARAMETER_VALUE FORMAT a30`
				`SELECT PARAMETER_NAME, PARAMETER_VALUE`

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				`FROM DBA_ADVISOR_PARAMETERS WHERE TASK_NAME='  p_task_name ' AND PARAMETER_VALUE != 'UNUSED' ORDER BY PARAMETER_NAME;`

Le programme PL/SQL suivant définit `ACCEPT_PLANS` à `FALSE` pour `SYS_AUTO_SPM_EVOLVE_TASK`. La tâche automatisée SQL Plan Management vérifie les plans et génère un rapport de résultats, mais ne fait pas évoluer les plans automatiquement. Vous pouvez utiliser un rapport pour identifier de nouvelles lignes de base de SQL Plan et les accepter manuellement.

```
BEGIN
  rdsadmin.rdsadmin_util.advisor_task_set_parameter(
    p_task_name => 'SYS_AUTO_SPM_EVOLVE_TASK',
    p_parameter => 'ACCEPT_PLANS',
    p_value     => 'FALSE');
END;
```

Le programme PL/SQL suivant définit `EXECUTION_DAYS_TO_EXPIRE` à `10` pour `AUTO_STATS_ADVISOR_TASK`. La tâche prédéfinie `AUTO_STATS_ADVISOR_TASK` s'exécute dans la fenêtre de maintenance une fois par jour automatiquement. Dans l'exemple, la période de rétention pour l'exécution de la tâche est définie à 10 jours.

```
BEGIN
  rdsadmin.rdsadmin_util.advisor_task_set_parameter(
    p_task_name => 'AUTO_STATS_ADVISOR_TASK',
    p_parameter => 'EXECUTION_DAYS_TO_EXPIRE',
    p_value     => '10');
END;
```

Désactivation de AUTO_STATS_ADVISOR_TASK

Pour désactiver `AUTO_STATS_ADVISOR_TASK`, utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_util.advisor_task_drop`. La procédure `advisor_task_drop` accepte les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_task_name	varchar2	—	Oui	Nom de la tâche de conseiller qui doit être désactivée. La seule valeur valide est AUTO_STATS_ADVISOR_TASK .

La commande suivante désactive AUTO_STATS_ADVISOR_TASK.

```
EXEC rdsadmin.rdsadmin_util.advisor_task_drop('AUTO_STATS_ADVISOR_TASK')
```

Vous pouvez réactiver AUTO_STATS_ADVISOR_TASK en utilisant rdsadmin.rdsadmin_util.dbms_stats_init.

Réactivation de AUTO_STATS_ADVISOR_TASK

Pour réactiver AUTO_STATS_ADVISOR_TASK, utilisez la procédure Amazon RDS rdsadmin.rdsadmin_util.dbms_stats_init. La procédure dbms_stats_init n'accepte aucun paramètre.

La commande suivante réactive AUTO_STATS_ADVISOR_TASK.

```
EXEC rdsadmin.rdsadmin_util.dbms_stats_init()
```

Transport des espaces de table

Utilisez le RDS package Amazon rdsadmin.rdsadmin_transport_util pour copier un ensemble de tablespaces d'une base de données Oracle sur site vers une RDS instance de base de données Oracle. Au niveau physique, la fonctionnalité d'espace de table transportable copie de manière incrémentielle les fichiers de données et de métadonnées sources vers votre instance cible. Vous pouvez transférer les fichiers à l'aide d'Amazon EFS ou d'Amazon S3. Pour de plus amples informations, veuillez consulter Migration à l'aide des espaces de table transportables Oracle.

Rubriques

- Importation des espaces de table transportés dans votre instance de base de données
- Importation des métadonnées d'espaces de table transportables dans votre instance de base de données

- Établissement de la liste des fichiers orphelins après une importation d'espace de table
- Suppression des fichiers de données devenus orphelins après une importation d'espace de table

Importation des espaces de table transportés dans votre instance de base de données

Utilisez la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces` pour restaurer les espaces de table que vous avez précédemment exportés depuis une instance de base de données source. Dans la phase de transport, vous sauvegardez vos espaces de table en lecture seule, exportez les métadonnées Data Pump, transférez ces fichiers vers votre instance de base de données cible, puis importez les espaces de table. Pour de plus amples informations, veuillez consulter Phase 4 : Transport des espaces de table.

Syntaxe

```
FUNCTION import_xtts_tablespaces(
    p_tablespace_list IN CLOB,
    p_directory_name  IN VARCHAR2,
    p_platform_id     IN NUMBER DEFAULT 13,
    p_parallel        IN INTEGER DEFAULT 0) RETURN VARCHAR2;
```

Paramètres

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_tablespace_list`	`CLOB`	—	Oui	Liste des espaces de table à importer.
`p_directory_name`	`VARCHAR2`	—	Oui	Répertoire qui contient les sauvegardes d'espaces de table.
`p_platform_id`	`NUMBER`	`13`	Non	Fournissez un ID de plateforme qui correspon d à celui spécifié durant la phase de sauvegarde. Pour obtenir la liste des

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				plateformes, interrogez V$TRANSPORTABLE_PLATFORM . La plateforme par défaut est Linux x86 64 bits, qui est au format Little Endian.
p_parallel	INTEGER	0	Non	Degré de parallélisme. Par défaut, le parallélisme est désactivé.

Exemples

L'exemple suivant importe les tablespaces *TBS1*, *TBS2*, et *TBS3* depuis le répertoire *DATA_PUMP_DIR*. La plate-forme source est AIX Based Systems (64 bits), dont l'ID de plate-forme est6. Vous pouvez trouver la plateforme IDs en V$TRANSPORTABLE_PLATFORM interrogeant.

```
VAR task_id CLOB

BEGIN
  :task_id:=rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces(
        'TBS1,TBS2,TBS3',
        'DATA_PUMP_DIR',
        p_platform_id => 6);
END;
/

PRINT task_id
```

Importation des métadonnées d'espaces de table transportables dans votre instance de base de données

Utilisez la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_metadata` pour importer des métadonnées d'espace disque logique transportables dans votre instance de base de données RDS pour Oracle. Pendant l'opération, le statut de l'importation des métadonnées est indiqué dans la table `rdsadmin.rds_xtts_operation_info`. Pour de plus amples informations,

veuillez consulter Étape 5 : Importer les métadonnées d'espace de table dans votre instance de base de données cible.

Syntaxe

```
PROCEDURE import_xtts_metadata(
    p_datapump_metadata_file IN SYS.DBA_DATA_FILES.FILE_NAME%TYPE,
    p_directory_name         IN VARCHAR2,
    p_exclude_stats          IN BOOLEAN DEFAULT FALSE,
    p_remap_tablespace_list  IN CLOB DEFAULT NULL,
    p_remap_user_list        IN CLOB DEFAULT NULL);
```

Paramètres

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_datapump_metadata_file	SYS.DBA_DATA_FILES.FILE_NAME%TYPE	—	Oui	Nom du fichier Data Pump d'Oracle qui contient les métadonnées de vos espaces de table transportables.
p_directory_name	VARCHAR2	—	Oui	Répertoire qui contient le fichier Data Pump.
p_exclude_stats	BOOLEAN	FALSE	Non	Indicateur qui indique s'il convient d'exclure les statistiques.
p_remap_tablespace_list	CLOB	NULL	Non	Liste des espaces de table à remapper lors de l'importation des métadonnées. Utilisez le format *from_tbs*:*to_tbs*. Par exemple,

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				spécifiez `users:user_data` .
`p_remap_user_list`	`CLOB`	NULL	Non	Liste des schémas utilisateur à remapper lors de l'importation des métadonnées. Utilisez le format *`from_schema_name`* `:`*`to_schema_name`* . Par exemple, spécifiez `hr:human_resources` .

Exemples

L'exemple importe les métadonnées de l'espace disque logique à partir du fichier. *`xttdump.dmp`*, qui se trouve dans le répertoire *`DATA_PUMP_DIR`*.

```
BEGIN
  rdsadmin.rdsadmin_transport_util.import_xtts_metadata('xttdump.dmp','DATA_PUMP_DIR');
END;
/
```

Établissement de la liste des fichiers orphelins après une importation d'espace de table

Utilisez la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files` pour répertorier les fichiers de données devenus orphelins après une importation d'espace de table. Après avoir identifié les fichiers de données, vous pouvez les supprimer en appelant `rdsadmin.rdsadmin_transport_util.cleanup_incomplete_xtts_import`.

Syntaxe

```
FUNCTION list_xtts_orphan_files RETURN xtts_orphan_files_list_t PIPELINED;
```

Exemples

L'exemple suivant exécute la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files`. La sortie montre deux fichiers de données devenus orphelins.

```
SQL> SELECT * FROM TABLE(rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files);

FILENAME       FILESIZE
-------------- ---------
datafile_7.dbf 104865792
datafile_8.dbf 104865792
```

Suppression des fichiers de données devenus orphelins après une importation d'espace de table

Utilisez la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files` pour supprimer les fichiers de données devenus orphelins après une importation d'espace de table. L'exécution de cette commande génère un fichier journal qui utilise le format de noms `rds-xtts-delete_xtts_orphaned_files-`*`YYYY-MM-DD.HH24-MI-SS.FF`*`.log` dans le répertoire BDUMP. Utilisez la procédure `rdsadmin.rdsadmin_transport_util.cleanup_incomplete_xtts_import` pour trouver les fichiers orphelins. Vous pouvez lire le fichier journal en appelant la procédure `rdsadmin.rds_file_util.read_text_file`. Pour de plus amples informations, veuillez consulter Phase 6 : Nettoyage des fichiers restants.

Syntaxe

```
PROCEDURE cleanup_incomplete_xtts_import(
    p_directory_name IN VARCHAR2);
```

Paramètres

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_directory_name`	`VARCHAR2`	—	Oui	Répertoire qui contient les fichiers de données orphelins.

Exemples

L'exemple suivant supprime les fichiers de données orphelins dans *DATA_PUMP_DIR*.

```
BEGIN
  rdsadmin.rdsadmin_transport_util.cleanup_incomplete_xtts_import('DATA_PUMP_DIR');
END;
/
```

L'exemple suivant lit le fichier journal généré par la commande précédente.

```
SELECT *
FROM TABLE(rdsadmin.rds_file_util.read_text_file(
       p_directory => 'BDUMP',
       p_filename  => 'rds-xtts-
delete_xtts_orphaned_files-2023-06-01.09-33-11.868894000.log'));

TEXT
--------------------------------------------------------------------------------
orphan transported datafile datafile_7.dbf deleted.
orphan transported datafile datafile_8.dbf deleted.
```

Configuration des fonctions avancées RDS for Oracle

RDS for Oracle prend en charge diverses fonctions avancées, notamment HugePages, un stockage d'instances et des types de données étendus.

Rubriques

- Stockage de données temporaires dans un stockage d'instances RDS for Oracle
- Activation de HugePages pour une instance RDS for Oracle
- Activation de types de données étendus dans RDS Oracle

Stockage de données temporaires dans un stockage d'instances RDS for Oracle

Utilisez un stockage d'instances pour les espaces de table temporaires et le cache Smart Flash de la base de données (le cache flash) sur les classes d'instances de base de données RDS for Oracle prises en charge.

Rubriques

- Présentation du stockage d'instances RDS for Oracle
- Activation d'un stockage d'instances RDS for Oracle
- Configuration d'un magasin d'instances RDS pour Oracle
- Utilisation d'un stockage d'instances sur un réplica en lecture Oracle
- Configuration d'un groupe de tablespaces temporaires sur un magasin d'instances et Amazon EBS
- Suppression d'un stockage d'instances RDS for Oracle

Présentation du stockage d'instances RDS for Oracle

Un stockage d'instances fournit un stockage temporaire de niveau bloc pour votre instance de base de données RDS for Oracle. Vous pouvez utiliser un stockage d'instances pour stocker temporairement des informations qui changent fréquemment.

Un magasin d'instance est basé sur des périphériques Non-Volatile Memory Express (NVMe) physiquement connectés à l'ordinateur hôte. Le stockage est optimisé pour une faible latence, des I/O performances aléatoires et un débit de lecture séquentiel.

La taille du stockage d'instances varie selon le type d'instance de base de données. Pour plus d'informations sur le magasin d'instances, consultez le magasin d' EC2 instances Amazon dans le guide de l'utilisateur Amazon Elastic Compute Cloud pour les instances Linux.

Rubriques

- Types de données dans le stockage d'instances RDS for Oracle
- Avantages du stockage d'instances RDS for Oracle
- Classes d'instances prises en charge pour le stockage d'instances RDS for Oracle
- Versions de moteur pris en charge pour le stockage d'instances RDS for Oracle
- Pris en charge Régions AWS pour le magasin d'instances RDS pour Oracle
- Coût du stockage d'instances RDS for Oracle

Types de données dans le stockage d'instances RDS for Oracle

Vous pouvez placer les types de données temporaires RDS for Oracle suivants dans un stockage d'instances :

Un espace de table temporaire

Oracle Database utilise des espaces de table temporaires pour stocker les résultats de requêtes intermédiaires qui ne tiennent pas dans la mémoire. Les requêtes plus volumineuses peuvent générer de grandes quantités de données intermédiaires qui doivent être mises en cache temporairement, mais qui n'ont pas besoin de persister. Un espace de table temporaire est particulièrement utile pour les tris, les agrégations par hachage et les jointures. Si votre instance de base de données RDS for Oracle utilise Enterprise Edition ou Standard Edition 2, vous pouvez placer un espace de table temporaire dans un stockage d'instances.

Le cache flash

Le cache flash améliore les performances des lectures aléatoires à bloc unique dans le chemin conventionnel. Il est recommandé de dimensionner le cache de manière à ce qu'il puisse contenir la majeure partie de votre jeu de données actif. Si votre instance de base de données RDS for Oracle utilise Enterprise Edition, vous pouvez placer le cache flash dans un stockage d'instances.

Par défaut, un stockage d'instances est configuré pour un espace de table temporaire, mais pas pour le cache flash. Vous ne pouvez pas placer les fichiers de données Oracle et les fichiers journaux de base de données dans un stockage d'instances.

Avantages du stockage d'instances RDS for Oracle

Vous pouvez envisager d'utiliser un stockage d'instances pour stocker des fichiers et des caches temporaires que vous pouvez vous permettre de perdre. Si vous souhaitez améliorer les performances de votre base de données ou si l'augmentation de la charge de travail entraîne des problèmes de performances pour votre stockage Amazon EBS, envisagez de passer à une classe d'instance qui prend en charge un stockage d'instances.

En plaçant votre espace de table temporaire et votre cache flash sur un stockage d'instances, vous bénéficiez des avantages suivants :

- Latences de lecture inférieures
- Débit supérieur
- Réduction de la charge sur vos volumes Amazon EBS
- Coûts de stockage et d'instantanés réduits grâce à la réduction de la charge Amazon EBS
- Moins de besoin d'approvisionner des IOPS élevées, ce qui peut réduire votre coût global

En plaçant votre espace de table temporaire sur le stockage d'instances, vous augmentez immédiatement les performances des requêtes qui utilisent de l'espace temporaire. Lorsque vous placez le cache flash sur le stockage d'instances, les lectures de blocs en cache ont généralement une latence bien inférieure à celle des lectures Amazon EBS. Le cache flash doit être « préparé » avant d'offrir des avantages en termes de performances. Le cache se prépare tout seul car la base de données écrit des blocs dans le cache flash à mesure qu'ils sortent du cache de la base de données.

> **Note**
>
> Dans certains cas, le cache flash entraîne une surcharge des performances en raison de la gestion du cache. Avant d'activer le cache flash dans un environnement de production, nous vous recommandons d'analyser votre charge de travail et de tester le cache dans un environnement de test.

Classes d'instances prises en charge pour le stockage d'instances RDS for Oracle

Amazon RDS prend en charge le stockage d'instances pour les classes d'instances de base de données suivantes :

- db.m5d
- db.m6id
- db.r5d
- db.r6id
- db.x2idn
- db.x2iedn

RDS for Oracle prend en charge les classes d'instances de base de données précédentes uniquement pour le modèle de licence BYOL. Pour plus d'informations, consultez RDS pris en charge pour les classes d'instance de base de données Oracle et Bring Your Own License (BYOL) pour EE et SE2.

Pour connaître le stockage d'instance total pour les types d'instances de base de données pris en charge, exécutez la commande suivante dans la AWS CLI.

Example

```
aws ec2 describe-instance-types \
  --filters "Name=instance-type,Values=*5d.*large*,*6id.*large*" \
  --query "InstanceTypes[?contains(InstanceType,'m5d')||contains(InstanceType,'r5d')||
contains(InstanceType,'m6id')||contains(InstanceType,'r6id')][InstanceType,
 InstanceStorageInfo.TotalSizeInGB]" \
  --output table
```

La commande précédente renvoie la taille brute du périphérique pour le stockage d'instances. RDS for Oracle utilise une petite partie de cet espace pour la configuration. L'espace disponible dans le stockage d'instances pour les espaces de table temporaires ou le cache flash est légèrement plus petit.

Versions de moteur pris en charge pour le stockage d'instances RDS for Oracle

Le stockage d'instances est pris en charge pour les versions de moteur RDS for Oracle suivantes :

- Versions 21.0.0.0.ru-2022-01.rur-2022-01.r1 ou versions ultérieures d'Oracle Database 21c
- Versions 19.0.0.0.ru-2021-10.rur-2021-10.r1 ou versions ultérieures d'Oracle Database 19c

Pris en charge Régions AWS pour le magasin d'instances RDS pour Oracle

Le magasin d'instances est disponible partout Régions AWS où un ou plusieurs de ces types d'instances sont pris en charge. Pour plus d'informations sur les classes d'instance db.m5d et db.r5d, consultez Classes d'instances de base de données . Pour plus d'informations sur les classes d'instance prises en charge par Amazon RDS for Oracle, consultez RDS pour les classes d'instance de base de données Oracle.

Coût du stockage d'instances RDS for Oracle

Le coût du stockage d'instances est intégré au coût du stockage d'instances activé sur les instances. Vous n'encourez aucun coût supplémentaire en activant un stockage d'instances sur une instance de base de données RDS for Oracle. Pour plus d'informations sur le stockage d'instances activé sur les instances, consultez Classes d'instances prises en charge pour le stockage d'instances RDS for Oracle.

Activation d'un stockage d'instances RDS for Oracle

Pour activer le stockage d'instances pour les données temporaires RDS for Oracle, effectuez l'une des opérations suivantes :

- Créez une instance de base de données RDS for Oracle à l'aide d'une classe d'instance prise en charge. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données RDS for Oracle à l'aide d'une classe d'instance prise en charge. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Configuration d'un magasin d'instances RDS pour Oracle

Par défaut, 100 % de l'espace de stockage d'instances est alloué à l'espace de table temporaire. Pour configurer le stockage d'instances afin d'allouer de l'espace au cache flash et à l'espace de table temporaire, définissez les paramètres suivants dans le groupe de paramètres de votre instance :

db_flash_cache_size= {* {0,2,4,6,8,10} /10} DBInstanceStore

Ce paramètre indique la quantité d'espace de stockage allouée au cache flash. Ce paramètre n'est valide que pour Oracle Database Enterprise Edition. La valeur par défaut

est `{DBInstanceStore*0/10}`. Si vous définissez une valeur différente de zéro pour`db_flash_cache_size`, votre instance RDS for Oracle active le cache flash après le redémarrage de l'instance.

rds.instance_store_temp_size= {* {0,2,4,6,8,10} /10} DBInstanceStore

Ce paramètre indique la quantité d'espace de stockage allouée à l'espace de table temporaire. La valeur par défaut est `{DBInstanceStore*10/10}`. Ce paramètre est modifiable pour Oracle Database Enterprise Edition et en lecture seule pour Standard Edition 2. Si vous définissez une valeur différente de zéro pour`rds.instance_store_temp_size`, Amazon RDS alloue de l'espace dans le magasin d'instances pour le tablespace temporaire.

Vous pouvez définir les paramètres `db_flash_cache_size` et `rds.instance_store_temp_size` pour les instances de base de données qui n'utilisent pas de stockage d'instances. Dans ce cas, les deux paramètres sont évalués sur `0`, ce qui désactive la fonction. Dans ce cas, vous pouvez utiliser le même groupe de paramètres pour différentes tailles d'instance et pour les instances qui n'utilisent pas de stockage d'instances. Si vous modifiez ces paramètres, veillez à redémarrer les instances associées afin que les modifications puissent prendre effet.

> **Important**
>
> Si vous allouez de l'espace à un espace disque logique temporaire, Amazon RDS ne le crée pas automatiquement. Pour savoir comment créer l'espace de table temporaire sur le stockage d'instances, consultez Création d'un espace de table temporaire sur le stockage d'instances.

La valeur combinée des paramètres précédents ne doit pas dépasser 10/10 ou 100 %. Le tableau suivant présente les paramètres valides et non valides.

db_flash_cache_size setting	rds.instance_store_temp_size setting	Explication
db_flash_cache_size= {*0/10} DBInstanceStore	rds.instance_store_temp_size= {*10/10} DBInstanceStore	Il s'agit d'une configuration valide pour toutes les éditions d'Oracle

db_flash_cache_size setting	rds.instance_store_temp_size setting	Explication
		Database. Amazon RDS alloue 100 % de l'espace de stockage de l'instance au tablespace temporaire. Il s’agit de l’option par défaut.
db_flash_cache_size= {*10/10} DBInstanceStore	rds.instance_store_temp_size= {*0/10} DBInstanceStore	Cette configuration n'est valide que pour Oracle Database Enterprise Edition. Amazon RDS alloue 100 % de l'espace de stockage de l'instance au cache flash.

db_flash_cache_size setting	rds.instance_store_temp_size setting	Explication
db_flash_cache_size= {*2/10} DBInstanceStore	rds.instance_store_temp_size= {*8/10} DBInstanceStore	Cette configuration n'est valide que pour Oracle Database Enterprise Edition. Amazon RDS alloue 20 % de l'espace de stockage d'instance au cache flash et 80 % de l'espace de stockage d'instance au tablespace temporaire.

db_flash_cache_size setting	rds.instance_store_temp_size setting	Explication
db_flash_cache_size= {*6/10} DBInstanceStore	rds.instance_store_temp_size= {*4/10} DBInstanceStore	Cette configuration n'est valide que pour Oracle Database Enterprise Edition. Amazon RDS alloue 60 % de l'espace de stockage d'instance au cache flash et 40 % de l'espace de stockage d'instance au tablespace temporaire.

db_flash_cache_size setting	rds.instance_store_temp_size setting	Explication
db_flash_cache_size= {*2/10} DBInstanceStore	rds.instance_store_temp_size= {*4/10} DBInstanceStore	Cette configuration n'est valide que pour Oracle Database Enterprise Edition. Amazon RDS alloue 20 % de l'espace de stockage d'instance au cache flash et 40 % de l'espace de stockage d'instance au tablespace temporaire.
db_flash_cache_size= {*8/10} DBInstanceStore	rds.instance_store_temp_size= {*8/10} DBInstanceStore	Cette configuration n'est pas valide car le pourcentage combiné d'espace de stockage d'instances dépasse 100 %. Dans ce cas, Amazon RDS échoue.

Considérations relatives à la modification du type d'instance de base de données

Si vous modifiez votre type d'instance de base de données, cela peut affecter la configuration du cache flash ou de l'espace de table temporaire sur le stockage d'instances. Tenez compte des modifications suivantes et de leurs effets :

Vous augmentez ou réduisez la taille de l'instance de base de données qui prend en charge le stockage d'instances.

Les valeurs suivantes augmentent ou diminuent proportionnellement à la nouvelle taille du stockage d'instances :

- La nouvelle taille du cache flash.
- L'espace alloué aux espaces de table temporaires qui se trouvent dans le stockage d'instances.

Par exemple, le paramètre `db_flash_cache_size={DBInstanceStore*6/10}` d'une instance db.m5d.4xlarge fournit environ 340 Go d'espace de cache flash. Si vous augmentez le type d'instance à db.m5d.8xlarge, l'espace de cache flash augmente jusqu'à environ 680 Go.

Vous modifiez une instance de base de données qui n'utilise pas de stockage d'instances en une instance qui utilise un stockage d'instances.

Si `db_flash_cache_size` est défini sur une valeur supérieure à `0`, le cache flash est configuré. Si `rds.instance_store_temp_size` est défini sur une valeur supérieure à `0`, l'espace de stockage d'instances est alloué pour être utilisé par un espace de table temporaire. RDSfor Oracle ne déplace pas automatiquement les fichiers temporaires vers le magasin d'instance. Pour plus d'informations sur l'utilisation de l'espace alloué, consultez Création d'un espace de table temporaire sur le stockage d'instances ou Ajout d'un fichier temporaire au stockage d'instances sur un réplica en lecture.

Vous modifiez une instance de base de données qui utilise un stockage d'instances en une instance qui n'utilise pas de stockage d'instances.

Dans ce cas, RDS for Oracle supprime le cache flash. RDSrecrée le fichier temporaire qui se trouve actuellement dans le magasin d'instance d'un volume AmazonEBS. La taille maximale du nouveau fichier temporaire est l'ancienne taille du paramètre `rds.instance_store_temp_size`.

Utilisation d'un stockage d'instances sur un réplica en lecture Oracle

Les réplicas en lecture prennent en charge le cache flash et les espaces de table temporaires sur un stockage d'instances. Bien que le cache flash fonctionne de la même manière que sur l'instance de base de données principale, notez les différences suivantes pour les espaces de table temporaires :

- Vous ne pouvez pas créer un espace de table temporaire sur un réplica en lecture. Si vous créez un nouvel espace disque logique temporaire sur l'instance principale, Oracle réplique RDS les informations du tablespace sans fichiers temporaires. Pour ajouter un nouveau fichier temporaire, utilisez l'une des techniques suivantes :
  - Utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.add_inst_store_tempfile`. RDSfor Oracle crée un fichier temporaire dans le magasin d'instances de votre réplique de lecture et l'ajoute au tablespace temporaire spécifié.
  - Exécutez la commande `ALTER TABLESPACE … ADD TEMPFILE`. RDSfor Oracle place le fichier temporaire sur Amazon EBS Storage.

  > **Note**
  >
  > La taille du fichier temporaire et les types de stockage peuvent être différents sur l'instance de base de données principale et sur le réplica en lecture.

- Vous pouvez gérer le paramètre d'espace de table temporaire par défaut uniquement sur l'instance de base de données principale. RDSfor Oracle réplique le paramètre sur toutes les répliques en lecture.
- Vous pouvez configurer le paramètre d'espace de table temporaire par défaut uniquement sur l'instance de base de données principale. RDSfor Oracle réplique le paramètre sur toutes les répliques en lecture.

Configuration d'un groupe de tablespaces temporaires sur un magasin d'instances et Amazon EBS

Vous pouvez configurer un groupe d'espaces disque logiques temporaires pour inclure des espaces disque logiques temporaires à la fois sur un magasin d'instances et sur Amazon. EBS Cette technique est utile lorsque vous souhaitez disposer d'un espace de stockage temporaire supérieur à celui autorisé par le paramètre maximum de `rds.instance_store_temp_size`.

Lorsque vous configurez un groupe d'espaces disque logiques temporaire à la fois sur un magasin d'instances et sur AmazonEBS, les deux espaces disque logiques présentent des caractéristiques de performance très différentes. Oracle Database choisit l'espace de table pour traiter les requêtes en fonction d'un algorithme interne. Par conséquent, les requêtes similaires peuvent varier en termes de performances.

En général, vous créez un espace de table temporaire dans le stockage d'instances comme suit :

1. Créez un espace de table temporaire sur le stockage d'instances.
2. Définissez le nouvel espace de table comme l'espace de table temporaire par défaut de la base de données.

Si la taille de l'espace de table dans le stockage d'instances est insuffisante, vous pouvez créer un espace de stockage temporaire supplémentaire comme suit :

1. Attribuez l'espace de table temporaire du stockage d'instances à un groupe d'espaces de table temporaires.
2. Créez un nouveau tablespace temporaire dans Amazon EBS s'il n'en existe pas.
3. Attribuez le tablespace temporaire d'Amazon EBS au même groupe d'espaces disque logiques qui inclut le tablespace du magasin d'instance.
4. Définissez le groupe d'espaces de table comme l'espace de table temporaire par défaut.

L'exemple suivant suppose que la taille de l'espace de table temporaire dans le stockage d'instances ne répond pas aux exigences de votre application. L'exemple crée le tablespace temporaire `temp_in_inst_store` dans le magasin d'instances, l'affecte au groupe d'espaces disque logiques, `temp_group` ajoute le tablespace Amazon existant nommé `temp_in_ebs` à ce groupe et définit ce groupe comme EBS espace disque logique temporaire par défaut.

```
SQL> EXEC rdsadmin.rdsadmin_util.create_inst_store_tmp_tblspace('temp_in_inst_store');

PL/SQL procedure successfully completed.

SQL> ALTER TABLESPACE temp_in_inst_store TABLESPACE GROUP temp_group;

Tablespace altered.

SQL> ALTER TABLESPACE temp_in_ebs TABLESPACE GROUP temp_group;
```

```
Tablespace altered.

SQL> EXEC rdsadmin.rdsadmin_util.alter_default_temp_tablespace('temp_group');

PL/SQL procedure successfully completed.

SQL> SELECT * FROM DBA_TABLESPACE_GROUPS;

GROUP_NAME                     TABLESPACE_NAME
------------------------------ ------------------------------
TEMP_GROUP                     TEMP_IN_EBS
TEMP_GROUP                     TEMP_IN_INST_STORE

SQL> SELECT PROPERTY_VALUE FROM DATABASE_PROPERTIES WHERE
 PROPERTY_NAME='DEFAULT_TEMP_TABLESPACE';

PROPERTY_VALUE
--------------
TEMP_GROUP
```

Suppression d'un stockage d'instances RDS for Oracle

Pour supprimer le stockage d'instances, modifiez votre instance de base de données RDS for Oracle pour utiliser un type d'instance qui ne prend pas en charge le stockage d'instances, tel que db.m5 ou db.r5.

Activation de HugePages pour une instance RDS for Oracle

Amazon RDS for Oracle prend en charge la fonctionnalité HugePages du noyau Linux pour obtenir une base de données plus évolutive. HugePages réduit la taille des tables de page et le temps UC de gestion de la mémoire, ce qui augmente les performances des instances de bases de données volumineuses. Pour plus d'informations, consultez Overview of HugePages (Présentation des grandes pages) dans la documentation Oracle.

Vous pouvez utiliser HugePages avec toutes les versions et éditions de RDS for Oracle prises en charge.

Le paramètre `use_large_pages` contrôle si HugePages est activé pour une instance de base de données. Les valeurs possibles pour ce paramètre sont `ONLY`, `FALSE` et `{DBInstanceClassHugePagesDefault}`. Le paramètre `use_large_pages` est défini sur

`{DBInstanceClassHugePagesDefault}` dans le groupe de paramètres DB par défaut pour Oracle.

Pour contrôler si HugePages est activé automatiquement pour une instance de base de données, vous pouvez utiliser la variable de formule `DBInstanceClassHugePagesDefault` dans les groupes de paramètres. La valeur est déterminée comme suit :

- Pour les classes d'instance de base de données indiquées dans le tableau suivant, `DBInstanceClassHugePagesDefault` a toujours la valeur `FALSE` par défaut, et `use_large_pages` a la valeur `FALSE`. Vous pouvez activer HugePages manuellement pour ces instances de base de données si la classe d'instance de bases de données dispose d'au moins 14 Gio de mémoire.
- Pour les classes d'instance de base de données non mentionnées dans le tableau suivant, si la classe d'instance de base de données possède au moins 14 Gio de mémoire, `DBInstanceClassHugePagesDefault` a toujours la valeur `FALSE`. En outre, `use_large_pages` a la valeur `FALSE`.
- Pour les classes d'instance de base de données non mentionnées dans le tableau suivant, si la classe d'instance possède au moins 14 Gio de mémoire et moins de 100 Gio de mémoire, `DBInstanceClassHugePagesDefault` a la valeur `TRUE` par défaut. En outre, `use_large_pages` a la valeur `ONLY`. Vous pouvez désactiver HugePages manuellement en définissant `use_large_pages` sur `FALSE`.
- Pour les classes d'instance de base de données non mentionnées dans le tableau suivant, si la classe d'instance possède au moins 100 Gio de mémoire, `DBInstanceClassHugePagesDefault` a toujours la valeur `TRUE`. En outre, `use_large_pages` a la valeur `ONLY` et HugePages ne peut pas être désactivé.

HugePages n'est pas activé par défaut pour les classes d'instance de base de données suivantes.

Famille de classes d'instance de base de données	Classes d'instance de base de données avec HugePages non activé par défaut
db.m5	db.m5.large
db.m4	db.m4.large, db.m4.xlarge, db.m4.2xlarge, db.m4.4xlarge, db.m4.10xlarge
db.t3	db.t3.micro, db.t3.small, db.t3.medium, db.t3.large

Pour plus d'informations sur les classes d'instance DB, consultez Spécifications matérielles pour les classes d'instance de base de données .

Pour activer manuellement HugePages pour des instances de bases de données nouvelles ou existantes, définissez le paramètre `use_large_pages` sur `ONLY`. Vous ne pouvez pas utiliser HugePages avec Oracle Automatic Memory Management (AMM). Si vous définissez le paramètre `use_large_pages` sur `ONLY`, vous devez aussi définir `memory_target` et `memory_max_target` sur `0`. Pour plus d'informations sur la définition des paramètres de votre instance de base de données, consultez Groupes de paramètres pour (Amazon RDS).

Vous pouvez aussi définir les paramètres `sga_target`, `sga_max_size` et `pga_aggregate_target`. Lorsque vous définissez les paramètres de mémoire des zones SGA (System Global Area) et PGA (Program Global Area), ajoutez les valeurs ensemble. Soustrayez ce total de votre mémoire d'instance disponible (`DBInstanceClassMemory`) pour déterminer l'espace mémoire libre restant au-delà de l'attribution par HugePages. Vous devez conserver une mémoire libre d'au moins 2 Gio ou égale à 10 pourcent de la mémoire totale disponible de l'instance, la plus petite des deux valeurs étant retenue.

Après avoir configuré vos paramètres, vous devez redémarrer votre instance de base de données pour que les modifications soient effectives. Pour plus d'informations, consultez Redémarrage d'une instance de base de données.

> **Note**
>
> L'instance de base de données Oracle diffère les modifications apportées aux paramètres d'initialisation liés à la SGA jusqu'à ce que vous redémarriez l'instance sans basculement. Dans la console Amazon RDS, choisissez Redémarrer, mais ne choisissez pas Redémarrer avec basculement. Dans la AWS CLI, appelez la commande `reboot-db-instance` avec le paramètre `--no-force-failover`. L'instance de base de données ne traite pas les paramètres liés à la SGA pendant le basculement ou lors d'autres opérations de maintenance qui provoquent le redémarrage de l'instance.

Voici un exemple de configuration de paramètres pour HugePages activant manuellement HugePages. Vous devez définir les valeurs qui répondent à vos besoins.

```
memory_target            = 0
memory_max_target        = 0
```

```
pga_aggregate_target     = {DBInstanceClassMemory*1/8}
sga_target               = {DBInstanceClassMemory*3/4}
sga_max_size             = {DBInstanceClassMemory*3/4}
use_large_pages          = ONLY
```

Supposons que les valeurs des paramètres suivantes sont définies dans un groupe de paramètres.

```
memory_target            = IF({DBInstanceClassHugePagesDefault}, 0,
 {DBInstanceClassMemory*3/4})
memory_max_target        = IF({DBInstanceClassHugePagesDefault}, 0,
 {DBInstanceClassMemory*3/4})
pga_aggregate_target     = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*1/8}, 0)
sga_target               = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*3/4}, 0)
sga_max_size             = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*3/4}, 0)
use_large_pages          = {DBInstanceClassHugePagesDefault}
```

Le groupe de paramètres est utilisé par une classe d'instance de base de données db.r4 dotée de moins de 100 Gio de mémoire. Avec ces paramètres et `use_large_pages` défini sur `{DBInstanceClassHugePagesDefault}`, HugePages est activé sur l'instance db.r4.

Supposons que les valeurs des paramètres suivantes sont définies dans un groupe de paramètres.

```
memory_target           = IF({DBInstanceClassHugePagesDefault}, 0,
 {DBInstanceClassMemory*3/4})
memory_max_target       = IF({DBInstanceClassHugePagesDefault}, 0,
 {DBInstanceClassMemory*3/4})
pga_aggregate_target    = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*1/8}, 0)
sga_target              = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*3/4}, 0)
sga_max_size            = IF({DBInstanceClassHugePagesDefault},
 {DBInstanceClassMemory*3/4}, 0)
use_large_pages         = FALSE
```

Le groupe de paramètres est utilisé par une classe d'instance de base de données db.r4 et une classe d'instance de base de données db.r5, toutes deux avec plus de 100 Gio de mémoire. Avec ces paramètres, HugePages est désactivé sur les instances db.r4 et db.r5.

> **Note**
>
> Si ce groupe de paramètres est utilisé par une classe d'instance de base de données db.r4 ou db.r5 avec au moins 100 Gio de mémoire, la valeur de `FALSE` pour `use_large_pages` est remplacée et définie sur `ONLY`. Dans ce cas, une notification concernant le remplacement est envoyée au client.

Lorsque HugePages est activé sur votre instance de base de données, vous pouvez afficher les informations sur HugePages en activant la surveillance améliorée. Pour de plus amples informations, veuillez consulter Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Activation de types de données étendus dans RDS Oracle

Amazon RDS pour Oracle prend en charge les types de données étendus. Avec les types de données étendus, la taille maximale est de 32 767 octets pour les types de données `VARCHAR2`, `NVARCHAR2` et `RAW`. Pour utiliser les types de données étendus, définissez le paramètre `MAX_STRING_SIZE` sur `EXTENDED`. Pour plus d'informations, consultez Extended Data Types dans la documentation Oracle.

Si vous ne souhaitez pas utiliser les types de données étendus, gardez le paramètre `MAX_STRING_SIZE` défini sur `STANDARD` (par défaut). Dans ce cas, les limites de taille sont de 4 000 octets pour les types de `NVARCHAR2` données `VARCHAR2` et et de 2 000 octets pour le type de RAW données.

Vous pouvez activer les types de données étendus sur une instance de base de données nouvelle ou existante. Pour les nouvelles instances de base de données, la durée nécessaire pour créer une instance de base de données est généralement plus longue lorsque vous activez les types de données étendus. Pour les instances de base de données existantes, l'instance de base de données n'est pas disponible pendant le processus de conversion.

Considérations relatives aux types de données étendus

Tenez compte des points suivants lorsque vous activez des types de données étendus pour votre instance de base de données :

- Lorsque vous activez les types de données étendus pour une instance de base de données nouvelle ou existante, vous devez redémarrer l'instance pour que la modification prenne effet.

- Une fois que vous avez activé les types de données étendus, vous ne pouvez pas modifier à nouveau l'instance de base de données pour utiliser la taille standard pour les types de données. Si vous redéfinissez le `MAX_STRING_SIZE` paramètre`STANDARD`, cela entraîne le `incompatible-parameters` statut.
- Lorsque vous restaurez une instance de base de données qui utilise des types de données étendus, vous devez spécifier un groupe de paramètres avec le paramètre `MAX_STRING_SIZE` défini sur `EXTENDED`. Pendant la restauration, si vous spécifiez le groupe de paramètres par défaut ou tout autre groupe de paramètres avec le paramètre `MAX_STRING_SIZE` défini sur `STANDARD` le statut devient `incompatible-parameters`.
- Lorsque l'état de l'instance de base de données est `incompatible-parameters` à cause du paramètre `MAX_STRING_SIZE`, l'instance de base de données reste indisponible jusqu'à ce que le paramètre `MAX_STRING_SIZE` soit défini sur `EXTENDED` et que l'instance de base de données soit redémarrée.

Activation des types de données étendus pour une nouvelle instance de base de données

Lorsque vous créez une instance de base de données avec `MAX_STRING_SIZE` set to`EXTENDED`, l'instance affiche la valeur `MAX_STRING_SIZE` définie par défaut`STANDARD`. Redémarrez l'instance pour activer la modification.

Activer les types de données étendus pour une nouvelle instance de base de données

1. Définissez le paramètre `MAX_STRING_SIZE` sur `EXTENDED` dans un groupe de paramètres.

   Pour définir le paramètre, vous pouvez créer un groupe de paramètres de base de données ou modifier un groupe de paramètres existant.

   Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).
2. Créez une nouvelle instance de base de données RDS pour Oracle.

   Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.
3. Associez le groupe de paramètres avec `MAX_STRING_SIZE` défini sur `EXTENDED` à l'instance de base de données.

Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.

4. Redémarrez l'instance de base de données pour que la modification des paramètres prenne effet.

   Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Activation des types de données étendus pour une instance de base de données existante

Lorsque vous modifiez une instance de base de données pour activer des types de données étendus, elle RDS convertit les données de la base de données pour utiliser les tailles étendues. La conversion et les temps d'arrêt interviennent lors du prochain redémarrage de la base de données après la modification du paramètre. L'instance de base de données n'est pas disponible pendant la conversion.

La durée nécessaire pour convertir les données dépend de la classe d'instances de base de données, de la taille de la base de données et de l'heure du dernier instantané de base de données. Pour réduire les temps d'arrêt, pensez à prendre un instantané juste avant le redémarrage. Cela permet de raccourcir la durée de la sauvegarde qui a lieu pendant le flux de travail de conversion.

> **Note**
>
> Une fois que vous avez activé les types de données étendus, vous ne pouvez pas effectuer de point-in-time restauration à un moment donné pendant la conversion. Vous pouvez restaurer au moment précédant immédiatement la conversion ou au moment suivant immédiatement la conversion.

Activer les types de données étendus pour une instance de base de données existante

1. Créez un instantané de la base de données.

   Si la base de données contient des objets non valides, Amazon RDS essaie de les recompiler. La conversion en types de données étendus peut échouer si Amazon RDS ne parvient pas à recompiler un objet non valide. L'instantané vous permet de restaurer la base de données en

cas de problème avec la conversion. Vérifiez toujours la présence d'objets non valides avant la conversion afin d'y apporter une solution ou de les supprimer. Pour les bases de données de production, nous vous conseillons d'abord de tester le processus de conversion sur une copie de votre instance de base de données.

Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

2. Définissez le paramètre `MAX_STRING_SIZE` sur `EXTENDED` dans un groupe de paramètres.

   Pour définir le paramètre, vous pouvez créer un groupe de paramètres de base de données ou modifier un groupe de paramètres existant.

   Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).

3. Modifiez l'instance de base de données afin de l'associer au groupe de paramètres avec le paramètre `MAX_STRING_SIZE` défini sur `EXTENDED`.

   Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

4. Redémarrez l'instance de base de données pour que la modification des paramètres prenne effet.

   Pour plus d'informations, consultez Redémarrage d'une instance de base de données.

Importation de données dans Oracle sur Amazon RDS

La manière dont vous importez des données dans une instance de base de données Amazon RDS pour Oracle dépend des éléments suivants :

- La quantité de données dont vous disposez
- Le nombre d'objets de base de données dans votre base de données
- La variété d'objets de base de données dans votre base de données

Par exemple, vous pouvez utiliser les outils suivants, en fonction de vos besoins :

- Oracle SQL Developer — Importez une base de données simple de 20 Mo.
- Oracle Data Pump – Importez des bases de données complexes ou des bases de données de centaines de mégaoctets ou de plusieurs téraoctets. Par exemple, vous pouvez transporter des tablespaces d'une base de données locale vers votre RDS instance de base de données Oracle. Vous pouvez utiliser Amazon S3 ou Amazon EFS pour transférer les fichiers de données et les métadonnées. Pour plus d'informations, consultez Migration à l'aide des espaces de table transportables Oracle, EFSIntégration avec Amazon et Intégration Amazon S3.
- AWS Database Migration Service (AWS DMS) — Migrez les bases de données sans interruption de service. Pour plus d'informations sur ce sujet AWS DMS, consultez What is AWS Database Migration Service et le billet de blog Migration des bases de données Oracle avec un temps d'arrêt quasi nul grâce à. AWS DMS

> **Important**
>
> Avant d'utiliser les techniques de migration précédentes, nous vous recommandons de sauvegarder votre base de données. Après avoir importé les données, vous pouvez sauvegarder vos instances de base RDS de données Oracle en créant des instantanés. Vous pouvez restaurer ultérieurement les instantanés. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

Pour de nombreux moteurs de base de données, la réplication en cours peut se poursuivre jusqu'à ce que vous soyez prêt à basculer sur la base de données cible. Vous pouvez l'utiliser AWS DMS RDS pour migrer vers Oracle à partir du même moteur de base de données ou d'un autre moteur.

Si vous migrez depuis un autre moteur de base de données, vous pouvez utiliser le AWS Schema Conversion Tool pour migrer des objets de schéma qui AWS DMS ne migrent pas.

Rubriques

- Importation à l'aide d'Oracle SQL Developer
- Migration à l'aide des espaces de table transportables Oracle
- Importation à l'aide d'Oracle Data Pump
- Importation avec les utilitaires d’importation/importation d’Oracle
- Importation avec Oracle SQL*Loader
- Migration avec les vues matérialisées d'Oracle

Importation à l'aide d'Oracle SQL Developer

Oracle SQL Developer est un outil graphique Java distribué gratuitement par Oracle. SQL Developer offre des options pour la migration des données entre deux bases de données Oracle, ou pour la migration des données d'autres bases de données, telles que MySQL, vers une base de données Oracle. Cet outil est idéal pour la migration de petites bases de données.

Vous pouvez installer cet outil sur votre ordinateur de bureau (Windows, Linux ou Mac) ou l'un de vos serveurs. Après avoir installé SQL Developer, vous pouvez l'utiliser pour vous connecter à vos bases de données source et cible. Utilisez la commande Database Copy du menu Outils pour copier vos données sur votre instance de base de données RDS pour Oracle.

Pour télécharger SQL Developer, consultez http://www.oracle.com/technetwork/developer-tools/sql-developer.

Nous vous recommandons de lire la documentation de produit Oracle SQL Developer avant de commencer à migrer vos données. Oracle possède également une documentation sur la façon de migrer depuis d'autres bases de données dont MySQL et SQL Server. Pour plus d'informations, consultez http://www.oracle.com/technetwork/database/migration dans la documentation Oracle.

Migration à l'aide des espaces de table transportables Oracle

Vous pouvez utiliser la fonctionnalité d'espaces de table transportables Oracle pour copier un ensemble d'espaces de table à partir d'une base de données Oracle sur site vers une instance de base de données RDS for Oracle. Au niveau physique, vous transférez les fichiers de données source et les fichiers de métadonnées vers votre instance de base de données cible à l'aide

d'Amazon EFS ou d'Amazon S3. La fonctionnalité d'espaces de table transportables utilise le package. `rdsadmin.rdsadmin_transport_util` Pour la syntaxe et la sémantique de ce package, consultez. Transport des espaces de table

Pour les articles de blog expliquant comment transporter des espaces disque logiques, consultez les sections Migrer les bases de données Oracle vers des tablespaces transportables et Amazon RDS for Oracle Transportable Tablespaces à l' AWS aide de RMAN.

Rubriques

- Vue d'ensemble des espaces de table transportables Oracle
- Phase 1 : Configuration de votre hôte source
- Phase 2 : Préparation de la sauvegarde complète des espaces de table
- Phase 3 : Création et transfert de sauvegardes incrémentielles
- Phase 4 : Transport des espaces de table
- Phase 5 : Validation des espaces de table transportés
- Phase 6 : Nettoyage des fichiers restants

Vue d'ensemble des espaces de table transportables Oracle

Un ensemble d'espaces de table transportables se compose de fichiers de données pour l'ensemble d'espaces de table en cours de transport et d'un fichier de vidage d'exportation contenant les métadonnées des espaces de table. Dans une solution de migration physique telle que les espaces de table transportables, vous transférez des fichiers physiques : fichiers de données, fichiers de configuration et fichiers de vidage Data Pump.

Rubriques

- Avantages et inconvénients des espaces de table transportables
- Limitations applicables aux espaces de table transportables
- Prérequis pour les espaces de table transportables

Avantages et inconvénients des espaces de table transportables

Nous vous recommandons d'utiliser des espaces de table transportables lorsque vous devez migrer un ou plusieurs espaces de table volumineux vers RDS avec un minimum de temps d'arrêt. Les espaces de table transportables offrent les avantages suivants par rapport à la migration logique :

- Les temps d'arrêt sont inférieurs à ceux de la plupart des autres solutions de migration Oracle.
- Comme la fonctionnalité des espaces de table transportables copie uniquement les fichiers physiques, elle évite les erreurs d'intégrité des données et la corruption logique qui peuvent survenir lors d'une migration logique.
- Aucune licence supplémentaire n'est requise.
- Vous pouvez migrer un ensemble d'espaces de table entre différentes plateformes et différents types d'endianisme, par exemple d'une plateforme Oracle Solaris vers Linux. Toutefois, le transport des espaces de table vers et depuis des serveurs Windows n'est pas pris en charge.

> **Note**
>
> Linux est entièrement testé et pris en charge. Toutes les variantes UNIX n'ont pas été testées.

Si vous utilisez des espaces de table transportables, vous pouvez transporter les données à l'aide d'Amazon S3 ou d'Amazon EFS :

- Lorsque vous utilisez EFS, vos sauvegardes restent dans le système de fichiers EFS pendant toute la durée de l'importation. Vous pouvez ensuite supprimer les fichiers. Dans cette technique, vous n'avez pas besoin de provisionner le stockage EBS pour votre instance de base de données. C'est pourquoi nous vous recommandons d'utiliser Amazon EFS plutôt que S3. Pour de plus amples informations, veuillez consulter EFSIntégration avec Amazon.
- Lorsque vous utilisez S3, vous téléchargez des sauvegardes RMAN sur le stockage EBS attaché à votre instance de base de données. Les fichiers restent dans votre stockage EBS pendant l'importation. Après l'importation, vous pouvez libérer cet espace, qui reste alloué à votre instance de base de données.

Le principal inconvénient des espaces de table transportables est que vous nécessitez une connaissance relativement avancée d'Oracle Database. Pour plus d'informations, consultez Transport des espaces de table entre bases de données dans le Guide de l'administrateur Oracle Database (langue française non garantie).

Limitations applicables aux espaces de table transportables

Les limitations Oracle Database pour les espaces de table transportables s'appliquent lorsque vous utilisez cette fonctionnalité dans RDS for Oracle. Pour plus d'informations, consultez Limitations

relatives aux espaces de table transportables et Limitations générales relatives au transport de données dans le Guide de l'administrateur Oracle Database (langue française non garantie). Notez les limitations supplémentaires suivantes pour les espaces de table transportables dans RDS for Oracle :

- Ni la base de données source ni la base de données cible ne peuvent utiliser Standard Edition 2 (SE2). Seule Enterprise Edition est prise en charge.
- Vous ne pouvez pas utiliser une base de données Oracle Database 11g comme source. La fonctionnalité d'espaces de table transportables multiplateformes RMAN repose sur le mécanisme de transport RMAN, qu'Oracle Database 11g ne prend pas en charge.
- Vous ne pouvez pas migrer des données depuis une instance de base de données RDS for Oracle en utilisant des espaces de table transportables. Vous pouvez uniquement utiliser des espaces de table transportables pour migrer des données vers une instance de base de données RDS for Oracle.
- Le système d'exploitation Windows n'est pas pris en charge.
- Vous ne pouvez pas transporter des espaces de table vers une base de données à un niveau de version inférieur. La base de données cible doit être à un niveau de version égal ou supérieur à celui de la base de données source. Par exemple, vous ne pouvez pas transporter des espaces de table d'Oracle Database 21c vers Oracle Database 19c.
- Vous ne pouvez pas transporter des espaces de table administratifs tels que `SYSTEM` et `SYSAUX`.
- Vous ne pouvez pas transporter d'objets autres que des données tels que des packages PL/SQL, des classes Java, des vues, des déclencheurs, des séquences, des utilisateurs, des rôles et des tables temporaires. Pour transporter des objets autres que des données, créez-les manuellement ou utilisez l'exportation et l'importation de métadonnées Data Pump. Pour plus d'informations, consultez la note de support My Oracle 1454872.1.
- Vous ne pouvez pas transporter des espaces de table chiffrés ou qui utilisent des colonnes chiffrées.
- Si vous transférez des fichiers à l'aide d'Amazon S3, la taille de fichier maximale prise en charge est de 5 Tio.
- Si la base de données source utilise des options Oracle telles que Spatial, vous ne pouvez pas transporter des espaces de table à moins que les mêmes options soient configurées sur la base de données cible.
- Vous ne pouvez pas transporter des espaces de table vers une instance de base de données RDS for Oracle dans une configuration de réplicas Oracle. Pour contourner ce problème, vous pouvez supprimer tous les réplicas, transporter les espaces de table, puis recréer les réplicas.

Prérequis pour les espaces de table transportables

Avant de commencer, effectuez les tâches suivantes :

- Passez en revue les exigences en matière d'espaces de table transportables, décrites dans les documents suivants figurant dans My Oracle Support :
  - Réduire les temps d'arrêt des espaces de table transportables en utilisant la sauvegarde incrémentielle multiplateforme (ID de document 2471245.1) (langue française non garantie)
  - Restrictions et limites relatives aux espaces de table transportables (TTS) : détails, référence et version, le cas échéant (ID de document 1454872.1) (langue française non garantie)
  - Note principale concernant les espaces de table transportables (TTS) -- Questions et problèmes courants (ID de document 1166564.1) (langue française non garantie)
- Prévoyez la conversion de l'endianisme. Si vous spécifiez l'identifiant de la plateforme source, RDS for Oracle convertit automatiquement l'endianisme. Pour savoir comment trouver une plateforme IDs, consultez Data Guard Support for Heterogenous Primary and Physical Standbys in Same Data Guard configuration (Doc ID 413484.1).
- Assurez-vous que la fonctionnalité des espaces de table transportables est activée sur votre instance de base de données cible. Cette fonctionnalité est activée uniquement si vous n'obtenez pas une erreur `ORA-20304` lorsque vous exécutez la requête suivante :

  ```
  SELECT * FROM TABLE(rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files);
  ```

  Si la fonctionnalité des espaces de table transportables n'est pas activée, redémarrez votre instance de base de données. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.
- Assurez-vous que le fichier de fuseau horaire est le même dans les bases de données source et cible.
- Assurez-vous que les jeux de caractères des bases de données source et cible répondent à l'une des exigences suivantes :
  - Les jeux de caractères sont les mêmes.
  - Les jeux de caractères sont compatibles. Pour obtenir la liste des exigences de compatibilité, consultez la section Limitations générales relatives au transport de données dans la documentation de la base de données Oracle.
- Si vous envisagez de transférer des fichiers à l'aide d'Amazon S3, procédez comme suit :

  - Assurez-vous qu'un compartiment Amazon S3 est disponible pour les transferts de fichiers et qu'il se trouve dans la même AWS région que votre instance de base de données. Pour plus d'informations, veuillez consulter Créer un compartiment dans le Guide de démarrage d'Amazon Simple Storage Service.
  - Préparez le compartiment Amazon S3 pour l'intégration d'Amazon RDS en suivant les instructions fournies dans Configuration des autorisations IAM pour l'intégration de RDS for Oracle à Amazon S3.
- Si vous envisagez de transférer des fichiers à l'aide d'Amazon EFS, assurez-vous d'avoir configuré EFS conformément aux instructions fournies dans EFSIntégration avec Amazon.
- Nous vous recommandons vivement d'activer les sauvegardes automatiques dans votre instance de base de données cible. Comme l'étape d'importation des métadonnées peut échouer, il est important de pouvoir restaurer votre instance de base de données dans son état d'avant l'importation, afin d'éviter d'avoir à sauvegarder, transférer et importer à nouveau vos espaces de table.

Phase 1 : Configuration de votre hôte source

Au cours de cette étape, vous copiez les scripts de transport des espaces de table fournis par My Oracle Support et configurez les fichiers de configuration nécessaires. Dans les étapes suivantes, l'hôte source exécute la base de données qui contient les espaces de table à transporter vers votre instance cible.

Pour configurer votre hôte source

1. Connectez-vous à votre hôte source en tant que propriétaire de votre répertoire de base Oracle.
2. Assurez-vous que vos variables d'environnement `ORACLE_HOME` et `ORACLE_SID` pointent vers votre base de données source.
3. Connectez-vous à votre base de données en tant qu'administrateur et vérifiez que la version de fuseau horaire, le jeu de caractères de base de données et le jeu de caractères national sont identiques à ceux de votre base de données cible.

   ```
   SELECT * FROM V$TIMEZONE_FILE;
   SELECT * FROM NLS_DATABASE_PARAMETERS
     WHERE PARAMETER IN ('NLS_CHARACTERSET','NLS_NCHAR_CHARACTERSET');
   ```

4. Configurez l'utilitaire d'espace de table transportable comme décrit dans la note de support Oracle 2471245.1 (langue française non garantie).

La configuration inclut la modification du fichier `xtt.properties` sur votre hôte source. L'exemple de fichier `xtt.properties` suivant spécifie les sauvegardes de trois espaces de table dans le répertoire `/dsk1/backups`. Il s'agit des espaces de table que vous souhaitez transporter vers votre instance de base de données cible. Il spécifie également l'identifiant de la plateforme source pour convertir automatiquement l'endianisme.

> **Note**
>
> Pour une plate-forme valide IDs, consultez Data Guard Support pour les systèmes de secours primaires et physiques hétérogènes dans la même configuration Data Guard (Doc ID 413484.1).

```
#linux system
platformid=13
#list of tablespaces to transport
tablespaces=TBS1,TBS2,TBS3
#location where backup will be generated
src_scratch_location=/dsk1/backups
#RMAN command for performing backup
usermantransport=1
```

Phase 2 : Préparation de la sauvegarde complète des espaces de table

Au cours de cette phase, vous sauvegardez vos espaces de table pour la première fois, vous transférez les sauvegardes vers votre hôte cible, puis vous les restaurez à l'aide de la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces`. Lorsque cette phase est terminée, les sauvegardes initiales des espaces de table résident sur votre instance de base de données cible et peuvent être mises à jour avec des sauvegardes incrémentielles.

Rubriques

- Étape 1 : Sauvegarder les espaces de table sur votre hôte source
- Étape 2 : Transférer les fichiers de sauvegarde vers votre instance de base de données cible
- Étape 3 : Importer les espaces de table dans votre instance de base de données cible

Étape 1 : Sauvegarder les espaces de table sur votre hôte source

Au cours de cette étape, vous utilisez le script `xttdriver.pl` pour effectuer une sauvegarde complète de vos espaces de table. La sortie de `xttdriver.pl` est stockée dans la variable d'environnement `TMPDIR`.

Pour sauvegarder vos espaces de table

1. Si vos espaces de table sont en mode lecture seule, connectez-vous à votre base de données source en tant qu'utilisateur disposant du privilège `ALTER TABLESPACE` et placez vos espaces de table en mode lecture/écriture. Sinon, passez à l'étape suivante.

   L'exemple suivant place `tbs1`, `tbs2` et `tbs3` en mode lecture/écriture.

   ```
   ALTER TABLESPACE tbs1 READ WRITE;
   ALTER TABLESPACE tbs2 READ WRITE;
   ALTER TABLESPACE tbs3 READ WRITE;
   ```

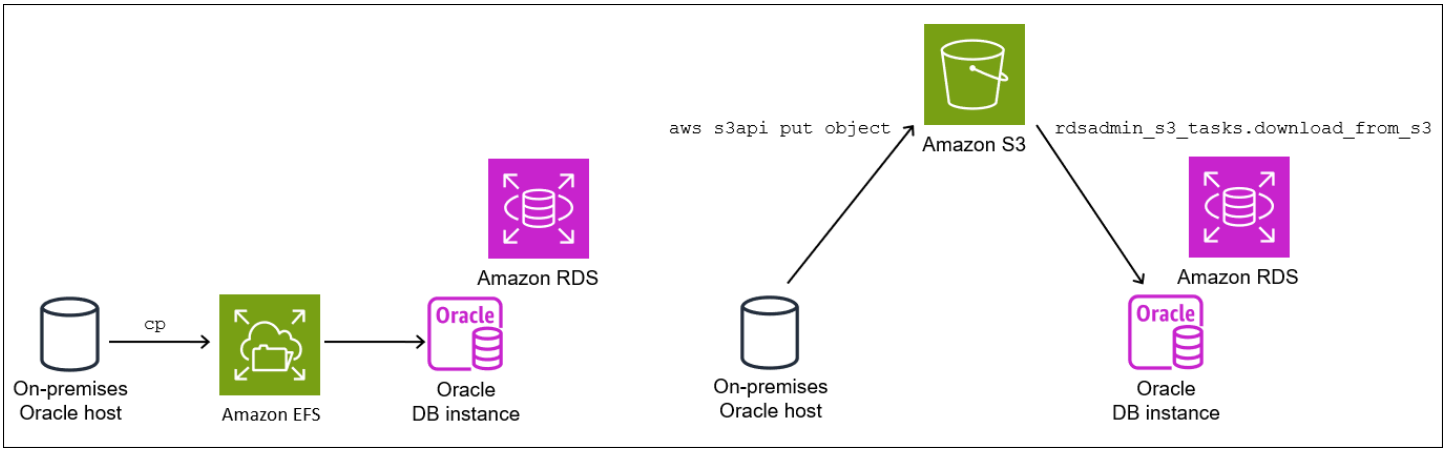
2. Sauvegardez vos espaces de table à l'aide du script `xttdriver.pl`. Vous pouvez éventuellement spécifier `--debug` pour exécuter le script en mode débogage.

   ```
   export TMPDIR=location_of_log_files
   cd location_of_xttdriver.pl
   $ORACLE_HOME/perl/bin/perl xttdriver.pl --backup
   ```

Étape 2 : Transférer les fichiers de sauvegarde vers votre instance de base de données cible

Au cours de cette étape, copiez les fichiers de sauvegarde et de configuration à partir de votre emplacement zéro vers votre instance de base de données cible. Choisissez l'une des options suivantes :

- Si les hôtes source et cible partagent un système de fichiers Amazon EFS, utilisez un utilitaire de système d'exploitation tel que `cp` pour copier vos fichiers de sauvegarde et le fichier `res.txt` à partir de votre emplacement zéro vers un répertoire partagé. Passez ensuite à Étape 3 : Importer les espaces de table dans votre instance de base de données cible.
- Si vous devez organiser vos sauvegardes vers un compartiment Amazon S3, procédez comme suit.



Étape 2.2 : Charger les sauvegardes dans votre compartiment Amazon S3

Chargez vos sauvegardes et le fichier `res.txt` depuis votre répertoire zéro vers votre compartiment Amazon S3. Pour plus d'informations, consultez Chargement d'objets dans le Guide de l'utilisateur Amazon Simple Storage Service.

Étape 2.3 : Télécharger les sauvegardes depuis votre compartiment Amazon S3 vers votre instance de base de données cible

Au cours de cette étape, vous utilisez la procédure `rdsadmin.rdsadmin_s3_tasks.download_from_s3` pour télécharger vos sauvegardes dans votre instance de base de données RDS for Oracle.

Pour télécharger vos sauvegardes depuis votre compartiment Amazon S3

1. Lancez SQL*Plus ou Oracle SQL Developer et connectez-vous à votre instance de base de données RDS for Oracle.
2. Téléchargez les sauvegardes depuis le compartiment Amazon S3 vers votre instance de base de données cible en utilisant la procédure Amazon RDS `rdsadmin.rdsadmin_s3_tasks.download_from_s3`. L'exemple suivant télécharge tous les fichiers d'un compartiment Amazon S3 nommé *`amzn-s3-demo-bucket`* vers le répertoire *`DATA_PUMP_DIR`*.

```
EXEC UTL_FILE.FREMOVE ('DATA_PUMP_DIR', 'res.txt');
SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
  p_bucket_name    =>  'amzn-s3-demo-bucket',
  p_directory_name =>  'DATA_PUMP_DIR')
AS TASK_ID FROM DUAL;
```

L'instruction SELECT renvoie l'ID de la tâche dans un type de données VARCHAR2. Pour de plus amples informations, veuillez consulter Téléchargement des fichiers d'un compartiment Amazon S3 vers une instance de base de données Oracle.

Étape 3 : Importer les espaces de table dans votre instance de base de données cible

Pour restaurer vos espaces de table sur votre instance de base de données cible, suivez la procédure. `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces` Cette procédure convertit automatiquement les fichiers de données au format endian approprié.

Si vous importez depuis une plate-forme autre que Linux, spécifiez la plate-forme source à l'aide du paramètre `p_platform_id` lorsque vous appelez`import_xtts_tablespaces`. Assurez-vous que l'identifiant de plate-forme que vous spécifiez correspond à celui indiqué dans le `xtt.properties` fichier dansÉtape 2 : Exporter les métadonnées d'espace de table sur votre hôte source.

Pour importer les espaces de table dans votre instance de base de données cible

1. Démarrez un client Oracle SQL et connectez-vous en tant qu'utilisateur principal à votre instance de base de données RDS for Oracle cible.
2. Exécutez la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces` en spécifiant les espaces de table à importer et le répertoire contenant les sauvegardes.

   L'exemple suivant importe les tablespaces *TBS1TBS2*, et *TBS3* depuis le répertoire. *DATA_PUMP_DIR* La plate-forme source est AIX Based Systems (64 bits), dont l'ID de 6 plate-forme est. Vous pouvez trouver la plateforme IDs en `V$TRANSPORTABLE_PLATFORM` interrogeant.

```
VAR task_id CLOB

BEGIN
  :task_id:=rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces(
        'TBS1,TBS2,TBS3',
        'DATA_PUMP_DIR',
        p_platform_id => 6);
END;
/
```

```
PRINT task_id
```

3. (Facultatif) Surveillez la progression en interrogeant la table `rdsadmin.rds_xtts_operation_info`. La colonne `xtts_operation_state` indique la valeur `EXECUTING`, `COMPLETED` ou `FAILED`.

```
SELECT * FROM rdsadmin.rds_xtts_operation_info;
```

> **Note**
>
> Pour les opérations de longue durée, vous pouvez également interroger `V$SESSION_LONGOPS`, `V$RMAN_STATUS` et `V$RMAN_OUTPUT`.

4. Consultez le journal de l'importation terminée en utilisant l'ID de tâche issu de l'étape précédente.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||'&task_id'||'.log'));
```

Assurez-vous de la réussite de l'importation avant de passer à l'étape suivante.

Phase 3 : Création et transfert de sauvegardes incrémentielles

Dans cette phase, vous effectuez et transférez régulièrement des sauvegardes incrémentielles pendant que la base de données source est active. Cette technique réduit la taille de la sauvegarde finale de vos espaces de table. Si vous effectuez plusieurs sauvegardes incrémentielles, vous devez copier le fichier `res.txt` après la dernière sauvegarde incrémentielle afin de pouvoir l'appliquer à l'instance cible.

Les étapes sont les mêmes que dans Phase 2 : Préparation de la sauvegarde complète des espaces de table, si ce n'est que l'étape d'importation est facultative.

Phase 4 : Transport des espaces de table

Dans cette phase, vous sauvegardez vos espaces de table en lecture seule et exportez les métadonnées Data Pump, transférez ces fichiers vers votre hôte cible et importez à la fois les espaces de table et les métadonnées.

Rubriques

- Étape 1 : Sauvegarder vos espaces de table en lecture seule
- Étape 2 : Exporter les métadonnées d'espace de table sur votre hôte source
- Étape 3 : (Amazon S3 uniquement) Transférer les fichiers de sauvegarde et d'exportation vers votre instance de base de données cible
- Étape 4 : Importer les espaces de table dans votre instance de base de données cible
- Étape 5 : Importer les métadonnées d'espace de table dans votre instance de base de données cible

Étape 1 : Sauvegarder vos espaces de table en lecture seule

Cette étape est identique à Étape 1 : Sauvegarder les espaces de table sur votre hôte source, mais présente une différence clé : vous placez vos espaces de table en mode lecture seule avant de les sauvegarder pour la dernière fois.

L'exemple suivant place `tbs1`, `tbs2` et `tbs3` en mode lecture seule.

```
ALTER TABLESPACE tbs1 READ ONLY;
ALTER TABLESPACE tbs2 READ ONLY;
ALTER TABLESPACE tbs3 READ ONLY;
```

Étape 2 : Exporter les métadonnées d'espace de table sur votre hôte source

Exportez les métadonnées de vos espaces de table en exécutant l'utilitaire `expdb` sur votre hôte source. L'exemple suivant exporte des tablespaces *TBS1TBS2*, et *TBS3* vers un fichier de sauvegarde *xttdump.dmp* dans un répertoire. *DATA_PUMP_DIR*

```
expdp username/pwd \
dumpfile=xttdump.dmp \
directory=DATA_PUMP_DIR \
statistics=NONE \
transport_tablespaces=TBS1,TBS2,TBS3 \
transport_full_check=y \
logfile=tts_export.log
```

S'il s'*DATA_PUMP_DIR*agit d'un répertoire partagé dans Amazon EFS, passez directement àÉtape 4 : Importer les espaces de table dans votre instance de base de données cible.

Étape 3 : (Amazon S3 uniquement) Transférer les fichiers de sauvegarde et d'exportation vers votre instance de base de données cible

Si vous utilisez Amazon S3 pour organiser vos sauvegardes d'espaces de table et votre fichier d'exportation Data Pump, procédez comme suit.

Étape 3.1 : Charger les sauvegardes et le fichier de vidage depuis votre hôte source dans votre compartiment Amazon S3

Chargez vos fichiers de sauvegarde et de vidage depuis votre hôte source dans votre compartiment Amazon S3. Pour plus d'informations, consultez Chargement d'objets dans le Guide de l'utilisateur Amazon Simple Storage Service.

Étape 3.2 : Télécharger les sauvegardes et le fichier de vidage depuis votre compartiment Amazon S3 vers votre instance de base de données cible

Au cours de cette étape, vous utilisez la procédure `rdsadmin.rdsadmin_s3_tasks.download_from_s3` pour télécharger vos sauvegardes et le fichier de vidage vers votre instance de base de données RDS for Oracle. Suivez les étapes de Étape 2.3 : Télécharger les sauvegardes depuis votre compartiment Amazon S3 vers votre instance de base de données cible.

Étape 4 : Importer les espaces de table dans votre instance de base de données cible

Utilisez la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces` pour restaurer les espaces de table. Pour la syntaxe et la sémantique de cette procédure, consultez Importation des espaces de table transportés dans votre instance de base de données

> **Important**
>
> Une fois l'importation finale de vos espaces de table terminée, l'étape suivante consiste à importer les métadonnées Oracle Data Pump. Si l'importation échoue, il est important de rétablir l'état de votre instance de base de données avant l'échec. Nous vous recommandons donc de créer un instantané de base de données de votre instance de base de données en suivant les instructions fournies dans Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS. Cet instantané contiendra tous les espaces de table importés. Ainsi, si l'importation échoue, vous n'avez pas besoin de répéter le processus de sauvegarde et d'importation.
> Si les sauvegardes automatiques de votre instance de base de données cible sont activées et qu'Amazon RDS ne détecte pas qu'un instantané valide a été lancé avant que vous

importiez les métadonnées, RDS tente de créer un instantané. En fonction de l'activité de votre instance, cet instantané peut réussir ou non. Si aucun instantané valide n'est détecté ou si un instantané ne peut pas être lancé, l'importation des métadonnées se termine avec des erreurs.

Pour importer les espaces de table dans votre instance de base de données cible

1. Démarrez un client Oracle SQL et connectez-vous en tant qu'utilisateur principal à votre instance de base de données RDS for Oracle cible.
2. Exécutez la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces` en spécifiant les espaces de table à importer et le répertoire contenant les sauvegardes.

   L'exemple suivant importe les tablespaces *TBS1TBS2*, et *TBS3* depuis le répertoire. *DATA_PUMP_DIR*

   ```
   BEGIN

    :task_id:=rdsadmin.rdsadmin_transport_util.import_xtts_tablespaces('TBS1,TBS2,TBS3','DATA_
   END;
   /
   PRINT task_id
   ```

3. (Facultatif) Surveillez la progression en interrogeant la table `rdsadmin.rds_xtts_operation_info`. La colonne `xtts_operation_state` indique la valeur `EXECUTING`, `COMPLETED` ou `FAILED`.

   ```
   SELECT * FROM rdsadmin.rds_xtts_operation_info;
   ```

   > **Note**
   >
   > Pour les opérations de longue durée, vous pouvez également interroger `V$SESSION_LONGOPS`, `V$RMAN_STATUS` et `V$RMAN_OUTPUT`.

4. Consultez le journal de l'importation terminée en utilisant l'ID de tâche issu de l'étape précédente.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file('BDUMP',
 'dbtask-'||'&task_id'||'.log'));
```

Assurez-vous de la réussite de l'importation avant de passer à l'étape suivante.

5. Prenez un instantané de base de données manuel en suivant les instructions fournies dans Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Étape 5 : Importer les métadonnées d'espace de table dans votre instance de base de données cible

Au cours de cette étape, vous importez les métadonnées d'espace de table transportable dans votre instance de base de données RDS for Oracle en utilisant la procédure `rdsadmin.rdsadmin_transport_util.import_xtts_metadata`. Pour la syntaxe et la sémantique de cette procédure, consultez Importation des métadonnées d'espaces de table transportables dans votre instance de base de données. Pendant l'opération, le statut de l'importation est indiqué dans la table `rdsadmin.rds_xtts_operation_info`.

> **Important**
>
> Avant d'importer les métadonnées, nous vous recommandons vivement de confirmer qu'un instantané de base de données a bien été créé après l'importation de vos espaces de table. Si l'étape d'importation échoue, restaurez votre instance de base de données, corrigez les erreurs d'importation, puis retentez l'importation.

Pour importer les métadonnées Data Pump dans votre instance de base de données RDS for Oracle

1. Démarrez votre client Oracle SQL et connectez-vous en tant qu'utilisateur principal à votre instance de base de données cible.
2. Créez les utilisateurs propriétaires des schémas dans vos espaces de table transportés, si ces utilisateurs n'existent pas encore.

```
CREATE USER tbs_owner IDENTIFIED BY password;
```

3. Importez les métadonnées en spécifiant le nom du fichier de vidage et l'emplacement du répertoire.

```
BEGIN

 rdsadmin.rdsadmin_transport_util.import_xtts_metadata('xttdump.dmp','DATA_PUMP_DIR');
END;
/
```

4. (Facultatif) Interrogez la table d'historique des espaces de table transportables pour voir le statut de l'importation des métadonnées.

   ```
   SELECT * FROM rdsadmin.rds_xtts_operation_info;
   ```

   Une fois l'opération terminée, vos espaces de table sont en mode lecture seule.

5. (Facultatif) Affichez le fichier journal.

   L'exemple suivant répertorie le contenu du répertoire BDUMP, puis interroge le journal d'importation.

   ```
   SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir(p_directory => 'BDUMP'));

   SELECT * FROM TABLE(rdsadmin.rds_file_util.read_text_file(
     p_directory => 'BDUMP',
     p_filename => 'rds-xtts-
   import_xtts_metadata-2023-05-22.01-52-35.560858000.log'));
   ```

Phase 5 : Validation des espaces de table transportés

Au cours de cette étape facultative, vous validez vos espaces de table transportés à l'aide de la procédure`rdsadmin.rdsadmin_rman_util.validate_tablespace`, puis vous placez vos espaces de table en mode lecture/écriture.

Pour valider les données transportées

1. Lancez SQL*Plus ou SQL Developer et connectez-vous en tant qu'utilisateur principal à votre instance de base de données cible.
2. Validez les espaces de table à l'aide de la procédure `rdsadmin.rdsadmin_rman_util.validate_tablespace`.

   ```
   SET SERVEROUTPUT ON
   ```

```
BEGIN
    rdsadmin.rdsadmin_rman_util.validate_tablespace(
        p_tablespace_name     => 'TBS1',
        p_validation_type     => 'PHYSICAL+LOGICAL',
        p_rman_to_dbms_output => TRUE);
    rdsadmin.rdsadmin_rman_util.validate_tablespace(
        p_tablespace_name     => 'TBS2',
        p_validation_type     => 'PHYSICAL+LOGICAL',
        p_rman_to_dbms_output => TRUE);
    rdsadmin.rdsadmin_rman_util.validate_tablespace(
        p_tablespace_name     => 'TBS3',
        p_validation_type     => 'PHYSICAL+LOGICAL',
        p_rman_to_dbms_output => TRUE);
END;
/
```

3. Placez vos espaces de table en mode lecture/écriture.

```
ALTER TABLESPACE TBS1 READ WRITE;
ALTER TABLESPACE TBS2 READ WRITE;
ALTER TABLESPACE TBS3 READ WRITE;
```

Phase 6 : Nettoyage des fichiers restants

Au cours de cette étape facultative, vous supprimez tous les fichiers inutiles. Utilisez la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files` pour répertorier les fichiers de données devenus orphelins après une importation d'espace de table, puis utilisez la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files` pour les supprimer. Pour la syntaxe et la sémantique de ces procédures, consultez Établissement de la liste des fichiers orphelins après une importation d'espace de table et Suppression des fichiers de données devenus orphelins après une importation d'espace de table.

Pour nettoyer les fichiers restants

1. Supprimez les anciennes sauvegardes *DATA_PUMP_DIR* comme suit :

    a. Répertoriez les fichiers de sauvegarde en exécutant `rdsadmin.rdsadmin_file_util.listdir`.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir(p_directory =>
 'DATA_PUMP_DIR'));
```

b. Supprimez les sauvegardes une par une en appelant `UTL_FILE.FREMOVE`.

```
EXEC UTL_FILE.FREMOVE ('DATA_PUMP_DIR', 'backup_filename');
```

2. Si vous avez importé des espaces de table mais n'avez pas importé de métadonnées pour ces espaces de table, vous pouvez supprimer les fichiers de données orphelins comme suit :

a. Répertoriez les fichiers de données orphelins que vous devez supprimer. L'exemple suivant exécute la procédure `rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files`.

```
SQL> SELECT * FROM
 TABLE(rdsadmin.rdsadmin_transport_util.list_xtts_orphan_files);

FILENAME       FILESIZE
-------------- ---------
datafile_7.dbf 104865792
datafile_8.dbf 104865792
```

b. Supprimez les fichiers orphelins en exécutant la procédure `rdsadmin.rdsadmin_transport_util.cleanup_incomplete_xtts_import`.

```
BEGIN

 rdsadmin.rdsadmin_transport_util.cleanup_incomplete_xtts_import('DATA_PUMP_DIR');
END;
/
```

L'opération de nettoyage génère un fichier journal qui utilise le format de nom `rds-xtts-delete_xtts_orphaned_files-`*`YYYY-MM-DD.HH24-MI-SS.FF`*`.log` dans le répertoire BDUMP.

c. Lisez le fichier journal généré à l'étape précédente. L'exemple suivant lit le journal `rds-xtts-delete_xtts_orphaned_files-`*`2023-06-01.09-33-11.868894000`*`.log`.

```
SELECT *
FROM TABLE(rdsadmin.rds_file_util.read_text_file(
```

```
       p_directory => 'BDUMP',
       p_filename  => 'rds-xtts-
delete_xtts_orphaned_files-2023-06-01.09-33-11.868894000.log'));

TEXT
--------------------------------------------------------------------------------
orphan transported datafile datafile_7.dbf deleted.
orphan transported datafile datafile_8.dbf deleted.
```

3. Si vous avez importé des espaces de table et importé des métadonnées pour ces espaces de table, mais que vous avez rencontré des erreurs de compatibilité ou d'autres problèmes liés à Oracle Data Pump, nettoyez les fichiers de données partiellement transportés comme suit :

   a. Répertoriez les espaces de table qui contiennent des fichiers de données partiellement transportés en interrogeant `DBA_TABLESPACES`.

   ```
   SQL> SELECT TABLESPACE_NAME FROM DBA_TABLESPACES WHERE PLUGGED_IN='YES';

   TABLESPACE_NAME
   --------------------------------------------------------------------------------
   TBS_3
   ```

   b. Supprimez les espaces de table et les fichiers de données partiellement transportés.

   ```
   DROP TABLESPACE TBS_3 INCLUDING CONTENTS AND DATAFILES;
   ```

Importation à l'aide d'Oracle Data Pump

Oracle Data Pump est un utilitaire qui vous permet d'exporter des données Oracle vers un fichier de vidage et de les importer dans une autre base de données Oracle. Il s'agit d'un remplacement à long terme des utilitaires d'importation/importation d'Oracle. Oracle Data Pump est le moyen recommandé pour déplacer de grandes quantités de données d'une base de données Oracle vers une instance de base de données Amazon RDS.

Les exemples de cette section montrent une façon d'importer des données dans une base de données Oracle, mais Oracle Data Pump prend en charge d'autres techniques. Pour plus d'informations, consultez la documentation Oracle Database.

Les exemples de cette section utilisent le package `DBMS_DATAPUMP`. Vous pouvez accomplir les mêmes tâches en utilisant les utilitaires de ligne de commande Oracle Data Pump `impdp` et `expdp`.

Vous pouvez installer ces utilitaires sur un hôte distant dans le cadre d'une installation de client Oracle, y compris Oracle Instant Client. Pour obtenir plus d'informations, consultez la section How do I use Oracle Instant Client to run Data Pump Import or Export for my Amazon RDS for Oracle DB instance? (Comment utiliser Oracle Instant Client pour exécuter Data Pump Import ou Export pour mon instance de base de données Amazon RDS for Oracle ?

Rubriques

- Présentation d'Oracle Data Pump
- Importation de données avec Oracle Data Pump et un compartiment Amazon S3
- Importation de données avec Oracle Data Pump et un lien de base de données

Présentation d'Oracle Data Pump

Oracle Data Pump est constitué des composants suivants :

- Clients de la ligne de commande `expdp` et `impdp`
- Le package PL/SQL `DBMS_DATAPUMP`
- Le package PL/SQL `DBMS_METADATA`

Vous pouvez utiliser Oracle Data Pump dans les scénarios suivants :

- Importez des données d'une base de données Oracle, sur site ou sur une EC2 instance Amazon, vers une instance de base de données RDS pour Oracle.
- Importez des données d'une instance de base de données RDS pour Oracle vers une base de données Oracle, sur site ou sur une instance Amazon EC2 .
- Importez des données entre des instances de base de données RDS pour Oracle, par exemple, pour migrer des données de EC2 -Classic vers VPC.

Pour télécharger les utilitaires Oracle Data Pump, veuillez consulter Oracle Database Software Downloads sur le site web Oracle Technology Network. Pour en savoir plus sur la compatibilité lors de la migration entre les versions d'Oracle Database, veuillez consulter la documentation Oracle Database.

Flux de travail Oracle Data Pump

En général, vous utilisez Oracle Data Pump pour les opérations suivantes :

1. Exportez vos données dans un fichier de vidage sur la base de données source.
2. Chargez votre fichier de vidage sur votre instance de base de données RDS for Oracle de destination. Vous pouvez effectuer le transfert à l'aide d'un compartiment Amazon S3 ou en utilisant un lien de base de données entre les deux bases de données.
3. Importez les données de votre fichier de vidage dans votre instance de base de données RDS for Oracle.

Bonnes pratiques d'Oracle Data Pump

Lorsque vous utilisez Oracle Data Pump pour importer des données dans une instance RDS for Oracle, nous vous recommandons de suivre les bonnes pratiques suivantes :

- Effectuez les importations en mode `schema` ou `table` pour importer des schémas et des objets spécifiques.
- Limitez les schémas que vous importez à ceux requis par votre application.
- N'importez pas en mode `full` ou importez des schémas pour les composants maintenus par le système.

  Comme RDS for Oracle n'autorise pas l'accès aux utilisateurs administratifs `SYS` ou `SYSDBA`, ces actions peuvent endommager le dictionnaire de données Oracle et affecter la stabilité de votre base de données.
- Lorsque vous chargez de grandes quantités de données, procédez comme suit :
  1. Transférez le fichier de vidage vers l'instance de base de données RDS for Oracle cible.
  2. Prenez un instantané de base de données de votre instance.
  3. Testez l'importation pour en vérifier le bon fonctionnement.

  Si les composants de la base de données sont invalidés, vous pouvez supprimer l'instance de base de données et la recréer à partir de l'instantané de base de données. L'instance de base de données restaurée inclut les fichiers de vidage intermédiaires sur l'instance de base de données lorsque vous avez pris l'instantané de base de données.
- N'importez pas de fichiers de vidage qui ont été créés à l'aide des paramètres d'exportation d'Oracle Data Pump `TRANSPORT_TABLESPACES`, `TRANSPORTABLE` ou `TRANSPORT_FULL_CHECK`. Les instances de base de données RDS for Oracle ne prennent pas en charge l'importation de ces fichiers de vidage.
- N'importez pas de fichiers de vidage contenant des objets Oracle Scheduler dans `SYS`, `SYSTEM`, `RDSADMIN`, `RDSSEC`, et `RDS_DATAGUARD` et appartenant aux catégories suivantes :

- Tâches
- Programmes
- Schedules
- Chaînes
- Règles
- Contextes d'évaluation
- Ensemble de règles

Les instances de base de données RDS for Oracle ne prennent pas en charge l'importation de ces fichiers de vidage.

- Pour exclure les objets Oracle Scheduler non pris en charge, utilisez des directives supplémentaires lors de l'exportation Data Pump. Si vous utilisez `DBMS_DATAPUMP`, vous pouvez ajouter un `METADATA_FILTER` supplémentaire avant le `DBMS_METADATA.START_JOB` :

```
DBMS_DATAPUMP.METADATA_FILTER(
  v_hdnl,
  'EXCLUDE_NAME_EXPR',
  q'[IN (SELECT NAME FROM SYS.OBJ$
         WHERE TYPE# IN (66,67,74,79,59,62,46)
         AND OWNER# IN
           (SELECT USER# FROM SYS.USER$
            WHERE NAME IN ('RDSADMIN','SYS','SYSTEM','RDS_DATAGUARD','RDSSEC')
            )
        )
  ]',
  'PROCOBJ'
);
```

Si vous utilisez `expdp`, créez un fichier de paramètres contenant la directive `exclude` indiquée dans l'exemple suivant. Ensuite, utilisez `PARFILE=`*`parameter_file`* avec votre commande `expdp`.

```
exclude=procobj:"IN
  (SELECT NAME FROM sys.OBJ$
   WHERE TYPE# IN (66,67,74,79,59,62,46)
   AND OWNER# IN
     (SELECT USER# FROM SYS.USER$
      WHERE NAME IN ('RDSADMIN','SYS','SYSTEM','RDS_DATAGUARD','RDSSEC')
```

```
    )
)"
```

Importation de données avec Oracle Data Pump et un compartiment Amazon S3

Le processus d'importation suivant utilise Oracle Data Pump et un compartiment Amazon S3. La procédure est la suivante :

1. Exportez les données de la base de données source à l'aide du package Oracle DBMS_DATAPUMP.
2. Placez le fichier de vidage dans un compartiment Amazon S3.
3. Téléchargez le fichier de vidage depuis le compartiment Amazon S3 vers le répertoire `DATA_PUMP_DIR` de l'instance de base de données RDS for Oracle cible.
4. Importez les données du fichier de vidage copié dans l'instance de base de données RDS for Oracle à l'aide du package `DBMS_DATAPUMP`.

Rubriques

- Conditions requises pour importer des données avec Oracle Data Pump et un compartiment Amazon S3
- Étape 1 : accordez des privilèges à l'utilisateur de la base de données sur l'instance de base de données cible RDS for Oracle.
- Étape 2 : exporter les données dans un fichier de vidage en utilisant DBMS_DATAPUMP
- Étape 3 : Charger le fichier de vidage dans votre compartiment Amazon S3
- Étape 4 : téléchargez le fichier de vidage depuis votre compartiment Amazon S3 vers votre instance de base de données cible.
- Étape 5 : importez votre fichier de vidage dans votre instance de base de données cible en utilisant DBMS_DATAPUMP.
- Étape 6 : nettoyer

Conditions requises pour importer des données avec Oracle Data Pump et un compartiment Amazon S3

Le processus est soumis aux exigences suivantes :

- Assurez-vous qu'un compartiment Amazon S3 est disponible pour les transferts de fichiers et que le compartiment Amazon S3 se trouve dans le même emplacement Région AWS que l'instance de base de données. Pour plus d'informations, veuillez consulter Créer un compartiment dans le Guide de démarrage d'Amazon Simple Storage Service.
- L'objet que vous téléchargez dans le compartiment Amazon S3 doit être d'une taille inférieure ou égale à 5 To. Pour plus d'informations sur l'utilisation des objets dans Amazon S3, consultez le Guide de l'utilisateur Amazon Simple Storage Service.

> **Note**
>
> Si le fichier de vidage dépasse 5 To, vous pouvez exécuter l'exportation Oracle Data Pump avec l'option parallèle. Cette opération répartit les données dans plusieurs fichiers de vidage de sorte que la taille de chaque fichier ne dépasse pas la limite de 5 To.

- Vous devez préparer le compartiment Amazon S3 pour l'intégration Amazon RDS en suivant les instructions de Configuration des autorisations IAM pour l'intégration de RDS for Oracle à Amazon S3.
- Vous devez veiller à disposer de suffisamment d'espace de stockage pour stocker le fichier de vidage sur l'instance source et l'instance de base de données cible.

> **Note**
>
> Ce processus importe un fichier de vidage dans le répertoire `DATA_PUMP_DIR`, qui est préconfiguré sur toutes les instances de bases de données Oracle. Ce répertoire est situé sur le même volume de stockage que vos fichiers de données. Lorsque vous importez le fichier de vidage, les fichiers de données Oracle existants utilisent davantage d'espace. Vous devez donc veiller à ce que votre instance de base de données puisse répondre aux besoins de cette utilisation d'espace supplémentaire. Le fichier de vidage importé n'est pas automatiquement supprimé ou purgé du répertoire `DATA_PUMP_DIR`. Pour supprimer le fichier de vidage importé, utilisez UTL_FILE.FREMOVE, disponible sur le site web d'Oracle.

Étape 1 : accordez des privilèges à l'utilisateur de la base de données sur l'instance de base de données cible RDS for Oracle.

Dans cette étape, vous créez les schémas dans lesquels vous prévoyez d'importer des données et vous accordez aux utilisateurs les privilèges nécessaires.

Pour créer des utilisateurs et accorder les privilèges nécessaires sur l'instance cible RDS for Oracle

1. Utilisez SQL*Plus ou Oracle SQL Developer pour vous connecter en tant qu'utilisateur principal à l'instance de la base de données RDS for Oracle dans laquelle les données seront importées. Pour en savoir plus sur la connexion à une instance de base de données, consultez Connexion à votre instance de base de données Oracle.
2. Créez les espaces de table requis avant d'importer les données. Pour de plus amples informations, veuillez consulter Création et dimensionnement des espaces de table.
3. Créez le compte utilisateur et accordez les autorisations et les rôles nécessaires si le compte utilisateur dans lequel les données sont importées n'existe pas. Si vous importez des données avec plusieurs schémas d'utilisateur, créez chaque compte d'utilisateur et accordez-lui les privilèges et rôles nécessaires.

   Par exemple, les instructions SQL suivantes créent un utilisateur et lui accordent les autorisations et rôles nécessaires pour importer les données dans le schéma de celui-ci : Remplacez *schema_1* par le nom de votre schéma dans cette étape et dans les étapes suivantes.

   ```
   CREATE USER schema_1 IDENTIFIED BY my_password;
   GRANT CREATE SESSION, RESOURCE TO schema_1;
   ALTER USER schema_1 QUOTA 100M ON users;
   ```

   > **Note**
   >
   > Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

   Les instructions précédentes accordent au nouvel utilisateur le privilège `CREATE SESSION` et le rôle `RESOURCE`. Vous pouvez avoir besoin de privilèges et de rôles supplémentaires en fonction des objets de la base de données que vous importez.

Étape 2 : exporter les données dans un fichier de vidage en utilisant DBMS_DATAPUMP

Pour créer un fichier de vidage, utilisez le package `DBMS_DATAPUMP`.

Pour exporter des données Oracle dans un fichier de vidage

1. Utilisez SQL Plus ou Oracle SQL Developer pour vous connecter à l'instance de base de données RDS for Oracle source avec un utilisateur administratif. Si la base de données source est une instance de base de données RDS for Oracle, connectez-vous avec l'utilisateur principal Amazon RDS.
2. Exportez les données en appelant des procédures `DBMS_DATAPUMP`.

   Le script suivant exporte le schéma *`SCHEMA_1`* dans un fichier de vidage nommé `sample.dmp` dans le répertoire `DATA_PUMP_DIR`. Remplacez *`SCHEMA_1`* par le nom du schéma que vous souhaitez exporter.

```
DECLARE
  v_hdnl NUMBER;
BEGIN
  v_hdnl := DBMS_DATAPUMP.OPEN(
    operation => 'EXPORT',
    job_mode  => 'SCHEMA',
    job_name  => null
  );
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl         ,
    filename  => 'sample.dmp'   ,
    directory => 'DATA_PUMP_DIR',
    filetype  => dbms_datapump.ku$_file_type_dump_file
  );
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample_exp.log',
    directory => 'DATA_PUMP_DIR' ,
    filetype  => dbms_datapump.ku$_file_type_log_file
  );
  DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'SCHEMA_EXPR','IN (''SCHEMA_1'')');
  DBMS_DATAPUMP.METADATA_FILTER(
    v_hdnl,
    'EXCLUDE_NAME_EXPR',
    q'[IN (SELECT NAME FROM SYS.OBJ$
           WHERE TYPE# IN (66,67,74,79,59,62,46)
           AND OWNER# IN
             (SELECT USER# FROM SYS.USER$
              WHERE NAME IN ('RDSADMIN','SYS','SYSTEM','RDS_DATAGUARD','RDSSEC')
             )
```

```
          )
    ]',
    'PROCOBJ'
  );
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

> **Note**
>
> Data Pump lance les tâches de manière asynchrone. Pour obtenir des informations sur la surveillance d'une tâche Data Pump, veuillez consulter Monitoring Job Status dans la documentation Oracle.

3. (Facultatif) Visualisez le contenu du journal d'exportation en appelant la procédure `rdsadmin.rds_file_util.read_text_file`. Pour de plus amples informations, veuillez consulter Lecture de fichiers dans un répertoire d'instance de base de données.

Étape 3 : Charger le fichier de vidage dans votre compartiment Amazon S3

Utilisez la procédure Amazon RDS `rdsadmin.rdsadmin_s3_tasks.upload_to_s3` pour copier le fichier de vidage dans le compartiment Amazon S3. L'exemple suivant charge tous les fichiers du répertoire `DATA_PUMP_DIR` dans le compartiment Amazon S3 nommé *`amzn-s3-demo-bucket`*.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
  p_bucket_name    =>  'amzn-s3-demo-bucket',
  p_directory_name =>  'DATA_PUMP_DIR')
AS TASK_ID FROM DUAL;
```

L'instruction `SELECT` renvoie l'ID de la tâche dans un type de données `VARCHAR2`. Pour de plus amples informations, veuillez consulter Téléchargement de fichiers depuis votre instance de base de données RDS pour Oracle vers un compartiment Amazon S3.

Étape 4 : téléchargez le fichier de vidage depuis votre compartiment Amazon S3 vers votre instance de base de données cible.

Effectuez cette étape en utilisant la procédure Amazon RDS `rdsadmin.rdsadmin_s3_tasks.download_from_s3`. Lorsque vous téléchargez un fichier dans un répertoire, la procédure `download_from_s3` ignore le téléchargement si un fichier de même

nom existe déjà dans le répertoire. Pour supprimer un fichier du répertoire de téléchargement, utilisez UTL_FILE.FREMOVE, que vous trouverez sur le site Web d'Oracle.

Pour télécharger votre fichier de vidage

1. Lancez SQL*Plus ou Oracle SQL Developer et connectez-vous en tant que maître sur votre instance de base de données Oracle cible Amazon RDS.
2. Téléchargez le fichier de vidage en utilisant la procédure Amazon RDS `rdsadmin.rdsadmin_s3_tasks.download_from_s3`.

   L'exemple suivant télécharge tous les fichiers d'un compartiment Amazon S3 nommé *amzn-s3-demo-bucket* dans le répertoire `DATA_PUMP_DIR`.

   ```
   SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
     p_bucket_name    =>  'amzn-s3-demo-bucket',
     p_directory_name =>  'DATA_PUMP_DIR')
   AS TASK_ID FROM DUAL;
   ```

   L'instruction `SELECT` renvoie l'ID de la tâche dans un type de données `VARCHAR2`. Pour de plus amples informations, veuillez consulter Téléchargement des fichiers d'un compartiment Amazon S3 vers une instance de base de données Oracle.

Étape 5 : importez votre fichier de vidage dans votre instance de base de données cible en utilisant DBMS_DATAPUMP.

Utilisez `DBMS_DATAPUMP` pour importer le schéma dans votre instance de base de données RDS for Oracle. Des options supplémentaires telles que `METADATA_REMAP` peuvent être nécessaires.

Pour importer des données dans votre instance de base de données cible

1. Lancez SQL*Plus ou SQL Developer et connectez-vous en tant qu'utilisateur principal à votre instance de base de données RDS for Oracle.
2. Importez les données en appelant `DBMS_DATAPUMP` des procédures.

   L'exemple suivant importe les *SCHEMA_1* données depuis votre `sample_copied.dmp` instance de base de données cible.

   ```
   DECLARE
     v_hdnl NUMBER;
   ```

```
BEGIN
  v_hdnl := DBMS_DATAPUMP.OPEN(
    operation => 'IMPORT',
    job_mode  => 'SCHEMA',
    job_name  => null);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample_copied.dmp',
    directory => 'DATA_PUMP_DIR',
    filetype  => dbms_datapump.ku$_file_type_dump_file);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample_imp.log',
    directory => 'DATA_PUMP_DIR',
    filetype  => dbms_datapump.ku$_file_type_log_file);
  DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'SCHEMA_EXPR','IN (''SCHEMA_1'')');
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

> **Note**
>
> Les tâches de vidage de données sont démarrées de façon asynchrone. Pour obtenir des informations sur la surveillance d'une tâche Data Pump, veuillez consulter Monitoring Job Status dans la documentation Oracle. Vous pouvez afficher le contenu du journal d'importation à l'aide de la procédure `rdsadmin.rds_file_util.read_text_file`. Pour de plus amples informations, veuillez consulter Lecture de fichiers dans un répertoire d'instance de base de données.

3. Vérifiez l'importation des données en listant les tables de schéma sur votre instance de base de données cible.

   Par exemple, la requête suivante renvoie le nombre de tables de *SCHEMA_1*.

   ```
   SELECT COUNT(*) FROM DBA_TABLES WHERE OWNER='SCHEMA_1';
   ```

Étape 6 : nettoyer

Après l'importation des données, vous pouvez supprimer les fichiers que vous ne souhaitez plus conserver.

Pour supprimer les fichiers inutiles

1. Lancez SQL*Plus ou SQL Developer et connectez-vous en tant qu'utilisateur principal à votre instance de base de données RDS for Oracle.
2. Listez les fichiers dans DATA_PUMP_DIR en utilisant la commande suivante.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir('DATA_PUMP_DIR')) ORDER BY
 MTIME;
```

3. Supprimez des fichiers dont vous n'avez plus besoin dans DATA_PUMP_DIR, utilisez la commande suivante :

```
EXEC UTL_FILE.FREMOVE('DATA_PUMP_DIR','filename');
```

Par exemple, la commande suivante supprime le fichier appelé sample_copied.dmp.

```
EXEC UTL_FILE.FREMOVE('DATA_PUMP_DIR','sample_copied.dmp');
```

Importation de données avec Oracle Data Pump et un lien de base de données

Le processus d'importation suivant utilise Oracle Data Pump et le package DBMS_FILE_TRANSFER. La procédure est la suivante :

1. Connectez-vous à une base de données Oracle source, qui peut être une base de données sur site, une EC2 instance Amazon ou une instance de base de données RDS pour Oracle.
2. Exportez des données à l'aide du package DBMS_DATAPUMP.
3. Utilisez DBMS_FILE_TRANSFER.PUT_FILE pour copier le fichier de vidage de la base de données Oracle dans le répertoire DATA_PUMP_DIR de l'instance de base de données RDS for Oracle cible qui est connectée à l'aide d'un lien de base de données.
4. Importez les données du fichier de vidage copié dans l'instance de base de données RDS for Oracle à l'aide du package DBMS_DATAPUMP.

Le processus d'importation utilisant Oracle Data Pump et le package DBMS_FILE_TRANSFER comporte les étapes suivantes :

Rubriques

- Conditions requises pour importer des données avec Oracle Data Pump et un lien vers une base de données
- Étape 1 : accorder des privilèges à l'utilisateur sur l'instance de base de données cible de RDS for Oracle
- Étape 2 : Accorder des privilèges à l'utilisateur sur la base de données source
- Étape 3 : créer un fichier de vidage en utilisant DBMS_DATAPUMP
- Étape 4 : créer un lien de base de données vers l'instance de base de données cible
- Étape 5 : copier le fichier de vidage exporté vers l'instance de base de données cible en utilisant DBMS_FILE_TRANSFER
- Étape 6 : importer le fichier de données vers l'instance de base de données cible en utilisant DBMS_DATAPUMP
- Étape 7 : nettoyer

Conditions requises pour importer des données avec Oracle Data Pump et un lien vers une base de données

Le processus est soumis aux exigences suivantes :

- Vous devez disposer des privilèges d'exécution sur les packages `DBMS_FILE_TRANSFER` et `DBMS_DATAPUMP`.
- Vous devez disposer de privilèges d'écriture sur le répertoire `DATA_PUMP_DIR` de l'instance de base de données source.
- Vous devez veiller à disposer de suffisamment d'espace de stockage pour stocker le fichier de vidage sur l'instance source et l'instance de base de données cible.

> **Note**
>
> Ce processus importe un fichier de vidage dans le répertoire `DATA_PUMP_DIR`, qui est préconfiguré sur toutes les instances de bases de données Oracle. Ce répertoire est situé sur le même volume de stockage que vos fichiers de données. Lorsque vous importez le fichier de vidage, les fichiers de données Oracle existants utilisent davantage d'espace. Vous devez donc veiller à ce que votre instance de base de données puisse répondre aux besoins de cette utilisation d'espace supplémentaire. Le fichier de vidage importé n'est pas

automatiquement supprimé ou purgé du répertoire `DATA_PUMP_DIR`. Pour supprimer le fichier de vidage importé, utilisez UTL_FILE.FREMOVE, disponible sur le site web d'Oracle.

Étape 1 : accorder des privilèges à l'utilisateur sur l'instance de base de données cible de RDS for Oracle

Pour accorder des privilèges à l'utilisateur sur l'instance de base de données cible RDS for Oracle, procédez comme suit :

1. Utilisez SQL Plus ou Oracle SQL Developer pour vous connecter à l'instance de la base de données RDS for Oracle dans laquelle vous souhaitez importer les données. Connectez-vous à l'utilisateur principal Amazon RDS. Pour plus d'informations sur la connexion à votre instance de base de données, consultez Connexion à votre instance de base de données Oracle.
2. Créez les espaces de table requis avant d'importer les données. Pour plus d'informations, consultez Création et dimensionnement des espaces de table.
3. Si le compte d'utilisateur dans lequel les données seront importées n'existe pas, créez-en un et accordez-lui les autorisations et rôles nécessaires. Si vous importez des données avec plusieurs schémas d'utilisateur, créez chaque compte d'utilisateur et accordez-lui les privilèges et rôles nécessaires.

   Par exemple, les commandes suivantes créent un nouvel utilisateur nommé *schema_1* et accordent les autorisations et les rôles nécessaires pour importer les données dans le schéma de cet utilisateur.

   ```
   CREATE USER schema_1 IDENTIFIED BY my-password;
   GRANT CREATE SESSION, RESOURCE TO schema_1;
   ALTER USER schema_1 QUOTA 100M ON users;
   ```

   > **Note**
   >
   > Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

   L'exemple précédent montre comment accorder au nouvel utilisateur le privilège `CREATE SESSION` et le rôle `RESOURCE`. Des privilèges et rôles supplémentaires peuvent être requis en fonction des objets de la base de données que vous allez importer.

> **Note**
>
> Remplacez *schema_1* par le nom de votre schéma dans cette étape et dans les étapes suivantes.

Étape 2 : Accorder des privilèges à l'utilisateur sur la base de données source

Utilisez SQL*Plus ou Oracle SQL Developer pour vous connecter à l'instance de la base de données RDS for Oracle qui contient les données à importer. Le cas échéant, créez un compte utilisateur et accordez les autorisations nécessaires.

> **Note**
>
> Si la base de données source est une instance Amazon RDS, vous pouvez ignorer cette étape. Vous allez utiliser votre compte d'utilisateur principal Amazon RDS for effectuer l'exportation.

Les commandes suivantes créent un nouvel utilisateur et accordent les autorisations nécessaires.

```
CREATE USER export_user IDENTIFIED BY my-password;
GRANT CREATE SESSION, CREATE TABLE, CREATE DATABASE LINK TO export_user;
ALTER USER export_user QUOTA 100M ON users;
GRANT READ, WRITE ON DIRECTORY data_pump_dir TO export_user;
GRANT SELECT_CATALOG_ROLE TO export_user;
GRANT EXECUTE ON DBMS_DATAPUMP TO export_user;
GRANT EXECUTE ON DBMS_FILE_TRANSFER TO export_user;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Étape 3 : créer un fichier de vidage en utilisant DBMS_DATAPUMP

Pour créer un fichier de vidage, procédez comme suit :

1. Utilisez SQL*Plus ou Oracle SQL Developer pour vous connecter à l'instance Oracle source avec un utilisateur administrateur ou avec l'utilisateur créé à l'étape 2. Si la base de données source est une instance de base de données Amazon RDS for Oracle, connectez-vous avec l'utilisateur principal Amazon RDS.
2. Créez un fichier de vidage à l'aide de l'utilitaire Oracle Data Pump.

   Le script suivant crée un fichier de vidage appelé sample.dmp dans le répertoire `DATA_PUMP_DIR`.

```
DECLARE
  v_hdnl NUMBER;
BEGIN
  v_hdnl := DBMS_DATAPUMP.OPEN(
    operation => 'EXPORT' ,
    job_mode  => 'SCHEMA' ,
    job_name  => null
  );
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample.dmp'    ,
    directory => 'DATA_PUMP_DIR' ,
    filetype  => dbms_datapump.ku$_file_type_dump_file
  );
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl           ,
    filename  => 'sample_exp.log' ,
    directory => 'DATA_PUMP_DIR'  ,
    filetype  => dbms_datapump.ku$_file_type_log_file
  );
  DBMS_DATAPUMP.METADATA_FILTER(
    v_hdnl              ,
    'SCHEMA_EXPR'       ,
    'IN (''SCHEMA_1'')'
  );
  DBMS_DATAPUMP.METADATA_FILTER(
    v_hdnl,
    'EXCLUDE_NAME_EXPR',
    q'[IN (SELECT NAME FROM sys.OBJ$
           WHERE TYPE# IN (66,67,74,79,59,62,46)
           AND OWNER# IN
             (SELECT USER# FROM SYS.USER$
              WHERE NAME IN ('RDSADMIN','SYS','SYSTEM','RDS_DATAGUARD','RDSSEC')
             )
```

```
          )
    ]',
    'PROCOBJ'
  );
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

> **Note**
>
> Les tâches de vidage de données sont démarrées de façon asynchrone. Pour obtenir des informations sur la surveillance d'une tâche Data Pump, veuillez consulter Monitoring Job Status dans la documentation Oracle. Vous pouvez afficher le contenu du journal d'exportation à l'aide de la procédure `rdsadmin.rds_file_util.read_text_file`. Pour plus d'informations, consultez Lecture de fichiers dans un répertoire d'instance de base de données.

Étape 4 : créer un lien de base de données vers l'instance de base de données cible

Créez un lien de base de données entre votre instance de base de données source et votre instance de base de données cible. Notez que votre instance Oracle locale doit avoir une connectivité réseau à l'instance de base de données pour créer un lien de base de données et transférer votre fichier de vidage.

Exécutez cette étape en étant connecté au même compte d'utilisateur qu'à l'étape précédente.

Si vous créez un lien de base de données entre deux instances de base de données au sein du même VPC ou peered VPCs, les deux instances de base de données doivent avoir une route valide entre elles. Le groupe de sécurité de chaque instance de base de données doit autoriser le trafic entrant dans l'autre instance de base de données et le trafic sortant de cette instance. Les règles entrantes et sortantes des groupes de sécurité peuvent faire référence à des groupes de sécurité à partir du même VPC ou d'un VPC appairé. Pour plus d'informations, consultez Ajustement des liens de base de données à utiliser avec des instances de base de données dans un VPC.

La commande suivante crée un lien de base de données appelé `to_rds` qui se connecte à l'utilisateur principal Amazon RDS au niveau de l'instance de base de données cible :

```
CREATE DATABASE LINK to_rds
  CONNECT TO <master_user_account> IDENTIFIED BY <password>
```

```
  USING '(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=<dns or ip address of remote db>)
         (PORT=<listener port>))(CONNECT_DATA=(SID=<remote SID>)))';
```

Étape 5 : copier le fichier de vidage exporté vers l'instance de base de données cible en utilisant DBMS_FILE_TRANSFER

Utilisez `DBMS_FILE_TRANSFER` pour copier le fichier de vidage depuis l'instance de base de données source vers l'instance de base de données cible. Le script suivant copie un fichier de vidage appelé sample.dmp depuis l'instance source vers un lien de base de données cible appelé to_rds (créé dans l'étape précédente) :

```
BEGIN
  DBMS_FILE_TRANSFER.PUT_FILE(
    source_directory_object       => 'DATA_PUMP_DIR',
    source_file_name              => 'sample.dmp',
    destination_directory_object  => 'DATA_PUMP_DIR',
    destination_file_name         => 'sample_copied.dmp',
    destination_database          => 'to_rds' );
END;
/
```

Étape 6 : importer le fichier de données vers l'instance de base de données cible en utilisant DBMS_DATAPUMP

Utilisez Oracle Data Pump pour importer le schéma dans l'instance de base de données. Notez que des options supplémentaires comme METADATA_REMAP pourraient être obligatoires.

Connectez-vous à l'instance de base de données avec le compte d'utilisateur principal Amazon RDS for effectuer l'importation.

```
DECLARE
  v_hdnl NUMBER;
BEGIN
  v_hdnl := DBMS_DATAPUMP.OPEN(
    operation => 'IMPORT',
    job_mode  => 'SCHEMA',
    job_name  => null);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample_copied.dmp',
    directory => 'DATA_PUMP_DIR',
    filetype  => dbms_datapump.ku$_file_type_dump_file );
```

```
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'sample_imp.log',
    directory => 'DATA_PUMP_DIR',
    filetype  => dbms_datapump.ku$_file_type_log_file);
  DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'SCHEMA_EXPR','IN (''SCHEMA_1'')');
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

> **Note**
>
> Les tâches de vidage de données sont démarrées de façon asynchrone. Pour obtenir des informations sur la surveillance d'une tâche Data Pump, veuillez consulter Monitoring Job Status dans la documentation Oracle. Vous pouvez afficher le contenu du journal d'importation à l'aide de la procédure `rdsadmin.rds_file_util.read_text_file`. Pour plus d'informations, consultez Lecture de fichiers dans un répertoire d'instance de base de données.

Vous pouvez vérifier l'importation des données en consultant les table de l'utilisateur sur l'instance de base de données. Par exemple, la requête suivante renvoie le nombre de tables de *schema_1*.

```
SELECT COUNT(*) FROM DBA_TABLES WHERE OWNER='SCHEMA_1';
```

Étape 7 : nettoyer

Après l'importation des données, vous pouvez supprimer les fichiers que vous ne souhaitez plus conserver. Vous pouvez répertorier les fichiers de DATA_PUMP_DIR à l'aide de la commande suivante.

```
SELECT * FROM TABLE(rdsadmin.rds_file_util.listdir('DATA_PUMP_DIR')) ORDER BY MTIME;
```

Pour supprimer des fichiers dont vous n'avez plus besoin dans DATA_PUMP_DIR, utilisez la commande suivante :

```
EXEC UTL_FILE.FREMOVE('DATA_PUMP_DIR','<file name>');
```

Par exemple, la commande suivante supprime le fichier appelé "sample_copied.dmp".

```
EXEC UTL_FILE.FREMOVE('DATA_PUMP_DIR','sample_copied.dmp');
```

Importation avec les utilitaires d’importation/importation d’Oracle

Vous pouvez envisager les utilitaires d’importation/importation d’Oracle pour les migrations dans les conditions suivantes :

- La taille de vos données est réduite.
- Les types de données tels que les nombres flottants binaires et doubles ne sont pas nécessaires.

Le processus d'importation crée les objets de schéma nécessaires. Ainsi, vous n'avez pas besoin d'exécuter un script pour créer les objets au préalable.

La méthode la plus simple pour installer les utilitaires d'exportation et d'importation d'Oracle est d'installer le client instantané Oracle. Pour télécharger le logiciel, accédez à https://www.oracle.com/database/technologies/instant-client.html. Pour la documentation, consultez Instant Client for SQL*Loader, Export, and Import (Instant Client pour SQL*Loader, export et importation) dans le manuel Oracle Database Utilities.

Pour exporter des tables et les importer ensuite

1. Exportez les tables depuis la base de données source en utilisant la commande `exp`.

   La commande suivante exporte les tables nommées `tab1`, `tab2` et `tab3`. Le fichier de vidage est `exp_file.dmp`.

   ```
   exp cust_dba@ORCL FILE=exp_file.dmp TABLES=(tab1,tab2,tab3) LOG=exp_file.log
   ```

   L'exportation crée un fichier de vidage binaire qui contient le schéma et les données pour les tables spécifiées.
2. Importez le schéma et les données dans une base de données cible à l'aide de la commande `imp`.

   La commande suivante importe les tables `tab1`, `tab2` et `tab3` depuis le fichier de vidage `exp_file.dmp`.

   ```
   imp cust_dba@targetdb FROMUSER=cust_schema TOUSER=cust_schema \
   TABLES=(tab1,tab2,tab3) FILE=exp_file.dmp LOG=imp_file.log
   ```

L'exportation et l'importation ont d'autres variantes qui pourraient être mieux adaptées à vos besoins. Consultez la documentation de la base de données Oracle pour plus de détails.

Importation avec Oracle SQL*Loader

Vous pouvez envisager Oracle SQL*Loader pour les grandes bases de données qui contiennent un nombre limité d'objets. Comme le processus d'exportation à partir d'une base de données source et de chargement dans une base de données cible est spécifique au schéma, l'exemple suivant crée les objets du schéma type, exporte à partir d'une source, puis charge les données dans une base de données cible.

La méthode la plus simple pour installer Oracle SQL*Loader est d'installer Oracle Instant Client. Pour télécharger le logiciel, accédez à https://www.oracle.com/database/technologies/instant-client.html. Pour la documentation, consultez Instant Client for SQL*Loader, Export, and Import (Instant Client pour SQL*Loader, export et importation) dans le manuel Oracle Database Utilities.

Pour importer des données avec Oracle SQL*Loader

1. Créez une table source type en utilisant l'instruction SQL suivante.

```
CREATE TABLE customer_0 TABLESPACE users
   AS (SELECT ROWNUM id, o.*
       FROM   ALL_OBJECTS o, ALL_OBJECTS x
       WHERE  ROWNUM <= 1000000);
```

2. Sur l'instance de base de données RDS for Oracle cible, créez une table de destination pour le chargement des données. La clause `WHERE 1=2` permet de copier la structure de `ALL_OBJECTS` sans copier aucune ligne.

```
CREATE TABLE customer_1 TABLESPACE users
  AS (SELECT 0 AS ID, OWNER, OBJECT_NAME, CREATED
      FROM   ALL_OBJECTS
      WHERE  1=2);
```

3. Exportez les données de la base de données source vers un fichier texte. L'exemple suivant utilise SQL*Plus. Pour vos données, vous pourrez avoir besoin de générer un script qui exporte tous les objets dans la base de données.

```
ALTER SESSION SET NLS_DATE_FORMAT = 'YYYY/MM/DD HH24:MI:SS'
```

```
SET LINESIZE 800 HEADING OFF FEEDBACK OFF ARRAY 5000 PAGESIZE 0
SPOOL customer_0.out
SET MARKUP HTML PREFORMAT ON
SET COLSEP ','

SELECT id, owner, object_name, created
FROM   customer_0;

SPOOL OFF
```

4. Créez un fichier de contrôle pour décrire les données. Vous devrez peut-être écrire un script pour réaliser cette étape.

```
cat << EOF > sqlldr_1.ctl
load data
infile customer_0.out
into table customer_1
APPEND
fields terminated by "," optionally enclosed by '"'
(
  id           POSITION(01:10)    INTEGER EXTERNAL,
  owner        POSITION(12:41)    CHAR,
  object_name  POSITION(43:72)    CHAR,
  created      POSITION(74:92)    date "YYYY/MM/DD HH24:MI:SS"
)
```

   Le cas échéant, copiez les fichiers générés par le code précédent vers une zone tampon comme une instance Amazon EC2.

5. Importez les données en utilisant SQL*Loader avec le nom d'utilisateur et le mot de passe appropriés pour la base de données cible.

```
sqlldr cust_dba@targetdb CONTROL=sqlldr_1.ctl BINDSIZE=10485760 READSIZE=10485760
 ROWS=1000
```

Migration avec les vues matérialisées d'Oracle

Pour migrer efficacement de grands jeux de données, vous pouvez utiliser la réplication de vues matérialisées Oracle. Avec la réplication, vous pouvez maintenir les tables cibles synchronisées avec les tables sources. Ainsi, vous pourrez passer à Amazon RDS ultérieurement, si nécessaire.

Avant de procéder à une migration à l'aide de vues matérialisées, assurez-vous que vous remplissez les conditions suivantes :

- Configurez l'accès de la base de données cible à la base de données source. Dans l'exemple suivant, les règles d'accès ont été activées sur la base de données source RDS pour permettre à la base de données cible d'Oracle de se connecter à la source via SQL *Net.
- Créez un lien de base de données entre l'instance RDS de base de données Oracle et la base de données source.

Pour migrer des données en utilisant des vues matérialisées

1. Créez un compte utilisateur sur les instances source et RDS cible Oracle qui peut s'authentifier avec le même mot de passe. L'exemple suivant crée un utilisateur nommé `dblink_user`.

```
CREATE USER dblink_user IDENTIFIED BY my-password
  DEFAULT TABLESPACE users
  TEMPORARY TABLESPACE temp;

GRANT CREATE SESSION TO dblink_user;

GRANT SELECT ANY TABLE TO dblink_user;

GRANT SELECT ANY DICTIONARY TO dblink_user;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

2. Créez un lien de base de données entre l'instance cible RDS pour Oracle et l'instance source à l'aide de votre nouvel utilisateur.

```
CREATE DATABASE LINK remote_site
  CONNECT TO dblink_user IDENTIFIED BY my-password
  USING '(description=(address=(protocol=tcp) (host=my-host)
    (port=my-listener-port)) (connect_data=(sid=my-source-db-sid)))';
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

3. Testez le lien :

```
SELECT * FROM V$INSTANCE@remote_site;
```

4. Créez un exemple de table avec une clé primaire et un journal des vues matérialisées sur l'instance source.

```
CREATE TABLE customer_0 TABLESPACE users
  AS (SELECT ROWNUM id, o.*
      FROM   ALL_OBJECTS o, ALL_OBJECTS x
      WHERE  ROWNUM <= 1000000);

ALTER TABLE customer_0 ADD CONSTRAINT pk_customer_0 PRIMARY KEY (id) USING INDEX;

CREATE MATERIALIZED VIEW LOG ON customer_0;
```

5. Sur la cible RDS de l'instance de base de données Oracle, créez une vue matérialisée.

```
CREATE MATERIALIZED VIEW customer_0
  BUILD IMMEDIATE REFRESH FAST
  AS (SELECT *
      FROM   cust_dba.customer_0@remote_site);
```

6. Sur la cible de l'instance RDS de base de données Oracle, actualisez la vue matérialisée.

```
EXEC DBMS_MVIEW.REFRESH('CUSTOMER_0', 'f');
```

7. Supprimez la vue matérialisée et ajoutez la clause `PRESERVE TABLE` pour conserver la table conteneur de la vue matérialisée et son contenu.

```
DROP MATERIALIZED VIEW customer_0 PRESERVE TABLE;
```

   La table conservée a le même nom que la vue matérialisée supprimée.

Utilisation de réplicas en lecture pour Amazon RDS for Oracle

Pour configurer la réplication entre les instances de base de données Oracle, vous pouvez créer des bases de données de réplicas. Pour obtenir une présentation des réplicas en lecture Amazon RDS, consultez Présentation des réplicas en lecture Amazon RDS. Pour obtenir un récapitulatif des différences entre les réplicas Oracle et les autres moteurs de base de données, consultez Différences entre les répliques de lecture pour les moteurs de base de données.

Rubriques

- Présentation des réplicas RDS for Oracle
- Exigences et considérations relatives aux réplicas RDS pour Oracle
- Préparation de la création d'un réplica Oracle
- Création d'un réplica RDS for Oracle en mode monté
- Modification du mode réplica RDS for Oracle
- Utilisation de RDS pour les sauvegardes de réplicas Oracle
- Exécution d'un basculement d'Oracle Data Guard
- Dépannage des réplicas RDS for Oracle

Présentation des réplicas RDS for Oracle

Un réplica Oracle de base de données est une copie physique de votre base de données primaire. Un réplica Oracle en lecture seule est appelé réplica en lecture. Un réplica Oracle en mode monté est appelé réplica monté. Oracle Database n'autorise pas les écritures dans un réplica, mais vous pouvez le promouvoir pour le rendre accessible en écriture. Le réplica en lecture promu a les données répliquées jusqu'au moment où la demande a été faite pour le promouvoir.

La vidéo suivante contient une présentation utile de la reprise après sinistre RDS for Oracle.

Pour plus d'informations, consultez les billets de blog Managed disaster recovery with Amazon RDS for Oracle cross-Region automated backups - Part 1 et Managed disaster recovery with Amazon RDS for Oracle cross-Region automated backups - Part 2.

Rubriques

- Réplicas en lecture seule et montés
- Réplicas en lecture de CDB
- Conservation du journal de reprise archivé

- Pannes pendant la réplication Oracle

Réplicas en lecture seule et montés

Lorsque vous créez ou modifiez un réplica Oracle, vous pouvez le placer dans l'un des modes suivants :

Lecture seule

Il s'agit de l'option par défaut. Active Data Guard transmet et applique les modifications de la base de données source à toutes les bases de données de réplicas en lecture.

Vous pouvez créer jusqu'à cinq réplicas en lecture à partir d'une seule instance de base de données source. Pour plus d'informations sur les réplicas en lecture qui s'appliquent à tous les moteurs de base de données, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données. Pour plus d'informations sur Oracle Data Guard, consultez Concepts et administration d'Oracle Data Guard dans la documentation Oracle.

Monté

Dans ce cas, la réplication utilise Oracle Data Guard, mais la base de données du réplica n'accepte pas les connexions utilisateur. L'utilisation principale des réplicas montés est la reprise après sinistre inter-région.

Un réplica monté ne peut pas servir de charge de travail en lecture seule. Le réplica monté supprime les fichiers de journalisation archivés après leur application, quelle que soit la stratégie de conservation des journaux archivés.

Vous pouvez créer une combinaison de réplicas de base de données montés et en lecture seule pour la même instance de base de données source. Vous pouvez changer un réplica en lecture seule en mode monté ou changer un réplica monté en mode lecture seule. Dans les deux cas, la base de données Oracle conserve le paramètre de conservation des journaux archivés.

Réplicas en lecture de CDB

RDS for Oracle prend en charge les réplicas en lecture Data Guard pour les bases de données CDB Oracle Database 19c et 21c seulement dans la configuration à locataire unique. Vous pouvez créer, gérer et promouvoir des réplicas en lecture dans une CDB, tout comme vous pouvez le faire dans une base de données non CDB. Les réplicas montés sont également pris en charge. Vous bénéficiez des avantages suivants :

- Reprise après sinistre gérée, haute disponibilité et accès en lecture seule à vos réplicas
- Possibilité de créer des répliques de lecture dans un autre format. Région AWS
- Intégration avec les API de réplication de lecture RDS existantes : CreateDB InstanceReadReplica, et PromoteReadReplicaSwitchoverReadReplica

Pour utiliser cette fonctionnalité, vous avez besoin d'une licence Active Data Guard et d'une licence Oracle Database Enterprise Edition pour les instances de base de données de réplica et principale. Il n'y a aucun coût supplémentaire lié à l'utilisation de l'architecture CDB. Vous ne payez que pour vos instances de base de données.

Pour plus d'informations sur les configurations à locataire unique et à locataires multiples de l'architecture CDB, consultez Présentation de RDS pour Oracle CDBs.

Conservation du journal de reprise archivé

Si une instance de base de données primaire ne possède aucun réplica en lecture entre régions, Amazon RDS for Oracle conserve un minimum de deux heures de journaux de reprise sur l'instance de base de données source. Cela est vrai quelle que soit la valeur définie pour `archivelog retention hours` dans `rdsadmin.rdsadmin_util.set_configuration`.

RDS purge les journaux de l'instance de base de données source au bout de deux heures ou à l'issue du délai de conservation des journaux d'archive défini, si celui-ci est plus long. RDS purge les journaux du réplica en lecture à l'issue du délai de conservation des journaux d'archive qui a été défini, uniquement s'ils ont été appliqués correctement à la base de données.

Dans certains cas, une instance de base de données primaire peut avoir un ou plusieurs réplicas en lecture entre régions. Dans ce cas, Amazon RDS for Oracle conserve les journaux de transaction sur l'instance de base de données source jusqu'à ce qu'ils aient été transmis et appliqués à tous les réplicas en lecture entre régions. Pour en savoir plus sur `rdsadmin.rdsadmin_util.set_configuration`, consultez Conservation des journaux redo archivés.

Pannes pendant la réplication Oracle

Lorsque vous créez un réplica en lecture, Amazon RDS prend un instantané de votre instance de base de données source et commence la réplication. L'instance de base de données source subit une très brève suspension des E/S lorsque l'opération de capture instantanée de base de données commence. La suspension des E/S dure généralement environ une seconde. Vous pouvez éviter la

suspension d'I/O si l'instance de base de données source est un déploiement multi-AZ, car dans ce cas l'instantané est pris à partir de l'instance de base de données secondaire.

L'instantané de base de données devient la réplique Oracle. Amazon RDS définit les paramètres et les autorisations nécessaires pour la base de données source et la réplique sans interruption de service. De même, si vous supprimez un réplica, aucune panne ne se produit.

Exigences et considérations relatives aux réplicas RDS pour Oracle

Avant de créer un réplica Oracle, familiarisez-vous avec les exigences et considérations suivantes.

Rubriques

- Exigences de version et de licence pour les réplicas RDS pour Oracle
- Limitations des groupes d'options pour les répliques RDS pour Oracle
- Considérations relatives à la sauvegarde et la restauration des réplicas RDS for Oracle
- Exigences et limites relatives à Oracle Data Guard pour les réplicas RDS for Oracle
- Considérations diverses relatives aux réplicas RDS for Oracle

Exigences de version et de licence pour les réplicas RDS pour Oracle

Avant de créer un réplica RDS pour Oracle, tenez compte des éléments suivants :

- Si le réplica est en mode lecture seule, assurez-vous que vous disposez d'une licence Active Data Guard. Si vous placez le réplica en mode monté, vous n'avez pas besoin d'une licence Active Data Guard. Seul le moteur Oracle DB prend en charge les réplicas montés.
- Les répliques Oracle ne sont prises en charge que pour Oracle Enterprise Edition (EE).
- Les répliques Oracle de non- ne CDBs sont prises en charge que pour les instances de base de données créées à l'aide d'instances non CDB exécutant Oracle Database 19c.
- Les répliques Oracle sont disponibles pour les instances de base de données exécutées uniquement sur des classes d'instances de base de données comportant au moins deux v. CPUs Une instance de base de données source ne peut pas utiliser la classe d'instance db.t3.small.
- La version du moteur de base de données Oracle de l'instance de base de données source et de toutes ses répliques doivent être identiques. Amazon RDS met à niveau les réplicas immédiatement après la mise à niveau de l'instance de base de données source, quelle que soit la fenêtre de maintenance d'un réplica. Pour les mises à niveau majeures de versions de réplicas inter-régions, Amazon RDS effectue automatiquement les opérations suivantes :

- Génère un groupe d'options pour la version cible
- Copie toutes les options et tous les paramètres d'option du groupe d'options d'origine vers le nouveau groupe d'options
- Associe le réplica en lecture inter-région mis à niveau au nouveau groupe d'options

Pour plus d'informations sur la mise à niveau de la version du moteur de base de données, veuillez consulter la section Mise à niveau du moteur de base de données RDS for Oracle.

Limitations des groupes d'options pour les répliques RDS pour Oracle

Lorsque vous utilisez des groupes d'options pour votre réplique RDS pour Oracle, tenez compte des points suivants :

- Vous ne pouvez pas utiliser un groupe d'options de réplication différent du groupe d'options d'instance de base de données source lorsque la source et le réplica se trouvent dans la même AWS région.

  Les modifications apportées au groupe d'options source ou à l'appartenance au groupe d'options source se propagent aux répliques Oracle. Ces modifications sont appliquées aux réplicas immédiatement après leur application à l'instance de base de données source, quelle que soit la fenêtre de maintenance du réplica. Pour plus d'informations sur les groupes d'options, consultez Utilisation de groupes d'options.
- Vous ne pouvez pas supprimer une réplique interrégionale RDS pour Oracle de son groupe d'options dédié, qui est automatiquement créé pour la réplique.
- Vous ne pouvez pas ajouter le groupe d'options dédié pour une réplique interrégionale RDS pour Oracle à une autre instance de base de données.
- Vous ne pouvez pas ajouter ou supprimer des options non répliquées d'un groupe d'options dédié pour une réplique interrégionale RDS for Oracle, à l'exception des options suivantes :
  - `NATIVE_NETWORK_ENCRYPTION`
  - `OEM`
  - `OEM_AGENT`
  - `SSL`

  Pour ajouter d'autres options à un réplica RDS for Oracle entre régions, ajoutez-les au groupe d'options de l'instance de base de données source. L'option est également installée sur tous les

réplicas de l'instance de base de données source. Pour les options sous licence, assurez-vous qu'il existe suffisamment de licences pour les réplicas.

Lorsque vous promouvez un réplica RDS for Oracle entre régions, le réplica promu se comporte de la même façon que d'autres instances de base de données Oracle, y compris pour la gestion de ses options. Vous pouvez promouvoir un réplica explicitement ou implicitement en supprimant son instance de base de données source.

Pour plus d'informations sur les groupes d'options, veuillez consulter Utilisation de groupes d'options.

- Vous ne pouvez pas ajouter `EFS_INTEGRATION` cette option à RDS pour les répliques interrégionales Oracle.

Considérations relatives à la sauvegarde et la restauration des réplicas RDS for Oracle

Avant de créer un réplica RDS pour Oracle, tenez compte des éléments suivants :

- Pour créer des instantanés des réplicas de RDS for Oracle ou activer les sauvegardes automatiques, veillez à définir manuellement la période de conservation des sauvegardes. Les sauvegardes automatiques ne sont pas activées par défaut.
- Lorsque vous restaurez une sauvegarde de réplica, vous rétablissez l'heure de la base de données, et non l'heure à laquelle la sauvegarde a été effectuée. L'heure de la base de données désigne la dernière heure de transaction appliquée des données dans la sauvegarde. La différence est importante car un réplica peut être en retard de plusieurs minutes ou heures par rapport à l'instance principale.

  Pour faire la distinction, utilisez la commande `describe-db-snapshots`. Comparez le paramètre `snapshotDatabaseTime`, qui correspond à l'heure de la base de données de la sauvegarde du réplica, et le champ `OriginalSnapshotCreateTime`, qui correspond à la dernière transaction appliquée sur la base de données principale.

Exigences et limites relatives à Oracle Data Guard pour les réplicas RDS for Oracle

Avant de créer un réplica RDS for Oracle, notez les exigences et limites suivantes :

- Si votre instance de base de données principale utilise la configuration à locataire unique de l'architecture mutualisée, tenez compte des points suivants :

  - Vous devez utiliser Oracle Database 19c ou version ultérieure avec la version Enterprise Edition.
  - Votre instance CDB principale doit se trouver dans un cycle de vie `ACTIVE`.
  - Vous ne pouvez pas convertir une instance principale non-CDB en instance CDB et convertir ses réplicas dans la même opération. Supprimez plutôt les réplicas non-CDB, convertissez l'instance de base de données principale en CDB, puis créez de nouveaux réplicas.
- Veillez à ce qu'un déclencheur de connexion sur une instance de base de données principale permette l'accès à l'utilisateur `RDS_DATAGUARD` et à tout utilisateur dont la valeur `AUTHENTICATED_IDENTITY` est `RDS_DATAGUARD` ou `rdsdb`. En outre, le déclencheur ne doit pas définir le schéma actuel pour l'utilisateur `RDS_DATAGUARD`.
- Pour éviter de bloquer les connexions du processus de l'agent Data Guard, n'activez pas les sessions restreintes. Pour plus d'informations sur les sessions restreintes, consultez Activation et désactivation de sessions restreintes.

Considérations diverses relatives aux réplicas RDS for Oracle

Avant de créer un réplica RDS pour Oracle, tenez compte des éléments suivants :

- Si votre instance de base de données est une source pour une ou plusieurs répliques entre régions, la base de données source conserve ses fichiers de journalisation archivés jusqu'à ce qu'ils soient appliqués à toutes les répliques entre régions. Les journaux redo archivés peuvent entraîner une augmentation de la consommation de stockage.
- Pour éviter d'interrompre l'automatisation RDS, les déclencheurs système doivent permettre à des utilisateurs spécifiques de se connecter à la base de données primaire et de réplication. Les déclencheurs système incluent les déclencheurs DDL, les déclencheurs de connexion et les déclencheurs de rôle de base de données. Nous vous recommandons d'ajouter du code à vos déclencheurs pour exclure les utilisateurs répertoriés dans l'exemple de code suivant :

```
-- Determine who the user is
SELECT SYS_CONTEXT('USERENV','AUTHENTICATED_IDENTITY') INTO CURRENT_USER FROM DUAL;
-- The following users should always be able to login to either the Primary or
 Replica
IF CURRENT_USER IN ('master_user', 'SYS', 'SYSTEM', 'RDS_DATAGUARD', 'rdsdb') THEN
RETURN;
END IF;
```

- Le suivi des modifications de bloc est pris en charge pour les réplicas en lecture seule, mais pas pour les réplicas montés. Vous pouvez convertir un réplica monté en réplica en lecture seule, puis

activer le suivi des modifications de bloc. Pour de plus amples informations, veuillez consulter Activation et désactivation du suivi des modifications de bloc.

- Vous ne pouvez pas créer de réplique Oracle Read lorsque la base de données source gère les informations d'identification de l'utilisateur principal avec Secrets Manager.

Préparation de la création d'un réplica Oracle

Avant de commencer à utiliser votre réplica, effectuez les tâches suivantes.

Rubriques

- Planification des sauvegardes automatiques
- Activation du mode de journalisation
- Modification de votre configuration de journalisation
- Définition du paramètre MAX_STRING_SIZE
- Planification des ressources de calcul et de stockage

Planification des sauvegardes automatiques

Avant qu'une instance de base de données puisse être utilisée comme instance de bases de données source, vous devez activer les sauvegardes automatiques sur l'instance de base de données source. Pour savoir comment effectuer cette procédure, veuillez consulter Activation des sauvegardes automatiques.

Activation du mode de journalisation

Nous vous recommandons d'activer le mode de journalisation forcée. En mode de journalisation forcée, la base de données Oracle écrit des enregistrements de journalisation même lorsque `NOLOGGING` est utilisée avec des instructions DDL (Data Definition Language).

Pour activer le mode de journalisation forcée

1. Connectez-vous à votre base de données Oracle à l'aide d'un outil client tel que SQL Developer.
2. Activez le mode de journalisation forcée en exécutant la procédure suivante.

```
exec rdsadmin.rdsadmin_util.force_logging(p_enable => true);
```

Pour plus d'informations sur cette procédure, veuillez consulter Configuration du mode FORCE LOGGING.

Modification de votre configuration de journalisation

Pour n journaux de restauration en ligne de taille m, RDS crée automatiquement n +1 journaux de secours de taille m sur l'instance de base de données principale et sur toutes les répliques. Chaque fois que vous modifiez la configuration de journalisation sur le serveur principal, les modifications se propagent automatiquement aux répliques.

Si vous modifiez la configuration de journalisation, tenez compte des directives suivantes :

- Nous vous recommandons d'effectuer les modifications avant de faire d'une instance de base de données la source des répliques. RDS pour Oracle prend également en charge la mise à jour de l'instance une fois qu'elle est devenue une source.
- Avant de modifier la configuration de journalisation sur l'instance de base de données principale, vérifiez que chaque réplique dispose d'un espace de stockage suffisant pour s'adapter à la nouvelle configuration.

Vous pouvez modifier la configuration de journalisation d'une instance de base de données à l'aide des procédures Amazon RDS `rdsadmin.rdsadmin_util.add_logfile` et`rdsadmin.rdsadmin_util.drop_logfile`. Pour plus d'informations, veuillez consulter Ajout de journaux redo en ligne et Suppression de journaux redo en ligne.

Définition du paramètre MAX_STRING_SIZE

Avant de créer un réplica Oracle, assurez-vous que le paramètre `MAX_STRING_SIZE` est le même sur l'instance de base de données source et sur le réplica. Pour cela, vous devez les associer au même groupe de paramètres. Si vous avez différents groupes de paramètres pour la source et le réplica, vous pouvez définir `MAX_STRING_SIZE` sur la même valeur. Pour plus d'informations sur ce paramètre, veuillez consulter Activation des types de données étendus pour une nouvelle instance de base de données.

Planification des ressources de calcul et de stockage

Assurez-vous que l'instance de base de données source et ses réplicas sont dimensionnés correctement, en termes de calcul et de stockage, en fonction de leur charge opérationnelle. Si un réplica atteint sa capacité en termes de ressources de calcul, de réseau ou de stockage, il arrête de

recevoir ou d'appliquer les modifications provenant de sa source. Amazon RDS for Oracle n'intervient pas pour atténuer un retard de réplica élevé entre une instance de base de données source et ses réplicas. Vous pouvez modifier les ressources de stockage et d'UC d'un réplica indépendamment de sa source et d'autres réplicas.

Création d'un réplica RDS for Oracle en mode monté

Par défaut, les réplicas Oracle sont en lecture seule. Pour créer un réplica en mode monté, utilisez la console, l'AWS CLI ou l'API RDS.

Console

Pour créer un réplica monté à partir d'une instance de base de données Oracle source

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance de base de données Oracle que vous souhaitez utiliser comme source pour un réplica monté.
4. Sous Actions, choisissez Créer un réplica.
5. Pour le Mode réplica, choisissez Monté.
6. Choisissez les paramètres que vous souhaitez utiliser. Sous Identifiant de l'instance DB, saisissez un nom pour le réplica en lecture. Ajustez les autres paramètres selon les besoins.
7. Pour Régions, choisissez la région dans laquelle le réplica monté sera lancé.
8. Choisissez la taille de votre instance et le type de stockage. Nous vous recommandons d'utiliser le même type de stockage et la même classe d'instance de base de données que l'instance de base de données source pour le réplica en lecture.
9. Pour Déploiement Multi-AZ, choisissez Créer une instance de secours pour créer une instance de secours de votre réplica dans une autre zone de disponibilité pour la prise en charge du basculement pour le réplica monté. La création de votre réplica monté en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.
10. Choisissez les autres paramètres que vous voulez utiliser.
11. Choisissez Créer un réplica.

Dans la page Bases de données, le réplica monté a le rôle Réplica.

AWS CLI

Pour créer une réplique Oracle en mode monté, définissez cette `--replica-mode` option `mounted` dans la AWS CLI commande create-db-instance-read-replica.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance-read-replica \
    --db-instance-identifier myreadreplica \
    --source-db-instance-identifier mydbinstance \
    --replica-mode mounted
```

Dans Windows :

```
aws rds create-db-instance-read-replica ^
    --db-instance-identifier myreadreplica ^
    --source-db-instance-identifier mydbinstance ^
    --replica-mode mounted
```

Pour passer d'un réplica en lecture seule à un état monté, définissez `--replica-mode` ce paramètre sur `mounted` dans la AWS CLI commande. modify-db-instance Pour convertir un réplica monté en réplica en lecture seule, définissez `--replica-mode` sur `open-read-only`.

API RDS

Pour créer une réplique Oracle en mode monté, spécifiez `ReplicaMode=mounted` dans l'API RDS l'opération InstanceReadReplicaCreateDB.

Modification du mode réplica RDS for Oracle

Pour modifier le mode de réplica d'un réplica existant, utilisez la console, AWS CLI ou l'API RDS. Lorsque vous passez en mode monté, le réplica déconnecte toutes les connexions actives. Lorsque vous passez en mode lecture seule, Amazon RDS initialise Active Data Guard.

L'opération de modification peut prendre quelques minutes. Au cours de l'opération, l'état de l'instance de base de données passe à en cours de modification. Pour plus d'informations sur les modifications d'état, veuillez consulter Affichage de l'état de l'instance de base de données dans un cluster Aurora.

Console

Pour changer le mode réplica d'un réplica Oracle de monté en lecture seule

1. Connectez-vous à la AWS Management Console et ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez la base de données de réplica montée.
4. Sélectionnez Modify.
5. Pour le mode Réplica, choisissez Lecture seule.
6. Choisissez les autres paramètres que vous souhaitez modifier.
7. Choisissez Continuer.
8. Pour Scheduling of Modifications (Planification des modifications), choisissez Appliquer immédiatement.
9. Choisissez Modifier l'instance DB.

AWS CLI

Pour passer en mode monté d'une réplique en lecture, `--replica-mode` définissez-le sur `mounted` dans la AWS CLI commande modify-db-instance. Pour convertir un réplica monté en réplica en lecture seule, définissez `--replica-mode` sur `open-read-only`.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier myreadreplica \
    --replica-mode mode
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier myreadreplica ^
    --replica-mode mode
```

API RDS

Pour changer un réplica en lecture seule en mode monté, définissez `ReplicaMode=mounted` dans ModifyDBInstance. Pour changer un réplica monté en mode lecture seule, définissez `ReplicaMode=read-only`.

Utilisation de RDS pour les sauvegardes de réplicas Oracle

Vous pouvez créer et restaurer des sauvegardes d'un réplica RDS for Oracle. Les sauvegardes automatiques et les instantanés manuels sont tous deux pris en charge. Pour de plus amples informations, veuillez consulter Sauvegarde, restauration et exportation de données. Les sections suivantes décrivent les principales différences entre la gestion des sauvegardes d'un réplica principal et d'un réplica RDS for Oracle.

Activation des sauvegardes de réplica RDS for Oracle

Les sauvegardes automatiques ne sont pas activées par défaut sur un réplica Oracle. Pour activer les sauvegardes automatiques, vous devez définir la période de rétention des sauvegardes sur une valeur positive différente de zéro.

console

Pour activer immédiatement les sauvegardes automatiques

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données ou le cluster de base de données multi-AZ que vous souhaitez modifier.
3. Sélectionnez Modifier.
4. Pour la période de conservation des sauvegardes, choisissez une valeur positive différente de zéro, par exemple trois jours.
5. Choisissez Continuer.
6. Choisissez Apply immediately (Appliquer immédiatement).
7. Choisissez Modifier l'instance de base de données ou Modifier le cluster pour enregistrer vos modifications et activer les sauvegardes automatisées.

AWS CLI

Pour activer les sauvegardes automatisées, utilisez la modify-db-clustercommande AWS CLI modify-db-instanceor.

Incluez les paramètres suivants :

- `--db-instance-identifier` (ou `--db-cluster-identifier` pour un cluster de base de données multi-AZ)
- `--backup-retention-period`
- `--apply-immediately` ou `--no-apply-immediately`

Dans l'exemple suivant, nous activons les sauvegardes automatiques en définissant la période de rétention des sauvegardes sur trois jours. Les modifications sont appliquées immédiatement.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier my_db_instance  \
    --backup-retention-period 3 \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier my_db_instance  ^
    --backup-retention-period 3 ^
    --apply-immediately
```

API RDS

Pour activer les sauvegardes automatisées, utilisez l'DBClusteropération Modify DBInstance ou Modify de l'API RDS avec les paramètres obligatoires suivants :

- `DBInstanceIdentifier` ou `DBClusterIdentifier`
- `BackupRetentionPeriod`

Restauration d'une sauvegarde de réplica RDS for Oracle

Vous pouvez restaurer une sauvegarde de réplica Oracle de la même manière que vous pouvez restaurer une sauvegarde de l'instance principale. Pour plus d'informations, consultez les ressources suivantes :

- Restauration vers une instance de base de données
- Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

Le principal élément à prendre en compte lorsque vous restaurez une sauvegarde de réplica est de déterminer l'instant dans le passé auquel vous effectuez la restauration. L'heure de la base de données désigne la dernière heure de transaction appliquée des données dans la sauvegarde. Lorsque vous restaurez une sauvegarde de réplica, vous rétablissez l'heure de la base de données, et non l'heure à laquelle la sauvegarde s'est terminée. La différence est importante car un réplica de RDS for Oracle peut accuser un retard de plusieurs minutes ou heures par rapport à l'instance principale. Ainsi, l'heure de la base de données d'une sauvegarde de réplica, et donc l'instant dans le passé auquel vous le restaurez, peut être bien antérieure à l'heure de création de la sauvegarde.

Pour faire la distinction entre l'heure de la base de données et l'heure de création, utilisez la commande `describe-db-snapshots`. Comparez le paramètre `SnapshotDatabaseTime`, qui correspond à l'heure de la base de données de la sauvegarde du réplica, et le champ `OriginalSnapshotCreateTime`, qui correspond à la dernière transaction appliquée sur la base de données principale. L'exemple suivant montre la différence entre les deux temps :

```
aws rds describe-db-snapshots \
    --db-instance-identifier my-oracle-replica
    --db-snapshot-identifier my-replica-snapshot

{
    "DBSnapshots": [
        {
            "DBSnapshotIdentifier": "my-replica-snapshot",
            "DBInstanceIdentifier": "my-oracle-replica",
            "SnapshotDatabaseTime": "2022-07-26T17:49:44Z",
            ...
            "OriginalSnapshotCreateTime": "2021-07-26T19:49:44Z"
        }
    ]
}
```

Exécution d'un basculement d'Oracle Data Guard

Un basculement est une inversion de rôle entre une base de données principale et une base de données secondaire. Lors d'un basculement, la base de données principale d'origine passe à un rôle secondaire, tandis que la base de données secondaire d'origine passe au rôle principal.

Dans un environnement Oracle Data Guard, une base de données principale prend en charge une ou plusieurs bases de données secondaire. Vous pouvez effectuer une transition de rôle gérée, basée sur le basculement, d'une base de données principale vers une base de données secondaire. Un basculement est une inversion de rôle entre une base de données principale et une base de données secondaire. Lors d'un basculement, la base de données principale d'origine passe à un rôle secondaire, tandis que la base de données secondaire d'origine passe au rôle principal.

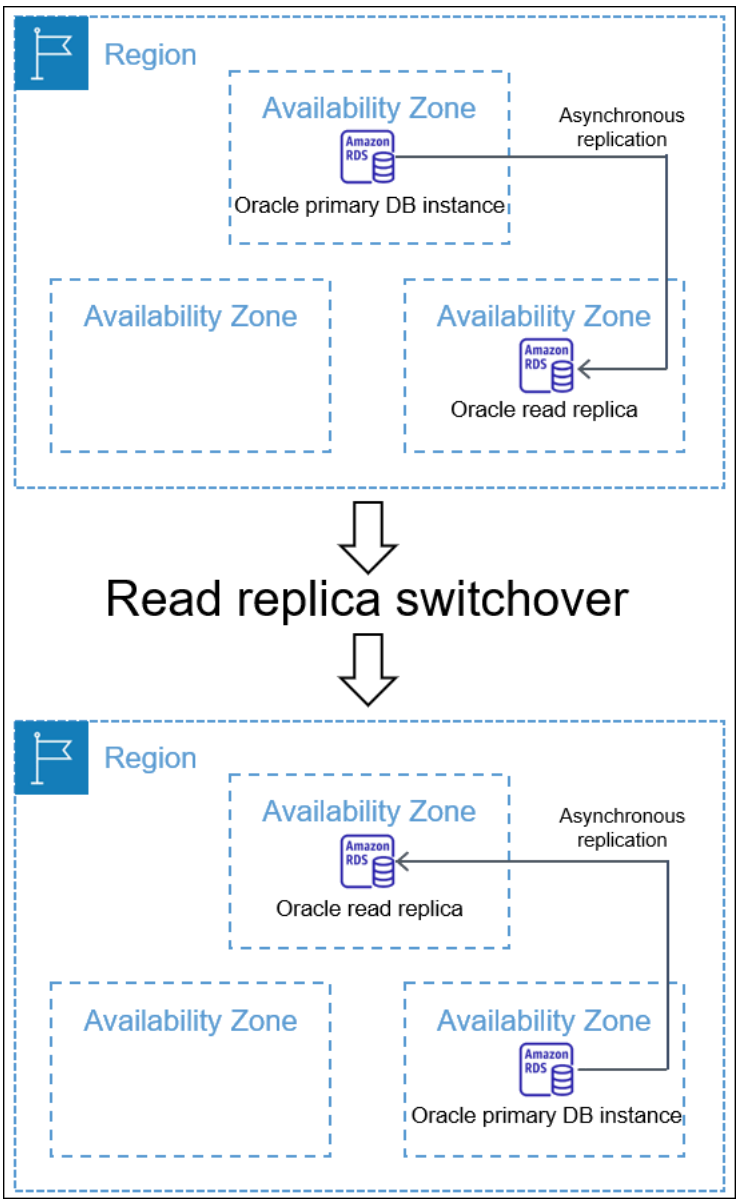
Rubriques

- Présentation du basculement d'Oracle Data Guard
- Conditions requises pour le passage à Oracle Data Guard
- Lancement du basculement d'Oracle Data Guard
- Surveillance du basculement d'Oracle Data Guard

Présentation du basculement d'Oracle Data Guard

Amazon RDS prend en charge une transition de rôle entièrement gérée et basée sur le changement pour les répliques de bases de données Oracle. Vous pouvez uniquement initier un basculement vers une base de données secondaire qui est montée ou ouverte en lecture seule.

Les répliques peuvent résider dans des zones de disponibilité distinctes Régions AWS ou différentes (AZs) d'une même région. Tous Régions AWS sont pris en charge.



Le passage au numérique est différent d'une promotion de lecture de répliques. Lors d'un basculement, les instances de base de données source et répliquée changent de rôle. Dans le cadre d'une promotion, une réplique en lecture devient une instance de base de données source, mais l'instance de base de données source ne devient pas une réplique. Pour de plus amples informations, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.

Rubriques

- Avantages du basculement vers Oracle Data Guard
- Versions de Oracle Database prises en charge
- Coût du basculement d'Oracle Data Guard

- Comment fonctionne le basculement d'Oracle Data Guard

Avantages du basculement vers Oracle Data Guard

Tout comme RDS pour les répliques de lecture Oracle, un basculement géré repose sur Oracle Data Guard. L'opération est conçue pour qu'il n'y ait aucune perte de données. Amazon RDS automatise les aspects suivants du passage au numérique :

- Inverse les rôles de votre base de données principale et de la base de données secondaire spécifiée, en plaçant la nouvelle base de données secondaire dans le même état (montée ou en lecture seule) que la base de données secondaire d'origine
- Garantit la cohérence des données
- Maintient votre configuration de réplication après la transition
- Prend en charge les inversions répétées, ce qui permet à votre nouvelle base de données secondaire de reprendre son rôle principal initial

Versions de Oracle Database prises en charge

Le passage à Oracle Data Guard est pris en charge pour les versions d'Oracle Database 19c et supérieures.

Coût du basculement d'Oracle Data Guard

La fonction de basculement d'Oracle Data Guard n'entraîne pas de coûts supplémentaires. Oracle Database Enterprise Edition inclut la prise en charge de bases de données de secours en mode monté. Pour ouvrir des bases de données de secours en mode lecture seule, vous devez disposer de l'option Oracle Active Data Guard.

Comment fonctionne le basculement d'Oracle Data Guard

Le basculement d'Oracle Data Guard est une opération entièrement gérée. Vous initiez le basculement d'une base de données de secours en émettant la CLI commande. `switchover-read-replica` Amazon RDS modifie ensuite les rôles principal et de secours dans votre configuration de réplication.

La base de données secondaire d'origine et la base de données principale d'origine sont les rôles qui existent avant le basculement. La nouvelle base de données secondaire et la nouvelle base de données principale sont les rôles qui existent après le basculement. Un réplica de secours est une

base de données de réplica qui sert de base de données secondaire dans l'environnement Oracle Data Guard mais qui ne change pas de rôle.

Rubriques

- Étapes du basculement d'Oracle Data Guard
- Après le basculement d'Oracle Data Guard

Étapes du basculement d'Oracle Data Guard

Pour effectuer le changement, Amazon RDS doit suivre les étapes suivantes :

1. Bloquer les nouvelles transactions sur la base de données principale d'origine. Pendant le passage au numérique, Amazon RDS interrompt la réplication de toutes les bases de données de votre configuration Oracle Data Guard. Pendant le basculement, la base de données principale d'origine ne peut pas traiter les requêtes d'écriture.
2. Envoyez les transactions non appliquées à la base de données secondaire d'origine, et appliquez-les.
3. Redémarrez la nouvelle base de données secondaire en mode lecture seule ou montée. Le mode dépend de l'état d'ouverture de la base de données secondaire d'origine avant le basculement.
4. Ouvrez la nouvelle base de données principale en mode lecture/écriture.

Après le basculement d'Oracle Data Guard

Amazon RDS change le rôle de la base de données principale et de la base de données de secours. Il vous incombe de reconnecter votre application et d'effectuer toute autre configuration souhaitée.

Rubriques

- Critères de réussite
- Connexion à la nouvelle base de données principale
- Configuration de la nouvelle base de données principale

Critères de réussite

Le basculement d'Oracle Data Guard est réussi lorsque la base de données secondaire d'origine :

- Effectue la transition vers son rôle de nouvelle base de données principale

- Termine sa reconfiguration

Pour limiter les temps d'arrêt, votre nouvelle base de données principale devient active dès que possible. Amazon RDS configurant les répliques externes de manière asynchrone, ces répliques peuvent devenir actives après la base de données principale d'origine.

Connexion à la nouvelle base de données principale

Amazon RDS ne propagera pas vos connexions de base de données actuelles à la nouvelle base de données principale après le passage au numérique. Une fois le basculement d'Oracle Data Guard terminé, reconnectez votre application à la nouvelle base de données principale.

Configuration de la nouvelle base de données principale

Pour effectuer un basculement vers la nouvelle base de données principale, Amazon RDS modifie le mode d'ouverture de la base de données de secours d'origine. Le changement de rôle est le seul changement apporté à la base de données. Amazon RDS ne configure pas de fonctionnalités telles que la réplication multi-AZ.

Si vous effectuez un basculement vers un réplica inter-régions avec des options différentes, la nouvelle base de données principale conserve ses propres options. Amazon RDS ne migrera pas les options de la base de données principale d'origine. Si la base de données principale d'origine comportait des options telles que SSL NNEOEM,, et OEM _AGENT, Amazon RDS ne les propage pas vers la nouvelle base de données principale.

Conditions requises pour le passage à Oracle Data Guard

Avant de lancer le basculement d'Oracle Data Guard, assurez-vous que votre environnement de réplication répond aux exigences suivantes :

- La base de données secondaire d'origine est montée ou ouverte en lecture seule.
- Les sauvegardes automatiques sont activées sur la base de données secondaire d'origine.
- La base de données principale d'origine et la base de données de secours d'origine sont dans `available` cet état.
- La base de données principale d'origine et la base de données de secours d'origine n'ont aucune action de maintenance en attente dans les états suivants : `requirednext window`, ou`in progress`. Les actions dans ces états bloquent le passage au numérique. Pour savoir comment vérifier l'état des mises à jour de maintenance en attente, consultezAfficher les mises à jour de maintenance en attente.

Les actions de maintenance en attente dans l'`available`État ne bloquent pas le passage au numérique. RDS for Oracle publie fréquemment des mises à jour du système d'exploitation (OS) dans l'`available`État. Ces mises à jour du système d'exploitation en attente ne bloqueront pas un changement, sauf si vous les planifiez pour la fenêtre de maintenance suivante, ce qui les place dans cet état. `next window`

> **Note**
>
> Si vous souhaitez différer une action de maintenance planifiée afin de pouvoir exécuter un changement, choisissez Actions, puis Différer la mise à niveau dans la console RDS. Vous pouvez également empêcher le blocage d'une commutation en appliquant une action de maintenance en attente ou en déplaçant la fenêtre de maintenance sur un intervalle avant votre transition. Pour plus d'informations, consultez l'article Re:post Comment supprimer les éléments RDS en attente de maintenance.

- La base de données secondaire d'origine est en état de réplication.
- Vous n'essayez pas de lancer un basculement lorsque la base de données principale ou la base de données secondaire est actuellement dans un cycle de vie de basculement. Si une base de données de réplica est en train de se reconfigurer après un basculement, Amazon RDS vous empêche de lancer un nouveau basculement.

> **Note**
>
> Un réplica de secours est un réplica dans la configuration Oracle Data Guard qui n'est pas la cible du basculement. Les réplicas de secours peuvent se trouver dans n'importe quel état pendant le basculement.

- La configuration de la base de données secondaire d'origine est aussi proche que possible de celle de la base de données principale d'origine. Prenons un scénario dans lequel les bases de données principale et secondaire d'origine ont des options différentes. Une fois le basculement terminé, Amazon RDS ne reconfigure pas automatiquement la nouvelle base de données principale pour qu'elle dispose des mêmes options que la base de données principale d'origine.
- Vous configurez le déploiement multi-AZ que vous souhaitez avant de lancer un basculement. Amazon RDS ne gère pas le déploiement multi-AZ dans le cadre du basculement. Le déploiement multi-AZ reste tel quel.

Supposons que db_maz soit la base de données principale dans un déploiement multi-AZ et que db_saz soit un réplica mono-AZ. Vous lancez un basculement de db_maz vers db_saz. Par la suite, db_maz est une base de données de réplica multi-AZ et db_saz est une base de données principale mono-AZ. La nouvelle base de données principale n'est désormais pas protégée par un déploiement multi-AZ.

- En prévision d'un basculement entre régions, la base de données principale n'utilise pas le même groupe d'options qu'une instance de base de données en dehors de la configuration de réplication. Pour qu'un basculement entre régions réussisse, la base de données principale actuelle et ses réplicas en lecture doivent être les seules instances de base de données à utiliser le groupe d'options de la base de données principale actuelle. Dans le cas contraire, Amazon RDS empêche le basculement.

Lancement du basculement d'Oracle Data Guard

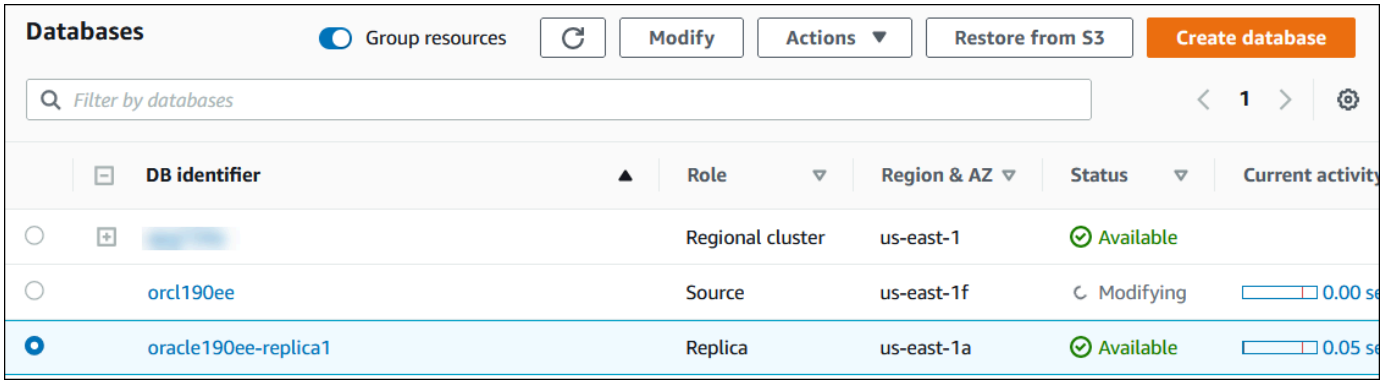
Vous pouvez passer d'une réplique de lecture RDS pour Oracle au rôle principal, et l'ancienne instance de base de données principale à un rôle de réplique.

Console

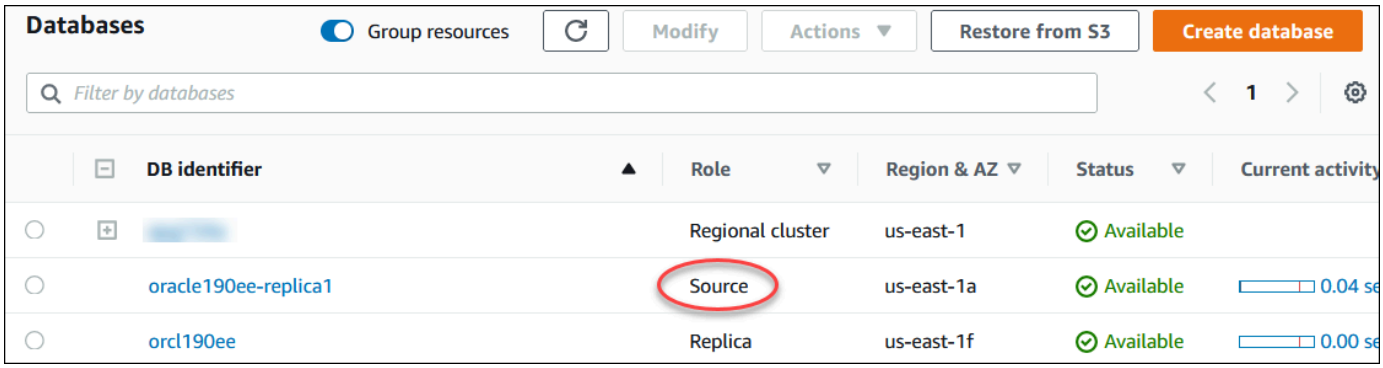
Pour basculer un réplica en lecture Oracle vers le rôle de base de données principale

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la RDS console Amazon, sélectionnez Databases.

   Le volet Bases de données s'affiche. Chaque réplica en lecture affiche Réplica dans la colonne Rôle.
3. Sélectionnez la réplica en lecture que vous souhaitez faire basculer vers le rôle principal.
4. Pour Actions, sélectionnez Switch over replica (Basculer le réplica).
5. Sélectionnez I acknowledge (Je confirme). Sélectionnez ensuite Switch over replica (Basculer le réplica).
6. Sur la page Databases (Bases de données), surveillez la progression du basculement.



Lorsque le basculement est terminé, le rôle de la cible du basculement passe de Replica (Réplica) à Source.



AWS CLI

Pour passer d'une réplique Oracle au rôle de base de données principal, utilisez la AWS CLI `switchover-read-replica`commande. Dans les exemples suivants, le réplica Oracle est nommé *`replica-to-be-made-primary`* dans la nouvelle base de données principale.

Example

Dans Linux, macOS, ou Unix:

```
aws rds switchover-read-replica \
    --db-instance-identifier replica-to-be-made-primary
```

Dans Windows:

```
aws rds switchover-read-replica ^
    --db-instance-identifier replica-to-be-made-primary
```

RDS API

Pour passer d'une réplique Oracle au rôle de base de données principal, appelez l'RDSAPI`SwitchoverReadReplica`opération Amazon avec le paramètre requis`DBInstanceIdentifier`. Ce paramètre spécifie le nom du réplica Oracle qui doit assumer le rôle de base de données principale.

Surveillance du basculement d'Oracle Data Guard

Pour vérifier l'état de vos instances, utilisez la AWS CLI commande`describe-db-instances`. La commande suivante vérifie l'état de l'instance de base de données *`orcl2`*. Cette base de données était une base de données de secours avant le basculement, mais c'est la nouvelle base de données principale après le basculement.

```
aws rds describe-db-instances \
    --db-instance-identifier orcl2
```

Pour confirmer que le basculement s'est bien déroulé, interrogez `V$DATABASE.OPEN_MODE`. Vérifiez que la valeur de la nouvelle base de données principale est `READ WRITE`.

```
SELECT OPEN_MODE FROM V$DATABASE;
```

Pour rechercher des événements liés au basculement, utilisez la commande. AWS CLI `describe-events` L'exemple suivant recherche des événements sur *`orcl2`* instance.

```
aws rds describe-events \
    --source-identifier orcl2 \
    --source-type db-instance
```

Dépannage des réplicas RDS for Oracle

Cette section décrit les problèmes de réplication et leurs résolutions possibles.

Rubriques

- Surveillance du retard de réplication Oracle
- Dépannage de l'échec de la réplication Oracle après l'ajout ou la modification de déclencheurs

Surveillance du retard de réplication Oracle

Pour surveiller le décalage de réplication dans Amazon CloudWatch, affichez la métrique Amazon RDS `ReplicaLag`. Pour obtenir plus d'informations sur la durée du retard de réplication, consultez Supervision de la réplication en lecture et CloudWatch Métriques Amazon pour Amazon RDS.

Pour un réplica en lecture, si le temps de latence est trop long, interrogez les vues suivantes :

- `V$ARCHIVED_LOG` – Indique quelles validations ont été appliquées au réplica en lecture.
- `V$DATAGUARD_STATS` – Indique la répartition détaillée des composants qui font partie de la métrique `ReplicaLag`.
- `V$DATAGUARD_STATUS` – Indique la sortie de journal des processus de réplication internes d'Oracle.

Pour un réplica monté, si le temps de latence est trop long, vous ne pouvez pas interroger les vues `V$`. Dans ce cas, procédez comme suit :

- Consultez la métrique `ReplicaLag` dans CloudWatch.
- Consultez le fichier journal des alertes du réplica dans la console. Recherchez les erreurs dans les messages de récupération. Les messages comprennent le numéro de séquence du journal, que vous pouvez comparer au numéro de séquence principal. Pour plus d'informations, consultez Fichiers journaux RDS de base de données Amazon pour Oracle.

Dépannage de l'échec de la réplication Oracle après l'ajout ou la modification de déclencheurs

Si vous ajoutez ou modifiez des déclencheurs et que la réplication échoue, les déclencheurs peuvent être en cause. Assurez-vous que le déclencheur exclut les comptes suivants, qui sont requis par RDS pour la réplication :

- Comptes d'utilisateurs avec privilèges d'administrateur
- `SYS`
- `SYSTEM`
- `RDS_DATAGUARD`
- `rdsdb`

Pour plus d'informations, consultez Considérations diverses relatives aux réplicas RDS for Oracle.

Ajout d'options aux instances de base de données Oracle

Dans Amazon RDS, une option est une fonctionnalité supplémentaire. Voici une description des options que vous pouvez ajouter aux instances Amazon RDS exécutant le moteur de base de données Oracle.

Rubriques

- Présentation des options de base de données Oracle
- Intégration Amazon S3
- Oracle Application Express (APEX)
- EFSIntégration avec Amazon
- Oracle Java Virtual Machine
- Oracle Enterprise Manager
- Oracle Label Security
- Oracle Locator
- Oracle NNE (Native Network Encryption)
- Oracle OLAP
- Oracle Secure Sockets Layer (SSL)
- Oracle Spatial
- Oracle SQLT
- Oracle Statspack
- Fuseau horaire Oracle
- Mise à niveau automatique du fichier sur le fuseau horaire Oracle
- Oracle Transparent Data Encryption
- Oracle UTL _ MAIL
- Oracle XML DB

Présentation des options de base de données Oracle

Pour activer ces options pour votre base de données Oracle, ajoutez-les à un groupe d'options, puis associez celui-ci à votre instance de base de données. Pour plus d'informations, consultez Utilisation de groupes d'options.

Rubriques

- Résumé des options Oracle Database
- Options prises en charge pour les différentes éditions
- Exigences de mémoire pour les options spécifiques

Résumé des options Oracle Database

Vous pouvez ajouter les options suivantes pour les instances de bases de données Oracle.

Option	ID d'option
Intégration Amazon S3	`S3_INTEGRATION`
Oracle Application Express (APEX)	`APEX` `APEX-DEV`
Oracle Enterprise Manager	`OEM` `OEM_AGENT`
Oracle Java Virtual Machine	`JVM`
Oracle Label Security	`OLS`
Oracle Locator	`LOCATOR`
Oracle NNE (Native Network Encryption)	`NATIVE_NETWORK_ENCRYPTION`
Oracle OLAP	`OLAP`
Oracle Secure Sockets Layer (SSL)	`SSL`
Oracle Spatial	`SPATIAL`
Oracle SQLT	`SQLT`
Oracle Statspack	`STATSPACK`

Option	ID d'option
Fuseau horaire Oracle	`Timezone`
Mise à niveau automatique du fichier sur le fuseau horaire Oracle	`TIMEZONE_FILE_AUTOUPGRADE`
Oracle Transparent Data Encryption	`TDE`
Oracle UTL _ MAIL	`UTL_MAIL`
Oracle XML DB	`XMLDB`

Options prises en charge pour les différentes éditions

RDS for Oracle vous empêche d'ajouter des options à une édition si elles ne sont pas prises en charge. Pour savoir quelles options RDS sont prises en charge dans les différentes éditions Oracle Database, utilisez la commande `aws rds describe-option-group-options`. L'exemple suivant répertorie les options prises en charge pour l'Enterprise Edition d'Oracle Database 19c.

```
aws rds describe-option-group-options \
    --engine-name oracle-ee \
    --major-engine-version 19
```

Pour plus d'informations, consultez describe-option-groups dans la Référence des commandes de CLI AWS .

Exigences de mémoire pour les options spécifiques

Certaines options nécessitent que de la mémoire supplémentaire s'exécute sur votre instance de base de données. Par exemple, Oracle Enterprise Manager Database Control utilise environ 300 Mo de RAM. Si vous activez cette option pour une petite instance de base de données, vous pourrez rencontrer des problèmes de performance liés aux contraintes de mémoire. Vous pouvez ajuster les paramètres Oracle afin que la base de données nécessite moins de RAM. Sinon, vous pouvez augmenter la taille de l'instance de base de données.

Intégration Amazon S3

Vous pouvez transférer des fichiers entre votre instance de base de données RDS pour Oracle et un compartiment Amazon S3. Vous pouvez utiliser l'intégration Amazon S3 avec les fonctionnalités Oracle Database telles que Oracle Data Pump. Par exemple, vous pouvez télécharger des fichiers Data Pump depuis Amazon S3 vers votre instance RDS de base de données Oracle. Pour de plus amples informations, veuillez consulter Importation de données dans Oracle sur Amazon RDS.

> **Note**
>
> Votre instance de base de données et votre compartiment Amazon S3 doivent se trouver dans la même Région AWS.

Rubriques

- Configuration des autorisations IAM pour l'intégration de RDS for Oracle à Amazon S3
- Ajout de l'option d'intégration Amazon S3
- Transfert de fichiers entre Amazon RDS pour Oracle et un compartiment Amazon S3
- Résolution des problèmes d'intégration avec Amazon S3
- Suppression de l'option d'intégration Amazon S3

Configuration des autorisations IAM pour l'intégration de RDS for Oracle à Amazon S3

Pour que RDS for Oracle s'intègre à Amazon S3, votre instance de base de données doit avoir accès à un compartiment Amazon S3. Le Amazon VPC utilisé par votre instance de base de données n'a pas besoin de fournir d'accès aux points de terminaison Amazon S3.

RDS for Oracle prend en charge le transfert de fichiers entre une instance de base de données d'un compte et un compartiment Amazon S3 d'un autre compte. Lorsque des étapes supplémentaires sont requises, elles sont indiquées dans les sections suivantes.

Rubriques

- Étape 1 : Créer une politique IAM pour votre rôle Amazon RDS
- Étape 2 (facultative) : Créer une politique IAM pour votre compartiment Amazon S3
- Étape 3 : Créer un rôle IAM pour votre instance de base de données et y attacher votre politique

- Étape 4 : associer votre rôle IAM à votre instance de base de données RDS for Oracle.

Étape 1 : Créer une politique IAM pour votre rôle Amazon RDS

Au cours de cette étape, vous créez une politique AWS Identity and Access Management (IAM) avec les autorisations requises pour transférer des fichiers entre votre compartiment Amazon S3 et votre instance de base de données RDS. Cette étape suppose également que vous avez déjà créé un compartiment S3.

Avant de créer la politique, notez les informations suivantes :

- ARN (Amazon Resource Name) de votre compartiment
- L'ARN de votre AWS KMS clé, si votre compartiment utilise le chiffrement SSE-KMS ou SSE-S3

> **Note**
>
> Une instance de base de données RDS for Oracle ne peut pas accéder aux compartiments Amazon S3 chiffrés avec SSE-C.

Pour plus d'informations, consultez la section Protection des données à l'aide du chiffrement côté serveur dans le Guide de l'utilisateur Amazon Simple Storage Service.

console

Pour créer une politique IAM afin d'autoriser Amazon RDS à accéder à votre compartiment Amazon S3

1. Ouvrez IAM Management Console.
2. Sous Access Management (Gestion des accès), choisissez Policies (Stratégies).
3. Choisissez Create Policy (Créer une politique).
4. Sous l'onglet Visual editor (Éditeur visuel), choisissez Choose a service (Choisir un service), puis S3.
5. Pour Actions, choisissez Expand all (Développer tout), puis choisissez les autorisations de compartiment et d'objet nécessaires pour transférer des fichiers d'un compartiment Amazon S3 vers Amazon RDS. Par exemple, procédez comme suit :
   - Développez la liste, puis sélectionnez ListBucket.

- Développez Lire, puis sélectionnez GetObject.
- Développez Write, puis sélectionnez PutObjectDeleteObject, AbortMultipartUpload, et ListMultipartUploadParts. Les autorisations de téléchargement partitionné sont requises lors du chargement de fichiers volumineux (100 Mo ou plus) vers Amazon S3.
- Développez la gestion des autorisations, puis sélectionnez PutObjectAcl. Cette autorisation est nécessaire si vous envisagez de charger des fichiers dans un compartiment appartenant à un autre compte (ce compte doit avoir un contrôle total du contenu du compartiment).

Les autorisations d'objet sont des autorisations pour les opérations sur les objets dans Amazon S3. Vous devez les accorder pour des objets d'un compartiment et non pour le compartiment lui-même. Pour plus d'informations, consultez Permissions for object operations.

6. Choisissez Ressources et procédez comme suit :

   a. Choisissez Spécifique.

   b. Pour Compartiment, choisissez Ajouter un ARN. Saisissez l'ARN de votre compartiment. Le nom du compartiment est renseigné automatiquement. Choisissez ensuite Ajouter.

   c. Si la ressource de l'objet est affichée, choisissez Ajouter un ARN pour ajouter des ressources manuellement ou choisissez Tous.

   > **Note**
   >
   > Vous pouvez affecter à Amazon Resource Name (ARN) une valeur d'ARN plus spécifique afin d'autoriser Amazon RDS à accéder uniquement à des fichiers ou des dossiers spécifiques dans un compartiment Amazon S3. Pour plus d'informations sur la définition d'une stratégie d'accès pour Amazon S3, consultez Gestion des autorisations d'accès de vos ressources Amazon S3.

7. (Facultatif) Choisissez Add additional permissions (Ajouter des autorisations supplémentaires) pour ajouter des ressources à la stratégie. Par exemple, procédez comme suit :

   a. Si votre compartiment est chiffré avec une clé KMS personnalisée, sélectionnez KMS pour le service.

   b. Pour Actions manuelles, sélectionnez ce qui suit :

      - Encrypt
      - ReEncrypt de et ReEncrypt vers

- Decrypt
- DescribeKey
- GenerateDataKey

c. Pour Resources (Ressources), choisissez Specific (Spécifique).

d. Pour Clé, choisissez Ajouter un ARN. Saisissez l'ARN de votre clé personnalisée en tant que ressource, puis choisissez Ajouter.

Pour plus d'informations, consultez la section Protection des données à l'aide du chiffrement côté serveur avec des clés KMS stockées dans AWS Key Management Service (SSE-KMS) dans le guide de l'utilisateur d'Amazon Simple Storage Service.

e. Si vous souhaitez qu'Amazon RDS accède à d'autres compartiments, ajoutez le ARNs pour ces compartiments. Si vous le souhaitez, vous pouvez également accorder l'accès à tous les compartiments et à tous les objets dans Amazon S3.

8. Choisissez Suivant : Balises, puis Suivant : Vérification.
9. Dans Name (Name), attribuez un nom à votre stratégie IAM, par exemple `rds-s3-integration-policy`. Vous utilisez ce nom lorsque vous créez un rôle IAM à associer à votre instance de base de données. Vous pouvez également ajouter une valeur Description facultative.
10. Choisissez Create Policy (Créer une politique).

AWS CLI

Créez une politique AWS Identity and Access Management (IAM) qui accorde à Amazon RDS l'accès à un compartiment Amazon S3. Après avoir créé la politique, notez son ARN. Vous en aurez besoin lors d'une étape ultérieure.

Incluez les actions appropriées dans la politique en fonction du type d'accès requis :

- `GetObject` – Obligatoire pour transférer les fichiers d'un compartiment Amazon S3 vers Amazon RDS.
- `ListBucket` – Obligatoire pour transférer les fichiers d'un compartiment Amazon S3 vers Amazon RDS.
- `PutObject` – Obligatoire pour transférer les fichiers de Amazon RDS vers un compartiment Amazon S3.
- `AbortMultipartUpload`— Nécessaire pour les téléchargements partitionnés lors du transfert de fichiers volumineux (100 Mo ou plus) d'Amazon RDS vers un compartiment Amazon S3.

- `ListMultipartUploadParts`— Nécessaire pour les téléchargements partitionnés lors du transfert de fichiers volumineux (100 Mo ou plus) d'Amazon RDS vers un compartiment Amazon S3.

La AWS CLI commande suivante crée une politique IAM nommée *`rds-s3-integration-policy`* avec ces options. Elle accorde un accès à un compartiment nommé *`amzn-s3-demo-bucket`*.

Example

Pour LinuxmacOS, ou Unix :

```
aws iam create-policy \
   --policy-name rds-s3-integration-policy \
   --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "s3integration",
         "Action": [
           "s3:GetObject",
           "s3:ListBucket",
           "s3:PutObject",
           "s3:AbortMultipartUpload",
           "s3:ListMultipartUploadParts"
         ],
         "Effect": "Allow",
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket",
           "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
       }
     ]
   }'
```

L'exemple suivant inclut des autorisations pour les clés KMS personnalisées.

```
aws iam create-policy \
   --policy-name rds-s3-integration-policy \
   --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
```

```
        "Sid": "s3integration",
        "Action": [
          "s3:GetObject",
          "s3:ListBucket",
          "s3:PutObject",
          "kms:Decrypt",
          "kms:Encrypt",
          "kms:ReEncrypt*",
          "kms:GenerateDataKey",
          "kms:DescribeKey",
        ],
        "Effect": "Allow",
        "Resource": [
          "arn:aws:s3:::amzn-s3-demo-bucket",
          "arn:aws:s3:::amzn-s3-demo-bucket/*",
          "arn:aws:kms:::your-kms-arn"
        ]
      }
    ]
  }'
```

Dans Windows :

```
aws iam create-policy ^
   --policy-name rds-s3-integration-policy ^
   --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "s3integration",
         "Action": [
           "s3:GetObject",
           "s3:ListBucket",
           "s3:PutObject",
           "s3:AbortMultipartUpload",
           "s3:ListMultipartUploadParts"
         ],
         "Effect": "Allow",
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket",
           "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
       }
```

```
    ]
  }'
```

L'exemple suivant inclut des autorisations pour les clés KMS personnalisées.

```
aws iam create-policy ^
   --policy-name rds-s3-integration-policy ^
   --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "s3integration",
         "Action": [
           "s3:GetObject",
           "s3:ListBucket",
           "s3:PutObject",
           "kms:Decrypt",
           "kms:Encrypt",
           "kms:ReEncrypt",
           "kms:GenerateDataKey",
           "kms:DescribeKey",
         ],
         "Effect": "Allow",
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket",
           "arn:aws:s3:::amzn-s3-demo-bucket/*",
           "arn:aws:kms:::your-kms-arn"
         ]
       }
     ]
   }'
```

Étape 2 (facultative) : Créer une politique IAM pour votre compartiment Amazon S3

Cette étape n'est nécessaire que dans les conditions suivantes :

- Vous prévoyez de charger des fichiers dans un compartiment Amazon S3 à partir d'un compte (compte A) et d'y accéder depuis un autre compte (compte B).
- Le compte A est le propriétaire du compartiment.
- Le compte B nécessite un contrôle total des objets chargés dans le compartiment.

Si les conditions précédentes ne s'appliquent pas à votre cas, passez à Étape 3 : Créer un rôle IAM pour votre instance de base de données et y attacher votre politique.

Pour créer votre politique de compartiment, assurez-vous de disposer des éléments suivants :

- ID de compte du compte A
- Nom d'utilisateur du compte A
- Valeur ARN du compartiment Amazon S3 dans le compte B

console

Pour créer ou modifier une politique de compartiment

1. Connectez-vous à la console Amazon S3 AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/s3/.
2. Dans la liste Buckets (Compartiments), choisissez le nom du compartiment pour lequel vous souhaitez créer une stratégie de compartiment ou modifier la stratégie de compartiment existante.
3. Choisissez Permissions.
4. Sous Politique de compartiment, choisissez Modifier. La page Modifier la stratégie de compartiment s'ouvre.
5. Dans la page Edit bucket policy (Modifier la politique de compartiment), explorez Policy examples (Exemples de politiques) dans le Guide de l'utilisateur Simple Storage Service (Amazon S3), choisissez Policy generator (Générateur de politiques) pour générer automatiquement une politique, ou modifiez le JSON dans la section Policy (Politique).

   Si vous choisissez le générateur de AWS politiques, celui-ci s'ouvre dans une nouvelle fenêtre :

   a. Sur la page AWS Policy Generator (Générateur de politiques), dans Select Type of Policy (Sélectionner le type de politique), sélectionnez S3 Bucket Policy (Politique de compartiment S3).

   b. Ajoutez une instruction en saisissant les informations dans les champs fournis, puis choisissez Add Statement (Ajouter une instruction). Répétez l'opération pour autant d'instructions que vous souhaitez ajouter. Pour plus d'informations sur ces champs, consultez la Référence des éléments de stratégie IAM JSON dans le Guide de l'utilisateur IAM.

> **Note**
>
> Pour plus de commodité, la page Edit Bucket Policy (Modifier la stratégie de compartiment) affiche l'ARN (Amazon Resource Name) de compartiment du compartiment actuel au-dessus du champ de texte Policy (Stratégie). Vous pouvez copier cet ARN pour l'utiliser dans les instructions de la page AWS Policy Generator (Générateur de politique).

c. Une fois que vous avez fini d'ajouter des instructions, choisissez Generate Policy (Générer une stratégie).

d. Copiez le texte de stratégie généré, choisissez Fermer et revenez à la page Modifier la stratégie de compartiment dans la console Amazon S3.

6. Dans la boîte Policy (Stratégie), modifiez la stratégie existante ou collez la stratégie de compartiment à partir du générateur de stratégies. Veillez à résoudre les avertissements de sécurité, les erreurs, les avertissements généraux et les suggestions avant d'enregistrer votre stratégie.

   JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ExamplePermissions",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::123456789012:user/account-A-user"
      },
      "Action": [
        "s3:PutObject",
        "s3:PutObjectAcl"
      ],
      "Resource": [
        "arn:aws:s3:::amzn-s3-demo-destination-bucket",
        "arn:aws:s3:::amzn-s3-demo-destination-bucket/*"
      ]
    }
  ]
```

```
}
```

7. Choisissez Save Changes (Enregistrez les modifications), qui vous renvoie à la page Autorisations du compartiment.

Étape 3 : Créer un rôle IAM pour votre instance de base de données et y attacher votre politique

Cette étape suppose que vous avez créé la politique IAM dans Étape 1 : Créer une politique IAM pour votre rôle Amazon RDS. Au cours de cette étape, vous créez un rôle pour votre instance de base de données RDS for Oracle, puis attachez votre politique au rôle.

console

Pour créer un rôle IAM afin d'autoriser Amazon RDS à accéder à un compartiment Amazon S3

1. Ouvrez IAM Management Console.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Sélectionnez Create role (Créer un rôle).
4. Choisissez Service AWS .
5. Pour les cas d'utilisation d'autres AWS services : choisissez RDS, puis RDS — Ajouter un rôle à la base de données. Ensuite, sélectionnez Suivant.
6. Pour Rechercher sous Politiques d'autorisations, saisissez le nom de la politique IAM que vous avez créée dans Étape 1 : Créer une politique IAM pour votre rôle Amazon RDS et choisissez la politique lorsqu'elle apparaît dans la liste. Ensuite, sélectionnez Suivant.
7. Pour Nom du rôle, indiquez le nom de votre rôle IAM, par exemple `rds-s3-integration-role`. Vous pouvez également ajouter une valeur Description facultative.
8. Choisissez Créer un rôle.

AWS CLI

Pour créer un rôle et y attacher votre politique

1. Créez un rôle IAM qu'Amazon RDS peut endosser en votre nom pour accéder à vos compartiments Amazon S3.

   Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des relations d'approbation basées sur les ressources pour

limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la relation d'approbation, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'Amazon Resource Name (ARN) complet des ressources qui accèdent au rôle.

La AWS CLI commande suivante crée le rôle nommé *`rds-s3-integration-role`* à cet effet.

Example

Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
   --role-name rds-s3-integration-role \
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
                 "aws:SourceAccount": "my_account_ID",
                 "aws:SourceArn": "arn:aws:rds:Region:my_account_ID:db:dbname"
             }
         }
       }
     ]
```

```
   }'
```

Dans Windows :

```
aws iam create-role ^
   --role-name rds-s3-integration-role ^
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
                 "aws:SourceAccount": "my_account_ID",
                 "aws:SourceArn": "arn:aws:rds:Region:my_account_ID:db:dbname"
             }
         }
       }
     ]
   }'
```

Pour de plus amples informations, veuillez consulter Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

2. Une fois le rôle créé, notez son ARN. Vous en aurez besoin lors d'une étape ultérieure.
3. Attachez la politique que vous avez créée au rôle que vous avez créé.

   La AWS CLI commande suivante associe la politique au rôle nommé*rds-s3-integration-role*.

   Example

   Pour LinuxmacOS, ou Unix :

```
aws iam attach-role-policy \
   --policy-arn your-policy-arn \
   --role-name rds-s3-integration-role
```

Dans Windows :

```
aws iam attach-role-policy ^
   --policy-arn your-policy-arn ^
   --role-name rds-s3-integration-role
```

Remplacez *your-policy-arn* par l'ARN de stratégie que vous avez noté lors d'une étape précédente.

Étape 4 : associer votre rôle IAM à votre instance de base de données RDS for Oracle.

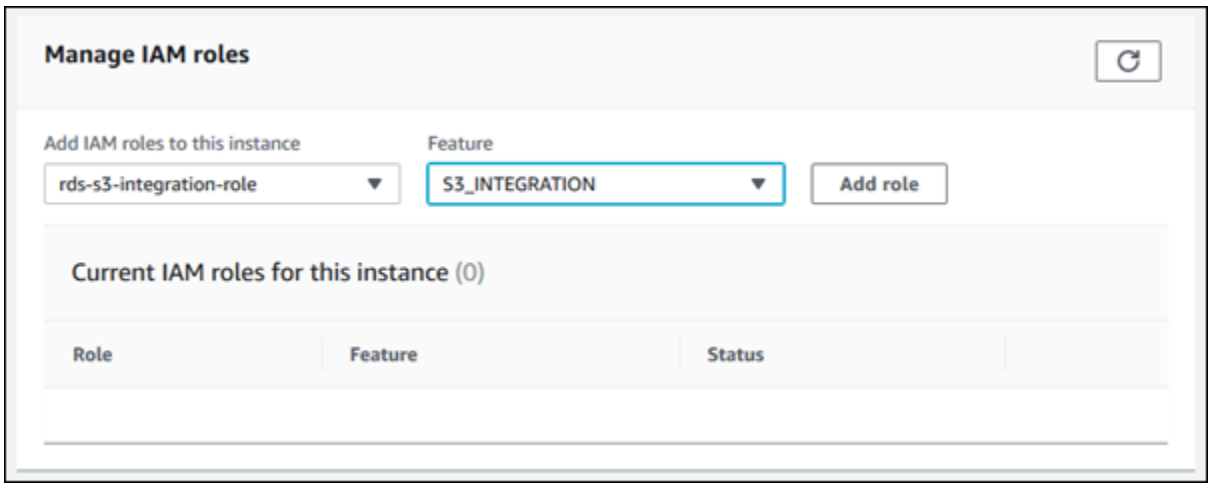
La dernière étape de la configuration des autorisations pour l'intégration d'Amazon S3 consiste à associer votre rôle IAM à votre instance de base de données. Notez les critères suivants :

- Vous devez avoir accès à un rôle IAM auquel est associée la politique d'autorisations Amazon S3 requise.
- Vous pouvez associer un seul rôle IAM à la fois avec votre instance de base de données RDS for Oracle.
- Votre instance de base de données doit être dans l'état Disponible.

console

Pour associer votre rôle IAM à votre instance de base de données RDS for Oracle

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez Databases (Bases de données) dans le volet de navigation.
3. Choisissez le nom de l'instance de base de données RDS for Oracle pour afficher ses détails.
4. Dans l'onglet Connectivity & security (Connectivité & sécurité), faites défiler l'écran jusqu'à l'onglet Manage IAM roles (Gérer les rôles IAM) au bas de la page.
5. Pour Ajouter des rôles IAM à cette instance, choisissez le rôle que vous avez créé dansÉtape 3 : Créer un rôle IAM pour votre instance de base de données et y attacher votre politique.
6. Pour Fonction, choisissez S3_INTEGRATION.



7. Choisissez Add role (Ajouter un rôle).

AWS CLI

La AWS CLI commande suivante ajoute le rôle à une instance de base de données Oracle nommée*mydbinstance*.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds add-role-to-db-instance \
   --db-instance-identifier mydbinstance \
   --feature-name S3_INTEGRATION \
   --role-arn your-role-arn
```

Dans Windows :

```
aws rds add-role-to-db-instance ^
   --db-instance-identifier mydbinstance ^
   --feature-name S3_INTEGRATION ^
   --role-arn your-role-arn
```

Remplacez *your-role-arn* par l'ARN du rôle que vous avez noté lors d'une étape précédente. `S3_INTEGRATION` doit être spécifié pour l'option `--feature-name`.

Ajout de l'option d'intégration Amazon S3

Pour intégrer Amazon RDS for Oracle à Amazon S3, votre instance de base de données doit être associée à un groupe d'options qui inclut l'`S3_INTEGRATION`option.

Console

Pour configurer un groupe d'options pour l'intégration Amazon S3

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `S3_INTEGRATION`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'option `S3_INTEGRATION` au groupe d'options.

   Pour de plus amples informations sur l'ajout d'une option à un groupe d'options, veuillez consulter Ajout d'une option à un groupe d'options.
3. Créez une nouvelle instance RDS de base de données Oracle et associez-y le groupe d'options, ou modifiez une instance de base de données Oracle RDS pour y associer le groupe d'options.

   Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.

   Pour plus d'informations sur la modification d'une instance de base de données , consultez Modification d'une RDS instance de base de données Amazon.

AWS CLI

Pour configurer un groupe d'options pour l'intégration Amazon S3

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `S3_INTEGRATION`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'option `S3_INTEGRATION` au groupe d'options.

   Par exemple, la AWS CLI commande suivante ajoute l'`S3_INTEGRATION`option à un groupe d'options nommé**`myoptiongroup`**.

Example

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
   --option-group-name myoptiongroup \
   --options OptionName=S3_INTEGRATION,OptionVersion=1.0
```

Dans Windows:

```
aws rds add-option-to-option-group ^
   --option-group-name myoptiongroup ^
   --options OptionName=S3_INTEGRATION,OptionVersion=1.0
```

3. Créez une nouvelle instance RDS de base de données Oracle et associez-y le groupe d'options, ou modifiez une instance de base de données Oracle RDS pour y associer le groupe d'options.

   Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.

   Pour plus d'informations sur la modification d'une instance de base de données RDS pour Oracle, consultezModification d'une RDS instance de base de données Amazon.

Transfert de fichiers entre Amazon RDS pour Oracle et un compartiment Amazon S3

Pour transférer des fichiers entre une instance de base de données RDS pour Oracle et un compartiment Amazon S3, vous pouvez utiliser le RDS package Amazon`rdsadmin_s3_tasks`. Vous pouvez compresser des fichiers GZIP lors de leur téléchargement et les décompresser lors du téléchargement.

Rubriques

- Exigences et limites relatives aux transferts de fichiers
- Téléchargement de fichiers depuis votre instance de base de données RDS pour Oracle vers un compartiment Amazon S3
- Téléchargement des fichiers d'un compartiment Amazon S3 vers une instance de base de données Oracle
- Surveillance du statut d'un transfert de fichiers

Exigences et limites relatives aux transferts de fichiers

Avant de transférer des fichiers entre votre instance de base de données et un compartiment Amazon S3, notez ce qui suit :

- Le `rdsadmin_s3_tasks` package transfère les fichiers situés dans un seul répertoire. Vous ne pouvez pas inclure de sous-répertoires dans un transfert.
- La taille maximale d'un objet dans un compartiment Amazon S3 est de 5 To.
- Les tâches créées par `rdsadmin_s3_tasks` s'exécutent de manière asynchrone.
- Vous pouvez télécharger des fichiers depuis le répertoire Data Pump, tel que`DATA_PUMP_DIR`, ou depuis n'importe quel répertoire créé par l'utilisateur. Vous ne pouvez pas télécharger de fichiers depuis un répertoire utilisé par les processus d'arrière-plan Oracle, tel que les `adump` `trace` répertoiresbdump, ou.
- La limite de téléchargement est de 2000 fichiers par appel de procéduredownload_from_s3. Si vous devez télécharger plus de 2 000 fichiers d'Amazon S3, divisez votre téléchargement en actions distinctes, avec un maximum de 2 000 fichiers par appel de procédure.
- Si un fichier existe dans votre dossier de téléchargement et que vous tentez de télécharger un fichier portant le même nom, `download_from_s3` ignore le téléchargement. Pour supprimer un fichier du répertoire de téléchargement, utilisez la SQL procédure PL/ UTL_FILE. FREMOVE.

Téléchargement de fichiers depuis votre instance de base de données RDS pour Oracle vers un compartiment Amazon S3

Pour charger les fichiers de votre instance de base de données vers un compartiment Amazon S3, utilisez la procédure `rdsadmin.rdsadmin_s3_tasks.upload_to_s3`. Par exemple, vous pouvez télécharger des fichiers de sauvegarde Oracle Recovery Manager (RMAN) ou des fichiers Oracle Data Pump. Pour plus d'informations sur l'utilisation des objets, consultez le Guide de l'utilisateur Amazon Simple Storage Service. Pour plus d'informations sur l'exécution de RMAN sauvegardes, consultezExécution de RMAN tâches courantes pour les instances de base de données Oracle.

La procédure `rdsadmin.rdsadmin_s3_tasks.upload_to_s3` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_bucket_name	VARCHAR2	–	obligatoire	Nom du compartiment Amazon S3 dans lequel charger les fichiers.
p_directory_name	VARCHAR2	–	obligatoire	Nom de l'objet de répertoire Oracle à partir duquel les fichiers doivent être chargés. Le répertoire peut être n'importe quel objet de répertoire créé par l'utilisateur ou répertoire Data Pump, par exemple DATA_PUMP_DIR . Vous ne pouvez pas télécharger de fichiers depuis un répertoire utilisé par des processus d'arrière-planadump, tels quebdump, ettrace. **Note** Vous pouvez uniquement charger les fichiers à partir du répertoire spécifié. Vous ne pouvez pas charger les fichiers des sous-répertoires dans

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				le répertoire spécifié.
`p_s3_prefix`	VARCHAR2	–	obligatoire	Préfixe de nom de fichier Amazon S3 en fonction duquel les fichiers sont chargés. Un préfixe vide charge tous les fichiers au niveau supérieur dans le compartiment Amazon S3 spécifié et n'ajoute pas de préfixe aux noms de fichier. Par exemple, si le préfixe est `folder_1/oradb` , les fichiers sont chargés dans `folder_1`. Dans ce cas, le préfixe `oradb` est ajouté à chaque fichier.
`p_prefix`	VARCHAR2	–	obligatoire	Préfixe de nom de fichier que les noms de fichier doivent contenir pour être chargés. Un préfixe vide charge tous les fichiers du répertoire spécifié.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_compression_level`	NUMBER	`0`	facultatif	Le niveau de GZIP compression. La plage des valeurs valides est comprise entre `0` et 9 : • `0` – Pas de compression • 1 – Compression la plus rapide • 9 – Compression la plus élevée
`p_bucket_owner_full_control`	VARCHAR2	–	facultatif	Paramètre de contrôle d'accès pour le compartiment. Les seules valeurs valides sont null ou `FULL_CONTROL`. Ce paramètre est requis uniquement si vous chargez des fichiers depuis un compte (compte A) dans un compartiment appartenant à un autre compte (compte B). Le compte B doit avoir un contrôle total des fichiers.

La valeur renvoyée pour la procédure `rdsadmin.rdsadmin_s3_tasks.upload_to_s3` est un ID de tâche.

L'exemple suivant télécharge tous les fichiers du *`DATA_PUMP_DIR`* répertoire dans le compartiment Amazon S3 nommé *`amzn-s3-demo-bucket`*. Les fichiers ne sont pas compressés.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
      p_bucket_name    =>  'amzn-s3-demo-bucket',
      p_prefix         =>  '',
      p_s3_prefix      =>  '',
      p_directory_name =>  'DATA_PUMP_DIR')
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge tous les fichiers ayant le préfixe *db* dans le répertoire*DATA_PUMP_DIR* du compartiment Amazon S3 nommé *amzn-s3-demo-bucket*. Amazon RDS applique le plus haut niveau de GZIP compression aux fichiers.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
      p_bucket_name       =>  'amzn-s3-demo-bucket',
      p_prefix            =>  'db',
      p_s3_prefix         =>  '',
      p_directory_name    =>  'DATA_PUMP_DIR',
      p_compression_level =>  9)
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge tous les fichiers d'un répertoire *DATA_PUMP_DIR* dans le compartiment Amazon S3 nommé *amzn-s3-demo-bucket*. Les fichiers sont chargés dans un dossier `dbfiles`. Dans cet exemple, le niveau GZIP de compression est *1*, qui est le niveau de compression le plus rapide.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
      p_bucket_name       =>  'amzn-s3-demo-bucket',
      p_prefix            =>  '',
      p_s3_prefix         =>  'dbfiles/',
      p_directory_name    =>  'DATA_PUMP_DIR',
      p_compression_level =>  1)
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge tous les fichiers d'un répertoire *DATA_PUMP_DIR* dans le compartiment Amazon S3 nommé *amzn-s3-demo-bucket*. Les fichiers sont chargés dans un dossier `dbfiles` et `ora` est ajouté au début de chaque nom de fichier. Aucune compression n'est appliquée.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
      p_bucket_name    =>  'amzn-s3-demo-bucket',
      p_prefix         =>  '',
      p_s3_prefix      =>  'dbfiles/ora',
```

```
      p_directory_name =>  'DATA_PUMP_DIR')
   AS TASK_ID FROM DUAL;
```

L'exemple suivant suppose que la commande est exécutée dans le compte A, mais le compte B exige un contrôle total du contenu du compartiment. La commande `rdsadmin_s3_tasks.upload_to_s3` transfère tous les fichiers dans le répertoire *DATA_PUMP_DIR* du compartiment nommé *s3bucketOwnedByAccountB*. Le contrôle d'accès est défini sur `FULL_CONTROL` afin que le compte B puisse accéder aux fichiers du compartiment. Le niveau GZIP de compression est *6*, qui équilibre la vitesse et la taille du fichier.

```
SELECT rdsadmin.rdsadmin_s3_tasks.upload_to_s3(
      p_bucket_name               =>  's3bucketOwnedByAccountB',
      p_prefix                    =>  '',
      p_s3_prefix                 =>  '',
      p_directory_name            =>  'DATA_PUMP_DIR',
      p_bucket_owner_full_control =>  'FULL_CONTROL',
      p_compression_level         =>  6)
   AS TASK_ID FROM DUAL;
```

Dans chaque exemple, l'instruction `SELECT` renvoie l'ID de la tâche dans un type de données `VARCHAR2`.

Vous pouvez afficher le résultat en affichant le fichier de sortie de la tâche.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-task-id.log'));
```

Remplacez *identifiant de tâche* avec l'ID de tâche renvoyé par la procédure.

> **Note**
>
> Les tâches sont exécutées de manière asynchrone.

Téléchargement des fichiers d'un compartiment Amazon S3 vers une instance de base de données Oracle

Pour télécharger des fichiers depuis un compartiment Amazon S3 vers une instance RDS pour Oracle, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_s3_tasks.download_from_s3`.

La procédure `download_from_s3` possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_bucket_name`	VARCHAI	–	Obligatoire	Nom du compartiment Amazon S3 à partir duquel télécharger les fichiers.
`p_directory_name`	VARCHAI	–	Obligatoire	Nom de l'objet de répertoire Oracle vers lequel les fichiers doivent être téléchargés. Le répertoire peut être n'importe quel objet de répertoire créé par l'utilisateur ou répertoire Data Pump, par exemple `DATA_PUMP_DIR` .
`p_error_on_zero_downloads`	VARCHAI	FALSE	Facultatif	Un indicateur qui détermine si la tâche génère une erreur quand aucun objet du compartiment Amazon S3 ne correspond au préfixe. Si ce paramètre n'est pas défini ou défini sur FALSE (par défaut), la tâche affiche un message indiquant qu'aucun objet n'a été trouvé, mais ne déclenche aucune exception ni n'échoue. Si ce paramètre est le casTRUE, la tâche déclenche une exception et échoue. Les exemples de spécifications de préfixe qui peuvent échouer aux tests de correspondance sont les espaces dans les préfixes, comme dans `' import/test9.log'` , et les différences de casse, comme dans `test9.log` et `test9.LOG` .
`p_s3_prefix`	VARCHAI	–	Obligatoire	Préfixe de nom de fichier que les noms de fichier doivent contenir pour être

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
				téléchargés. Un préfixe vide télécharge tous les fichiers au niveau le plus haut dans le compartiment Amazon S3 spécifié, mais pas les dossiers dans le compartiment. La procédure télécharge les objets Amazon S3 du dossier de premier niveau correspondant au préfixe. Les structures de répertoire imbriquées correspondant au préfixe spécifié ne sont pas téléchargées. Par exemple, supposons qu'un compartiment Amazon S3 ait la structure de dossiers `folder_1/folder_2/folder_3` . Vous spécifiez le préfixe `'folder_1/folder_2/'` . Dans ce cas, seuls les fichiers de `folder_2` sont téléchargés, et non ceux de `folder_1` ou `folder_3`. Si à la place vous spécifiez le préfixe `'folder_1/folder_2'` , tous les fichiers dans `folder_1` qui correspondent au préfixe `'folder_2'` sont téléchargés et aucun fichier dans `folder_2` n'est téléchargé.
`p_decompression_format`	VARCHAI	–	Facultatif	Format de décompression. Les valeurs valides sont `NONE` (pas de décompression) et `GZIP` (décompression).

La valeur renvoyée pour la procédure `rdsadmin.rdsadmin_s3_tasks.download_from_s3` est un ID de tâche.

L'exemple suivant télécharge tous les fichiers du compartiment Amazon S3 nommé *amzn-s3-demo-bucket* vers le répertoire *DATA_PUMP_DIR*. Les fichiers n'étant pas compressés, aucune décompression n'est appliquée.

```
SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
      p_bucket_name    =>  'amzn-s3-demo-bucket',
      p_directory_name =>  'DATA_PUMP_DIR')
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge tous les fichiers ayant le préfixe *db* dans le compartiment Amazon S3 nommé *amzn-s3-demo-bucket* du répertoire *DATA_PUMP_DIR*. Les fichiers sont compressés avecGZIP, donc la décompression est appliquée. Le paramètre `p_error_on_zero_downloads` active la vérification des erreurs de préfixe. Ainsi, si le préfixe ne correspond à aucun fichier du compartiment, la tâche lève une exception et échoue.

```
SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
      p_bucket_name               =>  'amzn-s3-demo-bucket',
      p_s3_prefix                 =>  'db',
      p_directory_name            =>  'DATA_PUMP_DIR',
      p_decompression_format      =>  'GZIP',
      p_error_on_zero_downloads   =>  'TRUE')
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge tous les fichiers du dossier *myfolder/* du compartiment Amazon S3 nommé *amzn-s3-demo-bucket* vers le répertoire *DATA_PUMP_DIR*. Utilisez le paramètre `p_s3_prefix` pour spécifier le dossier Amazon S3. Les fichiers téléchargés sont compressés avecGZIP, mais ne sont pas décompressés pendant le téléchargement.

```
SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
      p_bucket_name          =>  'amzn-s3-demo-bucket',
      p_s3_prefix            =>  'myfolder/',
      p_directory_name       =>  'DATA_PUMP_DIR',
      p_decompression_format =>  'NONE')
   AS TASK_ID FROM DUAL;
```

L'exemple suivant télécharge le fichier *mydumpfile.dmp* du compartiment Amazon S3 nommé *amzn-s3-demo-bucket* dans le répertoire *DATA_PUMP_DIR*. Aucune décompression n'est appliquée.

```
SELECT rdsadmin.rdsadmin_s3_tasks.download_from_s3(
      p_bucket_name    =>  'amzn-s3-demo-bucket',
      p_s3_prefix      =>  'mydumpfile.dmp',
      p_directory_name =>  'DATA_PUMP_DIR')
   AS TASK_ID FROM DUAL;
```

Dans chaque exemple, l'instruction `SELECT` renvoie l'ID de la tâche dans un type de données `VARCHAR2`.

Vous pouvez afficher le résultat en affichant le fichier de sortie de la tâche.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-task-id.log'));
```

Remplacez *identifiant de tâche* avec l'ID de tâche renvoyé par la procédure.

> **Note**
>
> Les tâches sont exécutées de manière asynchrone.
> Vous pouvez utiliser la procédure Oracle `UTL_FILE.FREMOVE` pour supprimer les fichiers d'un répertoire. Pour plus d'informations, consultez FREMOVEla procédure dans la documentation Oracle.

Surveillance du statut d'un transfert de fichiers

Les tâches de transfert de fichiers publient les RDS événements Amazon au début et à la fin. Le message d'événement contient l'ID de tâche pour le transfert de fichiers. Pour de plus amples informations sur l'affichage des événements, veuillez consulter Consulter les RDS événements Amazon.

Vous pouvez consulter le statut d'une tâche en cours dans un fichier bdump. Les fichiers bdump se trouvent dans le répertoire `/rdsdbdata/log/trace`. Chaque nom de fichier bdump a le format suivant.

```
dbtask-task-id.log
```

Remplacez *task-id* par l'ID de la tâche que vous souhaitez surveiller.

> **Note**
>
> Les tâches sont exécutées de manière asynchrone.

Vous pouvez utiliser la procédure stockée `rdsadmin.rds_file_util.read_text_file` pour afficher le contenu des fichiers bdump. Par exemple, la requête suivante renvoie le contenu du fichier bdump *dbtask-1234567890123-1234.log*.

```
SELECT text FROM
 table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-1234567890123-1234.log'));
```

L'exemple suivant montre le fichier journal correspondant à un transfert ayant échoué.

```
TASK_ID









----------------------------------------------------------------------------------------------------
1234567890123-1234


TEXT
```

```
------------------------------------------------------------------------------------------
2023-04-17 18:21:33.993 UTC [INFO ] File #1: Uploading the file /rdsdbdata/datapump/
A123B4CDEF567890G1234567890H1234/sample.dmp to Amazon S3 with bucket name amzn-s3-demo-
bucket and key sample.dmp.
2023-04-17 18:21:34.188 UTC [ERROR] RDS doesn't have permission to write to Amazon S3
 bucket name amzn-s3-demo-bucket and key sample.dmp.
2023-04-17 18:21:34.189 UTC [INFO ] The task failed.
```

Résolution des problèmes d'intégration avec Amazon S3

Pour obtenir des conseils de résolution des problèmes, consultez l'article AWS Re:Post Comment résoudre les problèmes liés à l'intégration d'Amazon RDS pour Oracle à Amazon S3 ? .

Suppression de l'option d'intégration Amazon S3

Vous pouvez supprimer l'option d'intégration Amazon S3 d'une instance de base de données.

Pour supprimer l'option d'intégration Amazon S3 d'une instance de base de données, effectuez l'une des actions suivantes :

- Pour supprimer l'option d'intégration Amazon S3 de plusieurs instances de base de données, supprimez l'option `S3_INTEGRATION` du groupe d'options auquel celles-ci appartiennent. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.
- Pour supprimer l'option d'intégration Amazon S3 d'une instance de base de données particulière, modifiez l'instance et spécifiez un autre groupe d'options qui n'inclut pas l'option `S3_INTEGRATION`. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Oracle Application Express (APEX)

Amazon RDS prend en charge Oracle Application Express (APEX) à l'aide des options `APEX` et `APEX-DEV`. Vous pouvez déployer Oracle APEX en tant qu'environnement d'exécution ou en tant qu'environnement de développement complet pour les applications Web. A l'aide d'Oracle APEX, vous pouvez créer des applications entièrement dans le navigateur web. Pour plus d'informations, consultez Oracle Application Express dans la documentation d'Oracle.

Rubriques

- Composants Oracle APEX
- Exigences et limitations
- Configuration d'Oracle APEX et d'Oracle Rest Data Services (ORDS)
- Configuration d'Oracle Rest Data Services (ORDS)
- Mise à niveau et suppression d'Oracle APEX

Composants Oracle APEX

Oracle APEX comprend les principaux composants suivants :

- Référentiel qui stocke les métadonnées des applications et des composants Oracle APEX. Le référentiel se compose de tables, d'index et d'autres objets installés dans votre instance de base de données Amazon RDS.
- Un écouteur qui gère les communications HTTP avec les clients Oracle APEX. L'écouteur réside sur un hôte distinct tel qu'une EC2 instance Amazon, un serveur local de votre entreprise ou votre ordinateur de bureau. L'écouteur accepte les connexions entrantes des navigateurs web et les transmet à l'instance de base de données Amazon RDS for traitement, puis renvoie les résultats du référentiel aux navigateurs.

  RDS for Oracle prend en charge les types d'écouteurs suivants :

  - Pour Oracle APEX version 5.0 et versions ultérieures, utilisez Oracle REST Data Services (ORDS) version 19.1 et supérieure. Nous vous recommandons d'utiliser la dernière version prise en charge d'Oracle APEX et ORDS. Cette documentation décrit les anciennes versions à des fins de compatibilité descendante uniquement.
  - Pour Oracle APEX version 4.1.1, vous pouvez utiliser Oracle APEX Listener version 1.1.4.
  - Vous pouvez utiliser Oracle HTTP Server et les écouteurs `mod_plsql`.

> **Note**
>
> Amazon RDS ne prend pas en charge le serveur HTTP Oracle XML DB avec PL/SQL gateway as a listener for Oracle APEX. In general, Oracle recommends against using the embedded PL/SQL passerelle intégrée pour les applications exécutées sur Internet.

Pour plus d'informations sur ces types d'écouteur, consultez About Choosing a Web Listener dans la documentation Oracle.

Lorsque vous ajoutez les `APEX-DEV` options `APEX` et à votre instance de base de données RDS pour Oracle, Amazon RDS installe uniquement le référentiel Oracle APEX. Installez votre écouteur sur un hôte distinct.

Exigences et limitations

La rubrique suivante répertorie les exigences et les limites d'Oracle APEX et ORDS.

Exigences relatives à la version d'Oracle APEX

L'`APEX`option utilise le stockage sur la classe d'instance de base de données de votre instance de base de données. Voici les versions prises en charge et les besoins approximatifs en stockage pour Oracle APEX.

Version d'Oracle APEX	Besoins de stockage	Versions de Oracle Database prises en charge	Remarques
Oracle APEX version 24.2.v1	114 MiB	Tous	Cette version inclut le correctif 37885097 : PSE BUNDLE POUR APEX 24.2 (PSES ON TOP OF 24.2.0), PATCH_VERSION 4.
Oracle APEX version 24.1.v1	112 MiB	Tous	Cette version inclut le correctif 36695709 : PSE BUNDLE POUR APEX 24.1 (PSES ON TOP OF 24.1.0), PATCH_VERSION 3. Si vous avez besoin d'installer exactement la même

Version d'Oracle APEX	Besoins de stockage	Versions de Oracle Database prises en charge	Remarques
			version d'images APEX sur votre EC2 instance, téléchargez le correctif 37544819 : BUNDLE PSE 24.1.3 POUR APEX 24.1 (PSES ON TOP OF 24.1.0).
Oracle APEX version 23.2.v1	110 MiB	Tous	Cette version inclut le correctif 35895964 : PSE BUNDLE POUR APEX 23.2 (PSES ON TOP OF 23.2.0), PATCH_VERSION 6. Si vous avez besoin d'installer exactement la même version d'images APEX sur votre EC2 instance, téléchargez le correctif 37593125 : 23.2.6 PSE BUNDLE POUR APEX 23.2 (PSES ON TOP OF 23.2.0).
Oracle APEX version 23.1.v1	106 Mio	Tous	Cette version inclut le correctif 35283657 : PSE BUNDLE FOR APEX 23.1 (PSES ON TOP OF 23.1.0), PATCH_VERSION 2.
Oracle APEX version 22.2.v1	106 Mio	Tous	Cette version inclut le correctif 34628174 : PSE BUNDLE FOR APEX 22.2 (PSES ON TOP OF 22.2.0), PATCH_VERSION 4.
Oracle APEX version 21.1.v1	124 Mio	Tous	Cette version inclut le correctif 34020981 : PSE BUNDLE FOR APEX 22.1 (PSES ON TOP OF 22.1.0), PATCH_VERSION 6.
Oracle APEX version 21.2.v1	125 Mio	Tous	Cette version inclut le correctif 33420059 : PSE BUNDLE FOR APEX 21.2 (PSES ON TOP OF 21.2.0), PATCH_VERSION 8.
Oracle APEX version 21.1.v1	125 Mio	Tous	Cette version inclut le patch 32598392: PSE BUNDLE FOR APEX 21.1, PATCH_VERSION 3.

Version d'Oracle APEX	Besoins de stockage	Versions de Oracle Database prises en charge	Remarques
Oracle APEX version 20.2.v1	148 Mio	Tous sauf Oracle Database 21c	Cette version inclut le patch 32006852: PSE BUNDLE FOR APEX 20.2, PATCH_VERSION 2020.11.12. Vous pouvez voir le numéro et la date du correctif en exécutant la requête suivante : `SELECT PATCH_VERSION, PATCH_NUMBER FROM APEX_PATCHES;`
Oracle APEX version 20.1.v1	173 Mio	Tous sauf Oracle Database 21c	Cette version inclut le patch 30990551: PSE BUNDLE FOR APEX 20.1, PATCH_VERSION 2020.07.15.
Oracle APEX version 19.2.v1	149 Mio	Tous sauf Oracle Database 21c	
Oracle APEX version 19.1.v1	148 Mio	Tous sauf Oracle Database 21c	

Pour les fichiers .zip d'Oracle APEX téléchargeables, consultez les archives des versions antérieures d'Oracle APEX sur le site Web d'Oracle.

Prérequis pour Oracle APEX et ORDS

Notez les conditions préalables suivantes pour utiliser Oracle APEX et ORDS :

- Votre système doit utiliser l'environnement d'exécution Java (JRE).

- Votre installation client Oracle doit comprendre les éléments suivants :
  - SQL*Plus ou SQL Developer pour les tâches d'administration
  - Oracle Net Services pour configurer les connexions à votre instance de base de données RDS for Oracle

Limites d'Oracle APEX

Vous ne pouvez pas modifier le compte `APEX_version` utilisateur, qui est géré par Amazon RDS. Vous ne pouvez donc pas appliquer de profils de base de données ni appliquer de règles de mot de passe à cet utilisateur. Les profils et les paramètres de mot de passe `APEX_version` sont prédéfinis par Oracle AWS et sont conçus pour répondre aux exigences de sécurité d'Amazon RDS.

Configuration d'Oracle APEX et d'Oracle Rest Data Services (ORDS)

La rubrique suivante répertorie les étapes requises pour configurer Oracle APEX et ORDS.

Rubriques

- Ajout des options APEX et APEX-DEV à votre instance de base de données
- Déverrouillage du compte utilisateur public sur votre instance de base de données
- Configuration des RESTful services pour Oracle APEX
- Préparation à l'installation d'ORDS sur un hôte distinct
- Configuration du processus d'écoute Oracle APEX

Ajout des options APEX et APEX-DEV à votre instance de base de données

Pour ajouter les `APEX-DEV` options `APEX` et à votre instance de base de données RDS pour Oracle, procédez comme suit :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez les options `APEX` et `APEX-DEV` au groupe d'options.
3. Associez le groupe d'options à votre instance de base de données.

Lorsque vous ajoutez les `APEX-DEV` options `APEX` et, une brève interruption se produit pendant le redémarrage automatique de votre instance de base de données.

> **Note**
>
> `APEX_MAIL` est disponible lorsque l'option `APEX` est installée. Le privilège d'exécution pour le package `APEX_MAIL` est accordé à `PUBLIC` et vous n'avez donc pas besoin du compte administratif APEX pour l'utiliser.

Pour ajouter les options APEX et APEX-DEV à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, sélectionnez l'édition d'Oracle que vous voulez utiliser. Les `APEX-DEV` options `APEX` et sont prises en charge sur toutes les éditions.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour plus d'informations, consultez Création d'un groupe d'options.

2. Ajouter les options au groupe d'options. Si vous souhaitez déployer uniquement l'environnement d'exécution Oracle APEX, ajoutez uniquement l'`APEX`option. Pour déployer l'environnement de développement complet, ajoutez à la fois les `APEX-DEV` options `APEX` et.

   Dans Version, choisissez la version d'Oracle APEX que vous souhaitez utiliser.

   > **Important**
   >
   > Si vous ajoutez les `APEX-DEV` options `APEX` ou à un groupe d'options existant déjà attaché à une ou plusieurs instances de base de données, une brève interruption se produit. Pendant cette interruption, toutes les instances de base de données sont automatiquement redémarrées.

   Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

- Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Lorsque vous ajoutez les APEX-DEV options APEX or à une instance de base de données existante, une brève interruption se produit pendant le redémarrage automatique de votre instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Déverrouillage du compte utilisateur public sur votre instance de base de données

Après avoir installé les APEX-DEV options APEX ou sur votre instance de base de données, veillez à effectuer les opérations suivantes :

1. Modifiez le mot de passe du APEX_PUBLIC_USER compte.
2. Déverrouillez le compte.

Pour cela, vous pouvez utiliser l'utilitaire de ligne de commande SQL*Plus Oracle. Connectez-vous à votre instance de base de données en tant qu'utilisateur principal et exécutez les commandes suivantes. Remplacez new_password par un mot de passe de votre choix.

```
ALTER USER APEX_PUBLIC_USER IDENTIFIED BY new_password;
ALTER USER APEX_PUBLIC_USER ACCOUNT UNLOCK;
```

Configuration des RESTful services pour Oracle APEX

Pour configurer les RESTful services dans Oracle APEX (ce n'est pas nécessaire pour Oracle APEX 4.1.1.V1), utilisez SQL*Plus pour vous connecter à votre instance de base de données en tant qu'utilisateur principal. Ensuite, exécutez la procédure stockée rdsadmin.rdsadmin_run_apex_rest_config. Lorsque vous exécutez la procédure stockée, vous fournissez les mots de passe des utilisateurs suivants :

- APEX_LISTENER
- APEX_REST_PUBLIC_USER

La procédure stockée exécute le script `apex_rest_config.sql`, qui crée de nouveaux comptes de base de données pour ces utilisateurs.

> **Note**
>
> Aucune configuration n'est requise pour Oracle APEX version 4.1.1.v1. Pour cette version d'Oracle APEX uniquement, vous n'avez pas besoin d'exécuter la procédure stockée.

La commande suivante exécute la procédure stockée.

```
EXEC rdsadmin.rdsadmin_run_apex_rest_config('apex_listener_password',
 'apex_rest_public_user_password');
```

Préparation à l'installation d'ORDS sur un hôte distinct

Installez ORDS sur un hôte distinct tel qu'une EC2 instance Amazon, un serveur local de votre entreprise ou votre ordinateur de bureau. Les exemples présentés dans cette section supposent que votre hôte exécute Linux et qu'il porte un nom`myapexhost.example.com`.

Avant d'installer ORDS, vous devez créer un utilisateur du système d'exploitation non privilégié, puis télécharger et décompresser le fichier d'installation d'Oracle APEX.

Pour préparer l'installation d'ORDS

1. Connectez-vous à `myapexhost.example.com` en tant que `root`.
2. Créez un utilisateur du système d'exploitation non privilégié qui sera propriétaire de l'installation de l'écouteur. La commande suivante crée un utilisateur nommé apexuser.

   ```
   useradd -d /home/apexuser apexuser
   ```

   La commande suivante affecte un mot de passe au nouvel utilisateur.

   ```
   passwd apexuser;
   ```
3. Connectez-vous à `myapexhost.example.com` as `apexuser` et téléchargez le fichier d'installation d'Oracle APEX depuis Oracle `/home/apexuser` dans votre répertoire :

   - http://www.oracle. com/technetwork/developer-tools/apex/downloads/index.html
   - Archives de versions antérieures d'Oracle Application Express

4. Décompressez le fichier dans le répertoire `/home/apexuser`.

   ```
   unzip apex_version.zip
   ```

   Une fois le fichier décompressé, un répertoire `apex` se trouve dans le répertoire `/home/apexuser`.
5. Pendant que vous êtes toujours connecté en `myapexhost.example.com` tant que`apexuser`, téléchargez le fichier Oracle REST Data Services depuis Oracle `/home/apexuser` dans votre répertoire : http://www.oracle. com/technetwork/developer-tools/apex-listener/downloads/index.html.

Configuration du processus d'écoute Oracle APEX

> **Note**
>
> L'écouteur Oracle APEX est obsolète.

Amazon RDS pour Oracle continue de prendre en charge Oracle APEX version 4.1.1 et Oracle APEX Listener version 1.1.4. Nous vous recommandons d'utiliser les dernières versions prises en charge d'Oracle APEX et d'ORDS.

Installez Oracle APEX Listener sur un hôte distinct tel qu'une EC2 instance Amazon, un serveur local de votre entreprise ou votre ordinateur de bureau. Nous partons du principe que le nom de votre hôte est `myapexhost.example.com` et qu'il exécute Linux.

Préparation de l'installation d'un écouteur Oracle APEX

Avant d'installer Oracle APEX Listener, vous devez créer un utilisateur du système d'exploitation non privilégié, puis télécharger et décompresser le fichier d'installation d'Oracle APEX.

Pour préparer l'installation de l'écouteur Oracle APEX

1. Connectez-vous à `myapexhost.example.com` en tant que `root`.
2. Créez un utilisateur du système d'exploitation non privilégié qui sera propriétaire de l'installation de l'écouteur. La commande suivante crée un utilisateur nommé apexuser.

   ```
   useradd -d /home/apexuser apexuser
   ```

La commande suivante affecte un mot de passe au nouvel utilisateur.

```
passwd apexuser;
```

3. Connectez-vous à `myapexhost.example.com` as `apexuser` et téléchargez le fichier d'installation d'Oracle APEX depuis Oracle `/home/apexuser` dans votre répertoire :

   - http://www.oracle. com/technetwork/developer-tools/apex/downloads/index.html
   - Archives de versions antérieures d'Oracle Application Express

4. Décompressez le fichier dans le répertoire `/home/apexuser`.

   ```
   unzip apex_<version>.zip
   ```

   Une fois le fichier décompressé, un répertoire `apex` se trouve dans le répertoire `/home/apexuser`.

5. Pendant que vous êtes toujours connecté à `myapexhost.example.com` en tant que `apexuser`, téléchargez le fichier de l'écouteur Oracle APEX depuis Oracle vers votre répertoire `/home/apexuser`.

Installation et configuration de l'écouteur Oracle APEX

Avant de pouvoir utiliser Oracle APEX, vous devez télécharger le `apex.war` fichier, utiliser Java pour installer Oracle APEX Listener, puis démarrer l'écouteur.

Pour installer et configurer l'écouteur Oracle APEX

1. Créez un répertoire basé sur l'écouteur Oracle APEX et ouvrez le fichier de l'écouteur.

   Exécutez le code suivant :

   ```
   mkdir /home/apexuser/apexlistener
   cd /home/apexuser/apexlistener
   unzip ../apex_listener.version.zip
   ```

2. Exécutez le code suivant.

   ```
   java -Dapex.home=./apex -Dapex.images=/home/apexuser/apex/images -Dapex.erase -
   jar ./apex.war
   ```

3. Entrez les informations suivantes à l'invitation du programme :

   - Nom d'utilisateur de l'administrateur APEX Listener. Le paramètre par défaut est adminlistener.
   - Un mot de passe pour l'administrateur APEX Listener.
   - Nom d'utilisateur du gestionnaire APEX Listener. La valeur par défaut est managerlistener.
   - Un mot de passe pour l'administrateur APEX Listener.

   Le programme imprime une URL dont vous avez besoin pour terminer la configuration, comme ci-dessous.

   ```
   INFO: Please complete configuration at: http://localhost:8080/apex/
   listenerConfigure
   Database is not yet configured
   ```

4. Continuez à exécuter l'écouteur Oracle APEX pour pouvoir utiliser Oracle Application Express. Lorsque vous avez terminé la procédure de configuration, vous pouvez exécuter l'écouteur à l'arrière-plan.
5. Dans votre navigateur Web, accédez à l'URL fournie par le programme Oracle APEX Listener. La fenêtre d'administration d'Oracle Application Express Listener s'affiche. Entrez les informations suivantes :

   - Nom d'utilisateur – `APEX_PUBLIC_USER`
   - Mot de passe – le mot de passe pour APEX_PUBLIC_USER. Ce mot de passe est celui que vous avez spécifié précédemment lors de la configuration du référentiel Oracle APEX. Pour de plus amples informations, veuillez consulter Déverrouillage du compte utilisateur public sur votre instance de base de données.
   - Type de connexion – Basic
   - Nom d'hôte – le point de terminaison de votre instance de base de données Amazon RDS, par exemple `mydb.f9rbfa893tft.us-east-1.rds.amazonaws.com`.
   - Port – 1521
   - SID – le nom de la base de données sur votre instance de base de données Amazon RDS, tel que `mydb`.
6. Choisissez Apply. La fenêtre d'administration d'Oracle APEX s'affiche.

7. Définissez un mot de passe pour l'`admin`utilisateur Oracle APEX. Pour ce faire, utilisez SQL*Plus pour vous connecter à votre instance de base de données en tant qu'utilisateur principal, puis exécutez les commandes suivantes.

   ```
   EXEC rdsadmin.rdsadmin_util.grant_apex_admin_role;
   grant APEX_ADMINISTRATOR_ROLE to master;
   @/home/apexuser/apex/apxchpwd.sql
   ```

   Remplacez *`master`* par votre identifiant principal. Lorsque le script `apxchpwd.sql` vous y invite, entrez un nouveau mot de passe `admin`.

8. Revenez à la fenêtre d'administration d'Oracle APEX dans votre navigateur et sélectionnez Administration. Ensuite, choisissez Application Express Internal Administration. Lorsque vous êtes invité à saisir les informations d'identification, entrez les informations suivantes :

   - Nom d'utilisateur – `admin`
   - Mot de passe – Mot de passe que vous avez défini en utilisant le script `apxchpwd.sql`.

   ChoisissezLogin, puis définissez un nouveau mot de passe pour l'utilisateur `admin`.

L'écouteur est maintenant prêt à être utilisé.

Configuration d'Oracle Rest Data Services (ORDS)

La rubrique suivante répertorie les options de configuration pour ORDS 21 et 22 :

Rubriques

- Installation et configuration d'ORDS 21 et versions antérieures
- Installation et configuration d'ORDS 22 et versions ultérieures

Installation et configuration d'ORDS 21 et versions antérieures

Vous êtes maintenant prêt à installer et à configurer Oracle Rest Data Services (ORDS) pour une utilisation avec Oracle APEX. Pour Oracle APEX version 5.0 et versions ultérieures, utilisez les versions 19.1 à 21 d'ORDS. Pour savoir comment installer ORDS 22 ou version ultérieure, voirInstallation et configuration d'ORDS 22 et versions ultérieures.

Installez l'écouteur sur un hôte distinct tel qu'une EC2 instance Amazon, un serveur local de votre entreprise ou votre ordinateur de bureau. Pour les exemples de cette section, nous supposons que le nom de votre hôte est `myapexhost.example.com`, et que votre hôte exécute Linux.

Pour installer et configurer ORDS 21 et versions antérieures pour une utilisation avec Oracle APEX

1. Accédez aux services de données Oracle REST et examinez le fichier Readme. Assurez-vous que la version requise de Java est installée.
2. Créez un nouveau répertoire pour votre installation ORDS.

   ```
   mkdir /home/apexuser/ORDS
   cd /home/apexuser/ORDS
   ```

3. Téléchargez le fichier `ords.`*`version.number`*`.zip` à partir de Oracle REST data services.
4. Décompressez le fichier dans le répertoire `/home/apexuser/ORDS`.
5. Si vous installez ORDS dans une base de données multi-locataires, ajoutez la ligne suivante au fichier `/home/apexuser/ORDS/params/ords_params.properties` :

   ```
   pdb.disable.lockdown=false
   ```

6. Accordez à l'utilisateur principal les privilèges nécessaires à l'installation d'ORDS.

   Une fois les options pour Oracle APEX installées, accordez à l'utilisateur principal les privilèges requis pour installer le schéma ORDS. Pour ce faire, connectez-vous à la base de données et exécutez les commandes suivantes. Remplacez *`MASTER_USER`* par le nom en majuscules de votre utilisateur principal.

   > **Important**
   >
   > Lorsque vous entrez le nom d'utilisateur, utilisez des majuscules, sauf si vous avez créé l'utilisateur avec un identifiant sensible à la casse. Par exemple, si vous exécutez `CREATE USER myuser` ou `CREATE USER MYUSER`, le dictionnaire de données stocke `MYUSER`. Toutefois, si vous utilisez des guillemets doubles dans `CREATE USER "MyUser"`, le dictionnaire de données stocke `MyUser`. Pour plus d'informations, consultez Octroi SELECT de EXECUTE privilèges à SYS des objets.

```
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_OBJECTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_ROLE_PRIVS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_TAB_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_CONS_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_CONSTRAINTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_OBJECTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_PROCEDURES', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_TAB_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_TABLES', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_VIEWS', 'MASTER_USER', 'SELECT',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('WPIUTL', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_SESSION', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_UTILITY', 'MASTER_USER',
 'EXECUTE', true);
```

> **Note**
>
> Ces commandes s'appliquent à ORDS versions 19.1 et ultérieures.

7. Installez le schéma ORDS à l'aide du fichier téléchargé ords.war.

```
java -jar ords.war install advanced
```

Le programme vous demande les informations suivantes. Les valeurs par défaut sont placées entre crochets. Pour de plus amples informations, veuillez consulter Introduction to Oracle REST Data Services dans la documentation Oracle.

- Entrez l'emplacement de stockage des données de configuration :

  Saisissez */home/apexuser/ORDS*. Il s'agit de l'emplacement des fichiers de configuration ORDS.

- Spécifiez le type de connexion à la base de données à utiliser. Entrez le numéro pour [1] Basic [2] TNS [3] Custom URL [1] :

  Choisissez le type de connexion souhaité.

- Entrez le nom du serveur de base de données [localhost] : *DB_instance_endpoint*

  Choisissez la valeur par défaut ou entrez la valeur correcte.

- Entrez le port de l'écouteur de base de données [1521] : *DB_instance_port*

  Choisissez la valeur par défaut ou entrez la valeur correcte.

- Entrez 1 pour spécifier le nom du service de base de données ou 2 pour spécifier le SID de base de données [1] :

  Choisissez 2 pour spécifier le SID de la base de données.

- SID de base de données [xe]

  Choisissez la valeur par défaut ou entrez la valeur correcte.

- Entrez 1 si vous souhaitez vérifier/installer le schéma Oracle REST Data Services ou 2 pour ignorer cette étape [1] :

  Choisissez 1. Cette étape crée l'utilisateur proxy Oracle REST Data Services nommé ORDS_PUBLIC_USER.

- Entrez le mot de passe de base de données pour ORDS_PUBLIC_USER :

  Entrez le mot de passe, puis confirmez-le.

- Nécessite une connexion avec des privilèges d'administrateur pour vérifier le schéma Oracle REST Data Services.

  Entrez le nom d'utilisateur de l'administrateur : *master_user*

  Entrez le mot de passe de base de données pour *master_user* : *master_user_password*

  Confirmez le mot de passe : *master_user_password*

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

- Entrez l'espace de table par défaut pour ORDS_METADATA [SYSAUX].

  Entrez l'espace de table temporaire pour ORDS_METADATA [TEMP].

  Entrez l'espace de table par défaut pour ORDS_PUBLIC_USER [USERS].

  Entrez l'espace de table temporaire pour ORDS_PUBLIC_USER [TEMP].
- Entrez 1 si vous voulez utiliser PL/SQL Gateway ou 2 pour passer cette étape. Si vous utilisez Oracle Application Express ou migrez à partir de mod_plsql, vous devez entrer 1 [1].

  Choisissez la valeur par défaut.
- Entrez le nom d'utilisateur de base de données de la passerelle PL/SQL [APEX_PUBLIC_USER]

  Choisissez la valeur par défaut.
- Entrez le mot de passe de base de données pour APEX_PUBLIC_USER :

  Entrez le mot de passe, puis confirmez-le.
- Entrez 1 pour spécifier les mots de passe des utilisateurs de la base de données Application Express RESTful Services (APEX_LISTENER, APEX_REST_PUBLIC_USER) ou 2 pour ignorer cette étape [1] :

  Choisissez 2 pour APEX 4.1.1.V1 ou 1 pour toutes les autres versions d'APEX.
- [Non nécessaire pour APEX 4.1.1.v1] Mot de passe de base de données pour APEX_LISTENER

  Entrez le mot de passe (si nécessaire), puis confirmez-le.
- [Non nécessaire pour APEX 4.1.1.v1] Mot de passe de base de données pour APEX_REST_PUBLIC_USER

  Entrez le mot de passe (si nécessaire), puis confirmez-le.
- Entrez un nombre pour sélectionner une fonction à activer :

Entrez 1 pour activer toutes les fonctions : SQL Developer Web, REST Enabled SQL et Database API.

- Entrez 1 si vous souhaitez démarrer en mode autonome ou 2 pour quitter [1] :

  Saisissez 1.

- Entrez l'emplacement des ressources statiques APEX :

  Si vous avez décompressé les fichiers d'installation APEX dans `/home/apexuser`, entrez `/home/apexuser/apex/images`. Sinon*`unzip_path`*`/apex/images`, entrez où se *`unzip_path`* trouve le répertoire dans lequel vous avez décompressé le fichier.

- Entrez 1 si vous utilisez HTTP ou 2 si vous utilisez HTTPS [1] :

  Si vous entrez 1, spécifiez le port HTTP. Si vous entrez 2, spécifiez le port HTTPS et le nom d'hôte SSL. L'option HTTPS vous invite à spécifier comment vous fournirez le certificat :

  - Entrez 1 pour utiliser le certificat auto-signé.
  - Entrez 2 pour fournir votre propre certificat. Si vous entrez 2, spécifiez le chemin d'accès du certificat SSL et le chemin d'accès de la clé privée du certificat SSL.

8. Définissez un mot de passe pour l'utilisateur `admin` APEX. Pour ce faire, utilisez SQL*Plus pour vous connecter à votre instance de base de données en tant qu'utilisateur principal, puis exécutez les commandes suivantes.

   ```
   EXEC rdsadmin.rdsadmin_util.grant_apex_admin_role;
   grant APEX_ADMINISTRATOR_ROLE to master;
   @/home/apexuser/apex/apxchpwd.sql
   ```

   Remplacez *`master`* par votre identifiant principal. Lorsque le script `apxchpwd.sql` vous y invite, entrez un nouveau mot de passe `admin`.

9. Démarrez l'écouteur ORDS. Exécutez le code suivant.

   ```
   java -jar ords.war
   ```

   La première fois que vous démarrez ORDS, vous devez fournir l'emplacement des ressources statiques APEX. Ce dossier d'images se trouve dans le répertoire `/apex/images` du répertoire d'installation d'APEX.

10. Revenez à la fenêtre d'administration d'Oracle APEX dans votre navigateur et sélectionnez Administration. Ensuite, choisissez Application Express Internal Administration. Lorsque vous êtes invité à saisir les informations d'identification, entrez les informations suivantes :

    - Nom d'utilisateur – `admin`
    - Mot de passe – Mot de passe que vous avez défini en utilisant le script `apxchpwd.sql`.

    ChoisissezLogin, puis définissez un nouveau mot de passe pour l'utilisateur `admin`.

L'écouteur est maintenant prêt à être utilisé.

Installation et configuration d'ORDS 22 et versions ultérieures

Vous êtes maintenant prêt à installer et à configurer Oracle Rest Data Services (ORDS) pour une utilisation avec Oracle APEX. Pour les exemples présentés dans cette section, nous partons du principe que le nom de votre hôte distinct est et que celui-ci exécute Linux. `myapexhost.example.com` Les instructions d'ORDS 22 sont différentes de celles des versions précédentes.

Pour installer et configurer ORDS 22 ou version ultérieure afin de l'utiliser avec Oracle APEX

1. Accédez aux services de données Oracle REST et examinez le fichier Readme correspondant à la version ORDS que vous souhaitez télécharger. Assurez-vous que la version requise de Java est installée.
2. Créez un nouveau répertoire pour votre installation ORDS.

   ```
   mkdir /home/apexuser/ORDS
   cd /home/apexuser/ORDS
   ```

3. Téléchargez le fichier `ords.`*`version.number`*`.zip` ou `ords-latest.zip` depuis les services de données Oracle REST.
4. Décompressez le fichier dans le répertoire `/home/apexuser/ORDS`.
5. Accordez à l'utilisateur principal les privilèges nécessaires à l'installation d'ORDS.

   Une fois l'`APEX`option installée, accordez à l'utilisateur principal les privilèges requis pour installer le schéma ORDS. Vous pouvez le faire en vous connectant à la base de données et en exécutant les commandes suivantes. Remplacez *`MASTER_USER`* par le nom en majuscules de votre utilisateur principal.

> **Important**
>
> Lorsque vous entrez le nom d'utilisateur, utilisez des majuscules, sauf si vous avez créé l'utilisateur avec un identifiant sensible à la casse. Par exemple, si vous exécutez `CREATE USER myuser` ou `CREATE USER MYUSER`, le dictionnaire de données stocke `MYUSER`. Toutefois, si vous utilisez des guillemets doubles dans `CREATE USER "MyUser"`, le dictionnaire de données stocke `MyUser`. Pour de plus amples informations, veuillez consulter Octroi SELECT de EXECUTE privilèges à SYS des objets.

```
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_OBJECTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_ROLE_PRIVS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_TAB_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_CONS_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_CONSTRAINTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_OBJECTS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_PROCEDURES', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_TAB_COLUMNS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_TABLES', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('USER_VIEWS', 'MASTER_USER', 'SELECT',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('WPIUTL', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_SESSION', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_UTILITY', 'MASTER_USER',
 'EXECUTE', true);

exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_LOB', 'MASTER_USER', 'EXECUTE',
 true);
```

```
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_ASSERT', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_OUTPUT', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_SCHEDULER', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('HTP', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('OWA', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('WPG_DOCLOAD', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_CRYPTO', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_METADATA', 'MASTER_USER',
 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_SQL', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('UTL_SMTP', 'MASTER_USER', 'EXECUTE',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_NETWORK_ACL_ADMIN',
 'MASTER_USER', 'EXECUTE', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('SESSION_PRIVS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_USERS', 'MASTER_USER', 'SELECT',
 true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_NETWORK_ACL_PRIVILEGES',
 'MASTER_USER', 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_NETWORK_ACLS', 'MASTER_USER',
 'SELECT', true);
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_REGISTRY', 'MASTER_USER',
 'SELECT', true);
```

> **Note**
>
> Les commandes précédentes s'appliquent à ORDS 22 et versions ultérieures.

6. Installez le schéma ORDS à l'aide du `ords` script téléchargé. Spécifiez les répertoires qui contiendront les fichiers de configuration et les fichiers journaux. Oracle Corporation recommande de ne pas placer ces répertoires dans le répertoire contenant le logiciel du produit ORDS.

```
mkdir -p /home/apexuser/ords_config /home/apexuser/ords_logs

/home/apexuser/ORDS/bin/ords \
  --config /home/apexuser/ords_config \
  install --interactive --log-folder /home/apexuser/ords_logs
```

Pour les instances de base de données exécutant l'architecture de base de données de conteneurs (CDB), utilisez ORDS 23.2 ou version ultérieure et transmettez l'`--pdb-skip-disable-lockdown`argument lors de l'installation d'ORDS.

```
/home/apexuser/ORDS/bin/ords \
  --config /home/apexuser/ords_config \
  install --interactive --log-folder /home/apexuser/ords_logs --pdb-skip-disable-
lockdown
```

Le programme vous demande les informations suivantes. Les valeurs par défaut sont placées entre crochets. Pour de plus amples informations, veuillez consulter Introduction to Oracle REST Data Services dans la documentation Oracle.

- `Choose the type of installation:`

  Choisissez **2** d'installer les schémas ORDS dans la base de données et de créer un pool de connexions à la base de données dans les fichiers de configuration ORDS locaux.

- `Specify the database connection type to use. Enter number for [1] Basic [2] TNS [3] Custom URL:`

  Choisissez le type de connexion souhaité. Cet exemple suppose que vous choisissez**1**.

- `Enter the name of the database server [localhost]:` ***`DB_instance_endpoint`***

  Choisissez la valeur par défaut ou entrez la valeur correcte.

- `Enter the database listener port [1521]:` ***`DB_instance_port`***

  Choisissez la valeur par défaut **1521** ou entrez la valeur correcte.

- `Enter the database service name [orcl]:`

  Entrez le nom de base de données utilisé par votre instance de base de données RDS pour Oracle.

- `Provide database user name with administrator privileges`

  Entrez le nom d'utilisateur principal pour votre instance de base de données RDS pour Oracle.

- `Enter the database password for [username]:`

  Entrez le mot de passe de l'utilisateur principal pour votre instance de base de données RDS pour Oracle.

- `Enter the default tablespace for ORDS_METADATA and ORDS_PUBLIC_USER [SYSAUX]:`
- `Enter the temporary tablespace for ORDS_METADATA [TEMP]. Enter the default tablespace for ORDS_PUBLIC_USER [USERS]. Enter the temporary tablespace for ORDS_PUBLIC_USER [TEMP].`
- `Enter a number to select additional feature(s) to enable [1]:`
- `Enter a number to configure and start ORDS in standalone mode [1]:`

  Choisissez **2** de ne pas démarrer ORDS immédiatement en mode autonome.

- `Enter a number to select the protocol [1] HTTP`
- `Enter the HTTP port [8080]:`
- `Enter the APEX static resources location:`

  Entrez le chemin d'accès aux fichiers d'installation d'Oracle APEX (`/home/apexuser/apex/images`).

7. Définissez un mot de passe pour l'`admin`utilisateur Oracle APEX. Pour ce faire, utilisez SQL*Plus pour vous connecter à votre instance de base de données en tant qu'utilisateur principal, puis exécutez les commandes suivantes.

   ```
   EXEC rdsadmin.rdsadmin_util.grant_apex_admin_role;
   grant APEX_ADMINISTRATOR_ROLE to master;
   @/home/apexuser/apex/apxchpwd.sql
   ```

   Remplacez *`master`* par votre identifiant principal. Lorsque le script `apxchpwd.sql` vous y invite, entrez un nouveau mot de passe `admin`.

8. Exécutez ORDS en mode autonome à l'aide du `ords` script associé à la `serve` commande. Pour les déploiements de production, pensez à utiliser des serveurs d'applications Java EE compatibles tels qu'Apache Tomcat ou Oracle WebLogic Server. Pour plus d'informations, voir

Déploiement et surveillance des services de données Oracle REST dans la documentation de la base de données Oracle.

```
/home/apexuser/ORDS/bin/ords \
  --config /home/apexuser/ords_config serve \
  --port 8193 \
  --apex-images /home/apexuser/apex/images
```

Si ORDS est en cours d'exécution mais ne parvient pas à accéder à l'installation d'Oracle APEX, l'erreur suivante peut s'afficher, en particulier sur les instances non CDB.

```
The procedure named apex_admin could not be accessed, it may not be declared,
 or the user executing this request may not have been granted execute privilege
 on the procedure, or a function specified by security.requestValidationFunction
 configuration property has prevented access.
```

Pour corriger cette erreur, modifiez la fonction de validation des demandes utilisée par ORDS en exécutant le `ords` script avec la `config` commande. Par défaut, ORDS utilise la `ords_util.authorize_plsql_gateway` procédure, qui n'est prise en charge que sur les instances CDB. Pour les instances non CDB, vous pouvez modifier cette procédure pour le `wwv_flow_epg_include_modules.authorize` package. Consultez la documentation de la base de données Oracle et le support Oracle pour connaître les meilleures pratiques relatives à la configuration de la fonction de validation des demandes adaptée à votre cas d'utilisation.

9. Revenez à la fenêtre d'administration d'Oracle APEX dans votre navigateur et sélectionnez Administration. Ensuite, choisissez Application Express Internal Administration. Lorsque vous êtes invité à saisir les informations d'identification, entrez les informations suivantes :

   - Nom d'utilisateur – `admin`
   - Mot de passe – Mot de passe que vous avez défini en utilisant le script `apxchpwd.sql`.

   ChoisissezLogin, puis définissez un nouveau mot de passe pour l'utilisateur `admin`.

L'écouteur est maintenant prêt à être utilisé.

Mise à niveau et suppression d'Oracle APEX

Pour mettre à niveau ou supprimer Oracle APEX, suivez les instructions de cette rubrique :

Rubriques

- Mise à niveau de la version Oracle APEX
- Suppression des options APEX et APEX-DEV

Mise à niveau de la version Oracle APEX

> **Important**
>
> Sauvegardez votre instance de base de données avant de mettre à niveau Oracle APEX. Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS et Test d'une mise à niveau de base de données Oracle.

Pour mettre à niveau Oracle APEX avec votre instance de base de données, procédez comme suit :

- Créez un nouveau groupe d'options pour la version mise à niveau de votre instance de base de données.
- Ajoutez les versions mises à niveau des `APEX-DEV` options `APEX` et au nouveau groupe d'options. Assurez-vous d'ajouter toutes les autres options utilisées par votre instance de base de données. Pour plus d'informations, consultez Considérations relatives au groupe d'options.
- Lorsque vous mettez à niveau votre instance de base de données, spécifiez le nouveau groupe d'options pour l'instance de base de données mise à niveau.

Après la mise à niveau de votre version d'Oracle APEX, le schéma Oracle APEX de la version précédente peut toujours exister dans votre base de données. Si vous n'en avez plus besoin, vous pouvez supprimer l'ancien schéma Oracle APEX de votre base de données après la mise à niveau.

Si vous mettez à niveau la version d'Oracle APEX et que les RESTful services n'étaient pas configurés dans la version précédente d'Oracle APEX, nous vous recommandons de configurer les RESTful services. Pour de plus amples informations, veuillez consulter Configuration des RESTful services pour Oracle APEX.

Dans certains cas, lorsque vous prévoyez d'effectuer une mise à niveau de version majeure de votre instance de base de données, vous constaterez peut-être que vous utilisez une version d'Oracle APEX qui n'est pas compatible avec la version de votre base de données cible. Dans ces cas, vous pouvez mettre à niveau votre version d'Oracle APEX avant de mettre à niveau votre instance de base

de données. La mise à niveau initiale d'Oracle APEX peut réduire le temps nécessaire à la mise à niveau de votre instance de base de données.

> **Note**
>
> Après la mise à niveau d'Oracle APEX, installez et configurez un écouteur à utiliser avec la version mise à niveau. Pour obtenir des instructions, veuillez consulter Configuration du processus d'écoute Oracle APEX.

Suppression des options APEX et APEX-DEV

Vous pouvez supprimer les `APEX-DEV` options `APEX` et d'une instance de base de données. Pour supprimer ces options de votre instance de base de données, effectuez l'une des opérations suivantes :

- Pour supprimer les `APEX-DEV` options `APEX` et de plusieurs instances de base de données, supprimez les options du groupe d'options auquel elles appartiennent. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Lorsque vous supprimez les options d'un groupe d'options attaché à plusieurs instances de base de données, une brève interruption se produit lors du redémarrage des instances de base de données.

  Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.
- Pour supprimer les `APEX-DEV` options `APEX` et d'une seule instance de base de données, modifiez l'instance de base de données et spécifiez un autre groupe d'options qui n'inclut pas ces options. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Lorsque vous supprimez les options, une brève interruption se produit pendant le redémarrage automatique de votre instance de base de données.

  Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Lorsque vous supprimez les `APEX-DEV` options `APEX` et d'une instance de base de données, le schéma APEX est supprimé de votre base de données.

EFSIntégration avec Amazon

Amazon Elastic File System (AmazonEFS) fournit un stockage de fichiers entièrement élastique sans serveur afin que vous puissiez partager des données de fichiers sans provisionner ni gérer la capacité et les performances de stockage. Avec AmazonEFS, vous pouvez créer un système de fichiers, puis le monter dans votre système VPC via les protocoles des NFS versions 4.0 et 4.1 (NFSv4). Vous pouvez ensuite utiliser le système de EFS fichiers comme n'importe quel autre système de fichiers POSIX compatible. Pour des informations générales, consultez Qu'est-ce qu'Amazon Elastic File System ? et le AWS blog Integrate Amazon RDS for Oracle with Amazon EFS.

Rubriques

- Présentation de l'EFSintégration avec Amazon
- Configuration des autorisations réseau RDS pour l'intégration d'Oracle à Amazon EFS
- Configuration des autorisations IAM pour l'intégration de RDS for Oracle avec Amazon EFS
- Ajouter l'INTEGRATIONoption EFS _
- Configuration des autorisations EFS du système de fichiers Amazon
- Transfert de fichiers entre un système de fichiers RDS pour Oracle et un système de EFS fichiers Amazon
- Supprimer l'INTEGRATIONoption EFS _
- Résolution des problèmes liés à EFS l'intégration avec Amazon

Présentation de l'EFSintégration avec Amazon

Avec AmazonEFS, vous pouvez transférer des fichiers entre votre instance de base de données RDS for Oracle et un système de EFS fichiers. Par exemple, vous pouvez utiliser EFS pour prendre en charge les cas d'utilisation suivants :

- Partagez un système de fichiers entre des applications et plusieurs serveurs de base de données.
- Créez un répertoire partagé pour les fichiers liés à la migration, y compris les fichiers de données d'espace de table transportable. Pour de plus amples informations, veuillez consulter Migration à l'aide des espaces de table transportables Oracle.
- Stockez et partagez les fichiers journaux redo sans allouer d'espace de stockage supplémentaire sur le serveur.

- Utilisez les utilitaires Oracle Database, tels que `UTL_FILE` pour lire et écrire des fichiers.

Avantages de l'EFSintégration avec Amazon

Lorsque vous choisissez un système de EFS fichiers plutôt que d'autres solutions de transfert de données, vous bénéficiez des avantages suivants :

- Vous pouvez transférer des fichiers Oracle Data Pump entre Amazon EFS et votre instance de base de données RDS for Oracle. Il n'est pas nécessaire de copier ces fichiers localement car Data Pump importe directement depuis le système de EFS fichiers. Pour de plus amples informations, veuillez consulter Importation de données dans Oracle sur Amazon RDS.
- La migration des données est plus rapide que l'utilisation d'un lien de base de données.
- Vous évitez d'allouer de l'espace de stockage sur votre instance de base de données Oracle RDS pour stocker les fichiers.
- Les systèmes de EFS fichiers peuvent automatiquement redimensionner le stockage sans que vous ayez à le provisionner.
- EFSL'intégration d'Amazon n'entraîne aucun frais minimum ni aucun coût d'installation. Vous ne payez que ce que vous utilisez.
- EFSL'intégration avec Amazon prend en charge deux formes de chiffrement : le chiffrement des données en transit et le chiffrement au repos. Le chiffrement des données en transit est activé par défaut dans TLS la version 1.2. Vous pouvez activer le chiffrement des données au repos lors de la création d'un système de EFS fichiers Amazon. Pour plus d’informations, consultez Chiffrement des données au repos dans le guide de l’utilisateur Amazon Elastic File System.

Exigences relatives à l'EFSintégration avec Amazon

Veillez à respecter les exigences suivantes :

- Votre base de données doit exécuter la version 19.0.0.0.ru-05.07.rur-05.07.r1 ou supérieure.
- Votre instance de base de données et votre système de EFS fichiers doivent être identiques Région AWS,VPC, et Compte AWS. RDSpour Oracle ne prend pas en charge l'accès entre comptes et entre régions pour. EFS
- La DNSrésolution et les DNSnoms d'hôtes VPC doivent être activés. Pour plus d'informations, consultez DNSles attributs dans le guide de l'utilisateur d'Amazon Virtual Private Cloud. VPC
- Votre système de EFS fichiers doit utiliser la classe de stockage Standard ou Standard-IA.

- Si vous utilisez un DNS nom dans la `mount` commande, assurez-vous que vous êtes VPC configuré pour utiliser le DNS serveur fourni par Amazon. Les DNS serveurs personnalisés ne sont pas pris en charge.
- Vous devez utiliser RDS des non-solutions pour sauvegarder votre système de EFS fichiers. RDSfor Oracle ne prend pas en charge les sauvegardes automatisées ni les instantanés manuels de base de données d'un système de EFS fichiers. Pour plus d'informations, consultez la section Sauvegarde de vos systèmes de EFS fichiers Amazon.

Configuration des autorisations réseau RDS pour l'intégration d'Oracle à Amazon EFS

RDSPour qu'Oracle puisse s'intégrer à AmazonEFS, assurez-vous que votre instance de base de données dispose d'un accès réseau à un système de EFS fichiers. Pour plus d'informations, consultez la section Contrôle de l'accès réseau aux systèmes de EFS fichiers Amazon pour les NFS clients dans le manuel Amazon Elastic File System User Guide.

Rubriques

- Contrôle de l'accès réseau avec les groupes de sécurité
- Contrôle de l'accès réseau avec des politiques de système de fichiers

Contrôle de l'accès réseau avec les groupes de sécurité

Vous pouvez contrôler l'accès de votre instance de base de données aux systèmes de EFS fichiers à l'aide de mécanismes de sécurité de la couche réseau tels que les groupes VPC de sécurité. Pour autoriser l'accès à un système de EFS fichiers pour votre instance de base de données, assurez-vous que celui-ci répond aux exigences suivantes : EFS

- Une cible de EFS montage existe dans chaque zone de disponibilité utilisée par une instance RDS de base de données Oracle.

  Une cible de EFS montage fournit l'adresse IP d'un NFSv4 point de terminaison sur lequel vous pouvez monter un système de EFS fichiers. Vous montez votre système de fichiers en utilisant son DNS nom, qui correspond à l'adresse IP de la cible de EFS montage dans la zone de disponibilité utilisée par votre instance de base de données.

  Vous pouvez configurer les instances de base de données de manière différente AZs pour utiliser le même système de EFS fichiers. Pour multi-AZ, vous avez besoin d'un point de montage pour chaque zone de disponibilité de votre déploiement. Vous pouvez avoir besoin de déplacer une

instance de base de données vers une autre zone de disponibilité. Pour ces raisons, nous vous recommandons de créer un point de EFS montage dans chaque AZ de votreVPC. Par défaut, lorsque vous créez un nouveau système de EFS fichiers à l'aide de la console, RDS des cibles de montage sont créées pour tousAZs.

- Un groupe de sécurité est attaché à la cible de montage.
- Le groupe de sécurité dispose d'une règle entrante autorisant le sous-réseau ou le groupe de sécurité de l'instance de base RDS de données Oracle sur TCP /2049 (Type). NFS

Pour plus d'informations, consultez les sections Création de systèmes de EFS fichiers Amazon et Création et gestion de cibles de EFS montage et de groupes de sécurité dans le guide de l'utilisateur d'Amazon Elastic File System.

Contrôle de l'accès réseau avec des politiques de système de fichiers

EFSL'intégration d'Amazon RDS à for Oracle fonctionne avec la politique du système de EFS fichiers par défaut (vide). La politique par défaut n'est pas utilisée IAM pour l'authentification. À la place, elle accorde un accès complet à tout client anonyme pouvant se connecter au système de fichiers à l'aide d'une cible de montage. La politique par défaut est en vigueur quand aucune politique de système de fichiers configurée par l'utilisateur n'est en vigueur, y compris au moment de la création du système de fichiers. Pour plus d'informations, consultez la section Politique EFS du système de fichiers par défaut dans le manuel Amazon Elastic File System User Guide.

Pour renforcer l'accès à votre système de EFS fichiers pour tous les clients, y compris RDS pour Oracle, vous pouvez configurer IAM les autorisations. Dans cette approche, vous créez une politique de système de fichiers. Pour plus d'informations, consultez Creating file system policies (Création de politiques de système de fichiers) dans le Guide de l'utilisateur Amazon Elastic File System.

Configuration des autorisations IAM pour l'intégration de RDS for Oracle avec Amazon EFS

Par défaut, la fonctionnalité d'intégration Amazon EFS n'utilise pas de rôle IAM : le paramètre de l'`USE_IAM_ROLE`option est `FALSE` défini comme suit. Pour intégrer RDS for Oracle à Amazon EFS et à un rôle IAM, votre instance de base de données doit disposer des autorisations IAM pour accéder à un système de fichiers Amazon EFS.

Rubriques

- Étape 1 : créer un rôle IAM pour votre instance de base de données et attacher votre politique
- Étape 2 : créer une politique de système de fichiers pour votre système de fichiers Amazon EFS

- Étape 3 : associer votre rôle IAM à votre instance de base de données RDS for Oracle.

Étape 1 : créer un rôle IAM pour votre instance de base de données et attacher votre politique

Dans cette étape, vous créez un rôle pour votre instance de base de données RDS for Oracle afin d'autoriser Amazon RDS à accéder à votre système de fichiers EFS.

console

Pour créer un rôle IAM afin d'autoriser Amazon RDS à accéder à un système de fichiers EFS

1. Ouvrez IAM Management Console.
2. Dans le panneau de navigation, choisissez Roles (Rôles).
3. Sélectionnez Create role (Créer un rôle).
4. Pour AWS Service, choisissez RDS.
5. Pour Sélectionner votre cas d'utilisation, choisissez RDS – Ajouter un rôle à la base de données.
6. Choisissez Suivant.
7. N'ajoutez aucune politique d'autorisation. Choisissez Suivant.
8. DansNom du rôle, attribuez un nom à votre rôle IAM, par exemple, `rds-efs-integration-role`. Vous pouvez également ajouter une valeur Description facultative.
9. Choisissez Créer un rôle.

AWS CLI

Pour limiter les autorisations du service à une ressource spécifique, nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des relations d'approbation basées sur les ressources. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Vous pouvez utiliser les deux clés de contexte de condition globale et faire en sorte que la valeur `aws:SourceArn` contienne l'ID de compte. Dans ce cas, la valeur `aws:SourceAccount` et le compte dans la valeur `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'ils sont utilisés dans la même instruction.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la relation d'approbation, assurez-vous d'utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'Amazon Resource Name (ARN) complet des ressources qui accèdent au rôle.

La AWS CLI commande suivante crée le rôle nommé *`rds-efs-integration-role`* à cet effet.

Example

Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
   --role-name rds-efs-integration-role \
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
                 "aws:SourceAccount": my_account_ID,
                 "aws:SourceArn": "arn:aws:rds:Region:my_account_ID:db:dbname"
             }
         }
       }
     ]
   }'
```

Dans Windows :

```
aws iam create-role ^
   --role-name rds-efs-integration-role ^
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
```

```
        "Action": "sts:AssumeRole",
        "Condition": {
            "StringEquals": {
                "aws:SourceAccount": my_account_ID,
                "aws:SourceArn": "arn:aws:rds:Region:my_account_ID:db:dbname"
            }
        }
      }
    ]
  }'
```

Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

Étape 2 : créer une politique de système de fichiers pour votre système de fichiers Amazon EFS

Dans cette étape, vous créez une politique de système de fichiers pour votre système de fichiers EFS.

Pour créer ou modifier une politique de système de fichiers EFS

1. Ouvrez la console de gestion EFS.
2. Choisissez Systèmes de fichiers.
3. Sur la page Système de fichiers, choisissez le système de fichiers pour lequel vous souhaitez modifier ou créer une politique de système de fichiers. La page de détails de ce système de fichiers s'affiche.
4. Choisissez l'onglet File system policy (Politique de système de fichiers).

   Si la politique est vide, la politique de système de fichiers EFS par défaut est utilisée. Pour plus d'informations, consultez Default EFS file system policy (Politique de système de fichiers EFS par défaut) dans le Guide de l'utilisateur Amazon Elastic File System.
5. Choisissez Modifier. La page Politique du système de fichiers s'affiche.
6. Dans Policy editor (Éditeur de politique), entrez une politique telle que la suivante, puis choisissez Enregistrer.

   JSON

```
{
    "Version": "2012-10-17",
```

```
    "Id": "ExamplePolicy01",
    "Statement": [
        {
            "Sid": "ExampleStatement01",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::123456789012:role/rds-efs-integration-
role"
            },
            "Action": [
                "elasticfilesystem:ClientMount",
                "elasticfilesystem:ClientWrite",
                "elasticfilesystem:ClientRootAccess"
            ],
            "Resource": "arn:aws:elasticfilesystem:us-
east-1:123456789012:file-system/fs-1234567890abcdef0"
        }
    ]
}
```

Étape 3 : associer votre rôle IAM à votre instance de base de données RDS for Oracle.

Dans cette étape, vous associez votre rôle IAM à votre instance de base de données. Tenez compte des exigences suivantes :

- Vous devez avoir accès à un rôle IAM auquel est attachée la politique d'autorisations Amazon EFS requise.
- Vous pouvez associer un seul rôle IAM à la fois avec votre instance de base de données RDS for Oracle.
- Le statut de votre instance doit être Available (Disponible).

Pour plus d'informations, consultez Identity and access management for Amazon EFS (Gestion des identités et des accès pour Amazon EFS) dans le guide de l'utilisateur Amazon Elastic File System.

console

Pour associer votre rôle IAM à votre instance de base de données RDS for Oracle

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Choisissez Databases (Bases de données).
3. Si votre instance de base de données n'est pas disponible, choisissez Actions , puis Start (Démarrer). Lorsque le statut de l'instance affiche Started (Démarré), passez à l'étape suivante.
4. Choisissez le nom de l'instance de base de données Oracle pour afficher ses détails.
5. Dans l'onglet Connectivity & security (Connectivité & sécurité), faites défiler l'écran jusqu'à l'onglet Manage IAM roles (Gérer les rôles IAM) au bas de la page.
6. Choisissez le rôle à ajouter dans la partie Add IAM roles to this instance (Ajouter des rôles IAM à cette instance).
7. Pour Feature (Fonction), choisissez EFS_INTEGRATION.
8. Choisissez Ajouter un rôle.

AWS CLI

La AWS CLI commande suivante ajoute le rôle à une instance de base de données Oracle nommée*mydbinstance*.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds add-role-to-db-instance \
   --db-instance-identifier mydbinstance \
   --feature-name EFS_INTEGRATION \
   --role-arn your-role-arn
```

Dans Windows :

```
aws rds add-role-to-db-instance ^
   --db-instance-identifier mydbinstance ^
   --feature-name EFS_INTEGRATION ^
   --role-arn your-role-arn
```

Remplacez *your-role-arn* par l'ARN du rôle que vous avez noté lors d'une étape précédente. `EFS_INTEGRATION` doit être spécifié pour l'option `--feature-name`.

Ajouter l'INTEGRATIONoption EFS _

Pour intégrer Amazon RDS pour Oracle à AmazonEFS, votre instance de base de données doit être associée à un groupe d'options qui inclut l'`EFS_INTEGRATION`option.

Plusieurs instances de base de données Oracle appartenant au même groupe d'options partagent le même système de EFS fichiers. Différentes instances de base de données peuvent accéder aux mêmes données, mais l'accès peut être divisé en utilisant différents répertoires Oracle. Pour plus d'informations, voir Transfert de fichiers entre un système de fichiers RDS pour Oracle et un système de EFS fichiers Amazon.

Console

Pour configurer un groupe d'options pour l'EFSintégration à Amazon

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `EFS_INTEGRATION`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'option `EFS_INTEGRATION` au groupe d'options. Vous devez spécifier l'ID de système de fichiers `EFS_ID` et définir l'indicateur `USE_IAM_ROLE`.

   Pour de plus amples informations, veuillez consulter Ajout d'une option à un groupe d'options.
3. Associez le groupe d'options à votre instance de base de données de l'une des manières suivantes :

   - Créez une nouvelle instance de base de données Oracle et associez-lui le groupe d'options. Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.
   - Modifiez une instance de base de données Oracle pour lui associer le groupe d'options. Pour de plus amples informations sur la modification d'une instance de base de données Oracle, veuillez consulter Modification d'une RDS instance de base de données Amazon.

AWS CLI

Pour configurer un groupe d'options à des fins EFS d'intégration

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `EFS_INTEGRATION`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'option `EFS_INTEGRATION` au groupe d'options.

   Par exemple, la AWS CLI commande suivante ajoute l'`EFS_INTEGRATION`option à un groupe d'options nommé**`myoptiongroup`**.

   Example

   Dans Linux, macOS, ou Unix:

   ```
   aws rds add-option-to-option-group \
      --option-group-name myoptiongroup \
      --options "OptionName=EFS_INTEGRATION,OptionSettings=\
      [{Name=EFS_ID,Value=fs-1234567890abcdef0},{Name=USE_IAM_ROLE,Value=TRUE}]"
   ```

   Dans Windows:

   ```
   aws rds add-option-to-option-group ^
      --option-group-name myoptiongroup ^
      --options "OptionName=EFS_INTEGRATION,OptionSettings=^
      [{Name=EFS_ID,Value=fs-1234567890abcdef0},{Name=USE_IAM_ROLE,Value=TRUE}]"
   ```

3. Associez le groupe d'options à votre instance de base de données de l'une des manières suivantes :

   - Créez une nouvelle instance de base de données Oracle et associez-lui le groupe d'options. Pour de plus amples informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.
   - Modifiez une instance de base de données Oracle pour lui associer le groupe d'options. Pour de plus amples informations sur la modification d'une instance de base de données Oracle, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Configuration des autorisations EFS du système de fichiers Amazon

Par défaut, seul l'utilisateur root (UID0) dispose des autorisations de lecture, d'écriture et d'exécution pour un système de EFS fichiers nouvellement créé. Pour que d'autres utilisateurs puissent modifier le système de fichiers, l'utilisateur racine doit leur accorder explicitement l'accès. L'utilisateur de l'instance de base de données RDS for Oracle se trouve dans la `others` catégorie. Pour plus d'informations, consultez la section Travailler avec les utilisateurs, les groupes et les autorisations

au niveau du système de fichiers réseau (NFS) dans le guide de l'utilisateur d'Amazon Elastic File System.

Pour permettre à votre RDS instance de base de données Oracle de lire et d'écrire des fichiers sur un EFS système de fichiers, procédez comme suit :

- Montez un système de EFS fichiers localement sur votre instance Amazon EC2 ou sur site.
- Configurez des autorisations détaillées.

Par exemple, pour autoriser `other` les utilisateurs à écrire à la racine du système de EFS fichiers, `chmod 777` exécutez ce répertoire. Pour plus d'informations, consultez la section Exemples de cas d'utilisation et d'autorisations du système de EFS fichiers Amazon dans le guide de l'utilisateur Amazon Elastic File System.

Transfert de fichiers entre un système de fichiers RDS pour Oracle et un système de EFS fichiers Amazon

Pour transférer des fichiers entre une instance RDS pour Oracle et un système de EFS fichiers Amazon, créez au moins un répertoire Oracle et configurez les autorisations du système de EFS fichiers pour contrôler l'accès aux instances de base de données.

Rubriques

- Création d'un répertoire Oracle
- Transfert de données vers et depuis un système de EFS fichiers : exemples

Création d'un répertoire Oracle

Pour créer un répertoire Oracle, utilisez la procédure `rdsadmin.rdsadmin_util.create_directory_efs`. La procédure possède les paramètres suivants.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
`p_directory_name`	VARCHAR	–	Oui	Nom du répertoire Oracle.

Nom du paramètre	Type de données	Par défaut	Obligatoire	Description
p_path_on_efs	VARCHAR	–	Oui	Le chemin sur le système de EFS fichiers. Le préfixe du nom du chemin utilise le modèle/rdsefs-*fsid*/, où *fsid* est un espace réservé pour l'ID de votre système de EFS fichiers. Par exemple, si votre système de EFS fichiers est nommé fs-1234567890abcdef0 et que vous créez un sous-répertoire sur ce système de fichiers nommémydir, vous pouvez spécifier la valeur suivante : `/rdsefs-fs-1234567890abcdef0/mydir`

Supposons que vous créez un sous-répertoire nommé dans /datapump1 le système de EFS fichiers. fs-1234567890abcdef0 L'exemple suivant crée un répertoire Oracle DATA_PUMP_DIR_EFS qui pointe vers le /datapump1 répertoire du système de EFS fichiers. La valeur du chemin du système de fichiers pour le paramètre p_path_on_efs est préfixée par la chaîne /rdsefs-.

```
BEGIN
  rdsadmin.rdsadmin_util.create_directory_efs(
    p_directory_name => 'DATA_PUMP_DIR_EFS',
    p_path_on_efs    => '/rdsefs-fs-1234567890abcdef0/datapump1');
END;
/
```

Transfert de données vers et depuis un système de EFS fichiers : exemples

L'exemple suivant utilise Oracle Data Pump pour exporter la table nommée MY_TABLE dans le fichier datapump.dmp. Ce fichier réside dans un système de EFS fichiers.

```
DECLARE
  v_hdnl NUMBER;
BEGIN
```

```
  v_hdnl := DBMS_DATAPUMP.OPEN(operation => 'EXPORT', job_mode => 'TABLE',
 job_name=>null);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'datapump.dmp',
    directory => 'DATA_PUMP_DIR_EFS',
    filetype  => dbms_datapump.ku$_file_type_dump_file);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'datapump-exp.log',
    directory => 'DATA_PUMP_DIR_EFS',
    filetype  => dbms_datapump.ku$_file_type_log_file);
  DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'NAME_EXPR','IN (''MY_TABLE'')');
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

L'exemple suivant utilise Oracle Data Pump pour importer la table nommée `MY_TABLE` depuis le fichier `datapump.dmp`. Ce fichier réside dans un système de EFS fichiers.

```
DECLARE
  v_hdnl NUMBER;
BEGIN
  v_hdnl := DBMS_DATAPUMP.OPEN(
    operation => 'IMPORT',
    job_mode  => 'TABLE',
    job_name  => null);
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'datapump.dmp',
    directory => 'DATA_PUMP_DIR_EFS',
    filetype  => dbms_datapump.ku$_file_type_dump_file );
  DBMS_DATAPUMP.ADD_FILE(
    handle    => v_hdnl,
    filename  => 'datapump-imp.log',
    directory => 'DATA_PUMP_DIR_EFS',
    filetype  => dbms_datapump.ku$_file_type_log_file);
  DBMS_DATAPUMP.METADATA_FILTER(v_hdnl,'NAME_EXPR','IN (''MY_TABLE'')');
  DBMS_DATAPUMP.START_JOB(v_hdnl);
END;
/
```

Pour de plus amples informations, veuillez consulter Importation de données dans Oracle sur Amazon RDS.

Supprimer l'INTEGRATIONoption EFS _

Les étapes de suppression de l'`EFS_INTEGRATIONoption` varient selon que vous supprimez l'option de plusieurs instances de base de données ou d'une seule instance.

Nombre d'instances de base de données	Action	Informations connexes
Plusieurs	Supprimez l'`EFS_INTEGRATION` option du groupe d'options auquel appartiennent les instances de base de données. Cette modification concerne toutes les instances qui utilisent le groupe d'options.	Suppression d'une option d'un groupe d'options
Unique	Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option `EFS_INTEGRATION` . Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent.	Modification d'une RDS instance de base de données Amazon

Après avoir supprimé l'`EFS_INTEGRATIONoption`, vous pouvez éventuellement supprimer le système de EFS fichiers connecté à vos instances de base de données.

Résolution des problèmes liés à EFS l'intégration avec Amazon

Votre instance de base de données RDS for Oracle surveille la connectivité à un système de EFS fichiers Amazon. Lorsque la surveillance détecte un problème, elle peut essayer de le corriger et de publier un événement dans la RDS console. Pour plus d'informations, consultez la section Affichage RDS des événements Amazon.

Utilisez les informations de cette section pour vous aider à diagnostiquer et à résoudre les problèmes courants liés à EFS l'intégration avec Amazon.

Notification	Description	Action
`The EFS for RDS Oracle instance instance_name isn't available on the primary host. NFS port 2049 of your EFS isn't reachable.`	L'instance de base de données ne peut pas communiquer avec le système de EFS fichiers.	Vérifiez les points suivants : • Le système de EFS fichiers existe. • Le groupe de sécurité attaché à la cible de EFS montage dispose d'une règle entrante autorisant le groupe de sécurité ou le sous-réseau de l'RDSinstance de base de données Oracle sur TCP /2049 (Type). NFS
`The EFS isn't reachable.`	Une erreur s'est produite lors de l'installation de l'option `EFS_INTEGRATION` .	Vérifiez les points suivants : • Le système de EFS fichiers existe. • Le groupe de sécurité attaché à la cible de EFS montage dispose d'une règle entrante autorisant le groupe de sécurité ou le sous-réseau de l'RDSinstance de base de données Oracle sur TCP /2049 (Type). NFS • L'`enableDnsSupport` attribut est activé pour votreVPC. • Vous utilisez le DNS serveur fourni par Amazon dans votreVPC. EFSL'intégration Amazon ne fonctionne pas

Notification	Description	Action
		avec une personnalisation DHCPDNS.
`The associated role with your DB instance wasn't found.`	Une erreur s'est produite lors de l'installation de l'option `EFS_INTEGRATION` .	Assurez-vous d'avoir associé un IAM rôle à votre instance de base de données RDS for Oracle.
`The associated role with your DB instance wasn't found.`	Une erreur s'est produite lors de l'installation de l'option `EFS_INTEGRATION` . RDSpour Oracle a été restauré à partir d'un instantané de base de données avec le paramètre d'`USE_IAM_ROLE` option de`TRUE`.	Assurez-vous d'avoir associé un IAM rôle à votre instance de base de données RDS for Oracle.
`The associated role with your DB instance wasn't found.`	Une erreur s'est produite lors de l'installation de l'option `EFS_INTEGRATION` . RDSpour Oracle a été créé à partir d'un all-in-one CloudFormation modèle avec le paramètre d'`USE_IAM_ROLE` option de`TRUE`.	Pour contourner le problème, procédez comme suit : 1. Créez une instance de base de données avec le IAM rôle et le groupe d'options par défaut. 2. Lors d'une mise à jour ultérieure de la pile, ajoutez le groupe d'options personnalisé avec l'`EFS_INTEGRATION` option.
`PLS-00302: component 'CREATE_DIRECTORY_EFS' must be declared`	Cette erreur peut se produire lorsque vous utilisez une version de RDS for Oracle qui ne prend pas en charge AmazonEFS.	Assurez-vous que vous utilisez RDS pour l'instance de base de données Oracle la version 19.0.0.0.ru-05.07.rur-05.07.r1 ou supérieure.

Notification	Description	Action
`Read access of your EFS is denied. Check your file system policy.`	Votre instance de base de données ne peut pas lire le système de EFS fichiers.	Assurez-vous que votre système de EFS fichiers autorise l'accès en lecture via le IAM rôle ou au niveau du système de EFS fichiers.
N/A	Votre instance de base de données ne peut pas écrire dans le système de EFS fichiers.	Suivez les étapes suivantes: 1. Assurez-vous que votre système de EFS fichiers est monté sur une EC2 instance Amazon. 2. Donnez au `others` groupe le droit d'écritur e à votre RDS utilisate ur. La technique la plus simple consiste à exécuter la `chmod 777` commande dans le répertoire supérieur du système de EFS fichiers.

Oracle Java Virtual Machine

Amazon RDS prend en charge Oracle Java Virtual Machine (JVM) par l'intermédiaire de l'option `JVM`. Oracle Java fournit un schéma SQL et des fonctions facilitant les fonctionnalités Oracle Java dans une base de données Oracle. Pour de plus amples informations, veuillez consulter Introduction to Java in Oracle Database dans la documentation Oracle. Vous pouvez utiliser Oracle JVM avec toutes les versions d'Oracle Database 21c (21.0.0) et d'Oracle Database 19c (19.0.0).

Considérations relatives à Oracle JVM

L'implémentation Java dans Amazon RDS a un jeu d'autorisations limité. L'utilisateur principal dispose du rôle `RDS_JAVA_ADMIN`, qui attribue un sous-ensemble des privilèges associés au rôle `JAVA_ADMIN`. Afin de répertorier les privilèges attribués au rôle `RDS_JAVA_ADMIN`, exécutez la requête suivante sur votre instance de base de données :

```
SELECT * FROM dba_java_policy
   WHERE grantee IN ('RDS_JAVA_ADMIN', 'PUBLIC')
   AND enabled = 'ENABLED'
   ORDER BY type_name, name, grantee;
```

Prérequis pour Oracle JVM

Les conditions suivantes sont requises pour utiliser Oracle Java :

- Votre instance de base de données doit être d'une classe de taille suffisante. Oracle Java n'est pas pris en charge pour les classes d'instance de base de données db.t3.small. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .
- Votre instance de base de données doit avoir l'option Auto Minor Version Upgrade (Mise à niveau automatique des versions mineures) activée. Cette option permet à votre instance de base de données de recevoir automatiquement les mises à niveau des versions mineures du moteur dès qu'elles sont disponibles. Amazon RDS utilise cette option pour mettre à jour votre instance de base de données vers le dernier PSU (Patch Set Update) ou la dernière mise à jour (RU) Oracle. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Bonnes pratiques pour Oracle JVM

Les bonnes pratiques suivantes sont requises pour utiliser Oracle Java :

- Pour une sécurité maximale, utilisez l'option `JVM` avec Secure Sockets Layer (SSL). Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).
- Configurez votre instance de base de données afin de restreindre l'accès réseau. Pour plus d'informations, consultez Scénarios d'accès à un d'instances de base de données dans un VPC et Utilisation d'un(e) instance de base de données dans un VPC.
- Mettez à jour la configuration de vos points de terminaison HTTPS pour prendre en charge la version TLSv1 2 si vous remplissez les conditions suivantes :
  - Vous utilisez Oracle Java Virtual Machine (JVM) pour connecter un point de terminaison HTTPS via TLSv1 nos protocoles TLSv1 1.1.
  - Votre terminal ne prend pas en charge le protocole TLSv1 2.2.
  - Vous n'avez pas appliqué la mise à jour de version d'avril 2021 à votre base de données Oracle.

  En mettant à jour la configuration de votre point de terminaison, vous vous assurez que la connectivité de la machine virtuelle Java au point de terminaison HTTPS continuera de fonctionner. Pour plus d'informations sur les modifications du protocole TLS dans l'environnement d'exécution Java et le kit de développement Java d'Oracle, consultez la section Oracle JRE and JDK Cryptographic Roadmap (Feuille de route pour la cryptographie de l'environnement d'exécution Java et du kit de développement Java d'Oracle).

Ajout de l'option Oracle JVM

La procédure générale suivante permet d'ajouter l'option `JVM` à une instance de base de données :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

L'ajout de l'option `JVM` entraîne une brève indisponibilité. Une fois que vous ajoutez l'option, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, Oracle Java est disponible.

> **Note**
>
> Durant cette interruption, les fonctions de vérification de mot de passe sont brièvement désactivées. Vous pouvez également vous attendre à voir des événements liés aux fonctions de vérification de mot de passe durant l'interruption. Les fonctions de vérification de mot de

passe sont activées de nouveau avant que l'instance de base de données Oracle ne soit disponible.

Pour ajouter l'option JVM à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   - Pour Moteur, choisissez le moteur de base de données utilisé par l'instance de base de données (oracle-ee, oracle-se, oracle-se1 ou oracle-se2).
   - Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour plus d'informations, consultez Création d'un groupe d'options.
2. Ajouter l'option JVM au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
4. Accordez les autorisations requises aux utilisateurs.

   L'utilisateur principal Amazon RDS est autorisé à utiliser l'option `JVM` par défaut. Si d'autres utilisateurs ont besoin de ces permissions, connectez-vous à l'instance de base de données en tant qu'utilisateur principal dans un client SQL et accordez les autorisations aux utilisateurs.

   L'exemple suivant accorde les autorisation d'utiliser l'option `JVM` à l'utilisateur `test_proc`.

```
create user test_proc identified by password;
CALL dbms_java.grant_permission('TEST_PROC',
 'oracle.aurora.security.JServerPermission', 'LoadClassInPackage.*', '');
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Une fois que les autorisations sont accordées à l'utilisateur, la requête suivante doit renvoyer un résultat.

```
select * from dba_java_policy where grantee='TEST_PROC';
```

> **Note**
>
> Le nom d'utilisateur Oracle est sensible à la casse et comportent généralement seulement des caractères majuscules.

Suppression de l'option Oracle JVM

Vous pouvez supprimer l'option `JVM` d'une instance de base de données. La suppression de l'option entraîne une brève indisponibilité. Une fois que vous supprimez l'option `JVM`, vous n'avez pas besoin de redémarrer votre instance de base de données.

> **Warning**
>
> La suppression de l'option `JVM` peut entraîner une perte de données si l'instance de base de données utilise des types de données qui ont été activés avec cette option. Sauvegardez vos données avant de continuer. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

Pour supprimer l'option `JVM` d'une instance de base de données, effectuez l'une des actions suivantes :

- Supprimez l'option `JVM` du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.
- Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option `JVM`. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Oracle Enterprise Manager

Amazon RDS prend en charge Oracle Enterprise Manager (OEM). OEMest la gamme de produits Oracle pour la gestion intégrée des technologies de l'information d'entreprise.

Amazon RDS prend en charge OEM sur Oracle Database 19c non- CDBs ouCDBs. Le tableau suivant décrit les OEM options prises en charge.

Option	ID d'option	OEMVersions prises en charge
OEMBase de données Express	`OEM`	OEMBase de données Express 19c
OEM Management Agent	`OEM_AGENT`	OEMCloud Control pour 13c

> **Note**
>
> Vous pouvez utiliser une OEM base de données ou un agent OEM de gestion, mais pas les deux.

Oracle Enterprise Manager Database Express

Amazon RDS prend en charge Oracle Enterprise Manager Database Express (EM Express) grâce à l'utilisation de OEM cette option. Amazon RDS prend en charge EM Express pour Oracle Database 19c en utilisant CDB ou non CDB l'architecture.

EM Express est un outil de gestion de base de données basé sur le Web inclus dans votre base de données et disponible uniquement lorsqu'il est ouvert. Il prend en charge la gestion des performances clés et les fonctions d'administration de base de données de base. Pour plus d'informations, reportez-vous à la section Présentation d'Oracle Enterprise Manager Database Express dans la documentation d'Oracle Database.

> **Note**
>
> EM Express n'est pas pris en charge sur la classe d'instance de base de données db.t3.small. Pour plus d'informations sur les classes d'instance de base de données, veuillez consulter RDS pour les classes d'instance de base de données Oracle.

OEMparamètres des options

Amazon RDS prend en charge les paramètres suivants pour OEM cette option.

Paramètre d'option	Valeurs valides	Description
Port	Une valeur d'entier	Port de l'instance RDS de base de données Oracle qui écoute EM Express. La valeur par défaut est 5500.
Groupes de sécurité	—	Un groupe de sécurité qui a accès à Port.

Étape 1 : ajout de l'OEMoption

Le processus général pour ajouter l'OEMoption à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.

3. Associez le groupe d'options à votre instance de base de données.

Lorsque vous ajoutez OEM cette option, une brève interruption se produit pendant le redémarrage automatique de votre instance de base de données.

Pour ajouter l'OEMoption à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, choisissez l'édition oracle pour votre instance de base de données.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'OEMoption au groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options. Pour plus d'informations sur chaque paramètre, consultez OEMparamètres des options.

   > **Note**
   >
   > Si vous ajoutez l'OEMoption à un groupe d'options existant déjà attaché à une ou plusieurs instances de base de données, une brève interruption se produit alors que toutes les instances de base de données sont automatiquement redémarrées.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Lorsque vous ajoutez OEM cette option, une brève interruption se produit pendant le redémarrage automatique de votre instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

> **Note**
>
> Vous pouvez également utiliser le AWS CLI pour ajouter l'OEMoption. Pour obtenir des exemples, consultez Ajout d'une option à un groupe d'options.

Étape 2 : (CDBuniquement) Déverrouillage du compte DBSNMP utilisateur

Si votre instance de base de données utilise cette CDB architecture, vous devez vous connecter à EM Express en tant queDBSNMP. in aCDB, DBSNMP est un utilisateur courant. Par défaut, ce compte est verrouillé. Si votre instance de base de données n'utilise pas cette CDB architecture, ignorez cette étape.

Pour déverrouiller le compte DBSNMP utilisateur dans une CDB instance

1. Dans SQL *Plus ou dans une autre SQL application Oracle, connectez-vous à votre instance de base de données en tant qu'utilisateur principal.
2. Exécutez la procédure stockée suivante pour déverrouiller le DBSNMP compte :

   ```
   EXEC rdsadmin.rdsadmin_util.reset_oem_agent_password('new_password');
   ```

   Si vous recevez un message d'erreur indiquant que la procédure n'existe pas, redémarrez votre CDB instance pour l'installer automatiquement. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Étape 3 : Accès à EM Express via votre navigateur

Lorsque vous accédez à EM Express depuis votre navigateur Web, une fenêtre de connexion s'affiche et vous invite à saisir un nom d'utilisateur et un mot de passe.

Pour accéder à EM Express via votre navigateur

1. Identifiez le point de terminaison et le port EM Express pour votre RDS instance de base de données Amazon. Pour plus d'informations sur la recherche du point de terminaison de votre RDS instance de base de données Amazon, consultezRecherche du point de terminaison de votre instance de base de données RDS for Oracle.
2. Entrez un URL dans la barre de localisation de votre navigateur en utilisant le format suivant.

```
https://endpoint.rds.amazonaws.com:port/em
```

Par exemple, si le point de terminaison de votre RDS instance de base de données Amazon est `mydb.a1bcde234fgh.us-east-1.rds.amazonaws.com` et que votre port EM Express l'est`1158`, utilisez ce qui suit URL pour accéder à EM Express.

```
https://mydb.f9rbfa893tft.us-east-1.rds.amazonaws.com:1158/em
```

3. Lorsque vous êtes invité à saisir vos informations de connexion, effectuez l'une des actions suivantes, en fonction de l'architecture de votre base de données :

   Votre base de données n'est pas une base de donnéesCDB.

   Tapez le nom d'utilisateur principal et le mot de passe principal de votre instance de base de données.

   Votre base de données est uneCDB.

   Entrez `DBSNMP` l'utilisateur et le `DBSNMP` mot de passe. Laissez le `Container` champ vide.

Modification des paramètres OEM de base de données

Après avoir activé la OEM base de données, vous pouvez modifier le paramètre des groupes de sécurité pour cette option.

Vous ne pouvez pas modifier le numéro de OEM port après avoir associé le groupe d'options à une instance de base de données. Pour modifier le numéro de OEM port d'une instance de base de données, procédez comme suit :

1. Créez un nouveau groupe d'options.
2. Ajoutez l'OEMoption avec le nouveau numéro de port au nouveau groupe d'options.
3. Supprimez le groupe d'options existant de l'instance de base de données.
4. Ajoutez le nouveau groupe d'options à l'instance de base de données.

Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option. Pour plus d'informations sur chaque paramètre, consultez OEMparamètres des options.

Exécution OEM de tâches Database Express

Vous pouvez utiliser RDS les procédures Amazon pour exécuter certaines tâches OEM de Database Express. Ces procédures vous permettent d'exécuter les tâches indiquées ci-après.

> **Note**
>
> OEMLes tâches Database Express s'exécutent de manière asynchrone.

Tâches

- Passage de l'interface du site Web pour OEM Database Express à Adobe Flash
- Passage du front-end du site Web pour OEM Database Express à Oracle JET

Passage de l'interface du site Web pour OEM Database Express à Adobe Flash

> **Note**
>
> Cette tâche n'est disponible que pour Oracle Database 19c non-CDBs.

À partir d'Oracle Database 19c, Oracle a abandonné l'ancienne interface utilisateur OEM de Database Express, qui était basée sur Adobe Flash. OEMDatabase Express utilise désormais une interface créée avec OracleJET. Si vous rencontrez des problèmes avec la nouvelle interface, vous pouvez revenir à l'interface obsolète basée sur Flash. Les difficultés que vous pourriez rencontrer avec la nouvelle interface incluent le fait de rester bloqué sur un `Loading` écran après la connexion à OEM Database Express. Certaines fonctionnalités présentes dans la version Flash de OEM Database Express risquent également de vous manquer.

Pour passer de l'OEMinterface du site Web Database Express à Adobe Flash, exécutez la RDS procédure Amazon`rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_flash`. Cette procédure est équivalente à la `execemx emx` SQL commande.

Les bonnes pratiques en matière de sécurité déconseillent l'utilisation d'Adobe Flash. Bien que vous puissiez revenir à OEM Database Express basé sur Flash, nous vous recommandons d'utiliser les sites Web OEM Database Express JET basés sur Flash dans la mesure du possible. Si vous revenez à l'utilisation d'Adobe Flash et souhaitez revenir à OracleJET, suivez la

`rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_jet` procédure ci-dessous. Après une mise à niveau de la base de données Oracle, une version plus récente d'Oracle JET peut résoudre JET les problèmes connexes dans OEM Database Express. Pour plus d'informations sur le passage à OracleJET, consultezPassage du front-end du site Web pour OEM Database Express à Oracle JET.

> **Note**
>
> L'exécution de cette tâche à partir de l'instance de base de données source pour une réplique en lecture entraîne également le transfert de ses OEM interfaces du site Web Database Express vers Adobe Flash.

L'appel de procédure suivant crée une tâche pour faire passer le site Web OEM de Database Express à Adobe Flash et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_flash() as TASK_ID from DUAL;
```

Vous pouvez afficher le résultat en affichant le fichier de sortie de la tâche.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-task-id.log'));
```

Remplacez *`identifiant de tâche`* avec l'ID de tâche renvoyé par la procédure. Pour plus d'informations sur la RDS procédure Amazon`rdsadmin.rds_file_util.read_text_file`, consultez Lecture de fichiers dans un répertoire d'instance de base de données

Vous pouvez également consulter le contenu du fichier de sortie de la tâche dans le AWS Management Console en recherchant dans les entrées du journal dans la section Journaux et événements le`task-id`.

Passage du front-end du site Web pour OEM Database Express à Oracle JET

> **Note**
>
> Cette tâche n'est disponible que pour Oracle Database 19c non-CDBs.

Pour passer du front-end du site Web OEM Database Express à OracleJET, exécutez la RDS procédure Amazon`rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_jet`. Cette procédure est équivalente à la `execemx omx` SQL commande.

Par défaut, les sites Web OEM Database Express pour les instances de base de données Oracle exécutant la version 19c ou ultérieure utilisent OracleJET. Si vous avez utilisé `rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_flash` cette procédure pour passer OEM de l'interface du site Web Database Express à Adobe Flash, vous pouvez revenir à OracleJET. Pour ce faire, utilisez la procédure `rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_jet`. Pour de plus amples informations sur le basculement vers Adobe Flash, veuillez consulter Passage de l'interface du site Web pour OEM Database Express à Adobe Flash.

> **Note**
>
> L'exécution de cette tâche à partir de l'instance de base de données source pour une réplique en lecture entraîne également le transfert de ses OEM interfaces du site Web Database Express vers OracleJET.

L'appel de procédure suivant crée une tâche pour transférer le site Web OEM de Database Express vers Oracle JET et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_tasks.em_express_frontend_to_jet() as TASK_ID from DUAL;
```

Vous pouvez afficher le résultat en affichant le fichier de sortie de la tâche.

```
SELECT text FROM table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-task-id.log'));
```

Remplacez *`identifiant de tâche`* avec l'ID de tâche renvoyé par la procédure. Pour plus d'informations sur la RDS procédure Amazon`rdsadmin.rds_file_util.read_text_file`, consultez Lecture de fichiers dans un répertoire d'instance de base de données

Vous pouvez également consulter le contenu du fichier de sortie de la tâche dans le AWS Management Console en recherchant dans les entrées du journal dans la section Journaux et événements le`task-id`.

Supprimer l'option de OEM base de données

Vous pouvez supprimer l'OEMoption d'une instance de base de données. Lorsque vous supprimez OEM cette option, une brève interruption se produit pendant le redémarrage automatique de votre instance. Par conséquent, une fois l'OEMoption supprimée, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'OEMoption d'une instance de base de données, effectuez l'une des opérations suivantes :

- Supprimez l'OEMoption du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.
- Modifiez l'instance de base de données et spécifiez un autre groupe d'options qui n'inclut pas l'OEMoption. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Oracle Management Agent pour Enterprise Manager Cloud Control

Oracle Enterprise Manager (OEM) Management Agent est un composant logiciel qui surveille les cibles en cours d'exécution sur les hôtes et communique ces informations au Oracle Management Service (OMS) de niveau intermédiaire. Amazon RDS prend en charge Management Agen via l'utilisation due l'option `OEM_AGENT`.

Pour plus d'informations, consultez les présentations Overview of Oracle Enterprise Manager Cloud Control 12c et Overview of Oracle Enterprise Manager Cloud Control 13c dans la documentation Oracle.

Rubriques

- Exigences relatives à l'agent de gestion
- Prérequis à la communication avec l'hôte OMS
- Limitations pour Management Agent
- Paramètres de l'option Management Agent
- Activation de l'option d'agent de gestion pour votre instance de base de données
- Suppression de l'option Management Agent
- Exécution de tâches de base de données avec l'option Management Agent

Exigences relatives à l'agent de gestion

Les exigences générales relatives à l'utilisation de l'agent de gestion sont les suivantes :

- Votre instance de base de données doit exécuter Oracle Database 19c (19.0.0.0). Vous pouvez utiliser l'architecture CDB ou non-CDB.
- Vous devez utiliser un service de gestion Oracle (OMS) configuré pour vous connecter à votre instance de base de données. Notez les exigences OMS suivantes :
  - Les versions 13.5.0.0.v2 et 13.5.0.0.v3 de l'agent de gestion nécessitent la version 13.5.0.23 ou 24.1 d'OMS.
  - La version 13.5.0.0.v1 de l'agent de gestion nécessite la version 13.5.0.0 ou 24.1 d'OMS.
  - Les versions 13.4.0.9.v1 et 13.4.0.9.v2 de l'agent de gestion nécessitent la version 13.4.0.9 ou ultérieure d'OMS et le correctif 32198287.
- Dans la plupart des cas, vous devez configurer votre VPC de sorte à autoriser les connexions entre OMS et votre instance de base de données. Si vous n'avez pas l'habitude de Amazon Virtual

Private Cloud (Amazon VPC), nous vous recommandons d'effectuer les étapes de Tutoriel : Création d'un VPC à utiliser avec un) IPv4 avant de poursuivre.

- Vous pouvez utiliser l'agent de gestion avec Oracle Enterprise Manager Cloud Control pour 12c et 13c. Vérifiez que vous disposez d'un espace de stockage suffisant pour votre version OEM :
  - Au moins 8,5 Gio pour OEM 13c version 5
  - Au moins 8,5 Gio pour OEM 13c version 4
  - Au moins 8,5 Gio pour OEM 13c version 3
  - Au moins 5,5 Gio pour OEM 13c version 2
  - Au moins 4,5 GiB pour la version 1 de l'OEM 13c
  - Au moins 2,5 Gio pour OEM 12c
- Si vous utilisez des versions `OEM_AGENT 13.2.0.0.v3` de l'agent de gestion et que vous souhaitez utiliser la connectivité TCPS, suivez les instructions de la section Configuration de certificats CA tiers pour la communication avec les bases de données cibles dans la documentation Oracle. `13.3.0.0.v2` De même, mettez à jour le JDK sur votre OMS en suivant les instructions du document Oracle dont l'ID de document Oracle est : 2241358.1. Cette étape garantit qu'OMS prendra en charge toutes les suites de chiffrement prises en charge par la base de données.

> **Note**
>
> La connectivité TCPS entre Management Agent et l'instance de base de données est uniquement prise en charge pour les versions `OEM_AGENT 13.2.0.0.v3`, `13.3.0.0.v2`, `13.4.0.9.v1` et ultérieures de Management Agent.

Prérequis à la communication avec l'hôte OMS

Vérifiez que votre hôte OMS et votre instance de base de données Amazon RDS peuvent communiquer. Procédez comme suit :

- Pour vous connecter depuis l'agent de gestion à votre hôte OMS lorsque celui-ci se trouve derrière un pare-feu, ajoutez les adresses IP de vos instances de base de données au pare-feu. Assurez-vous que le pare-feu de l'OMS autorise le trafic réseau suivant :

De l'hôte OMS à votre instance de base de données

Configurez une règle de pare-feu unidirectionnelle qui autorise le trafic depuis l'hôte OMS vers le port d'écoute de base de données (par défaut 1521) et le port de l'agent OEM (par défaut 3872).

De votre instance de base de données à l'hôte OMS

Configurez une règle de pare-feu unidirectionnelle qui autorise le trafic entre l'instance de base de données et le port HTTP OMS (par défaut 4903).

- Pour établir une connexion entre votre instance OMS et Management Agent, si votre instance OMS possède un nom d'hôte publiquement résolu, ajoutez l'adresse OMS à un groupe de sécurité. Votre groupe de sécurité doit avoir des règles de trafic entrant qui autorisent l'accès au port de l'instance de base de données et au port de Management Agent. Pour obtenir un exemple de création de règles de sécurité et d'ajout de règles de trafic entrant, consultez Tutoriel : Création d'un VPC à utiliser avec un) IPv4 .
- Pour vous connecter depuis votre OMS à l'agent de gestion, si votre OMS ne possède pas de nom d'hôte publiquement résolu, procédez d'une des manières suivantes :
  - Si votre OMS est hébergé sur une instance Amazon Elastic Compute Cloud (Amazon EC2) dans un VPC privé, vous pouvez configurer le peering VPC pour vous connecter de l'OMS à l'agent de gestion. Pour de plus amples informations, veuillez consulter Un d'instances de base de données dans un VPC accès par une EC2 instance dans un autre VPC.
  - Si votre OMS est hébergé sur site, vous pouvez configurer une connexion VPN pour autoriser l'accès entre OMS et l'agent de gestion. Pour plus d'informations, consultez Un d'instances de base de données dans une application cliente VPC accessible via Internet ou Connexions VPN.
- Pour connecter l'agent de gestion OEM version 13.5.0.0 (v1-v3) à un hôte OMS 24.1, configurez pour utiliser le protocole TLS `MINIMUM_TLS_VERSION` 1.2 dans vos options de configuration. `TLSv1.2`

Limitations pour Management Agent

Voici quelques limitations quant à l'utilisation de Management Agent :

- Vous ne pouvez pas fournir d'images personnalisées de l'agent de gestion Oracle.
- Les tâches d'administration telles que l'exécution de tâches et l'application de correctifs de bases de données, qui nécessitent des informations d'identification de l'hôte, ne sont pas prises en charge.

- Les métriques de l'hôte et la liste des processus ne reflètent pas nécessairement l'état réel du système. Par conséquent, vous ne devez pas utiliser OEM pour surveiller le système de fichiers racine ou le système de fichiers de point de montage. Pour de plus amples informations sur la surveillance du système d'exploitation, veuillez consulter Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.
- La découverte automatique n'est pas prise en charge. Vous devez ajouter manuellement les cibles de base de données.
- La disponibilité du module OMS dépend de l'édition de votre base de données. Par exemple, le module de diagnostic et de réglage des performances des bases de données n'est disponible que pour Oracle Database Enterprise Edition.
- Management Agent consomme des ressources de mémoire et de calcul supplémentaires. Si vous rencontrez des problèmes de performances après avoir activé l'option `OEM_AGENT`, nous vous recommandons d'augmenter la taille de la classe d'instance de base de données Pour plus d'informations, consultez Classes d'instances de base de données et Modification d'une RDS instance de base de données Amazon.
- L'utilisateur exécutant `OEM_AGENT` sur l'hôte Amazon RDS n'a pas accès au journal des alertes par le système d'exploitation. Vous ne pouvez donc pas collecter de métriques pour `DB Alert Log` et `DB Alert Log Error Status` dans OEM.

Paramètres de l'option Management Agent

Amazon RDS prend en charge les paramètres suivants pour l'option Management Agent.

Paramètre d'option	Obligatoire	Valeurs valides	Description

Version (`AGENT_VERSION`)	Oui	`13.5.0.0.v3`
		`13.5.0.0.v2`
		`13.5.0.0.v1`
		`13.4.0.9.v2`
			La version du logiciel Management Agent. La version minimale prise en charge est`13.1.0.0.v1`.
			Le nom de AWS CLI l'option est`OptionVersion`.

Paramètre d'option	Obligatoire	Valeurs valides	Description

		`13.4.0.9.v1`
		`13.3.0.0.v2`
		`13.3.0.0.v1`
		`13.2.0.0.v3`
		`13.2.0.0.v2`
		`13.2.0.0.v1`
		`13.1.0.0.v1`
			Note
			Dans les AWS GovCloud (US) régions, les versions 13.1 ne sont pas disponibles.

Port (`AGENT_PORT`)	Oui	Une valeur d'entier	Le port de l'instance de base de données qui écoute l'hôte OMS. La valeur par défaut est 3872. Votre hôte OMS doit appartenir à un groupe de sécurité qui a accès à ce port. Le nom de AWS CLI l'option est`Port`.
Groupes de sécurité	Oui	Groupes de sécurité existants	Un groupe de sécurité qui a accès à Port. Votre hôte OMS doit appartenir à ce groupe de sécurité. Le nom de l' AWS CLI option est `VpcSecurityGroupMemberships` ou`DBSecurityGroupMemberships` .

Paramètre d'option	Obligatoire	Valeurs valides	Description
OMS_HOST	Oui	Une valeur de chaîne, par exemple *my.example.oms*	Le nom d'hôte accessible au public ou l'adresse IP de l'OMS. Le nom de AWS CLI l'option est`OMS_HOST`.
OMS_PORT	Oui	Une valeur d'entier	Le port de chargement HTTPS de l'hôte OMS qui écoute l'agent de gestion. Pour déterminer le port de chargement the HTTPS, connectez-vous à l'hôte OMS et exécutez la commande suivante (sui nécessite le mot de passe `SYSMAN`) : `emctl status oms -details` Le nom de AWS CLI l'option est`OMS_PORT`.
AGENT_REGISTRATION_PASSWORD	Oui	Une valeur de chaîne	Le mot de passe que l'agent de gestion utilise pour s'authentifier auprès de l'OMS. Nous vous recommandons de créer un mot de passe permanent dans votre OMS avant d'activer l'option `OEM_AGENT` . Avec un mot de passe permanent, vous pouvez partager un groupe particulier de l'option Management Agent entre plusieurs bases de données Amazon RDS. Le nom de AWS CLI l'option est`AGENT_REGISTRATION_PASSWORD` .

Paramètre d'option	Obligatoire	Valeurs valides	Description
ALLOW_TLS_ONLY	Non	`true`, `false` (par défaut)	Valeur qui configure l'agent OEM pour ne prendre en charge que le protocole `TLSv1` pendant que l'agent écoute en tant que serveur. Ce paramètre n'est plus pris en charge. Les versions 13.1.0.0.v1 et supérieures de l'agent de gestion prennent en charge le protocole TLS (Transport Layer Security) par défaut.
MINIMUM_TLS_VERSION	Non	`TLSv1` (par défaut), `TLSv1.2`	Valeur qui spécifie la version TLS minimale prise en charge par l'agent OEM pendant que l'agent écoute en tant que serveur. Les versions d'agent non prises en charge ne prennent en charge que ce paramètre. `TLSv1` Pour connecter 13.5.0.0 (v1-v3) à un hôte OMS 24.1, définissez ce paramètre sur. `TLSv1.2`
TLS_CIPHER_SUITE	Non	Consultez Paramètres de l'option Management Agent.	Valeur qui spécifie la suite de chiffrement TLS utilisée par l'agent OEM pendant que l'agent écoute en tant que serveur.

La table suivante répertorie les suites de chiffrement TLS prises en charge par l'option Management Agent.

Suite de chiffrement	Version d'Agent prise en charge	Conforme au programme FedRAMP
TLS_RSA_WITH_AES_128_CBC_SHA	Tous	Non
TLS_RSA_WITH_AES_128_CBC_ SHA256	13.1.0.0.v1 et versions ultérieures	Non
TLS_RSA_WITH_AES_256_CBC_SHA	13.2.0.0.v3 et versions ultérieures	Non
TLS_RSA_WITH_AES_256_CBC_ SHA256	13.2.0.0.v3 et versions ultérieures	Non
TLS_ECDHE_RSA_WITH_AES_128_ CBC_SHA	13.2.0.0.v3 et versions ultérieures	Oui
TLS_ECDHE_RSA_WITH_AES_256_ CBC_SHA	13.2.0.0.v3 et versions ultérieures	Oui
TLS_ECDHE_RSA_WITH_AES_128_ CBC_ SHA256	13.2.0.0.v3 et versions ultérieures	Oui
TLS_ECDHE_RSA_WITH_AES_256_ CBC_ SHA384	13.2.0.0.v3 et versions ultérieures	Oui

Activation de l'option d'agent de gestion pour votre instance de base de données

Pour activer l'option Agent de gestion, procédez comme suit :

Rubriques

- Étape 1 : ajout de l'option d'agent de gestion à votre instance de base de données
- Étape 2 : Déverrouillage du compte utilisateur DBSNMP
- Étape 3 : ajout de vos cibles à la console de l'agent de gestion

Étape 1 : ajout de l'option d'agent de gestion à votre instance de base de données

Pour ajouter l'option d'agent de gestion à votre instance de base de données, procédez comme suit :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Si des erreurs se produisent, vous pouvez consulter les documents My Oracle Support pour savoir comment résoudre des problèmes spécifiques.

Après avoir ajouté l'option Management Agent, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, l'agent OEM est actif.

Si votre hôte OMS utilise un certificat tiers non approuvé, Amazon RDS renvoie l'erreur suivante :

```
You successfully installed the OEM_AGENT option. Your OMS host is using an untrusted
 third party certificate.
Configure your OMS host with the trusted certificates from your third party.
```

S'il vous renvoie une erreur, l'option Management Agent n'est pas activée avant que le problème soit résolu. Pour plus d'informations sur la résolution des problèmes, consultez le document du Support My Oracle 2202569.1.

console

Pour ajouter l'option d'agent de gestion à votre instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, choisissez l'édition oracle pour votre instance de base de données.
   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour plus d'informations, consultez Création d'un groupe d'options.
2. Ajoutez l'option OEM_AGENT pour le groupe d'options et configurez les paramètres de l'option. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options. Pour plus d'informations sur chaque paramètre, consultez Paramètres de l'option Management Agent.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

AWS CLI

L'exemple suivant utilise la commande AWS CLI add-option-to-option-group pour ajouter l'`OEM_AGENT`option à un groupe d'options appelé`myoptiongroup`.

Pour LinuxmacOS, ou Unix :

```
aws rds add-option-to-option-group \
    --option-group-name "myoptiongroup" \
    --options
 OptionName=OEM_AGENT,OptionVersion=13.1.0.0.v1,Port=3872,VpcSecurityGroupMemberships=sg-123456
{Name=OMS_PORT,Value=4903},{Name=AGENT_REGISTRATION_PASSWORD,Value=password}] \
    --apply-immediately
```

Dans Windows :

```
aws rds add-option-to-option-group ^
    --option-group-name "myoptiongroup" ^
    --options
 OptionName=OEM_AGENT,OptionVersion=13.1.0.0.v1,Port=3872,VpcSecurityGroupMemberships=sg-123456
{Name=OMS_PORT,Value=4903},{Name=AGENT_REGISTRATION_PASSWORD,Value=password}] ^
    --apply-immediately
```

Étape 2 : Déverrouillage du compte utilisateur DBSNMP

L'agent de gestion utilise le compte `DBSNMP` utilisateur pour se connecter à la base de données et signaler les problèmes à Oracle Enterprise Manager. Dans un CDB, `DBSNMP` c'est un utilisateur ordinaire. Ce compte utilisateur est nécessaire à la fois pour l'agent de gestion et pour OEM Database Express. Par défaut, ce compte est verrouillé. La procédure de déverrouillage de ce compte varie selon que votre base de données utilise une architecture non CDB ou CDB.

Pour déverrouiller le compte utilisateur DBSNMP

1. Dans SQL*Plus ou dans une autre application Oracle SQL, connectez-vous à votre instance de base de données en tant qu'utilisateur principal.
2. Effectuez l'une des actions suivantes, en fonction de l'architecture de base de données :

   Votre base de données n'est pas une base de données CDB.

   Exécutez l'instruction SQL suivante :

   ```
   ALTER USER dbsnmp IDENTIFIED BY new_password ACCOUNT UNLOCK;
   ```

   Votre base de données est un CDB.

   Exécutez la procédure stockée suivante pour déverrouiller le DBSNMP compte :

   ```
   EXEC rdsadmin.rdsadmin_util.reset_oem_agent_password('new_password');
   ```

   Si vous recevez un message d'erreur indiquant que la procédure n'existe pas, redémarrez votre instance CDB pour l'installer automatiquement. Pour de plus amples informations, veuillez consulter Redémarrage d'une instance de base de données.

Étape 3 : ajout de vos cibles à la console de l'agent de gestion

Pour ajouter une instance de base de données en tant que cible, assurez-vous de connaître le point de terminaison et le port. Pour plus d'informations sur la recherche du point de terminaison de votre instance de base de données Amazon RDS, consultez Recherche du point de terminaison de votre instance de base de données RDS for Oracle. Si votre base de données utilise l'architecture CDB, ajoutez le CDB$ROOT conteneur séparément en tant que cible.

Pour ajouter des cibles à la console de l'agent de gestion

1. Dans la console OMS, choisissez Setup (Configuration), Add Target (Ajouter une cible), Add Targets Manually (Ajouter des cibles manuellement).
2. Choisissez Add Targets Declaratively by Specifying Target Monitoring Properties (Ajouter déclarativement des cibles en spécifiant les propriétés de surveillance cibles).
3. Pour Target Type (Type de cible), choisissez Database Instance (Instance de base de données).

4. Pour Monitoring Agent (Agent de surveillance), sélectionnez l'agent ayant le même identifiant que votre identifiant d'instance de base de données RDS.
5. Choisissez Add Manually (Ajouter manuellement).
6. Entrez le point de terminaison de votre instance de base de données Amazon RDS ou choisissez-le dans la liste des noms d'hôte. Vérifiez que le nom d'hôte spécifié correspond au point de terminaison de l'instance de base de données Amazon RDS.
7. Spécifiez les propriétés de base de données suivantes :
   - Pour Target Name (Nom de la cible), entrez un nom.
   - Pour Database system name (Nom du système de base de données), entrez un nom.
   - Pour Monitor username (Nom d'utilisateur de surveillance), entrez **`dbsnmp`**.
   - Dans le champ Mot de passe du moniteur, entrez le mot de passe deÉtape 2 : Déverrouillage du compte utilisateur DBSNMP.
   - Pour Role (Rôle), entrez normal.
   - Pour Oracle home path (Chemin d'origine Oracle), entrez **`/oracle`**.
   - Pour Listener Machine name (Nom de machine d'écouteur), l'identifiant de l'agent est déjà affiché.
   - Pour Port, entrez le port de la base de données. La port RDS par défaut est 1521.
   - Pour Database name (Nom de la base de données), entrez le nom de votre base de données. Si votre base de données est une base de données CDB, ce nom est`RDSCDB`.
8. Choisissez Test Connection (Connexion test).
9. Choisissez Suivant. La base de données cible apparaît dans votre liste de ressources surveillées.

Suppression de l'option Management Agent

Vous pouvez supprimer l'agent OEM d'une instance de base de données. Lorsque vous avez supprimé l'agent OEM, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'agent OEM d'une instance de base de données, effectuez l'une des actions suivantes :

- Supprimez l'option OEM Agent du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.

- Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option OEM Agent. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Exécution de tâches de base de données avec l'option Management Agent

Vous pouvez utiliser les procédures Amazon RDS for exécuter certaines commandes EMCTL sur Management Agent. Ces procédures vous permettent d'exécuter les tâches indiquées ci-après.

> **Note**
>
> Les tâches sont exécutées de manière asynchrone.

Tâches

- Sécurisation de l'agent de gestion
- Obtenir le statut de l'agent de gestion
- Redémarrage du Management Agent
- Liste des cibles surveillées par le Management Agent
- Constitution de la liste des threads de collecte surveillés par Management Agent
- Suppression de l'état du Management Agent
- Chargement de l'OMS par le Management Agent
- Envoi de la commande Ping vers l'OMS
- Affichage du statut d'une tâche en cours

Sécurisation de l'agent de gestion

Pour sécuriser l'agent de gestion, exécutez la procédure `rdsadmin.rdsadmin_oem_agent_tasks.secure_oem_agent` Amazon RDS. Cette procédure équivaut à exécuter la commande `emctl secure agent`.

La procédure suivante crée une tâche pour sécuriser l'agent de gestion et renvoie l'ID de la tâche :

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.secure_oem_agent as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Obtenir le statut de l'agent de gestion

Pour obtenir le statut de l'agent de gestion, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.get_status_oem_agent`. Cette procédure est équivalente à la commande `emctl status agent`.

La procédure suivante crée une tâche permettant d'obtenir le statut du Management Agent et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.get_status_oem_agent() as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Redémarrage du Management Agent

Pour redémarrer le Management Agent, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.restart_oem_agent`. Cette procédure équivaut à exécuter les commandes `emctl stop agent` et `emctl start agent`.

La procédure suivante crée une tâche permettant de redémarrer Management Agent et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.restart_oem_agent as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Liste des cibles surveillées par le Management Agent

Pour répertorier les cibles surveillées par le Management Agent, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.list_targets_oem_agent`. Cette procédure équivaut à exécuter la commande `emctl config agent listtargets`.

La procédure suivante crée une tâche pour répertorier les cibles surveillées par Management Agent et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.list_targets_oem_agent as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Constitution de la liste des threads de collecte surveillés par Management Agent

Pour répertorier tous les threads de collecte (en cours d'exécution, prêts et planifiés) surveillés par Management Agent, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.list_clxn_threads_oem_agent`. Cette procédure est équivalente à la commande `emctl status agent scheduler`.

La procédure suivante crée une tâche pour répertorier les threads de collecte et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.list_clxn_threads_oem_agent() as TASK_ID from
 DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Suppression de l'état du Management Agent

Pour supprimer l'état du Management Agent, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.clearstate_oem_agent`. Cette procédure équivaut à exécuter la commande `emctl clearstate agent`.

La procédure suivante crée une tâche permettant d'annuler le statut de Management Agent et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.clearstate_oem_agent() as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Chargement de l'OMS par le Management Agent

Pour que le Management Agent charge l'OMS (Oracle Management Server) qui lui est associé, exécutez la procédure Amazon RDS

`rdsadmin.rdsadmin_oem_agent_tasks.upload_oem_agent`. Cette procédure équivaut à exécuter la commande `emclt upload agent`.

La procédure suivante crée une tâche dans laquelle Management Agent charge son OMS associé et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.upload_oem_agent() as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Envoi de la commande Ping vers l'OMS

Pour envoyer la commande Ping à l'OMS du Management Agent, exécutez la procédure Amazon RDS `rdsadmin.rdsadmin_oem_agent_tasks.ping_oms_oem_agent`. Cette procédure équivaut à exécuter la commande `emctl pingOMS`.

La procédure suivante crée une tâche permettant de pinger l'OMS de Management Agent et renvoie l'ID de la tâche.

```
SELECT rdsadmin.rdsadmin_oem_agent_tasks.ping_oms_oem_agent() as TASK_ID from DUAL;
```

Pour afficher le fichier de sortie de la tâche et visualiser le résultat, voirAffichage du statut d'une tâche en cours.

Affichage du statut d'une tâche en cours

Vous pouvez consulter le statut d'une tâche en cours dans un fichier bdump. Les fichiers bdump se trouvent dans le répertoire `/rdsdbdata/log/trace`. Chaque nom de fichier bdump a le format suivant.

```
dbtask-task-id.log
```

Lorsque vous souhaitez surveiller une tâche, remplacez *`task-id`* par l'ID de la tâche que vous souhaitez surveiller.

Pour afficher le contenu des fichiers bdump, exécutez la procédure Amazon RDS `rdsadmin.rds_file_util.read_text_file`. La requête suivante renvoie le contenu du fichier bdump `dbtask-1546988886389-2444.log`.

```
SELECT text FROM
 table(rdsadmin.rds_file_util.read_text_file('BDUMP','dbtask-1546988886389-2444.log'));
```

Pour de plus amples informations sur la procédure Amazon RDS `rdsadmin.rds_file_util.read_text_file`, veuillez consulter Lecture de fichiers dans un répertoire d'instance de base de données.

Oracle Label Security

Amazon RDS prend en charge Oracle Label Security pour l'édition Enterprise d'Oracle Database grâce à l'utilisation de OLS cette option.

La plupart des contrôles de sécurité de base de données accèdent au niveau de l'objet. Oracle Label Security offre un contrôle d'accès précis aux lignes de table individuelles. Par exemple, vous pouvez utiliser Label Security pour assurer la conformité réglementaire avec un modèle d'administration basé sur politique. Vous pouvez utiliser des politiques Label Security pour contrôler l'accès aux données sensibles et limiter l'accès aux seuls utilisateurs dotés du niveau d'autorisation approprié. Pour plus d'informations, consultez Introduction to Oracle Label Security dans la documentation Oracle.

Rubriques

- Exigences relatives à Oracle Label Security
- Considérations relatives à l'utilisation d'Oracle Label Security
- Ajout de l'option Oracle Label Security
- Résolution des problèmes

Exigences relatives à Oracle Label Security

Familiarisez-vous avec les exigences suivantes pour Oracle Label Security :

- Votre instance de base de données doit utiliser le modèle Bring Your Own License. Pour plus d'informations, consultez RDSpour les options de licence Oracle.
- Vous devez disposer d'une licence valide pour Oracle Enterprise Edition avec la licence de mise à jour du logiciel et le support.
- Votre licence Oracle doit inclure l'option Label Security.

Considérations relatives à l'utilisation d'Oracle Label Security

Pour utiliser Oracle Label Security, vous créez des politiques qui contrôlent l'accès à des lignes spécifiques dans vos tables. Pour plus d'informations, consultez Creating an Oracle Label Security Policy (Créer une stratégie Oracle Label Security) dans la documentation Oracle.

Éléments à prendre en compte :

- Oracle Label Security est une option permanente et persistante. Comme cette option est permanente, vous ne pouvez pas la supprimer d'un groupe d'options. Si vous ajoutez Oracle Label Security à un groupe d'options et que vous l'associez à votre instance de base de données, vous pouvez ultérieurement associer un autre groupe d'options à votre instance de base de données, mais ce groupe doit également contenir l'option Oracle Label Security.
- Lorsque vous travaillez avec Label Security, vous effectuez toutes les actions en tant que `LBAC_DBA` rôle. Le `LBAC_DBA` rôle est attribué à l'utilisateur principal de votre instance de base de données. Vous pouvez attribuer le `LBAC_DBA` rôle à d'autres utilisateurs afin qu'ils puissent administrer les politiques de sécurité des étiquettes.
- Assurez-vous d'accorder l'accès au `OLS_ENFORCEMENT` package à tous les nouveaux utilisateurs qui ont besoin d'accéder à Oracle Label Security. Pour accorder l'accès au `OLS_ENFORCEMENT` package, connectez-vous à l'instance de base de données en tant qu'utilisateur principal et exécutez l'SQLinstruction suivante :

```
GRANT ALL ON LBACSYS.OLS_ENFORCEMENT TO username;
```

- Vous pouvez configurer Label Security via Oracle Enterprise Manager (OEM) Cloud Control. Amazon RDS prend en charge OEM le contrôle du cloud via l'option d'agent de gestion. Pour de plus amples informations, veuillez consulter Oracle Management Agent pour Enterprise Manager Cloud Control.

Ajout de l'option Oracle Label Security

Le processus général d'ajout de l'option Oracle Label Security à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.

> **Important**
>
> Oracle Label Security est une option permanente et persistante.

3. Associez le groupe d'options à l'instance de base de données.

Lorsque vous ajoutez l'option Label Security, dès que le groupe d'options est actif, l'option Label Security est active.

Pour ajouter l'option Label Security à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, choisissez oracle-ee.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'OLSoption au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.

   > **Important**
   >
   > Si vous ajoutez l'option Label Security dans un groupe d'options existant qui est déjà attaché à une ou plusieurs instances de base de données, toutes les instances de base de données sont redémarrées.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Lorsque vous ajoutez l'option Label Security à une instance de base de données existante, une brève panne se produit pendant le redémarrage automatique de votre instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Résolution des problèmes

Ci-après des problèmes que vous pouvez rencontrer en utilisant Oracle Label Security.

Problème	Suggestions de dépannage
Lorsque vous essayez de créer une stratégie, vous voyez un message d'erreur similaire à ce qui suit : `insufficient authorization for the SYSDBA package`.	Un problème connu avec la fonctionnalité Label Security d'Oracle empêche les utilisateurs avec des noms d'utilisateur de 16 ou 24 caractères d'exécuter des commandes Label Security. Vous pouvez créer un nouvel utilisateur avec un nombre de caractère s différent, attribuer LBAC _ DBA au nouvel utilisate ur, vous connecter en tant que nouvel utilisateur et exécuter les OLS commandes en tant que nouvel utilisateur. Pour plus d'informations, contactez le support Oracle.

Oracle Locator

Amazon RDS prend en charge Oracle Locator par l'intermédiaire de l'option `LOCATOR`. Oracle Locator fournit des capacités généralement requises pour prendre en charge les applications basées sur des services sans fil et sur Internet, ainsi que les solutions GIS basées sur le partenariat. Oracle Locator est un sous-ensemble limité d'Oracle Spatial. Pour plus d'informations, consultez Oracle Locator dans la documentation d'Oracle.

> **Important**
>
> Si vous utilisez Oracle Locator, Amazon RDS met automatiquement à jour votre instance de base de données vers le PSU Oracle le plus récent en cas de vulnérabilités de sécurité présentant un score de Common Vulnerability Scoring System (CVSS) égal ou supérieur à 9, ou d'autres vulnérabilités de sécurité annoncées.

Versions de base de données prises en charge pour Oracle Locator

RDS pour Oracle prend en charge Oracle Locator pour Oracle Database 19c. Oracle Locator n'est pas pris en charge par Oracle Database 21c, mais sa fonctionnalité est disponible dans l'option Oracle Spatial. Auparavant, l'option Spatial nécessitait des licences supplémentaires. Oracle Locator représentait un sous-ensemble des fonctionnalités de Oracle Spatial et ne nécessitait pas de licences supplémentaires. En 2019, Oracle a annoncé que toutes les fonctionnalités d'Oracle Spatial étaient incluses dans les licences Enterprise Edition et Standard Edition 2 sans frais supplémentaires. Par conséquent, l'option Oracle Spatial ne nécessite plus de licence supplémentaire. Pour plus d'informations, consultez l'article Machine Learning, Spatial and Graph - No License Required! (Machine Learning, Spatial et Graph — Aucune licence requise !) sur le blog Oracle Database Insider.

Prérequis pour Oracle Locator

Les conditions suivantes sont requises pour utiliser Oracle Locator :

- Votre instance de base de données doit être d'une classe suffisante. Oracle Locator n'est pas pris en charge pour les classes d'instance de base de données db.t3.small. Pour de plus amples informations, veuillez consulter RDS pour les classes d'instance de base de données Oracle.
- Votre instance de base de données doit avoir l'option Auto Minor Version Upgrade (Mise à niveau automatique des versions mineures) activée. Cette option permet à votre instance de base de

données de recevoir automatiquement des mises à niveau mineures de version du moteur de base de données quand elles sont disponibles, et est requise pour toutes les options qui installent la machine virtuelle Java (JVM) Oracle. Amazon RDS utilise cette option pour mettre à jour votre instance de base de données vers le dernier PSU (Patch Set Update) ou la dernière mise à jour (RU) Oracle. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Bonnes pratiques pour Oracle Locator

Les bonnes pratiques suivantes sont requises pour utiliser Oracle Locator :

- Pour une sécurité maximale, utilisez l'option `LOCATOR` avec Secure Sockets Layer (SSL). Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).
- Configurez votre instance de base de données pour en restreindre l'accès. Pour plus d'informations, consultez Scénarios d'accès à un d'instances de base de données dans un VPC et Utilisation d'un(e) instance de base de données dans un VPC.

Ajout de l'option Oracle Locator

La procédure générale suivante permet d'ajouter l'option `LOCATOR` à une instance de base de données :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Si Oracle Java Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option `LOCATOR` est ajoutée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous ajoutez l'option, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, Oracle Locator est disponible.

> **Note**
>
> Durant cette interruption, les fonctions de vérification de mot de passe sont brièvement désactivées. Vous pouvez également vous attendre à voir des événements liés aux fonctions

de vérification de mot de passe durant l'interruption. Les fonctions de vérification de mot de passe sont activées de nouveau avant que l'instance de base de données Oracle ne soit disponible.

Pour ajouter l'option **LOCATOR** à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, choisissez l'édition oracle pour votre instance de base de données.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour plus d'informations, consultez Création d'un groupe d'options.

2. Ajoutez l'option LOCATOR au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.

3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Utilisation d'Oracle Locator

Une fois que vous activez l'option Oracle Locator, vous pouvez commencer à l'utiliser. Utilisez uniquement les fonctions Oracle Locator. N'utilisez pas les fonctions Oracle Spatial, sauf si vous avez la licence correspondante.

Pour obtenir la liste des fonctions prises en charge pour Oracle Locator, consultez Fonctions incluses avec Locator dans la documentation d'Oracle.

Pour obtenir la liste des fonctions qui ne sont pas prises en charge pour Oracle Locator, consultez Fonctions non incluses avec Locator dans la documentation d'Oracle.

Suppression de l'option Oracle Locator

Après avoir supprimé tous les objets qui utilisent des types de données fournis par l'option LOCATOR, vous pouvez supprimer l'option d'une instance de base de données. Si Oracle Java Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option LOCATOR est supprimée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous supprimez l'option LOCATOR, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'option **LOCATOR**

1. Sauvegardez vos données.

   > **Warning**
   >
   > Si l'instance utilise des types de données qui ont été activés dans le cadre de l'option, et si vous supprimez l'option LOCATOR, vous pouvez perdre des données. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

2. Vérifiez si des objets existants font référence à des types de données ou à des fonctionnalités de l'option LOCATOR.

   Si des options LOCATOR existent, l'instance peut rester bloquée lors de l'application du nouveau groupe d'options qui n'a pas l'option LOCATOR. Vous pouvez identifier les objets à l'aide des requêtes suivantes :

```
SELECT OWNER, SEGMENT_NAME, TABLESPACE_NAME, BYTES/1024/1024 mbytes
FROM   DBA_SEGMENTS
WHERE  SEGMENT_TYPE LIKE '%TABLE%'
AND    (OWNER, SEGMENT_NAME) IN
       (SELECT DISTINCT OWNER, TABLE_NAME
        FROM   DBA_TAB_COLUMNS
        WHERE  DATA_TYPE='SDO_GEOMETRY'
        AND    OWNER <> 'MDSYS')
ORDER BY 1,2,3,4;

SELECT OWNER, TABLE_NAME, COLUMN_NAME
FROM   DBA_TAB_COLUMNS
```

```
WHERE  DATA_TYPE = 'SDO_GEOMETRY'
AND    OWNER <> 'MDSYS'
ORDER BY 1,2,3;
```

3. Supprimez tous les objets qui font référence à des types de données ou à des fonctionnalités de l'option LOCATOR.
4. Effectuez l'une des actions suivantes :

   - Supprimez l'option LOCATOR du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.
   - Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option LOCATOR. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Oracle NNE (Native Network Encryption)

Amazon RDS prend en charge le chiffrement réseau natif d'Oracle (NNE). Avec `NATIVE_NETWORK_ENCRYPTION` cette option, vous pouvez chiffrer les données lors de leur transfert vers et depuis une instance de base de données. Amazon RDS prend en NNE charge toutes les éditions d'Oracle Database.

Une présentation détaillée d'Oracle NNE dépasse le propos de ce guide, mais vous devez bien comprendre les points forts et les points faibles de chaque algorithme et de chaque clé avant de choisir une solution pour votre déploiement. Pour plus d'informations sur les algorithmes et les clés qui sont disponibles via Oracle NNE, consultez Configuration du chiffrement des données réseau dans la documentation Oracle. Pour plus d'informations sur la sécurité d' AWS , consultez le Centre de sécuritéAWS.

> **Note**
>
> Vous pouvez utiliser le chiffrement réseau natif ou le protocole SSL (Secure Sockets Layer), mais pas les deux. Pour de plus amples informations, veuillez consulter Oracle Secure Sockets Layer (SSL).

Rubriques

- NATIVE_ NETWORK _ paramètres des ENCRYPTION options
- Ajouter l'ENCRYPTIONoption NATIVE NETWORK _ _
- Définition des NNE valeurs dans le fichier sqlnet.ora
- Modification des paramètres de ENCRYPTION l'option NATIVE NETWORK _ _
- Supprimer l'ENCRYPTIONoption NATIVE NETWORK _ _

NATIVE_ NETWORK _ paramètres des ENCRYPTION options

Vous pouvez spécifier les exigences de chiffrement à la fois sur le serveur et le client. L'instance de base de données peut agir en tant que client lorsque, par exemple, elle utilise un lien de base de données pour se connecter à une autre base de données. Vous pouvez éviter de forcer le chiffrement côté serveur. Par exemple, vous pouvez ne pas forcer toutes les communications client à utiliser le chiffrement car le serveur en a besoin. Dans ce cas, vous pouvez forcer le chiffrement côté client à l'aide des options `SQLNET.*CLIENT`.

Amazon RDS prend en charge les paramètres suivants pour `NATIVE_NETWORK_ENCRYPTION` cette option.

> **Note**
>
> Lorsque vous utilisez des virgules pour séparer les valeurs d'un paramètre d'option, ne placez pas d'espace après la virgule.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
`SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS`	`TRUE`, `FALSE`	`TRUE`	Comportement du serveur lorsqu'un client utilisant un chiffrement non sécurisé tente de se connecter à la base de données. Si`TRUE`, les clients peuvent se connecter même s'ils n'ont pas reçu le correctif de juillet 2021. PSU Si le paramètre est défini`FALSE`, les clients ne peuvent se connecter à la base de données que lorsqu'ils reçoivent le correctif de juillet 2021PSU. Avant de paramétrer `SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS` à `FALSE`, assurez-vous que les conditions suivantes sont remplies : • `SQLNET.ENCRYPTION_TYPES_SERVER` et `SQLNET.ENCRYPTION_TYPES_CLIENT` possèdent une méthode de chiffrement correspondante qui n'est pas `DES`,`3DES` ou `RC4` (toutes les longueurs de clés).

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
			• SQLNET.CHECKSUM_TYPES_SERVER et SQLNET.CHECKSUM_TYPES_CLIENT disposent d'une méthode de total de contrôle sécurisée correspondante autre que MD5. • Le client est patché avec le juillet 2021. PSU Si le client n'est pas corrigé, il perd la connexion et reçoit l'erreur ORA-12269 .

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.ALLOW_WEAK_CRYPTO	TRUE, FALSE	TRUE	Comportement du serveur lorsqu'un client utilisant un chiffrement non sécurisé tente de se connecter à la base de données. Les chiffrements suivants sont considérés comme non sécurisés : • Méthode de chiffrement DES (toutes les longueurs de clés) • Méthode de chiffrement 3DES (toutes les longueurs de clés) • Méthode de chiffrement RC4 (toutes les longueurs de clés) • Méthode de total de contrôle MD5 Si le paramètre est TRUE, les clients peuvent se connecter lorsqu'ils utilisent les chiffrements non sécurisés précédents. Si le paramètre est FALSE, la base de données empêche les clients de se connecter lorsqu'ils utilisent les chiffrements non sécurisés précédent s. Avant de paramétrer SQLNET.AL LOW_WEAK_CRYPTO à FALSE, assurez-vous que les conditions suivantes sont remplies : • SQLNET.ENCRYPTION_ TYPES_SERVER et SQLNET.EN CRYPTION_TYPES_CLIENT

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
			possèdent une méthode de chiffrement correspondante qui n'est pas DES ,3DES ou RC4 (toutes les longueurs de clés). • SQLNET.CHECKSUM_TYPES_SERVER et SQLNET.CHECKSUM_TYPES_CLIENT disposent d'une méthode de total de contrôle sécurisée correspondante autre que MD5. • Le client est patché avec le juillet 2021. PSU Si le client n'est pas corrigé, il perd la connexion et reçoit l'erreur ORA-12269 .
SQLNET.CRYPTO_CHECKSUM_CLIENT	Accepted Rejected Requested , Required	Requested	Comportement d'intégrité des données quand une instance de base de données se connecte au client, ou à un serveur agissant en tant que client. Lorsqu'une instance DB utilise un lien de base de données, elle agit en tant que client. Requested indique que le client ne nécessite pas que l'instance de base de données effectue un total de contrôle.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.CRYPTO_CHECKSUM_SERVER	Accepted Rejected Requested , Required	Requested	Comportement d'intégrité des données quand un client, ou un serveur agissant en tant que client, se connecte à l'instance de base de données. Lorsqu'une instance DB utilise un lien de base de données, elle agit en tant que client. Requested indique que l'instance de base de données ne nécessite pas que le client effectue un total de contrôle.
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT	SHA256, SHA384, SHA512, SHA1, MD5	SHA256, SHA384, SHA512	Une liste d'algorithmes de somme de contrôle. Vous pouvez spécifier une valeur ou une liste de valeurs séparées par des virgules. Si vous spécifiez une virgule, n'insérez pas d'espace après celle-ci ; sinon, vous recevez une erreur InvalidParameterValue . Ce paramètre et SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER doivent avoir un chiffrement commun.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER	SHA256, SHA384, SHA512, SHA1, MD5	SHA256, SHA384, SHA512, SHA1, MD5	Une liste d'algorithmes de somme de contrôle. Vous pouvez spécifier une valeur ou une liste de valeurs séparées par des virgules. Si vous spécifiez une virgule, n'insérez pas d'espace après celle-ci ; sinon, vous recevez une erreur InvalidParameterValue . Ce paramètre et SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT doivent avoir un chiffrement commun.
SQLNET.ENCRYPTION_CLIENT	Accepted Rejected Requested , Required	Requested	Comportement de chiffrement du client quand un client, ou un serveur agissant en tant que client, se connecte à l'instance de base de données. Lorsqu'une instance DB utilise un lien de base de données, elle agit en tant que client. Requested indique que le client ne requiert pas que le trafic depuis l'instance de base de données soit chiffré.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.ENCRYPTION_SERVER	Accepted Rejected Requested , Required	Requested	Comportement de chiffrement du serveur quand un client, ou un serveur agissant en tant que client, se connecte à l'instance de base de données. Lorsqu'une instance DB utilise un lien de base de données, elle agit en tant que client. Requested indique que l'instance de base de données ne requiert pas que le trafic depuis le client soit chiffré.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.ENCRYPTION_TYPES_CLIENT	RC4_256, AES256, AES192, 3DES168, RC4_128, AES128, 3DES112, RC4_56, DES, RC4_40, DES40	RC4_256, AES256, AES192, 3DES168, RC4_128, AES128, 3DES112, RC4_56, DES, RC4_40, DES40	Liste d'algorithmes de chiffrement utilisés par le client. Le client tente de déchiffrer l'entrée du serveur en essayant chaque algorithme, dans l'ordre, jusqu'à ce qu'un algorithme réussisse ou que la fin de la liste soit atteinte. Amazon RDS utilise la liste par défaut suivante d'Oracle. RDScommence par RC4_256 et descend la liste dans l'ordre. Vous pouvez modifier l'ordre ou limiter les algorithmes que l'instance de base de données accepte. 1. RC4_256: RSA RC4 (taille de clé 256 bits) 2. AES256: AES (taille de clé 256 bits) 3. AES192: AES (taille de clé 192 bits) 4. 3DES168: Triple- à 3 touches DES (taille de clé effective de 112 bits) 5. RC4_128: RSA RC4 (taille de clé 128 bits) 6. AES128: AES (taille de clé 128 bits) 7. 3DES112: Triple- à 2 touches DES (taille de clé effective de 80 bits) 8. RC4_56: RSA RC4 (taille de clé 56 bits) 9. DES: Standard DES (taille de clé 56 bits)

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
			10`RC4_40`: RSA RC4 (taille de clé de 40 bits) 11`DES40`: DES4 0 (taille de clé de 40 bits) Vous pouvez spécifier une valeur ou une liste de valeurs séparées par des virgules. Si vous spécifiez une virgule, n'insérez pas d'espace après celle-ci ; sinon, vous recevez une erreur `InvalidParameterValue` . Ce paramètre et `SQLNET.SQLNET.ENCRYPTION_TYPES_SERVER` doivent avoir un chiffrement commun.

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
SQLNET.ENCRYPTION_TYPES_SERVER	RC4_256, AES256, AES192, 3DES168, RC4_128, AES128, 3DES112, RC4_56, DES, RC4_40, DES40	RC4_256, AES256, AES192, 3DES168, RC4_128, AES128, 3DES112, RC4_56, DES, RC4_40, DES40	Liste d'algorithmes de chiffrement utilisés par l'instance de base de données. L'instance de base de données utilise chaque algorithme, dans l'ordre, pour tenter de déchiffrer l'entrée du client jusqu'à ce qu'un algorithme réussisse ou que la fin de la liste soit atteinte. Amazon RDS utilise la liste par défaut suivante d'Oracle. Vous pouvez modifier l'ordre ou limiter les algorithmes que le client accepte. 1. RC4_256: RSA RC4 (taille de clé 256 bits) 2. AES256: AES (taille de clé 256 bits) 3. AES192: AES (taille de clé 192 bits) 4. 3DES168: Triple- à 3 touches DES (taille de clé effective de 112 bits) 5. RC4_128: RSA RC4 (taille de clé 128 bits) 6. AES128: AES (taille de clé 128 bits) 7. 3DES112: Triple- à 2 touches DES (taille de clé effective de 80 bits) 8. RC4_56: RSA RC4 (taille de clé 56 bits) 9. DES: Standard DES (taille de clé 56 bits) 10. RC4_40: RSA RC4 (taille de clé de 40 bits)

Paramètre d'option	Valeurs valides	Valeurs par défaut	Description
			11DES40: DES4 0 (taille de clé de 40 bits) Vous pouvez spécifier une valeur ou une liste de valeurs séparées par des virgules. Si vous spécifiez une virgule, n'insérez pas d'espace après celle-ci ; sinon, vous recevez une erreur InvalidParameterValue . Ce paramètre et SQLNET.SQLNET.ENCRYPTION_TYPES_SERVER doivent avoir un chiffrement commun.

Ajouter l'ENCRYPTIONoption NATIVE NETWORK _ _

Le processus général pour ajouter l'NATIVE_NETWORK_ENCRYPTIONoption à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Lorsque le groupe d'options est actif, NNE il est actif.

Pour ajouter l'ENCRYPTIONoption NATIVE _ NETWORK _ à une instance de base de données à l'aide du AWS Management Console

1. Pour Moteur, sélectionnez l'édition d'Oracle que vous voulez utiliser. NNEest compatible avec toutes les éditions.
2. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.

3. Ajoutez l'ENCRYPTIONoption NATIVENETWORK_ _ au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.

> **Note**
>
> Après avoir ajouté l'ENCRYPTIONoption NATIVE_ NETWORK _, vous n'avez pas besoin de redémarrer vos instances de base de données. Dès que le groupe d'options est actif, NNE il est actif.

4. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Après avoir ajouté l'ENCRYPTIONoption NATIVE_ NETWORK _, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, NNE il est actif. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Définition des NNE valeurs dans le fichier sqlnet.ora

Avec le chiffrement réseau natif Oracle, vous pouvez spécifier le chiffrement réseau côté serveur et côté client. Le client est l'ordinateur utilisé pour se connecter à l'instance de base de données. Vous pouvez spécifier les paramètres client suivants dans le fichier slqnet.ora :

- `SQLNET.ALLOW_WEAK_CRYPTO`
- `SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS`
- `SQLNET.CRYPTO_CHECKSUM_CLIENT`
- `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT`
- `SQLNET.ENCRYPTION_CLIENT`
- `SQLNET.ENCRYPTION_TYPES_CLIENT`

Pour plus d'informations, consultez Configuration du chiffrement des données réseau et intégrité des clients et serveurs Oracle dans la documentation Oracle.

Parfois, l'instance de base de données rejette une demande de connexion provenant d'une application. Par exemple, un rejet peut se produire lorsque les algorithmes de chiffrement sur le client et sur le serveur ne correspondent pas. Pour tester le chiffrement réseau natif Oracle, ajoutez les lignes suivantes dans le fichier sqlnet.ora sur le client :

```
DIAG_ADR_ENABLED=off
TRACE_DIRECTORY_CLIENT=/tmp
TRACE_FILE_CLIENT=nettrace
TRACE_LEVEL_CLIENT=16
```

Lors d'une tentative de connexion, ces lignes génèrent un fichier de trace sur le client appelé `/tmp/nettrace*`. Le fichier de trace contient des informations concernant la connexion. Pour plus d'informations sur les problèmes liés à la connexion lorsque vous utilisez le chiffrement de réseau natif Oracle, veuillez consulter À propos de la négociation du chiffrement et de l'intégrité dans la documentation base de données Oracle.

Modification des paramètres de ENCRYPTION l'option NATIVE NETWORK _ _

Après avoir activé l'option `NATIVE_NETWORK_ENCRYPTION`, vous pouvez modifier ses paramètres. Actuellement, vous ne pouvez modifier les paramètres des `NATIVE_NETWORK_ENCRYPTION` options qu'avec le AWS CLI ou RDSAPI. Vous ne pouvez pas utiliser la console. L'exemple suivant modifie deux paramètres de l'option.

```
aws rds add-option-to-option-group \
    --option-group-name my-option-group \
    --options
 "OptionName=NATIVE_NETWORK_ENCRYPTION,OptionSettings=[{Name=SQLNET.CRYPTO_CHECKSUM_TYPES_SERVE
{Name=SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER,Value=SHA256}]" \
    --apply-immediately
```

Pour savoir comment modifier les paramètres des options à l'aide duCLI, voirAWS CLI. Pour plus d'informations sur chaque paramètre, consultez NATIVE_ NETWORK _ paramètres des ENCRYPTION options.

Rubriques

- Modification des CRYPTO valeurs _ CHECKSUM _*

- Modification des paramètres ALLOW WEAK _ _ CRYPTO *

Modification des CRYPTO valeurs _ CHECKSUM _*

Si vous modifiez les paramètres de ENCRYPTION l'option NATIVENETWORK_ _, assurez-vous que les paramètres d'option suivants comportent au moins un chiffre commun :

- `SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER`
- `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT`

L'exemple suivant montre un scénario dans lequel vous modifiez `SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER`. La configuration est valide car `CRYPTO_CHECKSUM_TYPES_CLIENT` et `CRYPTO_CHECKSUM_TYPES_SERVER` utilisent tous les deux `SHA256`.

Paramètre d'option	Valeurs avant modification	Valeurs après modification
`SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT`	**`SHA256`**, `SHA384`, `SHA512`	Pas de modification
`SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER`	**`SHA256`**, `SHA384`, `SHA512`, `SHA1`, `MD5`	`SHA1,MD5,` **`SHA256`**

Pour un autre exemple, supposons que vous souhaitez modifier `SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER` à partir de son paramètre par défaut vers `SHA1,MD5`. Dans ce cas, veillez à définir `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT` sur `SHA1` ou `MD5`. Ces algorithmes ne sont pas inclus dans les valeurs par défaut pour `SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT`.

Modification des paramètres ALLOW WEAK _ _ CRYPTO *

Pour définir le paramètre `SQLNET.ALLOW_WEAK_CRYPTO*` de la valeur par défaut à `FALSE`, vérifiez que les conditions suivantes sont réunies :

- SQLNET.ENCRYPTION_TYPES_SERVER et SQLNET.ENCRYPTION_TYPES_CLIENT possèdent une méthode de chiffrement sécurisée correspondante. Une méthode est considérée comme sécurisée si elle n'est pas DES ,3DES ou RC4 (toutes les longueurs de clés).
- SQLNET.CHECKSUM_TYPES_SERVER et SQLNET.CHECKSUM_TYPES_CLIENT disposent d'une méthode de total de contrôle sécurisée correspondante. Une méthode est considérée comme sécurisée si elle n'est pas MD5.
- Le client est patché avec le juillet 2021. PSU Si le client n'est pas corrigé, il perd la connexion et reçoit l'erreur ORA-12269.

L'exemple suivant montre des exemples de NNE paramètres. Supposons que vous souhaitez définir SQLNET.ENCRYPTION_TYPES_SERVER et SQLNET.ENCRYPTION_TYPES_CLIENT faireFALSE, bloquant ainsi les connexions non sécurisées. Les paramètres de l'option de total de contrôle répondent aux conditions préalables car ils ont tous les deux SHA256. Cependant, SQLNET.ENCRYPTION_TYPES_CLIENT et SQLNET.ENCRYPTION_TYPES_SERVER utilisent les méthodes de chiffrement DES, 3DES et RC4, qui ne sont pas sécurisées. Par conséquent, pour définir les options SQLNET.ALLOW_WEAK_CRYPTO* à FALSE, définissez d'abord SQLNET.ENCRYPTION_TYPES_SERVER et SQLNET.ENCRYPTION_TYPES_CLIENT pour utiliser une méthode de chiffrement sécurisée telle que AES256.

Paramètre d'option	Valeurs
SQLNET.CRYPTO_CHECKSUM_TYPES_CLIENT	SHA256, SHA384, SHA512
SQLNET.CRYPTO_CHECKSUM_TYPES_SERVER	SHA1,MD5,SHA256
SQLNET.ENCRYPTION_TYPES_CLIENT	RC4_256, 3DES168, DES40
SQLNET.ENCRYPTION_TYPES_SERVER	RC4_256, 3DES168, DES40

Supprimer l'ENCRYPTIONoption NATIVE NETWORK _ _

Vous pouvez effectuer une suppression NNE d'une instance de base de données.

Pour supprimer l'option NATIVE_NETWORK_ENCRYPTION d'une instance de base de données, effectuez l'une des actions suivantes :

- Pour supprimer l'option de plusieurs instances de base de données, NATIVE_NETWORK_ENCRYPTION supprimez-la du groupe d'options auquel elles appartiennent. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Une fois l'NATIVE_NETWORK_ENCRYPTIONoption supprimée, vous n'avez pas besoin de redémarrer vos instances de base de données. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.
- Pour supprimer l'option d'une seule instance de base de données, modifiez l'instance de base de données et spécifiez un autre groupe d'options qui n'inclut pas l'NATIVE_NETWORK_ENCRYPTIONoption. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Une fois que vous supprimez l'option NATIVE_NETWORK_ENCRYPTION, vous n'avez pas besoin de redémarrer votre instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Oracle OLAP

Amazon RDS prend en charge Oracle OLAP par l'intermédiaire de l'option `OLAP`. L'option OLAP (Online Analytical Processing) est proposée pour les instances de base de données Oracle. Vous pouvez utiliser Oracle OLAP pour analyser de grandes quantités de données en créant des objets et des cubes dimensionnels conformément à la norme OLAP. Pour de plus amples informations, veuillez consulter la documentation Oracle.

> **Important**
>
> Si vous utilisez Oracle OLAP, Amazon RDS met automatiquement à jour votre instance de base de données vers le PSU Oracle le plus récent en cas de vulnérabilités de sécurité présentant un score CVSS (Common Vulnerability Scoring System) égal ou supérieur à 9, ou d'autres vulnérabilités de sécurité annoncées.

Amazon RDS prend en charge Oracle OLAP pour l'édition Enterprise d'Oracle Database 19c et versions ultérieures.

Prérequis pour Oracle OLAP

Les conditions suivantes sont requises pour utiliser Oracle OLAP :

- Vous devez disposer d'une licence Oracle OLAP d'Oracle. Pour de plus amples informations, veuillez consulter la section relative aux informations sur les licences dans la documentation Oracle.
- La classe d'instance de votre instance de base de données doit être suffisante. Oracle OLAP n'est pas pris en charge pour les classes d'instance de base de données db.t3.small. Pour de plus amples informations, veuillez consulter RDS pour les classes d'instance de base de données Oracle.
- Votre instance de base de données doit avoir l'option Auto Minor Version Upgrade (Mise à niveau automatique des versions mineures) activée. Cette option permet à votre instance de base de données de recevoir automatiquement des mises à niveau mineures de version du moteur de base de données quand elles sont disponibles, et est requise pour toutes les options qui installent la machine virtuelle Java (JVM) Oracle. Amazon RDS utilise cette option pour mettre à jour votre instance de base de données vers le dernier PSU (Patch Set Update) ou la dernière mise à jour (RU) Oracle. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

- Votre instance de base de données ne doit pas contenir d'utilisateur nommé `OLAPSYS`. Si votre instance en contient, l'installation de l'option OLAP échoue.

Bonnes pratiques pour Oracle OLAP

Les bonnes pratiques suivantes sont requises pour utiliser Oracle OLAP :

- Pour une sécurité maximale, utilisez l'option `OLAP` avec Secure Sockets Layer (SSL). Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).
- Configurez votre instance de base de données pour en restreindre l'accès. Pour plus d'informations, consultez Scénarios d'accès à un d'instances de base de données dans un VPC et Utilisation d'un(e) instance de base de données dans un VPC.

Ajout de l'option Oracle OLAP

La procédure générale suivante permet d'ajouter l'option `OLAP` à une instance de base de données :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Si Oracle Java Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option `OLAP` est ajoutée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous ajoutez l'option, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, Oracle OLAP est disponible.

Pour ajouter l'option OLAP à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :
   - Pour Moteur, choisissez l'édition Oracle de votre instance de base de données.
   - Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

Pour plus d'informations, consultez Création d'un groupe d'options.

2. Ajoutez l'option OLAP au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:
   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Utilisation d'Oracle OLAP

Une fois que vous activez l'option Oracle OLAP, vous pouvez commencer à l'utiliser. Pour obtenir la liste des fonctions prises en charge par Oracle OLAP, veuillez consulter la documentation Oracle.

Suppression de l'option Oracle OLAP

Après avoir supprimé tous les objets qui utilisent des types de données fournis par l'option `OLAP`, vous pouvez supprimer l'option d'une instance de base de données. Si Oracle Java Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option `OLAP` est supprimée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous supprimez l'option `OLAP`, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'option **`OLAP`**

1. Sauvegardez vos données.

> **Warning**
>
> Si l'instance utilise des types de données qui ont été activés dans le cadre de l'option, et si vous supprimez l'option `OLAP`, vous pouvez perdre des données. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

2. Vérifiez si des objets existants font référence à des types de données ou à des fonctionnalités de l'option OLAP.
3. Supprimez tous les objets qui font référence à des types de données ou à des fonctionnalités de l'option OLAP.
4. Effectuez l'une des actions suivantes :

   - Supprimez l'option OLAP du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.
   - Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option OLAP. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Oracle Secure Sockets Layer (SSL)

Pour activer SSL le chiffrement RDS pour une instance de base de données Oracle, ajoutez l'SSLoption Oracle au groupe d'options associé à l'instance de base de données. Amazon RDS utilise un deuxième port, comme l'exige Oracle, pour les SSL connexions. Cette approche permet à la fois d'établir une communication en texte clair et SSL cryptée entre une instance de base de données et SQL *Plus. Par exemple, vous pouvez utiliser le port avec une communication en texte clair pour communiquer avec d'autres ressources VPC pendant un certain temps en utilisant le port avec une communication SSL cryptée pour communiquer avec des ressources extérieures à. VPC

> **Note**
>
> Vous pouvez utiliser l'un ou l'autre SSL ou Native Network Encryption (NNE) de la même manière RDS pour une instance de base de données Oracle, mais pas les deux. Si vous utilisez SSL le chiffrement, assurez-vous de désactiver tout autre chiffrement de connexion. Pour de plus amples informations, veuillez consulter Oracle NNE (Native Network Encryption).

SSL/TLSet ne NNE font plus partie d'Oracle Advanced Security. Dans RDS for Oracle, vous pouvez utiliser le SSL chiffrement avec toutes les éditions sous licence des versions de base de données suivantes :

- Oracle Database 21c (21.0.0)
- Oracle Database 19c (19.0.0)

Rubriques

- TLSversions pour l'SSLoption Oracle
- Suites de chiffrement pour l'option Oracle SSL
- FIPSsoutien
- Ajouter l'SSLoption
- Configuration de SQL *Plus pour une utilisation SSL avec une instance de base RDS de données Oracle
- Connexion à une instance de base de données RDS for Oracle à l'aide de SSL
- Configuration d'une SSL connexion via JDBC

- Imposer une correspondance de DN avec une SSL connexion
- Résolution des problèmes de SSL connexions

TLSversions pour l'SSLoption Oracle

Amazon RDS pour Oracle prend en charge les versions 1.0 et 1.2 de Transport Layer Security (TLS). Lorsque vous ajoutez une nouvelle SSL option Oracle, définissez `SQLNET.SSL_VERSION` explicitement une valeur valide. Les valeurs suivantes sont autorisées pour ce paramètre d'option :

- `"1.0"`— Les clients peuvent se connecter à l'instance de base de données uniquement en utilisant TLS la version 1.0. Pour les SSL options Oracle existantes, `SQLNET.SSL_VERSION` est défini sur `"1.0"` automatiquement. Vous pouvez modifier au besoin ce paramètre.
- `"1.2"`— Les clients peuvent se connecter à l'instance de base de données en utilisant la TLS version 1.2 uniquement.
- `"1.2 or 1.0"`— Les clients peuvent se connecter à l'instance de base de données à l'aide de la version TLS 1.2 ou 1.0.

Suites de chiffrement pour l'option Oracle SSL

Amazon RDS pour Oracle prend en charge plusieurs suites de SSL chiffrement. Par défaut, l'SSLoption Oracle est configurée pour utiliser la suite de `SSL_RSA_WITH_AES_256_CBC_SHA` chiffrement. Pour spécifier une autre suite de chiffrement à utiliser sur SSL les connexions, utilisez le paramètre d'`SQLNET.CIPHER_SUITE`option.

Vous pouvez spécifier plusieurs valeurs pour`SQLNET.CIPHER_SUITE`. Cette technique est utile si vous avez des liens de base de données entre vos instances de base de données et que vous décidez de mettre à jour vos suites de chiffrement.

Le tableau suivant récapitule le SSL support RDS d'Oracle dans toutes les éditions d'Oracle Database 19c et 21c.

Suite de chiffrement (SQLNET. CIPHER_SUITE)	TLSsupport de version (SQLNET. SSL_VERSION)	FIPSsoutien	RAMPConforme à la Fed
SSL_ _ RSA WITH _ AES _256_ CBC _ SHA (par défaut)	1.0 et 1.2	Oui	Non
SSL_ _ RSA WITH _ AES CBC _256_ _ SHA256	1.2	Oui	Non
SSL_ _ RSA WITH _ AES GCM _256_ _ SHA384	1.2	Oui	Non
TLS_ ECDHE _ _ RSA WITH _ AES GCM _256_ _ SHA384	1.2	Oui	Oui
TLS_ ECDHE _ _ RSA WITH _ AES GCM _128_ _ SHA256	1.2	Oui	Oui
TLS_ ECDHE _ _ RSA WITH _ AES CBC _256_ _ SHA384	1.2	Oui	Oui
TLS_ ECDHE _ _ RSA WITH _ AES CBC _128_ _ SHA256	1.2	Oui	Oui
TLS_ ECDHE _ _ RSA WITH _ AES CBC _256_ _ SHA	1.2	Oui	Oui
TLS_ ECDHE _ _ RSA WITH _ AES CBC _128_ _ SHA	1.2	Oui	Oui

FIPSsoutien

RDSpour Oracle vous permet d'utiliser la norme Federal Information Processing Standard (FIPS) pour 140-2. FIPS140-2 est une norme du gouvernement des États-Unis qui définit les exigences de sécurité des modules cryptographiques. Vous activez la FIPS norme en définissant `FIPS.SSLFIPS_140` `TRUE` l'SSLoption Oracle sur. Lorsque FIPS 140-2 est configuré pourSSL, les

bibliothèques cryptographiques chiffrent les données entre le client et l'instance de base RDS de données Oracle.

Les clients doivent utiliser la suite de chiffrement conformeFIPS. Lors de l'établissement d'une connexion, le client et l'instance RDS de base de données Oracle négocient la suite de chiffrement à utiliser lors de la transmission de messages dans les deux sens. Le tableau ci-dessous Suites de chiffrement pour l'option Oracle SSL indique les suites de SSL chiffrement FIPS compatibles pour chaque TLS version. Pour plus d'informations, consultez les paramètres de la base de données Oracle FIPS 140-2 dans la documentation de la base de données Oracle.

Ajouter l'SSLoption

Pour être utiliséeSSL, votre instance de base de données RDS for Oracle doit être associée à un groupe d'options qui inclut l'`SSL`option.

Console

Pour ajouter l'SSLoption à un groupe d'options

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `SSL`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'option `SSL` au groupe d'options.

   Si vous souhaitez utiliser uniquement des suites de FIPS chiffrement vérifiées pour les SSL connexions, définissez l'option `FIPS.SSLFIPS_140` sur. `TRUE` Pour plus d'informations sur la FIPS norme, consultezFIPSsoutien.

   Pour de plus amples informations sur l'ajout d'une option à un groupe d'options, veuillez consulter Ajout d'une option à un groupe d'options.
3. Créez une nouvelle instance RDS de base de données Oracle et associez-y le groupe d'options, ou modifiez une instance de base de données Oracle RDS pour y associer le groupe d'options.

   Pour obtenir des informations sur la création d'une instance de base de données, consultez Création d'une instance de base de données Amazon RDS.

   Pour obtenir des informations sur la modification d'une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

AWS CLI

Pour ajouter l'SSLoption à un groupe d'options

1. Créez un groupe d'options ou identifiez un groupe d'options existant auquel vous pouvez ajouter l'option `SSL`.

   Pour de plus amples informations sur la création d'un groupe d'options, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'option `SSL` au groupe d'options.

   Spécifiez les paramètres d'option suivants :

   - `Port`— Le numéro SSL de port
   - `VpcSecurityGroupMemberships`— Le groupe VPC de sécurité pour lequel l'option est activée
   - `SQLNET.SSL_VERSION`— TLS Version que le client peut utiliser pour se connecter à l'instance de base de données

   Par exemple, la AWS CLI commande suivante ajoute l'`SSL`option à un groupe d'options nommé`ora-option-group`.

   Example

   Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group --option-group-name ora-option-group \
  --options
 'OptionName=SSL,Port=2484,VpcSecurityGroupMemberships="sg-68184619",OptionSettings=[{Name=
```

   Dans Windows:

```
aws rds add-option-to-option-group --option-group-name ora-option-group ^
  --options
 'OptionName=SSL,Port=2484,VpcSecurityGroupMemberships="sg-68184619",OptionSettings=[{Name=
```

3. Créez une nouvelle instance RDS de base de données Oracle et associez-y le groupe d'options, ou modifiez une instance de base de données Oracle RDS pour y associer le groupe d'options.

   Pour obtenir des informations sur la création d'une instance de base de données, consultez Création d'une instance de base de données Amazon RDS.

   Pour obtenir des informations sur la modification d'une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Configuration de SQL *Plus pour une utilisation SSL avec une instance de base RDS de données Oracle

Avant de pouvoir vous connecter à une instance RDS de base de données Oracle qui utilise l'SSLoption Oracle, vous devez configurer SQL *Plus avant de vous connecter.

> **Note**
>
> Pour permettre l'accès à l'instance de base de données à partir des clients appropriés, assurez-vous que vous groupes de sécurité soient bien configurés. Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité. Ces instructions s'adressent également à SQL *Plus et aux autres clients qui utilisent directement un répertoire d'origine Oracle Home. Pour JDBC les connexions, voirConfiguration d'une SSL connexion via JDBC.

Pour configurer SQL *Plus à utiliser pour se connecter SSL à une instance de base RDS de données Oracle

1. Configurez la variable d'environnement `ORACLE_HOME` sur l'emplacement de votre répertoire de base Oracle.

   Le chemin vers votre répertoire de base Oracle dépend de votre installation. L'exemple suivant définit la variable d'environnement `ORACLE_HOME`.

   ```
   prompt>export ORACLE_HOME=/home/user/app/user/product/19.0.0/dbhome_1
   ```

Pour plus d'informations sur la définition des variables d'environnement Oracle, voir Variables d'environnement SQL *Plus dans la documentation Oracle, ainsi que le guide d'installation d'Oracle pour votre système d'exploitation.

2. Ajoutez `$ORACLE_HOME/lib` à la variable d'environnement `LD_LIBRARY_PATH`.

   Voici un exemple qui définit la variable d'PATHenvironnement LIBRARY LD_ _.

   ```
   prompt>export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
   ```

3. Créez un répertoire pour le portefeuille Oracle dans `$ORACLE_HOME/ssl_wallet`.

   Voici un exemple qui crée le répertoire du portefeuille Oracle.

   ```
   prompt>mkdir $ORACLE_HOME/ssl_wallet
   ```

4. Téléchargez le fichier .pem du bundle de certificats qui fonctionne pour tous Régions AWS et placez le fichier dans le répertoire ssl_wallet. Pour plus d'informations, veuillez consulter .

5. Dans l'annuaire `$ORACLE_HOME/network/admin`, modifiez ou créez le fichier `tnsnames.ora` et incluez l'entrée suivante.

   ```
   net_service_name =
     (DESCRIPTION =
       (ADDRESS_LIST =
         (ADDRESS =
           (PROTOCOL = TCPS)
           (HOST = endpoint)
           (PORT = ssl_port_number)
         )
       )
       (CONNECT_DATA =
         (SID = database_name)
       )
       (SECURITY =
         (SSL_SERVER_CERT_DN =
    "C=US,ST=Washington,L=Seattle,O=Amazon.com,OU=RDS,CN=endpoint")
       )
     )
   ```

6. Dans le même répertoire, modifiez ou créez le fichier sqlnet.ora et incluez les paramètres suivants.

> **Note**
>
> Pour communiquer avec des entités via une connexion TLS sécurisée, Oracle a besoin d'un portefeuille contenant les certificats nécessaires à l'authentification. Vous pouvez utiliser l'ORAPKIutilitaire Oracle pour créer et gérer des portefeuilles Oracle, comme indiqué à l'étape 7. Pour plus d'informations, consultez la section Configuration d'un portefeuille Oracle ORAPKI à l'aide de la documentation Oracle.

```
WALLET_LOCATION = (SOURCE = (METHOD = FILE) (METHOD_DATA = (DIRECTORY =
 $ORACLE_HOME/ssl_wallet)))
SSL_CLIENT_AUTHENTICATION = FALSE
SSL_VERSION = 1.0
SSL_CIPHER_SUITES = (SSL_RSA_WITH_AES_256_CBC_SHA)
SSL_SERVER_DN_MATCH = ON
```

> **Note**
>
> Vous pouvez définir `SSL_VERSION` sur une valeur plus élevée si votre instance de base de données la prend en charge.

7. Exécutez la commande suivante pour créer le portefeuille Oracle.

```
prompt>orapki wallet create -wallet $ORACLE_HOME/ssl_wallet -auto_login_only
```

8. Extrayez chaque certificat du fichier groupé .pem dans un fichier .pem distinct à l'aide d'un utilitaire du système d'exploitation.
9. Ajoutez chaque certificat à votre portefeuille à l'aide de `orapki` commandes distinctes, en le *`certificate-pem-file`* remplaçant par le nom de fichier absolu du fichier .pem.

```
prompt>orapki wallet add -wallet $ORACLE_HOME/ssl_wallet -trusted_cert -cert
      certificate-pem-file -auto_login_only
```

Pour de plus amples informations, veuillez consulter Rotation de votre certificat SSL/TLS.

Connexion à une instance de base de données RDS for Oracle à l'aide de SSL

Une fois que vous avez configuré SQL*Plus pour utiliser SSL comme décrit précédemment, vous pouvez vous connecter à l'instance de base de données RDS for Oracle avec l'option SSL. Vous pouvez éventuellement d'abord exporter la valeur `TNS_ADMIN` qui pointe vers le répertoire qui contient les fichiers tnsnames.ora et sqlnet.ora. Vous vous assurez ainsi que SQL*Plus peut trouver ces fichiers de manière cohérente. L'exemple suivant exporte la valeur `TNS_ADMIN`.

```
export TNS_ADMIN=${ORACLE_HOME}/network/admin
```

Connectez-vous à l'instance de base de données. Par exemple, vous pouvez vous connecter à l'aide de SQL*Plus et d'un fichier *<net_service_name>* tnsnames.ora.

```
sqlplus mydbuser@net_service_name
```

Vous pouvez également vous connecter à l'instance de base de données à l'aide de SQL*Plus sans fichier tnsnames.ora, en utilisant la commande suivante.

```
sqlplus 'mydbuser@(DESCRIPTION = (ADDRESS = (PROTOCOL = TCPS)(HOST = endpoint) (PORT
 = ssl_port_number))(CONNECT_DATA = (SID = database_name)))'
```

Vous pouvez également vous connecter à l'instance de base de données RDS for Oracle sans utiliser SSL. Par exemple, la commande suivante se connecte à l'instance de base de données via le port en texte clair sans chiffrement SSL.

```
sqlplus 'mydbuser@(DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = endpoint) (PORT
 = port_number))(CONNECT_DATA = (SID = database_name)))'
```

Si vous voulez fermer l'accès au port TCP, créez un groupe de sécurité sans entrée d'adresse IP et ajoutez-le à l'instance. Cet ajout ferme les connexions sur le port TCP, tout en continuant à autoriser les connexions sur le port SSL spécifiées à partir d'adresses IP au sein de la plage autorisée par le groupe de sécurité de l'option SSL.

Configuration d'une SSL connexion via JDBC

Pour utiliser une SSL connexionJDBC, vous devez créer un keystore, faire confiance au certificat RDS racine de l'autorité de certification Amazon et utiliser l'extrait de code spécifié ci-dessous.

Pour créer le keystore au JKS format, vous pouvez utiliser la commande suivante. Pour plus d'informations sur la création du keystore, consultez la section Création d'un keystore dans la documentation Oracle. Pour des informations de référence, voir keytool dans le manuel Java Platform, Standard Edition Tools Reference.

```
keytool -genkey -alias client -validity 365 -keyalg RSA -keystore clientkeystore
```

Suivez les étapes ci-dessous pour faire confiance au certificat RDS racine de l'autorité de certification Amazon.

Pour faire confiance au certificat CA RDS racine d'Amazon

1. Téléchargez le fichier .pem du bundle de certificats qui fonctionne pour tous Régions AWS et placez le fichier dans le répertoire ssl_wallet.

   Pour plus d'informations sur le téléchargement de certificats, veuillez consulter .
2. Extrayez chaque certificat du fichier .pem dans un fichier distinct à l'aide d'un utilitaire du système d'exploitation.
3. Convertissez chaque certificat au format .der à l'aide d'une `openssl` commande distincte, en remplaçant *`certificate-pem-file`* avec le nom du fichier .pem du certificat (sans l'extension .pem).

   ```
   openssl x509 -outform der -in certificate-pem-file.pem -out certificate-pem-file.der
   ```
4. Importez chaque certificat dans le keystore à l'aide de la commande suivante.

   ```
   keytool -import -alias rds-root -keystore clientkeystore.jks -file certificate-pem-file.der
   ```

   Pour de plus amples informations, veuillez consulter Rotation de votre certificat SSL/TLS.
5. Vérifiez que le magasin de clés a été créé avec succès.

   ```
   keytool -list -v -keystore clientkeystore.jks
   ```

   Entrez le mot de passe du magasin de clés lorsque vous y êtes invité.

L'exemple de code suivant montre comment configurer la SSL connexion à l'aide deJDBC.

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.Properties;

public class OracleSslConnectionTest {
    private static final String DB_SERVER_NAME = "dns-name-provided-by-amazon-rds";
    private static final Integer SSL_PORT = "ssl-option-port-configured-in-option-
group";
    private static final String DB_SID = "oracle-sid";
    private static final String DB_USER = "user-name";
    private static final String DB_PASSWORD = "password";
    // This key store has only the prod root ca.
    private static final String KEY_STORE_FILE_PATH = "file-path-to-keystore";
    private static final String KEY_STORE_PASS = "keystore-password";

    public static void main(String[] args) throws SQLException {
        final Properties properties = new Properties();
        final String connectionString = String.format(
                "jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=%s)(PORT=
%d))(CONNECT_DATA=(SID=%s)))",
                DB_SERVER_NAME, SSL_PORT, DB_SID);
        properties.put("user", DB_USER);
        properties.put("password", DB_PASSWORD);
        properties.put("oracle.jdbc.J2EE13Compliant", "true");
        properties.put("javax.net.ssl.trustStore", KEY_STORE_FILE_PATH);
        properties.put("javax.net.ssl.trustStoreType", "JKS");
        properties.put("javax.net.ssl.trustStorePassword", KEY_STORE_PASS);
        final Connection connection = DriverManager.getConnection(connectionString,
 properties);
        // If no exception, that means handshake has passed, and an SSL connection can
 be opened
    }
}
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Imposer une correspondance de DN avec une SSL connexion

Vous pouvez utiliser le paramètre Oracle `SSL_SERVER_DN_MATCH` pour imposer que le nom unique du serveur de base de données corresponde à son nom de service. Si vous appliquez les vérifications de correspondanceSSL, assurez-vous que le certificat provient du serveur. Si vous n'appliquez pas la vérification des correspondances, SSL exécutez-la mais autorisez la connexion, qu'il y ait ou non une correspondance. Si vous n'appliquez pas la correspondance, vous autorisez le serveur à potentiellement falsifier son identité.

Pour appliquer la correspondance de nom unique, ajoutez la propriété DN Match et utilisez la chaîne de connexion spécifiée ci-dessous.

Ajoutez la propriété à la connexion client pour appliquer la correspondance de nom unique.

```
properties.put("oracle.net.ssl_server_dn_match", "TRUE");
```

Utilisez la chaîne de connexion suivante pour appliquer la correspondance DN lors de l'utilisationSSL.

```
final String connectionString = String.format(
    "jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCPS)(HOST=%s)(PORT=%d))" +
    "(CONNECT_DATA=(SID=%s))" +
    "(SECURITY = (SSL_SERVER_CERT_DN =
\"C=US,ST=Washington,L=Seattle,O=Amazon.com,OU=RDS,CN=%s\")))",
    DB_SERVER_NAME, SSL_PORT, DB_SID, DB_SERVER_NAME);
```

Résolution des problèmes de SSL connexions

Il se peut que vous interrogiez votre base de données et que vous receviez l'erreur `ORA-28860`.

```
ORA-28860: Fatal SSL error
28860. 00000 - "Fatal SSL error"
*Cause: An error occurred during the SSL connection to the peer. It is likely that this
 side sent data which the peer rejected.
*Action: Enable tracing to determine the exact cause of this error.
```

Cette erreur se produit lorsque le client tente de se connecter en utilisant une version non prise en charge par le serveur. TLS Pour éviter cette erreur, modifiez le fichier sqlnet.ora et définissez `SSL_VERSION` la version correcte. TLS Pour plus d'informations, consultez le document de support Oracle 2748438.1 dans My Oracle Support.

Oracle Spatial

Amazon RDS prend en charge Oracle Spatial par l'intermédiaire de l'option `SPATIAL`. Oracle Spatial fournit un schéma SQL et des fonctions qui facilitent le stockage, la récupération, la mise à jour et la requête de collections de données spatiales dans une base de données Oracle. Pour plus d'informations, consultez Concepts spatiaux dans la documentation d'Oracle. Amazon RDS prend en charge Oracle Spatial dans toutes les éditions de toutes les versions prises en charge.

Comment fonctionnent les Spatial Patch Bundles (SPBs)

Chaque trimestre, RDS for Oracle publie de nouvelles versions mineures du moteur pour chacun des principaux moteurs pris en charge. Une version du moteur Release Update (RU) intègre les corrections de bogues d'Oracle en incluant les correctifs RU pour le trimestre spécifié. Une version du moteur SPB (Spatial Patch Bundle) contient des correctifs RU ainsi que des correctifs spécifiques à Oracle Spatial. Par exemple, 19.0.0.0.ru-2025-01.spb-1.r1 est une version mineure du moteur qui contient les correctifs RU de la version 19.0.0.0.ru-2025-01.rur-2025-01.rur-2025-01.r1 ainsi que des correctifs spatiaux. SPBs ne sont pris en charge que pour Oracle Database 19c.

SPBs fonctionnent de la même manière que RUs, bien qu'ils soient nommés différemment. Un RU utilise le format de dénomination 19.0.0.0.ru-2025-01.rur-2025-01.r1. Un nom SPB inclut le texte « spb », comme dans 19.0.0.0.ru-2025-01.spb-1.r1. Généralement, un SPB est publié 2 à 3 semaines après le RU trimestriel correspondant. Par exemple, 19.0.0.0.ru-2025-01.spb-1.r1 est publié après 19.0.0.0.ru-2025-01.rur-2025-01.r1.

RDS pour Oracle dispose de chemins distincts pour les mises à niveau automatiques des versions mineures de RUs et SPBs. Si votre instance de base de données utilise un RU, RDS met automatiquement à niveau votre instance vers un RU. Si votre instance de base de données utilise un SPB, RDS met à niveau votre instance vers un SPB. Lorsque vous ajoutez l'option Oracle Spatial à votre instance de base de données et que vous activez les mises à niveau automatiques, RDS place votre instance dans le chemin de mise à niveau pour SPBs.

Pour plus d'informations sur RUs et SPBs, voirMises à niveau des versions mineures d'Oracle. Pour obtenir la liste des bases de données prises en charge RUs et SPBs pour Oracle Database 19c, consultez Amazon RDS pour Oracle Database 19c (19.0.0.0) dans les notes de version d'Amazon RDS pour Oracle.

Prérequis pour Oracle Spatial

Les conditions suivantes sont requises pour utiliser Oracle Spatial :

- Assurez-vous que votre instance de base de données est d'une classe d'instance suffisante. Oracle Spatial n'est pas pris en charge pour les classes d'instance de base de données db.t3.small. Pour de plus amples informations, veuillez consulter RDS pour les classes d'instance de base de données Oracle.
- Assurez-vous que Mise à niveau automatique des versions mineures est activée pour votre instance de base de données. Cette option permet à votre instance de base de données de recevoir automatiquement des mises à niveau mineures de version du moteur de base de données quand elles sont disponibles, et est requise pour toutes les options qui installent la machine virtuelle Java (JVM) Oracle. Amazon RDS utilise cette option pour mettre à jour votre instance de base de données vers le dernier PSU (Patch Set Update) ou la dernière mise à jour (RU) Oracle. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Bonnes pratiques pour Oracle Spatial

Les bonnes pratiques suivantes sont requises pour utiliser Oracle Spatial :

- Pour une sécurité maximale, utilisez l'option `SPATIAL` avec Secure Sockets Layer (SSL). Pour plus d'informations, consultez Oracle Secure Sockets Layer (SSL).
- Configurez votre instance de base de données pour en restreindre l'accès. Pour plus d'informations, consultez Scénarios d'accès à un d'instances de base de données dans un VPC et Utilisation d'un(e) instance de base de données dans un VPC.

Ajout de l'option Oracle Spatial

La procédure générale suivante permet d'ajouter l'option `SPATIAL` à une instance de base de données :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Si Oracle Java Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option `SPATIAL` est ajoutée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous ajoutez l'option, vous n'avez pas besoin de redémarrer votre instance de base de données. Dès que le groupe d'options est actif, Oracle Spatial est disponible.

> **Note**
>
> Durant cette interruption, les fonctions de vérification de mot de passe sont brièvement désactivées. Vous pouvez également vous attendre à voir des événements liés aux fonctions de vérification de mot de passe durant l'interruption. Les fonctions de vérification de mot de passe sont activées de nouveau avant que l'instance de base de données Oracle ne soit disponible.

Pour ajouter l'option **SPATIAL** à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, choisissez l'édition Oracle de votre instance de base de données.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour plus d'informations, consultez Création d'un groupe d'options.
2. Ajoutez l'option SPATIAL au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Suppression de l'option Oracle Spatial

Après avoir abandonné tous les objets qui utilisent des types de données fournis par l'option `SPATIAL`, vous pouvez supprimer l'option à partir d'une instance de base de données. Si Oracle Java

Virtual Machine (JVM) n'est pas installé sur l'instance de base de données, il y a une brève panne lorsque l'option SPATIAL est supprimée. Il n'y a pas de panne si Oracle Java Virtual Machine (JVM) est déjà installé sur l'instance de base de données. Une fois que vous supprimez l'option SPATIAL, vous n'avez pas besoin de redémarrer votre instance de base de données.

Pour supprimer l'option **SPATIAL**

1. Sauvegardez vos données.

   > **Warning**
   >
   > Si l'instance utilise des types de données qui ont été activés dans le cadre de l'option, et si vous supprimez l'option SPATIAL, vous pouvez perdre des données. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

2. Vérifiez si des objets existants font référence à des types de données ou à des fonctionnalités de l'option SPATIAL.

   Si des options SPATIAL existent, l'instance peut rester bloquée lors de l'application du nouveau groupe d'options qui n'a pas l'option SPATIAL. Vous pouvez identifier les objets à l'aide des requêtes suivantes :

   ```
   SELECT OWNER, SEGMENT_NAME, TABLESPACE_NAME, BYTES/1024/1024 mbytes
   FROM   DBA_SEGMENTS
   WHERE  SEGMENT_TYPE LIKE '%TABLE%'
   AND    (OWNER, SEGMENT_NAME) IN
          (SELECT DISTINCT OWNER, TABLE_NAME
           FROM   DBA_TAB_COLUMNS
           WHERE  DATA_TYPE='SDO_GEOMETRY'
           AND    OWNER <> 'MDSYS')
   ORDER BY 1,2,3,4;

   SELECT OWNER, TABLE_NAME, COLUMN_NAME
   FROM   DBA_TAB_COLUMNS
   WHERE  DATA_TYPE = 'SDO_GEOMETRY'
   AND    OWNER <> 'MDSYS'
   ORDER BY 1,2,3;
   ```

3. Supprimez tous les objets qui font référence à des types de données ou à des fonctionnalités de l'option SPATIAL.

4. Effectuez l'une des actions suivantes :

   - Supprimez l'option `SPATIAL` du groupe d'options auquel elle appartient. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour plus d'informations, consultez Suppression d'une option d'un groupe d'options.
   - Modifiez l'instance de base de données et spécifiez un groupe d'options différent qui n'inclut pas l'option `SPATIAL`. Ce changement affecte une seule instance de base de données. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Oracle SQLT

Amazon RDS prend en charge Oracle SQLTXPLAIN (SQLT) grâce à l'utilisation de SQLT cette option. Vous pouvez l'utiliser SQLT avec n'importe quelle édition d'Oracle Database 19c ou supérieure.

L'`EXPLAIN PLAN`instruction Oracle peut déterminer le plan d'exécution d'une SQL instruction. Elle peut vérifier si l'optimiseur d'Oracle choisit un plan d'exécution particulier, comme une jointure de boucles imbriquées. Elle vous aide également à comprendre les décisions de l'optimiseur, par exemple, pourquoi celui-ci a choisi une jointure de boucles imbriquées plutôt qu'une jointure de hachage. `EXPLAIN PLAN` vous aide donc à comprendre les performances de l'instruction.

SQLTest un utilitaire Oracle qui produit un rapport. Le rapport inclut les statistiques des objets, les métadonnées des objets, les paramètres d'initialisation liés à l'optimiseur et d'autres informations qu'un administrateur de base de données peut utiliser pour ajuster une SQL instruction afin d'optimiser les performances. SQLTproduit un HTML rapport contenant des hyperliens vers toutes les sections du rapport.

Contrairement aux rapports Automatic Workload Repository ou Statspack, il SQLT fonctionne sur des instructions individuellesSQL. SQLTest un ensemble de SQL fichiers PL/ SQL et SQL *Plus qui collectent, stockent et affichent des données de performance.

Vous trouverez ci-dessous les versions Oracle prises en charge pour chaque SQLT version.

SQLTversion	Oracle Database 21c	Oracle Database 19c
2018-07-25.v1	Pris en charge	Pris en charge
2018-03-31.v1	Non pris en charge	Non pris en charge
2016-04-29.v1	Non pris en charge	Non pris en charge

Pour le télécharger SQLT et accéder aux instructions d'utilisation, procédez comme suit :

- Connectez-vous à votre compte My Oracle Support et ouvrez les documents suivants :
- À télécharger SQLT : Document 215187.1
- Pour les instructions SQLT d'utilisation : Document 1614107.1
- Pour les questions fréquemment posées concernant SQLT : Document 1454160.1

- Pour plus d'informations sur la lecture SQLT de la sortie : Document 1456176.1
- Pour interpréter le rapport principal : Document 1922234.1

Amazon RDS ne prend pas en charge les SQLT méthodes suivantes :

- `XPLORE`
- `XHUME`

Prérequis pour SQLT

Les conditions préalables à l'utilisation SQLT sont les suivantes :

- Vous devez supprimer les utilisateurs et les rôles requis parSQLT, s'ils existent.

  L'SQLToption crée les utilisateurs et les rôles suivants sur une instance de base de données :

  - `SQLTXPLAIN` user
  - `SQLTXADMIN` user
  - `SQLT_USER_ROLE` rôle

  Si votre instance de base de données possède l'un de ces utilisateurs ou rôles, connectez-vous à l'instance de base de données à l'aide d'un SQL client et supprimez-le en utilisant les instructions suivantes :

  ```
  DROP USER SQLTXPLAIN CASCADE;
  DROP USER SQLTXADMIN CASCADE;
  DROP ROLE SQLT_USER_ROLE CASCADE;
  ```

- Vous devez supprimer les espaces disque logiques requis parSQLT, s'ils existent.

  L'SQLToption crée les tablespaces suivants sur une instance de base de données :

  - `RDS_SQLT_TS`
  - `RDS_TEMP_SQLT_TS`

  Si votre instance de base de données possède ces tablespaces, connectez-vous à l'instance de base de données à l'aide d'un SQL client et supprimez-les.

SQLTparamètres des options

SQLTpeut fonctionner avec les fonctionnalités sous licence fournies par le pack Oracle Tuning et le pack Oracle Diagnostics. Le pack Oracle Tuning inclut le SQL Tuning Advisor, et le pack Oracle Diagnostics inclut le référentiel de charge de travail automatique. Les SQLT paramètres activent ou désactivent l'accès à ces fonctionnalités depuisSQLT.

Amazon RDS prend en charge les paramètres suivants pour SQLT cette option.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
`LICENSE_PACK`	`T`, `D`, `N`	`N`	Les packs de gestion Oracle auxquels vous souhaitez accéderSQLT. Entrez l'une des valeurs suivantes : • Tindique que vous possédez une licence pour Oracle Tuning Pack et Oracle Diagnostics Pack, et que vous souhaitez accéder au SQL Tuning Advisor et au référentiel Automatic Workload depuisSQLT. • Dindique que vous possédez une licence pour le pack Oracle Diagnostics et que vous souhaitez accéder au référentiel de charge de travail automatique à partir deSQLT. • Nindique que vous ne possédez pas de licence pour Oracle Tuning Pack et Oracle Diagnostics Pack, ou que vous possédez une licence pour l'un ou les deux, mais que vous ne SQLT souhaitez pas y accéder.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
			Note Amazon RDS ne fournit pas de licences pour ces packs de gestion Oracle. Si vous indiquez que vous souhaitez utiliser un pack qui n'est pas inclus dans votre instance de base de données, vous pouvez l'utiliser SQLT avec l'instance de base de données. Cependant, je ne SQLT peux pas accéder au pack et le SQLT rapport n'inclut pas les données du pack. Par exemple, si vous le spécifiezT, mais que l'instance de base de données n'inclut pas le pack de réglage Oracle, cela SQLT fonctionne sur l'instance de base de données, mais que le rapport généré ne contient pas de données relatives au pack de réglage Oracle.

Paramètre d'option	Valeurs valides	Valeur par défaut	Description
VERSION	2016-04-29.v1 2018-03-31.v1 2018-07-25.v1	2016-04-29.v1	Version SQLT que vous souhaitez installer. **Note** Pour Oracle Database 19c et 21c, la seule version prise en charge est 2018-07-25.v1 . Il s'agit de la version par défaut pour ces versions.

Ajouter l'SQLToption

Le processus général d'ajout de l'SQLToption à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'SQLToption au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Une fois que vous avez ajouté l'SQLToption, elle est active dès que le groupe d'options SQLT est actif.

Pour ajouter l'SQLToption à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, sélectionnez l'édition d'Oracle que vous voulez utiliser. L'SQLToption est prise en charge sur toutes les éditions.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.

2. Ajoutez l'SQLToption au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

   - Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

   - Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

4. (Facultatif) Vérifiez l'SQLTinstallation sur chaque instance de base de données avec l'SQLToption.

   a. Utilisez un SQL client pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

      Pour plus d'informations sur la connexion à une instance de base de données Oracle à l'aide d'un SQL client, consultezConnexion à votre instance de base de données Oracle.

   b. Exécutez la requête suivante :

      ```
      SELECT sqltxplain.sqlt$a.get_param('tool_version') sqlt_version FROM DUAL;
      ```

      La requête renvoie la version actuelle de l'SQLToption sur AmazonRDS. `12.1.160429`est un exemple d'une version de SQLT celui-ci disponible sur AmazonRDS.

5. Modifiez les mots de passe des utilisateurs créés par l'SQLToption.

   a. Utilisez un SQL client pour vous connecter à l'instance de base de données en tant qu'utilisateur principal.

   b. Exécutez l'SQLinstruction suivante pour modifier le mot de passe de l'`SQLTXADMIN`utilisateur :

```
ALTER USER SQLTXADMIN IDENTIFIED BY new_password ACCOUNT UNLOCK;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

c. Exécutez l'SQLinstruction suivante pour modifier le mot de passe de l'`SQLTXPLAIN`utilisateur :

```
ALTER USER SQLTXPLAIN IDENTIFIED BY new_password ACCOUNT UNLOCK;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

> **Note**
>
> La mise à niveau SQLT nécessite de désinstaller une ancienne version de SQLT puis d'installer la nouvelle version. Ainsi, toutes les SQLT métadonnées peuvent être perdues lors de la mise à niveauSQLT. La mise à niveau d'une version majeure d'une base de données entraîne également la désinstallation et la réinstallation. SQLT La mise à niveau d'Oracle Database 19c vers Oracle Database 21c est un exemple de mise à niveau de version majeure.

En utilisant SQLT

SQLTfonctionne avec l'utilitaire Oracle SQL *Plus.

À utiliser SQLT

1. Téléchargez le fichier SQLT .zip à partir du document 215187.1 sur le site My Oracle Support.

   > **Note**
   >
   > Vous ne pouvez pas télécharger la version SQLT 12.1.160429 depuis le site My Oracle Support. Oracle a supprimé cette ancienne version.

2. Décompressez le SQLT fichier .zip.
3. Depuis une invite de commande et accédez au répertoire `sqlt/run` sur votre système de fichiers.
4. À partir de l'invite de commande, ouvrez SQL *Plus et connectez-vous à l'instance de base de données en tant qu'utilisateur principal.

   Pour plus d'informations sur la connexion à une instance de base de données à l'aide de SQL *Plus, consultezConnexion à votre instance de base de données Oracle.
5. Obtenez l'SQLidentifiant d'une SQL déclaration :

   ```
   SELECT SQL_ID FROM V$SQL WHERE SQL_TEXT='sql_statement';
   ```

   Votre sortie est similaire à ce qui suit :

   ```
   SQL_ID
   -------------
   chvsmttqjzjkn
   ```
6. Analysez une SQL déclaration avec SQLT :

   ```
   START sqltxtract.sql sql_id sqltxplain_user_password
   ```

   Par exemple, pour l'SQLID`chvsmttqjzjkn`, entrez ce qui suit :

```
START sqltxtract.sql chvsmttqjzjkn sqltxplain_user_password
```

SQLTgénère le HTML rapport et les ressources associées sous forme de fichier .zip dans le répertoire à partir duquel la SQLT commande a été exécutée.

7. (Facultatif) Pour permettre aux utilisateurs de l'application de diagnostiquer SQL les instructions avecSQLT, accordez SQLT_USER_ROLE à chaque utilisateur de l'application l'instruction suivante :

```
GRANT SQLT_USER_ROLE TO application_user_name;
```

> **Note**
>
> Oracle ne recommande pas d'exécuter SQLT avec l'SYSutilisateur ou avec les utilisateurs qui ont le DBA rôle. Il est recommandé d'exécuter des SQLT diagnostics à l'aide du compte de l'utilisateur de l'application, en l'accordant SQLT_USER_ROLE à l'utilisateur de l'application.

Mise à niveau de l'SQLToption

Avec Amazon RDS pour Oracle, vous pouvez passer SQLT de votre version existante à une version supérieure. Pour mettre à niveau l'SQLToption, suivez les étapes 1 à 3 En utilisant SQLT de la nouvelle version deSQLT. De même, si vous avez accordé des privilèges pour la version précédente SQLT ou à l'étape 7 de cette section, accordez-les à nouveau pour la nouvelle SQLT version.

La mise à niveau de l'SQLToption entraîne la perte des métadonnées de l'ancienne SQLT version. Le schéma de l'ancienne SQLT version et les objets associés sont supprimés, et la nouvelle version de SQLT est installée. Pour plus d'informations sur les modifications apportées à la dernière SQLT version, consultez le document 1614201.1 sur le site My Oracle Support.

> **Note**
>
> Les mises à niveau vers une version antérieure ne sont pas prises en charge.

Modification des SQLT paramètres

Après avoir activé l'optionSQLT, vous pouvez modifier les `VERSION` paramètres `LICENSE_PACK` et.

Pour plus d'informations sur la modification des paramètres d'options, consultez Modification d'un paramètre d'option. Pour plus d'informations sur chaque paramètre, consultez SQLTparamètres des options.

Supprimer l'SQLToption

Vous pouvez effectuer une suppression SQLT d'une instance de base de données.

Pour effectuer une suppression SQLT d'une instance de base de données, effectuez l'une des opérations suivantes :

- Pour supprimer SQLT de plusieurs instances de base de données, supprimez l'SQLToption du groupe d'options auquel les instances de base de données appartiennent. Ce changement affecte toutes les instances de bases de données qui utilisent le groupe d'options. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.

- Pour supprimer SQLT d'une seule instance de base de données, modifiez l'instance de base de données et spécifiez un autre groupe d'options qui n'inclut pas l'SQLToption. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Oracle Statspack

L'option Oracle Statspack installe et active la fonction de statistiques de performances Oracle Statspack. Oracle Statspack est un ensemble de SQL scripts PL/ et SQL *Plus qui collectentSQL, stockent et affichent des données de performance. Pour plus d'informations sur l'utilisation d'Oracle Statspack, consultez Oracle Statspack dans la documentation Oracle.

> **Note**
>
> Oracle Statspack n'est plus pris en charge par Oracle et a été remplacé par le référentiel automatique de charge de travail ()AWR, plus avancé. AWRest disponible uniquement pour les clients d'Oracle Enterprise Edition qui ont acheté le pack de diagnostic. Vous pouvez utiliser Oracle Statspack avec n'importe quel moteur de base de données Oracle sur Amazon. RDS Vous ne pouvez pas exécuter Oracle Statspack sur Amazon RDS read replicas.

Configurer Oracle Statspack

Pour exécuter des scripts Statspack, vous devez ajouter l'option Statspack.

Pour configurer Oracle Statspack

1. Dans un SQL client, connectez-vous à la base de données Oracle avec un compte administratif.
2. Procédez comme suit, selon que Statspack est installé ou pas :

   - Si Statspack est installé et que le compte `PERFSTAT` lui est associé, passez à l'étape 4.
   - Si Statspack n'est pas installé et que le compte `PERFSTAT` existe, supprimez le compte comme suit :

     ```
     DROP USER PERFSTAT CASCADE;
     ```

     Sinon, une tentative d'ajout de l'option Statspack génère une erreur et un `RDS-Event-0058`.
3. Ajoutez l'option Statspack à un groupe d'options. Consultez Ajout d'une option à un groupe d'options.

   Amazon installe RDS automatiquement les scripts Statspack sur l'instance de base de données, puis configure le compte. `PERFSTAT`

4. Réinitialisez le mot de passe à l'aide de l'SQLinstruction suivante, en remplaçant pwd par votre nouveau mot de passe :

```
ALTER USER PERFSTAT IDENTIFIED BY pwd ACCOUNT UNLOCK;
```

   Vous pouvez vous connecter à l'aide du compte d'utilisateur PERFSTAT et exécuter les scripts Statspack.

5. Accordez le CREATE JOB privilège au PERFSTAT compte en utilisant la déclaration suivante :

```
GRANT CREATE JOB TO PERFSTAT;
```

6. Assurez-vous que les événements d'attente inactifs dans la table PERFSTAT.STATS $IDLE_EVENT sont renseignés.

   En raison du bogue 28523746 d'Oracle, les événements d'attente inactifs dans PERFSTAT.STATS$IDLE_EVENT peuvent ne pas être renseignés. Pour vous assurer que tous les événements inactifs sont disponibles, exécutez l'instruction suivante :

```
INSERT INTO PERFSTAT.STATS$IDLE_EVENT (EVENT)
SELECT NAME FROM V$EVENT_NAME WHERE WAIT_CLASS='Idle'
MINUS
SELECT EVENT FROM PERFSTAT.STATS$IDLE_EVENT;
COMMIT;
```

Génération de rapports Statspack

Un rapport Statspack compare deux instantanés.

Pour générer des rapports Statspack

1. Dans un SQL client, connectez-vous à la base de données Oracle avec le PERFSTAT compte.
2. Créez un instantané à l'aide de l'une des techniques suivantes :

   - Créez manuellement un instantané Statspack.
   - Créez une tâche qui prend un instantané Statspack après un intervalle de temps donné. Par exemple, la tâche suivante crée un instantané Statspack chaque heure :

```
VARIABLE jn NUMBER;
```

```
exec dbms_job.submit(:jn, 'statspack.snap;',SYSDATE,'TRUNC(SYSDATE
+1/24,''HH24'')');
COMMIT;
```

3. Affichez les instantanés à l'aide de la requête suivante :

```
SELECT SNAP_ID, SNAP_TIME FROM STATS$SNAPSHOT ORDER BY 1;
```

4. Exécutez la RDS procédure `rdsadmin.rds_run_spreport` Amazon en remplaçant begin_snap et end_snap par le snapshot : IDs

```
exec rdsadmin.rds_run_spreport(begin_snap,end_snap);
```

Par exemple, la commande suivante crée un rapport basé sur l'intervalle entre les instantanés Statspack 1 et 2 :

```
exec rdsadmin.rds_run_spreport(1,2);
```

Le nom de fichier du rapport Statspack inclut le numéro des deux instantanés. Par exemple, un fichier de rapport créé à l'aide des instantanés Statspack 1 et 2 se nommera `ORCL_spreport_1_2.lst`.

5. Surveillez la sortie pour détecter des erreurs.

Oracle Statspack effectue des vérifications avant d'exécuter le rapport. Par conséquent, vous pouvez également voir des messages d'erreur dans la sortie de la commande. Vous pouvez par exemple essayer de générer un rapport basé sur une plage non valable, dans laquelle la valeur de l'instantané Statspack de début est supérieure à la valeur de fin. Dans ce cas, la sortie affiche le message d'erreur, mais le moteur de base de données ne génère pas de fichier d'erreur.

```
exec rdsadmin.rds_run_spreport(2,1);
*
ERROR at line 1:
ORA-20000: Invalid snapshot IDs. Find valid ones in perfstat.stats$snapshot.
```

Si vous utilisez un numéro erroné d'instantané Statspack, la sortie affiche une erreur. Par exemple, si vous essayez de générer un rapport pour les instantanés 1 et 50, mais que l'instantané 50 n'existe pas, la sortie affiche une erreur.

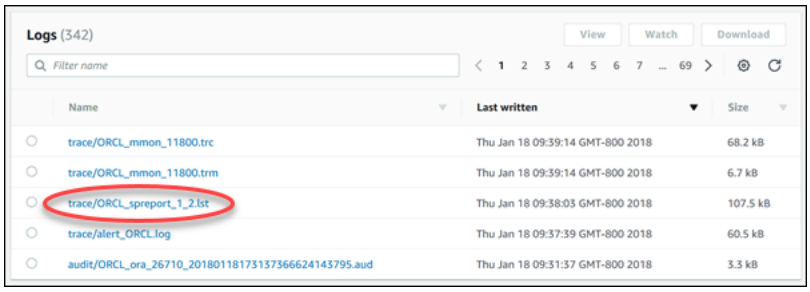
```
exec rdsadmin.rds_run_spreport(1,50);
```

```
*
ERROR at line 1:
ORA-20000: Could not find both snapshot IDs
```

6. (Facultatif)

   Pour récupérer le rapport, appelez les procédures du fichier de trace, selon les explications de la section Utilisation des fichiers de trace Oracle.

   Vous pouvez également télécharger le rapport Statspack depuis la RDS console. Accédez à la section Journal des détails de l'instance de base de données et choisissez Télécharger. L'exemple suivant montre `trace/ORCL_spreport_1_2.lst`



   Si une erreur se produit lors de la génération d'un rapport, le moteur de base de données utilise les mêmes conventions de dénomination que pour un rapport, mais avec une extension de `.err`. Par exemple, si une erreur s'est produite lors de la création d'un rapport à l'aide des instantanés Statspack 1 et 7, le fichier de rapport sera nommé `ORCL_spreport_1_7.err`. Vous pouvez télécharger le rapport d'erreurs en utilisant les mêmes techniques que pour un rapport d'instantané standard.

Suppression d'instantanés Statspack

Pour supprimer une plage d'instantanés Oracle Statspack, utilisez la commande suivante :

```
exec statspack.purge(begin snap, end snap);
```

Fuseau horaire Oracle

Utilisez l'option de fuseau horaire pour modifier le fuseau horaire système utilisé par votre instance de base de données Oracle. Par exemple, vous devrez peut-être modifier le fuseau horaire d'une instance de base de données afin qu'elle soit compatible avec un environnement sur site ou une application héritée. L'option de fuseau horaire change le fuseau horaire au niveau de l'hôte. La modification du fuseau horaire impacte toutes les valeurs et colonnes date, y compris `SYSDATE` et `SYSTIMESTAMP`.

L'option de fuseau horaire diffère de la commande `rdsadmin_util.alter_db_time_zone`. La commande `alter_db_time_zone` modifie le fuseau horaire uniquement pour certains types de données. L'option du fuseau horaire modifie le fuseau horaire de toutes les valeurs et colonnes date. Pour plus d'informations sur `alter_db_time_zone`, consultez Définition du fuseau horaire de la base de données. Pour de plus amples informations sur les VPC, veuillez consulter Considérations relatives au fuseau horaire.

Restrictions relatives au réglage du fuseau horaire

L'option de fuseau horaire est persistante et permanente. Par conséquent, vous ne pouvez pas effectuer les opérations suivantes :

- Supprimez l'option d'un groupe d'options après avoir ajouté l'option de fuseau horaire.
- Supprimer le groupe d'options d'une instance de base de données après l'avoir ajouté
- Remplacer le paramètre de fuseau horaire de l'option par un autre fuseau horaire

Recommandations pour le réglage du fuseau horaire

Avant d'ajouter l'option de fuseau horaire à votre base de données de production, nous vous recommandons vivement d'effectuer les opérations suivantes :

- Prenez un instantané de votre instance de base de données. Si vous définissez accidentellement le fuseau horaire de manière incorrecte, vous devez rétablir le paramètre de fuseau horaire précédent de votre instance de base de données. Pour de plus amples informations, veuillez consulter Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
- Ajoutez l'option de fuseau horaire à une instance de base de données de test. L'ajout de l'option de fuseau horaire peut entraîner des problèmes avec les tables qui utilisent la date système

pour ajouter des dates ou des heures. Nous vous recommandons d'analyser vos données et vos applications sur l'instance de test. Vous pouvez ainsi évaluer l'impact de la modification du fuseau horaire sur votre instance de production.

Si votre instance de base de données utilise le groupe d'options par défaut, procédez comme suit :

1. Prenez un instantané de votre instance de base de données.
2. Ajoutez l'option de fuseau horaire à votre instance de base de données.

Si votre instance de base de données utilise actuellement un groupe d'options autre que celui par défaut, procédez comme suit :

1. Prenez un instantané de votre instance de base de données.
2. Créez un nouveau groupe d'options.
3. Ajoutez-y l'option de fuseau horaire, ainsi que toutes les autres options actuellement associées au groupe d'options existant.

   Cela empêche la désinstallation des options existantes lors de l'activation de l'option de fuseau horaire.
4. Ajoutez le groupe d'options à votre instance de base de données.

Paramètres d'option de fuseau horaire

Amazon RDS prend en charge les paramètres suivants pour l'option de fuseau horaire.

Paramètre d'option	Valeurs valides	Description
`TIME_ZONE`	Un des fuseaux horaires disponibles. Pour obtenir la liste complète, consultez Fuseaux horaires disponibles.	Nouveau fuseau horaire de votre instance de base de données.

Ajout de l'option de fuseau horaire

Procédez comme suit pour ajouter l'option de fuseau horaire à votre instance de base de données :

1. (Recommandé) Prenez un instantané de votre instance de base de données.
2. Effectuez l'une des tâches suivantes :
   - Créez un nouveau groupe d'options à partir de zéro. Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
   - Copiez un groupe d'options existant à l'aide du AWS CLI ouAPI. Pour de plus amples informations, veuillez consulter Copie d'un groupe d'options.
   - Réutilisez un groupe d'options existant autre que celui par défaut. La meilleure pratique consiste à utiliser un groupe d'options qui n'est actuellement associé à aucune instance de base de données ni à aucun instantané.
3. Ajoutez la nouvelle option au groupe d'options de l'étape précédente.
4. Si le groupe d'options actuellement associé à votre instance de base de données possède des options activées, ajoutez ces options à votre nouveau groupe d'options. Cette stratégie empêche la désinstallation des options existantes lors de l'activation de la nouvelle option.
5. Ajoutez le nouveau groupe d'options à votre instance de base de données.

Lorsque vous ajoutez l'option de fuseau horaire, une brève interruption de service se produit pendant le redémarrage automatique de votre instance de base de données.

Console

Pour ajouter l'option de fuseau horaire à un groupe d'options et l'associer à une instance de base de données

1. Dans la RDS console, choisissez Groupes d'options.
2. Choisissez le nom du groupe d'options auquel vous souhaitez ajouter l'option.
3. Sélectionnez Ajouter une option.
4. Dans Nom de l'option, choisissez Fuseau horaire, puis configurez les paramètres de l'option.
5. Associez le groupe d'options à une instance de base de données nouvelle ou existante :
   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Lorsque vous ajoutez la nouvelle option à une instance de base de données existante, une brève interruption se produit pendant

le redémarrage automatique de votre instance de base de données. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

AWS CLI

L'exemple suivant utilise la commande AWS CLI add-option-to-option-group pour ajouter l'`Timezone`option et le paramètre d'`TIME_ZONE`option à un groupe d'options appelé`myoptiongroup`. Le fuseau horaire par défaut est défini sur `Africa/Cairo`.

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name "myoptiongroup" \
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=Africa/
Cairo}]" \
    --apply-immediately
```

Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name "myoptiongroup" ^
    --options "OptionName=Timezone,OptionSettings=[{Name=TIME_ZONE,Value=Africa/
Cairo}]" ^
    --apply-immediately
```

Modification des paramètres de fuseau horaire

L'option de fuseau horaire est persistante et permanente. Vous ne pouvez pas supprimer l'option d'un groupe d'options après l'avoir ajoutée. Vous ne pouvez pas supprimer le groupe d'options d'une instance de base de données après l'avoir ajouté. Vous ne pouvez pas remplacer le paramètre de fuseau horaire de l'option par un autre fuseau horaire. Si vous définissez le fuseau horaire de manière incorrecte, restaurez un instantané de votre instance de base de données antérieur à l'ajout de l'option de fuseau horaire.

Suppression de l'option de fuseau horaire

L'option de fuseau horaire est persistante et permanente. Vous ne pouvez pas supprimer l'option d'un groupe d'options après l'avoir ajoutée. Vous ne pouvez pas supprimer le groupe d'options d'une instance de base de données après l'avoir ajouté. Pour supprimer l'option de fuseau horaire,

restaurez un instantané de votre instance de base de données antérieur à l'ajout de l'option de fuseau horaire.

Fuseaux horaires disponibles

Vous pouvez utiliser les valeurs suivantes pour l'option de fuseau horaire.

disponibilité	Fuseau horaire
Afrique	Africa/Cairo, Africa/Casablanca, Africa/Harare, Africa/Lagos, Africa/Luanda, Africa/Monrovia, Africa/Nairobi, Africa/Tripoli, Africa/Windhoek
Amérique	America/Araguaina, America/Argentina/Buenos_Aires, America/Asuncion, America/Bogota, America/Caracas, America/Chicago, America/Chihuahua, America/Cuiaba, America/Denver, America/Detroit, America/Fortaleza, America/Godthab, America/Guatemala, America/Halifax, America/Lima, America/Los_Angeles, America/Manaus, America/Matamoros, America/Mexico_City, America/Monterrey, America/Montevideo, America/New_York, America/Phoenix, America/Santiago, America/Sao_Paulo, America/Tijuana, America/Toronto
Asie	Asia/Amman, Asia/Ashgabat, Asia/Baghdad, Asia/Baku, Asia/Bangkok, Asia/Beirut, Asia/Calcutta, Asia/Damascus, Asia/Dhaka, Asia/Hong_Kong, Asia/Irkutsk, Asia/Jakarta, Asia/Jerusalem, Asia/Kabul, Asia/Karachi, Asia/Kathmandu, Asia/Kolkata, Asia/Krasnoyarsk, Asia/Magadan, Asia/Manila, Asia/Muscat, Asia/Novosibirsk, Asia/Rangoon, Asia/Riyadh, Asia/Seoul, Asia/Shanghai, Asia/Singapore, Asia/Taipei, Asia/Tehran, Asia/Tokyo, Asia/Ulaanbaatar, Asia/Vladivostok, Asia/Yakutsk, Asia/Yerevan
Atlantique	Atlantic/Azores, Atlantic/Cape_Vert
Australie	Australia/Adelaide, Australia/Brisbane, Australia/Darwin, Australia/Eucla, Australia/Hobart, Australia/Lord_Howe, Australia/Perth, Australia/Sydney
Brésil	Brazil/DeNoronha, Brazil/East
Canada	Canada/Newfoundland, Canada/Saskatchewan
Etc	Etc/ -3 GMT

disponibilité	Fuseau horaire
Europe	Europe/Amsterdam, Europe/Athens, Europe/Berlin, Europe/Dublin, Europe/Helsinki, Europe/Kaliningrad, Europe/London, Europe/Madrid, Europe/Moscow, Europe/Paris, Europe/Prague, Europe/Rome, Europe/Sarajevo
Pacifique	Pacific/Apia, Pacific/Auckland, Pacific/Chatham, Pacific/Fiji, Pacific/Guam, Pacific/Honolulu, Pacific/Kiritimati, Pacific/Marquesas, Pacific/Samoa, Pacific/Tongatapu, Pacific/Wake
ETATS-UNIS	US/Alaska, US/Central, US/East-Indiana, US/Eastern, US/Pacific
UTC	UTC

Mise à niveau automatique du fichier sur le fuseau horaire Oracle

Avec `TIMEZONE_FILE_AUTOUPGRADE` cette option, vous pouvez mettre à niveau le fichier de fuseau horaire actuel vers la dernière version de votre instance RDS de base de données Oracle.

Rubriques

- Présentation des fichiers sur le fuseau horaire Oracle
- Politiques pour la mise à jour de votre fichier sur le fuseau horaire
- Temps d'arrêt pendant la mise à jour du fichier sur le fuseau horaire
- Préparation de la mise à jour du fichier sur le fuseau horaire
- Ajout de l'option de mise à niveau automatique du fichier sur le fuseau horaire
- Vérification de vos données après la mise à jour du fichier sur le fuseau horaire

Présentation des fichiers sur le fuseau horaire Oracle

Un fichier sur le fuseau horaire de la base de données Oracle contient les informations suivantes :

- Décalage par rapport au temps universel coordonné (UTC)
- Délais de transition pour l'heure d'été (DST)
- Abréviations pour l'heure normale et DST

La base de données Oracle fournit plusieurs versions de fichiers sur le fuseau horaire. Lorsque vous créez une base de données Oracle dans un environnement sur site, vous choisissez la version du fichier sur le fuseau horaire. Pour plus d'informations, consultez Choosing a Time Zone File (Choix d'un fichier sur le fuseau horaire) dans le Guide de prise en charge de la mondialisation de la base de données Oracle.

Si les règles changent pourDST, Oracle publie de nouveaux fichiers de fuseau horaire. Oracle publie ces nouveaux fichiers de fuseau horaire indépendamment du calendrier des mises à jour trimestrielles (RUs) et des révisions des mises à jour des versions (RURs). Les fichiers sur le fuseau horaire se trouvent sur l'hôte de la base de données dans le répertoire `$ORACLE_HOME/oracore/zoneinfo/`. Les noms des fichiers de fuseau horaire utilisent le format DSTv*`version`*, comme dansDSTv35.

Comment le fichier sur le fuseau horaire affecte le transfert des données

Dans Oracle Database, le type de données `TIMESTAMP WITH TIME ZONE` stocke les données relatives à l'horodatage et au fuseau horaire. Les données avec le type de données `TIMESTAMP WITH TIME ZONE` utilisent les règles de la version du fichier sur le fuseau horaire associée. Ainsi, les `TIMESTAMP WITH TIME ZONE` données existantes sont affectées lorsque vous mettez à jour le fichier de fuseau horaire.

Des problèmes peuvent survenir lorsque vous transférez des données entre des bases de données utilisant différentes versions du fichier sur le fuseau horaire. Par exemple, si vous importez des données depuis une base de données source dont la version de fichier de fuseau horaire est supérieure à celle de la base de données cible, la base de données émet l'`ORA-39405`erreur. Auparavant, vous deviez contourner cette erreur en utilisant l'une des techniques suivantes :

- Créez une instance de base de données RDS pour Oracle avec le fichier de fuseau horaire souhaité, exportez les données de votre base de données source, puis importez-les dans la nouvelle base de données.
- Utilisez AWS DMS la réplication logique pour migrer vos données.

Mises à jour automatiques à l'aide de l'AUTOUPGRADEoption TIMEZONE FILE _ _

Lorsque le groupe d'options attaché à votre instance de base de données RDS pour Oracle inclut l'`TIMEZONE_FILE_AUTOUPGRADE`option, RDS met automatiquement à jour vos fichiers de fuseau horaire. En veillant à ce que vos bases de données Oracle utilisent la même version de fichier de fuseau horaire, vous évitez les techniques manuelles fastidieuses lorsque vous déplacez des données entre différents environnements. L'`TIMEZONE_FILE_AUTOUPGRADE`option est prise en charge à la fois pour les bases de données de conteneurs (CDBs) et nonCDBs.

Lorsque vous ajoutez l'option `TIMEZONE_FILE_AUTOUPGRADE` à votre groupe d'options, vous pouvez choisir si vous souhaitez l'ajouter immédiatement ou pendant la fenêtre de maintenance. Une fois que votre instance de base de données a appliqué la nouvelle option, RDS vérifie si elle peut installer une version plus récente DSTv*`version`* dans le fichier. La cible DSTv*`version`* dépend de ce qui suit :

- La version de moteur mineure que votre instance de base de données exécute actuellement
- La version de moteur mineure vers laquelle vous souhaitez mettre à niveau votre instance de base de données

Par exemple, la version actuelle de votre fichier de fuseau horaire peut êtreDSTv33. Lorsque RDS la mise à jour est appliquée à votre groupe d'options, elle peut déterminer qu'il DSTv34 est actuellement disponible sur votre système de fichiers d'instance de base de données. RDSmettra alors DSTv34 automatiquement à jour votre fichier de fuseau horaire.

Pour trouver les DST versions disponibles dans les mises à jour prises en chargeRDS, consultez les correctifs dans les notes de publication d'Amazon Relational Database Service (RDSAmazon) pour Oracle. Par exemple, la version 19.0.0.0.ru-10.rur-05.rur-10.r1 répertorie le correctif 34533061 : - - 022C. RDBMS DSTV39 UPDATE TZDATA2

Politiques pour la mise à jour de votre fichier sur le fuseau horaire

La mise à niveau de votre moteur de base de données et l'ajout de l'`TIMEZONE_FILE_AUTOUPGRADE`option à un groupe d'options sont des opérations distinctes. L'ajout de `TIMEZONE_FILE_AUTOUPGRADE` cette option lance la mise à jour de votre fichier de fuseau horaire si un fichier plus récent est disponible. Vous exécutez les commandes suivantes (seules les options pertinentes sont affichées) immédiatement ou lors de la fenêtre de maintenance suivante :

- Mettez à niveau votre moteur de base de données uniquement à l'aide de la RDS CLI commande suivante :

```
modify-db-instance --engine-version name ...
```

- Ajoutez l'`TIMEZONE_FILE_AUTOUPGRADE`option uniquement à l'aide de la CLI commande suivante :

```
add-option-to-option-group --option-group-name name --options
 OptionName=TIMEZONE_FILE_AUTOUPGRADE ...
```

- Mettez à niveau votre moteur de base de données et ajoutez un nouveau groupe d'options à votre instance à l'aide de la CLI commande suivante :

```
modify-db-instance --engine-version name --option-group-name name ...
```

Votre stratégie de mise à jour varie selon que vous souhaitez mettre à niveau votre base de données et votre fichier de fuseau horaire en même temps ou effectuer une seule de ces opérations. N'oubliez pas que si vous mettez à jour votre groupe d'options puis que vous mettez à niveau votre moteur de base de données dans le cadre d'APIopérations distinctes, il est possible qu'une mise à jour du fichier

de fuseau horaire soit actuellement en cours lorsque vous mettez à niveau votre moteur de base de données.

Les exemples de cette section supposent ce qui suit :

- Vous n'avez pas encore été ajouté `TIMEZONE_FILE_AUTOUPGRADE` au groupe d'options actuellement associé à votre instance de base de données.
- Votre instance de base de données utilise la version 19.0.0.0.ru-2019-07.rur-2019-07.r1 et un fichier de fuseau horaire. DSTv33
- Le système de fichiers de votre instance de base de données inclut un fichierDSTv34.
- La mise à jour 19.0.0.0.ru-05.10.rur-2_10.r1 inclut. DSTv35

Pour mettre à jour votre fichier sur le fuseau horaire, vous pouvez utiliser les stratégies suivantes.

Rubriques

- Mettre à jour le fichier sur le fuseau horaire sans mettre à niveau le moteur
- Mettre à niveau le fichier sur le fuseau horaire et la version du moteur de base de données
- Mettre à niveau la version de votre moteur de base de données sans mettre à jour le fichier sur le fuseau horaire

Mettre à jour le fichier sur le fuseau horaire sans mettre à niveau le moteur

Dans ce scénario, votre base de données utiliseDSTv33, mais elle DSTv34 est disponible sur votre système de fichiers d'instance de base de données. Vous souhaitez mettre à jour le fichier de fuseau horaire utilisé par votre instance de base de données de DSTv33 àDSTv34, mais vous ne souhaitez pas mettre à niveau votre moteur vers une nouvelle version mineure, qui inclutDSTv35.

Dans une `add-option-to-option-group` commande, ajoutez `TIMEZONE_FILE_AUTOUPGRADE` au groupe d'options utilisé par votre instance de base de données. Indiquez si vous souhaitez ajouter l'option immédiatement ou la reporter à la fenêtre de maintenance. Après avoir appliqué l'`TIMEZONE_FILE_AUTOUPGRADE`option, RDS procédez comme suit :

1. Vérifie l'existence d'une nouvelle DST version.
2. Détermine ce qui DSTv34 est disponible sur le système de fichiers.
3. Met à jour immédiatement le fichier de fuseau horaire.

Mettre à niveau le fichier sur le fuseau horaire et la version du moteur de base de données

Dans ce scénario, votre base de données utiliseDSTv33, mais elle DSTv34 est disponible sur votre système de fichiers d'instance de base de données. Vous souhaitez mettre à niveau votre moteur de base de données vers la version mineure 19.0.0.0.ru-05.10.rur-05.r1, qui inclut DSTv35 et mettre à jour votre fichier de fuseau horaire pendant la mise à niveau du moteur. DSTv35 Ainsi, votre objectif est d'ignorer DSTv34 et de mettre à jour vos fichiers de fuseau horaire directement versDSTv35.

Pour mettre à niveau le moteur et le fichier de fuseau horaire en même temps, exécutez `modify-db-instance` les `--engine-version` options `--option-group-name` et. Vous pouvez exécuter la commande immédiatement ou la reporter à la fenêtre de maintenance. `In --option-group-name`, spécifiez un groupe d'options qui inclut l'`TIMEZONE_FILE_AUTOUPGRADE`option. Par exemple :

```
aws rds modify-db-instance
    --db-instance-identifier my-instance \
    --engine-version new-version \
    ----option-group-name og-with-timezone-file-autoupgrade \
    --apply-immediately
```

RDScommence la mise à niveau de votre moteur vers la version 19.0.0.0.ru-05.10.rur-05.r1. Après avoir appliqué l'`TIMEZONE_FILE_AUTOUPGRADE`option, RDS vérifie s'il existe une nouvelle DST version, vérifie qu'elle DSTv35 est disponible dans 19.0.0.0.ru-10.rur-05.r1 et lance immédiatement la mise à jour vers. DSTv35

Pour mettre à niveau votre moteur immédiatement, puis mettre à niveau votre fichier de fuseau horaire, effectuez les opérations dans l'ordre :

1. Mettez à niveau votre moteur de base de données uniquement à l'aide de la CLI commande suivante :

   ```
   aws rds modify-db-instance \
       --db-instance-identifier my-instance \
       --engine-version new-version \
       --apply-immediately
   ```

2. Ajoutez l'`TIMEZONE_FILE_AUTOUPGRADE`option au groupe d'options attaché à votre instance à l'aide de la CLI commande suivante :

   ```
   aws rds add-option-to-option-group \
   ```

```
    --option-group-name og-in-use-by-your-instance \
    --options OptionName=TIMEZONE_FILE_AUTOUPGRADE \
    --apply-immediately
```

Mettre à niveau la version de votre moteur de base de données sans mettre à jour le fichier sur le fuseau horaire

Dans ce scénario, votre base de données utiliseDSTv33, mais elle DSTv34 est disponible sur votre système de fichiers d'instance de base de données. Vous souhaitez mettre à niveau votre moteur de base de données vers la version 19.0.0.0.ru-05.10.rur-05.r1, qui inclut le fichier de fuseau horaire, mais le conserve. DSTv35 DSTv33 Vous pouvez être appelé choisir cette politique pour les raisons suivantes :

- Vos données n'utilisent pas le type de données `TIMESTAMP WITH TIME ZONE`.
- Vos données utilisent le type de données `TIMESTAMP WITH TIME ZONE`, mais vos données ne sont pas affectées par les changements de fuseau horaire.
- Vous souhaitez reporter la mise à jour du fichier sur le fuseau horaire, car vous ne pouvez pas tolérer de temps d'arrêt supplémentaire.

Votre stratégie dépend de si les possibilités suivantes sont vraies (true) :

- Votre instance de base de données n'est pas associée à un groupe d'options comprenant `TIMEZONE_FILE_AUTOUPGRADE`. Dans votre `modify-db-instance` commande, ne spécifiez pas de nouveau groupe d'options afin de RDS ne pas mettre à jour votre fichier de fuseau horaire.
- Votre instance de base de données est actuellement associée à un groupe d'options qui inclut`TIMEZONE_FILE_AUTOUPGRADE`. À l'aide d'une seule `modify-db-instance` commande, associez votre instance de base de données à un groupe d'options qui n'inclut pas `TIMEZONE_FILE_AUTOUPGRADE` et mettez à niveau votre moteur de base de données vers 19.0.0.0.ru-10.rur-10.r1.

Temps d'arrêt pendant la mise à jour du fichier sur le fuseau horaire

Lors de la RDS mise à jour de votre fichier de fuseau horaire, les données existantes utilisées `TIMESTAMP WITH TIME ZONE` peuvent changer. Dans ce cas, votre principale considération est les temps d'arrêt.

> **Warning**
>
> Si vous ajoutez l'option `TIMEZONE_FILE_AUTOUPGRADE`, la mise à niveau de votre moteur peut avoir un temps d'arrêt prolongé. La mise à jour des données de fuseau horaire d'une base de données volumineuse peut prendre des heures, voire des jours.

La durée de mise à jour du fichier sur le fuseau horaire dépend entre autres des facteurs suivants :

- La quantité de données `TIMESTAMP WITH TIME ZONE` dans votre base de données
- La configuration de l'instance de base de données
- La classe d'instance de base de données
- La configuration du stockage
- La configuration de la base de données
- Les réglages des paramètres de base de données

Des temps d'arrêt supplémentaires peuvent survenir lorsque vous effectuez les opérations suivantes :

- Ajout de l'option au groupe d'options lorsque l'instance de base de données utilise un fichier sur le fuseau horaire obsolète
- Mise à niveau du moteur de base de données Oracle lorsque la nouvelle version du moteur contient une nouvelle version du fichier sur le fuseau horaire

> **Note**
>
> Pendant la mise à jour du fichier de fuseau horaire, RDS pour les appels Oracle`PURGE DBA_RECYCLEBIN`.

Préparation de la mise à jour du fichier sur le fuseau horaire

Une mise à niveau de fichier sur le fuseau horaire comporte deux phases distinctes : la préparation et la mise à niveau. Même si ce n'est pas nécessaire, nous vous recommandons vivement de procéder à l'étape de préparation. Au cours de cette étape, vous découvrirez quelles données seront affectées par l'exécution de la SQL procédure `DBMS_DST.FIND_AFFECTED_TABLES` PL/. Pour plus

d'informations sur la fenêtre de préparation, consultez Mise à niveau du fichier sur le fuseau horaire et de l'horodatage avec les données de fuseau horaire dans la documentation Oracle Database.

Pour préparer la mise à jour du fichier sur le fuseau horaire

1. Connectez-vous à votre base de données Oracle à l'aide d'un SQL client.
2. Déterminez la version actuelle du fichier sur le fuseau horaire utilisé.

```
SELECT * FROM V$TIMEZONE_FILE;
```

3. Déterminez la dernière version de fichier sur le fuseau horaire disponible sur votre instance de base de données.

```
SELECT DBMS_DST.GET_LATEST_TIMEZONE_VERSION FROM DUAL;
```

4. Déterminez la taille totale des tables qui ont des colonnes de type `TIMESTAMP WITH LOCAL TIME ZONE` ou `TIMESTAMP WITH TIME ZONE`.

```
SELECT SUM(BYTES)/1024/1024/1024 "Total_size_w_TSTZ_columns_GB"
FROM   DBA_SEGMENTS
WHERE  SEGMENT_TYPE LIKE 'TABLE%'
AND    (OWNER, SEGMENT_NAME) IN
         (SELECT OWNER, TABLE_NAME
          FROM   DBA_TAB_COLUMNS
          WHERE  DATA_TYPE LIKE 'TIMESTAMP%TIME ZONE');
```

5. Déterminez les noms et les tailles des segments qui ont des colonnes de type `TIMESTAMP WITH LOCAL TIME ZONE` ou `TIMESTAMP WITH TIME ZONE`.

```
SELECT OWNER, SEGMENT_NAME, SUM(BYTES)/1024/1024/1024
 "SEGMENT_SIZE_W_TSTZ_COLUMNS_GB"
FROM   DBA_SEGMENTS
WHERE  SEGMENT_TYPE LIKE 'TABLE%'
AND    (OWNER, SEGMENT_NAME) IN
         (SELECT OWNER, TABLE_NAME
          FROM   DBA_TAB_COLUMNS
          WHERE  DATA_TYPE LIKE 'TIMESTAMP%TIME ZONE')
GROUP BY OWNER, SEGMENT_NAME;
```

6. Exécutez l'étape de préparation.

- La procédure `DBMS_DST.CREATE_AFFECTED_TABLE` crée une table pour stocker toutes les données affectées. Vous passez le nom de cette table à la procédure `DBMS_DST.FIND_AFFECTED_TABLES`. Pour plus d'informations, consultez la section CREATE_ AFFECTED _ TABLE Procédure dans la documentation de la base de données Oracle.
- Cette procédure `CREATE_ERROR_TABLE` crée une table pour journaliser les erreurs. Pour plus d'informations, consultez la section CREATE_ ERROR _ TABLE Procédure dans la documentation de la base de données Oracle.

L'exemple suivant crée les tables de données et d'erreurs affectées, et recherche toutes les tables affectées.

```
EXEC DBMS_DST.CREATE_ERROR_TABLE('my_error_table')
EXEC DBMS_DST.CREATE_AFFECTED_TABLE('my_affected_table')

EXEC DBMS_DST.BEGIN_PREPARE(new_version);
EXEC DBMS_DST.FIND_AFFECTED_TABLES('my_affected_table', TRUE, 'my_error_table');
EXEC DBMS_DST.END_PREPARE;

SELECT * FROM my_affected_table;
SELECT * FROM my_error_table;
```

7. Interrogez les tables affectées et les tables d'erreurs.

```
SELECT * FROM my_affected_table;
SELECT * FROM my_error_table;
```

Ajout de l'option de mise à niveau automatique du fichier sur le fuseau horaire

Lorsque vous ajoutez l'option à un groupe d'options, le groupe d'options se trouve dans l'un des états suivants :

- Un groupe d'options existant est actuellement attaché à au moins une instance de base de données. Lorsque vous ajoutez l'option, toutes les instances de base de données qui utilisent ce groupe d'options redémarrent automatiquement. Cela provoque une panne brève.

- Aucun groupe d'options existant n'est attaché à une instance de base de données. Vous prévoyez d'ajouter l'option, puis d'associer le groupe d'options existant à des instances de base de données existantes ou à une nouvelle instance de base de données.
- Vous créez un nouveau groupe d'options et ajoutez l'option. Vous prévoyez d'associer le nouveau groupe d'options à des instances de base de données existantes ou à une nouvelle instance de base de données.

Console

Pour ajouter l'option de mise à niveau automatique du fichier sur le fuseau horaire à une instance de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Groupes d'options.
3. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Engine (Moteur), choisissez l'édition de base de données Oracle de votre instance de base de données.
   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
4. Cochez la case pour le groupe d'options que vous souhaitez modifier, puis choisissez Ajouter une option.
5. Dans la fenêtre Ajouter une option, procédez comme suit :

   a. Choisissez TIMEZONE_ FILE _ AUTOUPGRADE.
   b. Pour activer l'option sur toutes les instances de base de données associées dès que vous l'ajoutez, pour Apply Immediately (Appliquer immédiatement), choisissez Oui. Si vous choisissez Non (valeur par défaut), l'option est activée pour chaque instance de base de données associée pendant sa fenêtre de maintenance suivante.
6. Lorsque les paramètres vous conviennent, choisissez Ajouter une option.

AWS CLI

L'exemple suivant utilise la commande AWS CLI add-option-to-option-group pour ajouter l'`TIMEZONE_FILE_AUTOUPGRADE`option à un groupe d'options appelé`myoptiongroup`.

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name "myoptiongroup" \
    --options "OptionName=TIMEZONE_FILE_AUTOUPGRADE" \
    --apply-immediately
```

Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name "myoptiongroup" ^
    --options "OptionName=TIMEZONE_FILE_AUTOUPGRADE" ^
    --apply-immediately
```

Vérification de vos données après la mise à jour du fichier sur le fuseau horaire

Nous vous recommandons de vérifier vos données après avoir mis à jour le fichier sur le fuseau horaire. Au cours de l'étape de préparation, RDS for Oracle crée automatiquement les tables suivantes :

- `rdsadmin.rds_dst_affected_tables` – Répertorie les tables qui contiennent des données affectées par la mise à jour
- `rdsadmin.rds_dst_error_table` – Répertorie les erreurs générées pendant la mise à jour

Ces tables sont indépendantes des tables que vous créez dans la fenêtre de préparation. Pour afficher les résultats de la mise à jour, interrogez les tables comme suit.

```
SELECT * FROM rdsadmin.rds_dst_affected_tables;
SELECT * FROM rdsadmin.rds_dst_error_table;
```

Pour plus d'informations sur le schéma des données concernées et des tables d'erreurs, voir FIND_ AFFECTED _ TABLES Procedure dans la documentation Oracle.

Oracle Transparent Data Encryption

Amazon RDS prend en charge Oracle Transparent Data Encryption (TDE), une fonctionnalité de l'option Oracle Advanced Security disponible dans Oracle Enterprise Edition. Cette fonction chiffre automatiquement les données avant qu'elles ne soient écrites dans le stockage et déchiffre automatiquement les données lorsqu'elles sont lues depuis le stockage. Cette option n'est prise en charge que pour le modèle Bring Your Own License (BYOL).

TDEest utile dans les scénarios où vous devez chiffrer des données sensibles au cas où des fichiers de données et des sauvegardes seraient obtenus par un tiers. TDEest également utile lorsque vous devez respecter les réglementations relatives à la sécurité.

Une explication détaillée de TDE la base de données Oracle dépasse le cadre de ce guide. Pour plus d'informations, consultez les ressources de base de données Oracle suivantes :

- Présentation du chiffrement transparent des données dans la documentation de la base de données Oracle
- Sécurité avancée d'Oracle dans la documentation d'Oracle Database
- Sécurité avancée Oracle : meilleures pratiques de chiffrement transparent des données, livre blanc d'Oracle

Pour plus d'informations sur l'utilisation de TDE with RDS for Oracle, consultez les blogs suivants :

- Options de chiffrement des bases de données Oracle sur Amazon RDS
- Migrez une instance de base de données TDE Amazon RDS pour Oracle compatible avec plusieurs comptes avec un temps d'arrêt réduit grâce à AWS DMS

TDEmodes de chiffrement

Oracle Transparent Data Encryption prend en charge deux modes de chiffrement : le chiffrement des TDE tablespaces et le chiffrement des TDE colonnes. TDEle chiffrement des tablespaces est utilisé pour chiffrer des tables d'applications entières. TDEle chiffrement par colonne est utilisé pour chiffrer des éléments de données individuels contenant des données sensibles. Vous pouvez également appliquer une solution de chiffrement hybride qui utilise à la fois le chiffrement des TDE tablespaces et des colonnes.

> **Note**
>
> Amazon RDS gère le portefeuille Oracle et la clé TDE principale de l'instance de base de données. Vous n'avez pas besoin de définir la clé de chiffrement à l'aide de la commande `ALTER SYSTEM set encryption key`.

Après avoir activé TDE cette option, vous pouvez vérifier l'état du portefeuille Oracle à l'aide de la commande suivante :

```
SELECT * FROM v$encryption_wallet;
```

Pour créer un espace de table chiffré, utilisez la commande suivante :

```
CREATE TABLESPACE encrypt_ts ENCRYPTION DEFAULT STORAGE (ENCRYPT);
```

Pour spécifier l'algorithme de chiffrement, utilisez la commande suivante :

```
CREATE TABLESPACE encrypt_ts ENCRYPTION USING 'AES256' DEFAULT STORAGE (ENCRYPT);
```

Les instructions précédentes pour chiffrer un tablespace sont les mêmes que celles que vous utiliseriez sur une base de données Oracle locale.

Restrictions relatives à l'TDEoption

L'TDEoption est permanente et persistante. Une fois que vous avez associé votre instance de base de données à un groupe d'options dont l'TDEoption est activée, vous ne pouvez pas effectuer les actions suivantes :

- Désactivez l'TDEoption dans le groupe d'options actuellement associé.
- Associez votre instance de base de données à un autre groupe d'options qui n'inclut pas l'TDEoption.
- Partagez un instantané de base de données qui utilise TDE cette option. Pour plus d'informations sur le partage d'instantanés de base de données, consultez Partage d'un instantané de base de données pour Amazon RDS.

Pour plus d'informations sur les options persistantes et permanentes, consultezOptions persistantes et permanentes.

Déterminer si votre instance de base de données utilise TDE

Vous souhaiterez peut-être déterminer si votre instance de base de données est associée à un groupe d'options dans lequel l'TDEoption est activée. Pour afficher le groupe d'options auquel une instance de base de données est associée, utilisez la RDS console, la describe-db-instance AWS CLI commande ou l'APIopération escribeDBInstancesD.

Ajouter l'TDEoption

Pour ajouter l'TDEoption à votre instance de base de données, procédez comme suit :

1. (Recommandé) Prenez un instantané de votre instance de base de données.
2. Effectuez l'une des tâches suivantes :
   - Créez un nouveau groupe d'options à partir de zéro. Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
   - Copiez un groupe d'options existant à l'aide du AWS CLI ouAPI. Pour de plus amples informations, veuillez consulter Copie d'un groupe d'options.
   - Réutilisez un groupe d'options existant autre que celui par défaut. La meilleure pratique consiste à utiliser un groupe d'options qui n'est actuellement associé à aucune instance de base de données ni à aucun instantané.
3. Ajoutez la nouvelle option au groupe d'options de l'étape précédente.
4. Si le groupe d'options actuellement associé à votre instance de base de données possède des options activées, ajoutez ces options à votre nouveau groupe d'options. Cette stratégie empêche la désinstallation des options existantes lors de l'activation de la nouvelle option.
5. Ajoutez le nouveau groupe d'options à votre instance de base de données.

Console

Pour ajouter l'TDEoption à un groupe d'options et l'associer à votre instance de base de données

1. Dans la RDS console, choisissez Groupes d'options.
2. Choisissez le nom du groupe d'options auquel vous souhaitez ajouter l'option.
3. Sélectionnez Ajouter une option.
4. Dans Nom de l'option, choisissez TDE, puis configurez les paramètres de l'option.
5. Sélectionnez Ajouter une option.

> **Important**
>
> Si vous ajoutez l'TDEoption à un groupe d'options actuellement attaché à une ou plusieurs instances de base de données, une brève interruption se produit alors que toutes les instances de base de données sont automatiquement redémarrées.

Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.

6. Associez le groupe d'options à une instance de base de données nouvelle ou existante :

   - Pour une nouvelle instance de base de données, appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
   - Pour une instance de base de données existante, appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. L'instance de base de données ne redémarre pas dans le cadre de cette opération. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

AWS CLI

Dans l'exemple suivant, vous utilisez la commande AWS CLI add-option-to-option-group pour ajouter l'TDEoption à un groupe d'options appelé`myoptiongroup`. Pour plus d'informations, consultez Getting started : Flink 1.13.2.

Dans Linux, macOS, ou Unix:

```
aws rds add-option-to-option-group \
    --option-group-name "myoptiongroup" \
    --options "OptionName=TDE" \
    --apply-immediately
```

Dans Windows:

```
aws rds add-option-to-option-group ^
    --option-group-name "myoptiongroup" ^
    --options "OptionName=TDE" ^
    --apply-immediately
```

Copier vos données vers une instance de base de données qui n'inclut pas l'TDEoption

Vous ne pouvez pas supprimer l'TDEoption d'une instance de base de données ou l'associer à un groupe d'options qui ne l'TDEinclut pas. Pour migrer vos données vers une instance qui n'inclut pas TDE cette option, procédez comme suit :

1. Déchiffrez les données de votre instance de base de données.
2. Copiez les données sur une nouvelle instance de base de données qui n'est pas associée à un groupe d'options `TDE` activé.
3. Supprimez votre instance de base de données d'origine.

Vous pouvez utiliser le même nom pour la nouvelle instance que l'instance de base de données précédente.

Considérations relatives à l'utilisation TDE avec Oracle Data Pump

Vous pouvez utiliser Oracle Data Pump pour importer ou exporter des fichiers de vidage chiffrés. Amazon RDS prend en charge le mode de cryptage des mots de passe `(ENCRYPTION_MODE=PASSWORD)` pour Oracle Data Pump. Amazon RDS ne prend pas en charge le mode de chiffrement transparent `(ENCRYPTION_MODE=TRANSPARENT)` pour Oracle Data Pump. Pour de plus amples informations, veuillez consulter Importation à l'aide d'Oracle Data Pump.

Oracle UTL _ MAIL

Amazon RDS prend en charge Oracle UTL _ MAIL grâce à l'utilisation de l'MAILoption UTL _ et de SMTP serveurs. Vous pouvez envoyer un e-mail directement depuis votre base de données en utilisant le MAIL package UTL _. Amazon RDS prend en charge UTL _ MAIL pour les versions suivantes d'Oracle :

- Oracle Database 21c (21.0.0.0), toutes les versions
- Oracle Database 19c (19.0.0.0), toutes les versions

Voici quelques limites à l'utilisation de UTL _ MAIL :

- UTL_ MAIL ne prend pas en charge le protocole Transport Layer Security (TLS) et les e-mails ne sont donc pas chiffrés.

  Pour vous connecter en toute sécurité à SSL des TLS ressources distantes en créant et en téléchargeant des portefeuilles Oracle personnalisés, suivez les instructions figurant dans. Configuration de l'accès UTL_HTTP à l'aide de certificats et d'un portefeuille Oracle

  Les certificats spécifiques requis pour votre portefeuille varient par service. Pour les AWS services, ceux-ci se trouvent généralement dans le référentiel des services de confiance Amazon.
- UTL_ MAIL ne prend pas en charge l'authentification auprès SMTP des serveurs.
- Vous ne pouvez envoyer qu'une seule pièce jointe dans un e-mail.
- Vous ne pouvez pas envoyer de pièces jointes de plus de 32 Ko.
- Vous ne pouvez utiliser que ASCII des codages de caractères en code d'échange décimal codé binaire étendu (EBCDIC).
- SMTPle port (25) est limité en fonction des politiques du propriétaire de l'interface elastic network.

Lorsque vous activez UTL _MAIL, seul l'utilisateur principal de votre instance de base de données dispose du privilège d'exécution. Si nécessaire, l'utilisateur principal peut accorder le privilège d'exécution à d'autres utilisateurs afin qu'ils puissent utiliser UTL _MAIL.

> **Important**
>
> Nous vous recommandons d'activer la fonctionnalité d'audit intégrée d'Oracle pour suivre l'utilisation des MAIL procédures UTL _.

Conditions requises pour Oracle _ UTL MAIL

Les conditions requises pour utiliser Oracle UTL _ MAIL sont les suivantes :

- Un ou plusieurs SMTP serveurs, ainsi que les adresses IP ou les noms de serveurs de noms de domaine (DNS) publics ou privés correspondants. Pour plus d'informations sur DNS les noms privés résolus via un DNS serveur personnalisé, consultezConfiguration d'un DNS serveur personnalisé.

Ajout de l'MAILoption Oracle UTL _

Le processus général pour ajouter l'MAILoption Oracle UTL _ à une instance de base de données est le suivant :

1. Créer un groupe d'options ou copier ou modifier un groupe existant.
2. Ajoutez l'option au groupe d'options.
3. Associez le groupe d'options à l'instance de base de données.

Après avoir ajouté l'MAILoption UTL _, dès que le groupe d'options est actif, UTL _ MAIL est actif.

Pour ajouter l'MAILoption UTL _ à une instance de base de données

1. Déterminez le groupe d'options que vous voulez utiliser. Vous pouvez créer un groupe d'options ou utiliser un groupe d'options existant. Si vous souhaitez utiliser un groupe d'options existant, passez à l'étape suivante. Sinon, créez un groupe d'options DB personnalisé avec les paramètres suivants :

   a. Pour Moteur, sélectionnez l'édition d'Oracle que vous voulez utiliser.

   b. Pour Version majeure du moteur, choisissez la version de votre instance de base de données.

   Pour de plus amples informations, veuillez consulter Création d'un groupe d'options.
2. Ajoutez l'MAILoption UTL_ au groupe d'options. Pour plus d'informations sur l'ajout d'options, consultez Ajout d'une option à un groupe d'options.
3. Appliquez le groupe d'options à une instance de base de données nouvelle ou existante:

- Pour une nouvelle instance de base de données, vous appliquez le groupe d'options lorsque vous lancez l'instance. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.
- Pour une instance de base de données existante, vous appliquez le groupe d'options en modifiant l'instance et en attachant le nouveau groupe d'options. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Utilisation d'Oracle UTL _ MAIL

Après avoir activé l'MAILoption UTL _, vous devez configurer le SMTP serveur avant de pouvoir commencer à l'utiliser.

Vous configurez le SMTP serveur en attribuant au SERVER paramètre SMTP OUT _ _ une adresse IP ou un DNS nom public valide. Pour le SERVER paramètre SMTP _ OUT _, vous pouvez spécifier une liste séparée par des virgules des adresses de plusieurs serveurs. Si le premier serveur n'est pas disponible, UTL _ MAIL essaie le serveur suivant, et ainsi de suite.

Vous pouvez définir le SMTP _ OUT _ par défaut SERVER pour une instance de base de données en utilisant un groupe de paramètres de base de données. Vous pouvez définir le SERVER paramètre SMTP _ OUT _ pour une session en exécutant le code suivant sur votre base de données sur votre instance de base de données.

```
ALTER SESSION SET smtp_out_server = mailserver.domain.com:25;
```

Une fois que l'MAILoption UTL _ est activée et que votre SMTP _ OUT _ SERVER est configuré, vous pouvez envoyer un e-mail à l'aide de la SEND procédure. Pour plus d'informations, consultez UTL_MAIL dans la documentation Oracle.

Supprimer l'MAILoption Oracle UTL _

Vous pouvez supprimer Oracle UTL _ d'une instance MAIL de base de données.

Pour supprimer UTL _ MAIL d'une instance de base de données, effectuez l'une des opérations suivantes :

- Pour supprimer UTL _ MAIL de plusieurs instances de base de données, supprimez l'MAILoption UTL _ du groupe d'options auquel elles appartiennent. Ce changement affecte toutes les instances

de bases de données qui utilisent le groupe d'options. Pour de plus amples informations, veuillez consulter Suppression d'une option d'un groupe d'options.

- Pour supprimer UTL _ d'une seule instance MAIL de base de données, modifiez l'instance de base de données et spécifiez un autre groupe d'options qui n'inclut pas l'MAILoption UTL _. Vous pouvez spécifier le groupe d'options (vide) par défaut, ou un groupe d'options personnalisées différent. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Résolution des problèmes

Vous pouvez rencontrer les problèmes suivants lorsque vous utilisez UTL _ MAIL avec AmazonRDS.

- Limitation. SMTPle port (25) est limité en fonction des politiques du propriétaire de l'interface elastic network. Si vous parvenez à envoyer un e-mail en utilisant UTL _ MAIL et que le message d'erreur s'affiche`ORA-29278: SMTP transient error: 421 Service not available`, il est possible que vous soyez limité. Si vous rencontrez une limitation dans le cadre de la remise d'e-mails, nous vous recommandons d'implémenter un algorithme d'interruption. Pour plus d'informations sur les algorithmes de backoff, consultez Nouvelles tentatives après erreur et backoff exponentiel dans AWS et Comment traiter une erreur « throttling – Maximum sending rate exceeded ».

  Vous pouvez demander que cette limitation soit supprimée. Pour plus d'informations, consultez Comment supprimer le régulateur sur le port 25 de mon EC2 instance ? .

Oracle XML DB

Oracle XML DB ajoute la prise en charge XML native à votre instance de base de données. Avec XML DB, vous pouvez stocker et récupérer des données XML et relationnelles structurées ou non structurées. Le serveur de protocole XML DB n'est pas pris en charge sur RDS pour Oracle.

La base de données XML est préinstallée sur Oracle Database 12c et versions ultérieures. Ainsi, vous n'avez pas besoin d'utiliser un groupe d'options pour installer explicitement la base de données XML en tant que fonctionnalité supplémentaire.

Pour savoir comment configurer et utiliser une base de données XML, consultez le guide du développeur de base de données Oracle XML dans la documentation de la base de données Oracle.

Mise à niveau du moteur de base de données RDS for Oracle

Quand Amazon RDS prend en charge une nouvelle version d'Oracle Database, vous pouvez mettre à niveau vos instances de base de données vers cette nouvelle version. Pour savoir quelles versions d'Oracle sont disponibles sur Amazon RDS, consultez Amazon RDS for Oracle Release Notes (Notes de mise à jour de Amazon RDS for Oracle).

> **Important**
>
> Les bases de données RDS for Oracle 11g, 12c et 18c ne sont plus prises en charge. Si vous conservez des instantanés de base de données Oracle 11g, 12c ou 18c, vous pouvez les mettre à niveau vers une version ultérieure. Pour plus d'informations, voir Mise à niveau d'un instantané de base de données Oracle.

Rubriques

- Présentation des mises à niveau du moteur RDS for Oracle
- Mises à niveau des versions majeures d'Oracle
- Mises à niveau des versions mineures d'Oracle
- Considérations relatives aux mises à niveau des bases de données
- Test d'une mise à niveau de base de données Oracle
- Mise à niveau de la version d'une instance de base de données RDS pour Oracle
- Mise à niveau d'un instantané de base de données Oracle

Présentation des mises à niveau du moteur RDS for Oracle

Avant de mettre à niveau votre instance de base de données RDS for Oracle, familiarisez-vous avec les concepts suivants.

Rubriques

- Mises à niveau des versions majeures et mineures
- Support technique et mises à niveau obligatoires pour RDS pour Oracle
- Gestion des versions du moteur Oracle
- Instantanés automatiques lors de mises à niveau du moteur
- Mises à niveau Oracle dans un déploiement multi-AZ

- Mises à niveau Oracle des réplicas en lecture

Mises à niveau des versions majeures et mineures

Les versions majeures sont des versions majeures d'Oracle Database publiées tous les 1 à 2 ans. Oracle Database 19c et Oracle Database 21c sont des versions majeures.

Chaque trimestre, RDS for Oracle publie de nouvelles versions mineures du moteur pour chacun des principaux moteurs pris en charge. Une version du moteur Release Update (RU) intègre les corrections de bogues d'Oracle en incluant les correctifs RU pour le trimestre spécifié. Par exemple, 21.0.0.0.ru-2024-10.rur-2024-10.r1 est une version mineure d'Oracle Database 21c qui intègre le RU d'octobre 2024.

Une version du moteur Spatial Patch Bundle (SPB) contient des correctifs RU et des correctifs spécifiques à Oracle Spatial. Par exemple, 19.0.0.0.ru-2025-01.spb-1.r1 est une version mineure du moteur qui contient les correctifs RU de la version 19.0.0.0.ru-2025-01.rur-2025-01.rur-2025-01.r1 ainsi que des correctifs spatiaux. En général, RDS pour Oracle est publié 2 SPBs à 3 semaines après le RU correspondant. Pour une explication des différences entre RUs et SPBs, voirMises à jour des versions (RUs) et ensembles de correctifs spatiaux () SPBs. Pour plus d'informations sur le RUs support SPBs, consultez les notes de version d'Amazon Relational Database Service (Amazon RDS) pour Oracle.

RDS pour Oracle prend en charge les mises à niveau suivantes vers une instance de base de données.

Type de mise à niveau	Compatibilité des applications	Méthodes de mise à niveau	Exemple de chemin de mise à niveau
Version majeure	La mise à niveau d'une version majeure peut introduire des modifications non compatibles avec les applications existantes.	Manuelles uniquement	Depuis Oracle Database 19c vers Oracle Database 21c
Version mineure	Une mise à niveau de version mineure contient uniquement des modifications rétrocompatibles avec les applications existantes.	Automatiques ou manuelles	Du 21.0.0.0.ru-2023-07.rur-05.07.r1 au 21.0.0.0.ru-2023-10.rur-2_10.r1

> **Important**
>
> La mise à niveau de votre moteur de base de données provoque une panne. La durée de cette dernière dépend de la version du moteur et de la taille de l'instance de base de données.
> Veillez à tester soigneusement toute mise à niveau pour vérifier que vos applications fonctionnent correctement avant d'appliquer la mise à niveau à vos bases de données de production. Pour de plus amples informations, veuillez consulter Test d'une mise à niveau de base de données Oracle.

Support technique et mises à niveau obligatoires pour RDS pour Oracle

Les versions de base de données de RDS pour Oracle ont des dates de support prévues. Lorsqu'une version majeure ou mineure d'un moteur de base de données RDS pour Oracle approche de sa end-of-support date limite, RDS commence les mises à niveau obligatoires, également appelées mises à niveau forcées. RDS publie les informations suivantes :

- Une recommandation vous invitant à commencer à mettre à niveau manuellement les instances des versions obsolètes vers les versions prises en charge
- Date après laquelle vous ne pourrez plus créer d'instances sur les versions non prises en charge
- Date à laquelle RDS commence à mettre automatiquement à niveau vos instances vers les versions prises en charge pendant les fenêtres de maintenance
- Date à laquelle RDS commence à mettre automatiquement à niveau vos instances vers les versions prises en charge en dehors des fenêtres de maintenance

> **Important**
>
> Les mises à niveau forcées peuvent avoir des conséquences inattendues sur les AWS CloudFormation cumuls. Si vous comptez sur RDS pour mettre à niveau automatiquement vos instances de base de données, vous pouvez rencontrer des problèmes avec AWS CloudFormation.

Cette section contient les rubriques suivantes :

Rubriques

- Support technique pour les versions majeures de RDS pour Oracle
- Support des versions mineures de RDS pour Oracle

Support technique pour les versions majeures de RDS pour Oracle

Les versions majeures de RDS for Oracle restent disponibles au moins jusqu'à la date de fin de prise en charge de la version correspondante de la base de données Oracle. Vous pouvez utiliser les dates suivantes pour planifier vos cycles de test et de mise à niveau. Ces dates représentent la première date à laquelle une mise à niveau vers une version plus récente pourrait être nécessaire. Si Amazon étend le support d'une version de RDS for Oracle au-delà de la date initialement prévue, ce tableau sera mis à jour pour refléter la nouvelle date.

> **Note**
>
> Vous pouvez consulter les versions principales de vos bases de données Oracle en exécutant la AWS CLI commande describe-db-major-engine-versions ou en utilisant l'opération d'API DBMajorEngineVersionsDescribe RDS.

Version majeure de la version d'Oracle Database	Date prévue pour la mise à niveau vers une version plus récente
Oracle Database 19c	31 décembre 2029 avec BYOL Premier Support (exonération des frais pour le support étendu) 31 décembre 2032 avec un support étendu BYOL (coût supplémentaire) ou un contrat de licence illimité 31 décembre 2029 avec licence incluse (LI)
Oracle Database 21c	31 juillet 2027 (non disponible pour le Support étendu)

RDS vous avertit au moins 12 mois avant que vous ne deviez effectuer une mise à niveau vers une version majeure plus récente. La notification décrit le processus de mise à niveau, y compris le calendrier des étapes importantes, l'effet sur vos instances de base de données et les actions recommandées. Nous vous recommandons de tester minutieusement vos applications avec les

nouvelles versions de RDS pour Oracle avant de mettre à niveau votre base de données vers une version majeure.

Après cette période de notification préalable, une mise à niveau automatique vers la version majeure suivante peut être appliquée à toute instance de base de données RDS for Oracle exécutant encore l'ancienne version. Dans ce cas, la mise à niveau démarre pendant les fenêtres de maintenance planifiées.

Pour plus d'informations, consultez le calendrier de publication des versions actuelles de la base de données dans My Oracle Support.

Support des versions mineures de RDS pour Oracle

Les versions mineures suivantes d'Oracle Database 19c approchent de leur end-of-support date d'expiration :

- 19.0.0.0.ru-2019-07.rur-2019-07.r1
- 19.0.0.0.ru-2019-10.rur-2019-10.r1
- 19.0.0.0.ru-2020-01.rur-2020-01.r1

Les versions mineures suivantes d'Oracle Database 21c approchent de leur end-of-support date d'expiration :

- 21.0.0.0.ru-2022-01.rur-2022-01.r1
- 21.0.0.0.ru-2022-04.rur-2022-04.r1
- 21.0.0.0.ru-2022-07.rur-2022-07.r1
- 21.0.0.0.ru-2022-10.rur-2022-10.r1
- 21.0.0.0.ru-2023-01.rur-2023-01.r1
- 21.0.0.0.ru-2023-01.rur-2023-01.r2

Le tableau suivant indique le calendrier des mises à niveau obligatoires des versions mineures pour les versions obsolètes précédentes.

Date	Action pour les versions obsolètes
25/02/07	Vous ne pouvez plus créer d'instances de base de données exécutant des versions obsolètes. Mettez à niveau vos instances de base de données

Date	Action pour les versions obsolètes
	manuellement vers la dernière mise à jour (RU) ou attendez que RDS les mette à niveau automatiquement.
08/01/25 — 15/09/25	RDS Force met à niveau les instances de base de données dans la fenêtre de maintenance planifiée vers la dernière RU. RDS met également à niveau toutes les instances restaurées à partir d'instantanés des versions obsolètes.
15/09/25	RDS commence à forcer la mise à niveau des instances de base de données vers le dernier RU quelle que soit la fenêtre de maintenance planifiée. Les mises à niveau forcées se poursuivent jusqu'à ce que toutes les instances exécutent des versions mineures prises en charge.

Si vous ne souhaitez pas que RDS mette automatiquement à niveau vos instances exécutant des versions obsolètes, procédez comme suit :

- Mettez à niveau manuellement les versions mineures de votre base de données Oracle 19c et 21c vers la dernière version d'Oracle Database ou vers une unité d'exploitation dont les mises à niveau obligatoires ne sont pas prévues. Par exemple, vous pouvez mettre à niveau 21.0.0.0.ru-2023-01.rur-2023-01.r2 vers 21.0.0.0.ru-2023-04.rur-2023-04.r2, ou mettre à niveau 19.0.0.0.ru-2019-07.rur-2019-07.r1 vers 19.0.0.0.ru-2025-01.rur-2025-01.r2.
- Mettez à niveau les instantanés de base de données qui utilisent des versions mineures obsolètes, puis restaurez les instantanés.

Pour plus d'informations sur les versions mineures, consultez les notes de publication d'Amazon Relational Database Service (Amazon RDS) pour Oracle.

Gestion des versions du moteur Oracle

La gestion des versions du moteur de base de données vous permet de contrôler quand et comment le moteur de base de données est corrigé et mis à niveau. Vous bénéficiez de la flexibilité nécessaire pour maintenir la compatibilité avec les versions de correctifs du moteur de base de données. Vous pouvez également tester les nouvelles versions correctives de RDS for Oracle pour vous assurer qu'elles fonctionnent avec votre application avant de les déployer en production. En outre, vous effectuez la mise à niveau des versions selon vos critères et vos calendriers.

> **Note**
>
> Amazon RDS agrège régulièrement des correctifs de base de données Oracle officiels en utilisant une version de moteur de base de données propre à Amazon RDS. Pour voir la liste des correctifs Oracle contenus dans une version du moteur spécifique à Amazon RDS for Oracle, consultez Amazon RDS for Oracle Release Notes (Notes de mise à jour de Amazon RDS for Oracle).

Instantanés automatiques lors de mises à niveau du moteur

Lors de mises à niveau d'une instance de base de données Oracle, les instantanés offrent une protection contre les problèmes de mise à niveau. Si la période de rétention des sauvegardes de votre instance de base de données est supérieure à 0, Amazon RDS prend les instantanés de base de données suivants au cours de la mise à niveau :

1. Un instantané de l'instance de base de données avant que toute modification de mise à niveau soit apportée. Si la mise à niveau échoue, vous pouvez restaurer cet instantané pour créer une instance de base de données exécutant l'ancienne version.
2. Un instantané de l'instance de base de données une fois la mise à niveau terminée.

> **Note**
>
> Pour modifier la période de rétention des sauvegardes, consultez Modification d'une RDS instance de base de données Amazon.

Après une mise à niveau, vous ne pouvez pas revenir à la version antérieure du moteur. Toutefois, vous pouvez créer une nouvelle instance de base de données Oracle en restaurant l'instantané antérieur à la mise à niveau.

Mises à niveau Oracle dans un déploiement multi-AZ

Si votre instance de base de données se trouve dans un déploiement Multi-AZ, Amazon RDS met à niveau les deux réplicas, principal et de secours. Si aucune mise à jour du système d'exploitation n'est requise, les mises à niveau principale et de secours se produisent simultanément. Les instances ne sont pas disponibles tant que la mise à niveau n'est pas terminée.

Si des mises à jour du système d'exploitation sont requises dans le cadre d'un déploiement multi-AZ, Amazon RDS applique les mises à jour lorsque vous demandez la mise à niveau de la base de données. Amazon RDS effectue les étapes suivantes :

1. Met à jour le système d'exploitation sur l'instance de base de données de secours actuelle.
2. Bascule entre l'instance de base de données principale et l'instance de base de données de secours.
3. Met à niveau la version de base de données sur la nouvelle instance de base de données principale, qui était auparavant l'instance de secours. La base de données principale n'est pas disponible pendant la mise à niveau.
4. Met à jour le système d'exploitation sur la nouvelle instance de base de données de secours, qui était auparavant l'instance de base de données principale.
5. Met à niveau la version de base de données sur la nouvelle instance de base de données de secours.
6. Fait basculer la nouvelle instance de base de données principale vers l'instance de base de données principale d'origine, et la nouvelle instance de base de données de secours revient à l'instance de base de données de secours d'origine. Ainsi, Amazon RDS rétablit la configuration de réplication dans son état d'origine.

Mises à niveau Oracle des réplicas en lecture

La version du moteur de base de données Oracle de l'instance de base de données source et tous ses réplicas en lecture doivent être identiques. Amazon RDS effectue la mise à niveau lors des étapes suivantes :

1. Mise à niveau de l'instance de base de données source. Les réplicas en lecture sont disponibles au cours de cette étape.
2. Mise à niveau des réplicas en lecture en parallèle, quelles que soient les fenêtres de maintenance des réplicas. La base de données source est disponible pendant cette étape.

Pour les mises à niveau de versions majeures des réplicas en lecture entre régions, Amazon RDS effectue des actions supplémentaires :

- Génération automatique d'un groupe d'options pour la version cible
- Copie de toutes les options et tous les paramètres d'option du groupe d'options d'origine vers le nouveau groupe d'options

- Association du réplica en lecture entre régions mis à niveau au nouveau groupe d'options

Mises à niveau des versions majeures d'Oracle

Pour effectuer une mise à niveau de version majeure, modifiez l'instance de base de données manuellement. Les mises à niveau de versions majeures ne sont pas effectuées automatiquement.

> **Important**
>
> Veillez à tester soigneusement toute mise à niveau pour vérifier que vos applications fonctionnent correctement avant d'appliquer la mise à niveau à vos bases de données de production. Pour plus d'informations, consultez Test d'une mise à niveau de base de données Oracle.

Rubriques

- Versions prises en charge pour les mises à niveau majeures
- Classes d'instance prises en charge pour les mises à niveau majeures
- Collecte de statistiques avant les mises à niveau majeures
- Autorisation d'installation des mises à niveau majeures

Versions prises en charge pour les mises à niveau majeures

Amazon RDS prend en charge les mises à niveau des versions majeures suivantes.

Version actuelle	Mise à niveau prise en charge
19.0.0.0 utilisant l'architecture CDB	21,0.0.0

Une mise à niveau de version majeure d'Oracle Database doit être mise à niveau vers une mise à jour de version publiée (RU) qui a été publiée le même mois ou plus tard. Le retour à une ancienne version majeure n'est pas pris en charge pour les versions Oracle Database.

Classes d'instance prises en charge pour les mises à niveau majeures

Il est possible que votre instance de base de données Oracle s'exécute sur une classe d'instance de base de données qui n'est pas prise en charge pour la version vers laquelle vous effectuez la mise à niveau. Dans ce cas, avant de procéder à la mise à niveau, vous devez migrer l'instance de base de données vers une classe prise en charge. Pour plus d'informations sur les classes d'instances de base de données prises en charge pour chaque version et édition de Amazon RDS for Oracle, consultez la section Classes d'instances de base de données .

Collecte de statistiques avant les mises à niveau majeures

Avant d'effectuer une mise à niveau de version majeure, Oracle vous recommande de recueillir des statistiques de l'optimiseur sur l'instance de base de données que vous mettez à jour. Cette action peut réduire les temps d'arrêt de l'instance de base de données lors de la mise à niveau.

Pour recueillir des statistiques de l'optimiseur, connectez-vous à l'instance de base de données en tant qu'utilisateur principal et exécutez la procédure `DBMS_STATS.GATHER_DICTIONARY_STATS`, comme dans l'exemple suivant.

```
EXEC DBMS_STATS.GATHER_DICTIONARY_STATS;
```

Pour plus d'informations, consultez la procédure GATHER_DICTIONARY_STATS dans la documentation Oracle.

Autorisation d'installation des mises à niveau majeures

Une mise à niveau majeure de la version du moteur peut être incompatible avec votre application. La mise à niveau est irréversible. Si vous spécifiez une version majeure pour le EngineVersion paramètre qui est différente de la version principale actuelle, vous devez autoriser les mises à niveau des versions majeures.

Si vous procédez à la mise à niveau d'une version majeure à l'aide de la commande CLI modify-db-instance, spécifiez `--allow-major-version-upgrade`. Dans la mesure où ce paramètre n'est pas persistant, vous devez spécifier `--allow-major-version-upgrade` chaque fois que vous effectuez une mise à niveau majeure. Ce paramètre n'a aucun impact sur les mises à niveau des versions mineures du moteur. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Si vous mettez à niveau une version majeure à l'aide de la console, vous n'avez pas besoin de choisir une option pour autoriser la mise à niveau. Au lieu de cela, la console affiche un avertissement indiquant que les mises à niveau majeures sont irréversibles.

Mises à niveau des versions mineures d'Oracle

Dans RDS pour Oracle, une mise à niveau de version mineure est une mise à jour d'une version majeure du moteur de base de données. Dans RDS, une version mineure du moteur est soit une Release Update (RU), soit un Spatial Patch Bundle (SPB). Par exemple, si votre instance de base de données exécute la version majeure d'Oracle Database 21c et la version mineure 21.0.0.0.ru-2024-10.rur-2024-10.r1, vous pouvez mettre à niveau votre moteur de base de données vers la version mineure 21.0.0.0.ru-2025-01.rur-2025-01.r1. RDS for Oracle ne prend pas en charge les rétrogradations de versions mineures.

Vous pouvez mettre à niveau manuellement ou automatiquement votre moteur de base de données vers une version mineure. Pour découvrir comment effectuer une mise à niveau manuelle, consultez Mise à niveau manuelle de la version du moteur. Pour découvrir comment configurer des mises à niveau automatiques, consultez Mise à niveau automatique de la version mineure du moteur. Que vous procédiez à une mise à niveau manuelle ou automatique, une mise à niveau de version mineure entraîne une interruption de service. Tenez compte de ce temps d'arrêt lorsque vous planifiez vos mises à niveau.

> **Important**
>
> Veillez à tester soigneusement toute mise à niveau pour vérifier que vos applications fonctionnent correctement avant d'appliquer la mise à niveau à vos bases de données de production. Pour de plus amples informations, veuillez consulter Test d'une mise à niveau de base de données Oracle.

Rubriques

- Mises à jour des versions (RUs) et ensembles de correctifs spatiaux () SPBs
- Activation des mises à niveau automatiques des versions mineures pour Oracle
- Notification des mises à niveau automatiques des versions mineures dans RDS pour Oracle
- Lorsque RDS planifie des mises à niveau automatiques de versions mineures dans RDS pour Oracle
- Gestion d'une mise à niveau automatique d'une version mineure dans RDS pour Oracle

Mises à jour des versions (RUs) et ensembles de correctifs spatiaux () SPBs

Dans RDS, une mise à jour (RU) est une version mineure trimestrielle du moteur qui inclut des correctifs de sécurité, des corrections de bogues et de nouvelles fonctionnalités pour Oracle Database. Un Spatial Patch Bundle (SPB) est une version du moteur RU qui inclut des correctifs conçus pour l'option Oracle Spatial. Par exemple, le SPB nommé 19.0.0.0.ru-2025-01.spb-1.r1 inclut tous les correctifs du RU 19.0.0.0.ru-2025-01.rur-2025-01.rur-2025-01.r1 ainsi que les correctifs spécifiques à Spatial. SPBs ne sont pris en charge que pour Oracle Database 19c.

Lorsque votre instance est configurée pour les mises à niveau automatiques de versions mineures RUs et qu'elle SPBs se trouve sur des chemins de mise à niveau distincts. Généralement, un SPB est publié 2 à 3 semaines après le RU correspondant. Le tableau suivant présente des exemples de versions mineures d'Oracle Database 19c.

Voie de mise à niveau standard de l'UE	chemin de mise à niveau du SPB
19.0.0.0.ru-2025-01.rur-2025-01.r1	19.0.0.0.ru-2025-01.spb-1.r1
19.0.0.0.ru-2025-04.rur-2025-04.r1	19.0.0.0.ru-2025-04.spb-1.r1
19.0.0.0.ru-2025-07.rur-2025-07.r1	19.0.0.0.ru-2025-07.spb-1.r1
19.0.0.0.ru-2025-10.rur-2025-10.r1	19.0.0.0.ru-2025-10.spb-1.r1

Si votre instance de base de données est configurée pour les mises à niveau automatiques, elle se trouve sur le chemin de mise à niveau correspondant à votre version actuelle. Par exemple, si votre instance de base de données exécute la version 19.0.0.0.ru-2025-01.rur-2025-01.r1, alors lorsque 19.0.0.0.ru-2025-04.rur-2025-04.r1 est publié, votre instance est automatiquement mise à niveau vers cette RU. De même, si votre instance de base de données exécute 19.0.0.0.ru-2025-01.spb-1.r1, alors lorsque 19.0.0.0.ru-2025-04.spb-1.r1 est publié, votre instance est automatiquement mise à niveau vers ce SPB. Une instance exécutant 19.0.0.0.ru-2025-01.rur-2025-01.r1, qui est un RU, ne sera pas automatiquement mise à niveau vers 19.0.0.0.ru-2025-04.spb-1.r1, qui est un SPB sur un chemin de mise à niveau distinct.

Vous pouvez mettre à niveau votre instance de base de données SPBs même si elle n'utilise pas Spatial, mais les correctifs Spatial ne s'appliquent qu'à Oracle Spatial. Vous pouvez effectuer une mise à niveau manuelle d'un RU vers un SPB avec la même version du moteur

ou une version supérieure. Par exemple, vous pouvez mettre à niveau votre instance de 19.0.0.0.ru-2025-01.rur-2025-01.r1 vers l'une des versions de moteur suivantes :

- 19.0.0.0.ru-2025-01.spb-1.r1
- 19.0.0.0.ru-2025-04.spb-1.r1

Vous pouvez mettre à niveau votre instance d'un SPB vers un RU uniquement si le RU est une version supérieure du moteur. Par exemple, vous pouvez passer de la version SPB 19.0.0.0.ru-2025-04.spb-1.r1 à une version ultérieure de RU 19.0.0.0.ru-2025-07.rur-2025-07.rur-2025-07.r1, mais pas à la même version de RU 19.0.0.0.ru-2025-04.rur-2025-04.r1.

Si votre instance de base de données est configurée pour des mises à niveau automatiques de versions mineures et que vous effectuez une mise à niveau manuelle d'un RU vers un SPB ou d'un SPB vers un RU, votre chemin de mise à niveau automatique change. Supposons que vous passiez manuellement la version 19.0.0.0.ru-2025-01.rur-2025-01.r1 à la version SPB 19.0.0.0.ru-2025-01.spb-1.r1. Votre prochaine mise à niveau automatique de la version mineure sera vers la version SPB 19.0.0.0.ru-2025-04.spb-1.r1.

Parce que SPBs fonctionnez en tant que RUs, les RDS APIs pour la mise à niveau de votre instance vers RUs et SPBs sont identiques. Les commandes suivantes illustrent la mise à niveau vers un RU et un SPB.

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --engine-version 19.0.0.0.ru-2025-01.rur-2025-01.r1

aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --engine-version 19.0.0.0.ru-2025-01.spb-1.r1
```

Pour plus d'informations sur l'option Oracle Spatial, consultezComment fonctionnent les Spatial Patch Bundles (SPBs). Pour les informations prises en charge RUs et SPBs pour Oracle Database 19c, consultez Amazon RDS pour Oracle Database 19c (19.0.0.0).

Activation des mises à niveau automatiques des versions mineures pour Oracle

Dans le cadre d'une mise à niveau automatique de version mineure, RDS applique la dernière version mineure disponible à votre base de données Oracle sans intervention manuelle. Une instance

de base de données Amazon RDS pour Oracle planifie votre mise à niveau dans la fenêtre de maintenance suivante, dans les circonstances suivantes :

- L'option Mise à niveau automatique des versions mineures est activée pour votre instance de base de données.
- Votre instance de base de données n'exécute pas encore la dernière version mineure du moteur de base de données.
- Votre instance de base de données n'a pas encore de mise à niveau planifiée en attente.

Pour découvrir comment activer des mises à niveau automatiques, consultez Mise à niveau automatique de la version mineure du moteur.

Notification des mises à niveau automatiques des versions mineures dans RDS pour Oracle

RDS publie un avis préalable avant de commencer à planifier des mises à niveau automatiques. Vous pouvez trouver la notification dans l'onglet Maintenance et sauvegardes de la page de détails de la base de données. Le message a le format suivant :

```
An automatic minor version upgrade to engine version will become available
 on availability-date and will be applied during a subsequent maintenance window.
```

La notification préalable est la date *availability-date* à laquelle RDS commence à planifier les mises à niveau pour les instances de base de données de votre Région AWS. Il ne s'agit pas de la date à laquelle la mise à niveau de votre instance de base de données est prévue. Par exemple, si *availability-date* c'est le 1er mars, à cette date, RDS peut planifier votre mise à niveau pour le 14 avril.

Vous pouvez également obtenir la date de disponibilité de la mise à niveau à l'aide de la `describe-pending-maintenance-actions` commande figurant dans le AWS CLI, comme illustré dans l'exemple suivant :

```
aws rds describe-pending-maintenance-actions

{
    "PendingMaintenanceActions": [
        {
            "ResourceIdentifier": "arn:aws:rds:us-east-1:123456789012:db:orclinst1",
            "PendingMaintenanceActionDetails": [
```

```
                {
                    "Action": "db-upgrade",
                    "Description": "Automatic minor version upgrade to
21.0.0.0.ru-2024-07.rur-2024-07.r1",
                    "CurrentApplyDate": "2024-12-02T08:10:00Z",
                    "OptInStatus": "next-maintenance"
                }
            ]
        }, ...
```

Le tableau ci-dessous décrit les options possibles pour chaque type de message d'action de maintenance en attente.

Message d'action de maintenance en attente	Moment où le message apparaît	Application à la prochaine fenêtre de maintenance possible ?	Application immédiate possible ?	Annulation de l'acceptation possible ?
Une mise à niveau automatique de la version mineure vers *engine-version* sera disponible *availability-date* et devrait être appliquée lors d'une fenêtre de maintenance ultérieure.	4 à 6 semaines avant la data prévue des mises à niveau automatiques.	Oui	Oui	Oui
Mise à niveau automatique de la version mineure vers *engine-version*	À ou après *availability-date* . RDS applique automatiquement cette mise à niveau lors de la prochaine fenêtre de maintenance de l'instance de base de données.	Oui	Oui	Non

Pour plus d'informations sur describe-pending-maintenance-actions, consultez la rubrique Référence des commandes de la AWS CLI .

Lorsque RDS planifie des mises à niveau automatiques de versions mineures dans RDS pour Oracle

Quand la date de disponibilité des mises à niveau automatiques arrive, RDS commence à planifier les mises à niveau. Dans la plupart des cas Régions AWS, RDS planifie votre mise à niveau vers le dernier RU trimestriel environ quatre à six semaines après la date de disponibilité. La date prévue varie en fonction de ces facteurs Région AWS et d'autres facteurs. Pour plus d'informations sur RUs et RURs, consultez les notes de mise à jour d'Amazon RDS for Oracle.

Quand RDS planifie la mise à niveau, la notification suivante apparaît dans l'onglet Maintenance et sauvegardes de la page de détails de la base de données :

```
Automatic minor version upgrade to engine-version
```

Le message précédent indique que RDS a planifié la mise à niveau de votre moteur de base de données lors de la prochaine fenêtre de maintenance.

Parfois, une nouvelle version mineure devient disponible avant que RDS n'applique une version mineure précédente. Par exemple, votre instance est en cours d'exécution *minor-version-1* lorsque les deux *minor-version-2* *minor-version-3* sont disponibles en tant que cibles de mise à niveau. Dans ce cas, afin d'éviter des temps d'arrêt inutiles pour vos instances de base de données, RDS planifie la mise à niveau automatique de la version mineure vers la version la plus récente, en ignorant la mise à niveau vers la version précédente. Dans cet exemple, RDS met à niveau votre instance *minor-version-1* directement vers*minor-version-3*.

Pour garantir une certaine fréquence des mises à niveau des versions mineures, vous pouvez mettre à niveau vos instances manuellement au lieu d'utiliser le mécanisme de mise à niveau automatique. Pour planifier une mise à niveau pour la prochaine fenêtre de maintenance, spécifiez à quel `--no-apply-immediately` moment vous effectuez la mise à niveau vers une version mineure en utilisant`modify-db-instance`. Pour effectuer une mise à niveau immédiate, spécifiez `--apply-immediately` plutôt. Pour de plus amples informations, veuillez consulter Mise à niveau manuelle de la version du moteur.

Gestion d'une mise à niveau automatique d'une version mineure dans RDS pour Oracle

Quand une nouvelle version mineure devient disponible, vous pouvez mettre à niveau manuellement votre instance de base de données vers cette version. L'exemple suivant met à niveau immédiatement l'instance de base de données nommée `orclinst1` :

```
aws rds apply-pending-maintenance-action \
    --resource-identifier arn:aws:rds:us-east-1:123456789012:db:orclinst1 \
    --apply-action db-upgrade \
    --opt-in-type immediate
```

Pour refuser une mise à niveau automatique de version mineure qui n'a pas encore été planifiée, définissez `--opt-in-type` sur `undo-opt-in`, comme dans l'exemple suivant :

```
aws rds apply-pending-maintenance-action \
    --resource-identifier arn:aws:rds:us-east-1:123456789012:db:orclinst1 \
    --apply-action db-upgrade \
    --opt-in-type undo-opt-in
```

Si RDS a déjà planifié une mise à niveau pour votre instance de base de données, vous ne pouvez pas utiliser `apply-pending-maintenance-action` pour l'annuler. En revanche, vous pouvez modifier votre instance de base de données pour désactiver la fonction de mise à niveau automatique de version mineure, ce qui annule la planification de la mise à niveau.

Pour découvrir comment activer des mises à niveau automatiques de version mineure, consultez Mise à niveau automatique de la version mineure du moteur. Pour plus d'informations sur apply-pending-maintenance-action, consultez la rubrique Référence des commandes de la AWS CLI .

Considérations relatives aux mises à niveau des bases de données

Avant de mettre à niveau votre instance Oracle, consultez les informations suivantes.

Rubriques

- Considérations relatives à Oracle Multitenant
- Considérations relatives au groupe d'options
- Considérations relatives au groupe de paramètres
- Considérations relatives au fuseau horaire
- Considérations relatives au Spatial Patch Bundle (SPB)

Considérations relatives à Oracle Multitenant

Le tableau suivant décrit les architectures de base de données Oracle prises en charge dans les différentes versions.

Version de Oracle Database	Statut de prise en charge de RDS	Architecture
Oracle Database 21c	Pris en charge	CDB uniquement
Oracle Database 19c	Pris en charge	CDB ou non CDB

Le tableau suivant décrit les chemins de mise à niveau pris en charge et non pris en charge.

Chemin de mise à niveau	Pris en charge ?
CDB à CDB	Oui
Non CDB à CDB	Non, mais vous pouvez convertir un objet non CDB en CDB, puis le mettre à niveau
CDB à non CDB	Non

Pour plus d'informations sur Oracle Multitenant dans RDS for Oracle, consultez Configuration à locataire unique de l'architecture CDB.

Considérations relatives au groupe d'options

Si votre instance de base de données utilise un groupe d'options personnalisé, Amazon RDS n'est pas toujours en mesure d'attribuer automatiquement un nouveau groupe d'options. C'est le cas, par exemple, lorsque vous procédez à une mise à niveau vers une nouvelle version majeure. Dans ce cas, spécifiez un nouveau groupe d'options lors de la mise à niveau. Nous vous recommandons de créer un nouveau groupe d'options et d'y ajouter les mêmes options qu'à votre groupe d'options personnalisé existant.

Pour plus d'informations, veuillez consulter Création d'un groupe d'options ou Copie d'un groupe d'options.

Si votre instance de base de données utilise un groupe d'options personnalisé contenant les `APEX-DEV` options `APEX` et, vous pouvez parfois réduire le temps de mise à niveau. Pour ce faire, mettez à niveau votre version d'Oracle APEX en même temps que votre instance de base de données. Pour de plus amples informations, veuillez consulter Mise à niveau de la version Oracle APEX.

Considérations relatives au groupe de paramètres

Si votre instance de base de données utilise un groupe de paramètres personnalisé, Amazon RDS n'est pas toujours en mesure d'attribuer automatiquement un nouveau groupe de paramètres à votre instance de base de données. C'est le cas, par exemple, lorsque vous procédez à une mise à niveau vers une nouvelle version majeure. Dans ce cas, vous devez spécifier un nouveau groupe de paramètres lors de la mise à niveau. Nous vous recommandons de créer un nouveau groupe de paramètres et de configurer les mêmes paramètres que ceux de votre groupe de paramètres personnalisé existant.

Pour plus d'informations, veuillez consulter Création d'un groupe de paramètres de base de données dans RDSAmazon ou Copier un groupe de paramètres de base de données dans RDSAmazon.

Considérations relatives au fuseau horaire

L'option de fuseau horaire vous permet de modifier le fuseau horaire système utilisé par votre instance de base de données Oracle. Par exemple, vous devrez peut-être modifier le fuseau horaire d'une instance de base de données afin qu'elle soit compatible avec un environnement sur site ou une application héritée. L'option de fuseau horaire change le fuseau horaire au niveau de l'hôte. Amazon RDS for Oracle met à jour automatiquement le fuseau horaire système tout au long de l'année. Pour plus d'informations sur le fuseau horaire système, veuillez consulter Fuseau horaire Oracle.

Lorsque vous créez une instance de base de données Oracle, la base de données définit automatiquement le fuseau horaire de la base de données. Le fuseau horaire de la base de données est également connu sous le nom de fuseau horaire pour l'heure d'été (DST). Le fuseau horaire de la base de données est distinct du fuseau horaire système.

Entre les versions de base de données Oracle, les jeux de correctifs ou les correctifs individuels peuvent inclure de nouvelles versions d'heure d'été. Ces correctifs reflètent les modifications apportées aux règles de transition pour diverses régions de fuseau horaire. Par exemple, un gouvernement peut modifier l'entrée en vigueur de l'heure d'été. Les modifications apportées aux règles d'heure d'été peuvent affecter les données existantes du type de données `TIMESTAMP WITH TIME ZONE`.

Si vous mettez à niveau une instance de base de données RDS for Oracle, Amazon RDS ne met pas à niveau le fuseau horaire de la base de données automatiquement. Pour mettre à niveau automatiquement le fichier sur le fuseau horaire, vous pouvez inclure l'option `TIMEZONE_FILE_AUTOUPGRADE` au groupe d'options associé à votre instance de base de données pendant ou après la mise à niveau de version du moteur. Pour plus d'informations, veuillez consulter Mise à niveau automatique du fichier sur le fuseau horaire Oracle.

Aussi, pour mettre à niveau manuellement le fichier sur le fuseau horaire de la base de données, créez une nouvelle instance de base de données Oracle dotée du correctif d'heure d'été souhaité. Nous vous recommandons toutefois de mettre à niveau le fichier sur le fuseau horaire de la base de données à l'aide de l'option `TIMEZONE_FILE_AUTOUPGRADE`.

Après la mise à niveau du fichier sur le fuseau horaire, procédez à la migration des données depuis votre instance actuelle vers la nouvelle instance. Vous pouvez migrer des données à l'aide de plusieurs techniques, dont les suivantes :

- AWS Database Migration Service
- Oracle GoldenGate
- Oracle Data Pump
- Exportation/importation d'origine (non prise en charge pour usage général)

> **Note**
>
> Lorsque vous migrez des données à l'aide d'Oracle Data Pump, l'utilitaire déclenche l'erreur ORA-39405 lorsque la version du fuseau horaire cible est inférieure à la version du fuseau horaire source.

Pour de plus amples informations, veuillez consulter TIMESTAMP WITH TIMEZONE Restrictions dans la documentation Oracle.

Considérations relatives au Spatial Patch Bundle (SPB)

Dans RDS pour Oracle, la mise à jour (RU) est une version mineure du moteur qui inclut des correctifs de sécurité, des corrections de bogues et de nouvelles fonctionnalités pour Oracle Database. Un Spatial Patch Bundle (SPB) est une version mineure du moteur qui inclut également des correctifs conçus pour l'option Oracle Spatial. Par exemple,

19.0.0.0.ru-2025-01.spb-1.r1 est une version mineure du moteur qui contient les correctifs RU de la version 19.0.0.0.ru-2025-01.rur-2025-01.rur-2025-01.r1 ainsi que des correctifs spatiaux.

Lorsque vous mettez à niveau votre base de données vers SPBs, tenez compte des points suivants :

- SPBs ne sont pris en charge que pour Oracle Database 19c.
- Généralement, un SPB est publié 2 à 3 semaines après le RU trimestriel correspondant.
- Vous pouvez mettre à niveau votre instance de base de données vers un SPB même si l'instance n'utilise pas l'option Oracle Spatial, mais les correctifs spatiaux de la version du moteur ne s'appliquent qu'à Oracle Spatial. Vous pouvez créer une nouvelle instance sur un SPB et installer l'option Oracle Spatial ultérieurement.
- Si vous activez la mise à niveau automatique des versions mineures pour votre instance de base de données, le chemin de mise à niveau dépend du fait que votre instance utilise actuellement un SPB ou un RU. Si votre instance utilise un SPB, RDS met automatiquement à niveau votre instance vers le dernier SPB. Si votre instance utilise un RU, RDS met automatiquement à niveau votre instance vers le dernier RU.
- Vous pouvez mettre à niveau manuellement votre instance de base de données d'un RU vers un SPB uniquement si le SPB est de la même version du moteur ou d'une version supérieure à votre RU actuel.
- Vous pouvez mettre à niveau manuellement votre instance de base de données d'un SPB vers un RU uniquement si le RU est une version supérieure.

Test d'une mise à niveau de base de données Oracle

Avant de procéder à la mise à niveau de votre instance de base de données vers une version majeure, vous devez tester la compatibilité de votre base de données et de toutes les applications qui y accèdent avec la nouvelle version. Nous vous recommandons d'utiliser la procédure suivante.

Pour tester une mise à niveau de version majeure

1. Passez en revue la documentation de la mise à niveau Oracle pour la nouvelle version du moteur de base de données pour voir si des problèmes de compatibilité peuvent affecter votre base de données ou vos applications. Pour plus d'informations, consultez le document Database Upgrade Guide (Guide de mise à niveau de base de données) dans la documentation d'Oracle.
2. Si votre instance de base de données utilise un groupe d'options personnalisé, créez un nouveau groupe d'options compatible avec la version vers laquelle vous procédez à la mise à niveau. Pour plus d'informations, consultez Considérations relatives au groupe d'options.

3. Si votre instance de base de données utilise un groupe de paramètres personnalisé, créez un nouveau groupe de paramètres compatible avec la version vers laquelle vous procédez à la mise à niveau. Pour plus d'informations, consultez Considérations relatives au groupe de paramètres.
4. Créez un instantané de base de données de l'instance de base de données à mettre à niveau. Pour plus d'informations, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.
5. Restaurez l'instantané de base de données pour créer une nouvelle instance de base de données de test. Pour plus d'informations, consultez Restauration vers une instance de base de données.
6. Modifiez cette nouvelle instance de base de données de test pour la mettre à niveau vers la nouvelle version, en utilisant l'une des méthodes suivantes :

   - console
   - AWS CLI
   - API RDS
7. Effectuez des tests :

   - Exécutez sur l'instance de base de données mise à niveau autant de tests d'assurance qualité que nécessaire pour garantir que votre base de données et votre application fonctionnent correctement avec la nouvelle version.
   - Implémentez tous les nouveaux tests requis pour évaluer l'impact des éventuels problèmes de compatibilité que vous avez identifiés à l'étape 1.
   - Testez l'ensemble des procédures stockées, fonctions et déclencheurs.
   - Dirigez les versions de test de vos applications vers l'instance de base de données mise à niveau. Vérifiez que les applications fonctionnent correctement avec la nouvelle version.
   - Évaluez le stockage utilisé par l'instance mise à niveau pour déterminer si la mise à niveau requiert un stockage supplémentaire. Vous devrez peut-être choisir une plus grande classe d'instance pour la prise en charge de la nouvelle version en production. Pour plus d'informations, consultez Classes d'instances de base de données .
8. Si tous les tests réussissent, mettez à niveau votre instance de base de données de production. Nous vous recommandons de confirmer que l'instance de base de données fonctionne correctement avant d'autoriser les opérations d'écriture sur l'instance de base de données.

Mise à niveau de la version d'une instance de base de données RDS pour Oracle

Pour mettre à niveau manuellement la version du moteur de base de données d'une instance de base de données RDS pour Oracle, utilisez l' AWS Management Console API AWS CLI, la ou l'API RDS. Pour des informations générales sur les mises à niveau de bases de données dans RDS, consultezMise à niveau de la version d'une instance de base de données RDS pour Oracle. Pour obtenir des cibles de mise à niveau valides, utilisez la AWS CLI describe-db-engine-versionscommande.

Console

Pour mettre à niveau la version du moteur d'une instance de base de données RDS pour Oracle à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Bases de données, puis l'instance de base de données que vous souhaitez mettre à niveau.
3. Sélectionnez Modify.
4. Pour la version du moteur de base de données, choisissez une version de base de données supérieure.
5. Choisissez Continuer et vérifiez le récapitulatif des modifications. Assurez-vous de bien comprendre les implications d'une mise à niveau de version de base de données. Vous ne pouvez pas reconvertir une instance de base de données mise à niveau vers la version précédente. Assurez-vous d'avoir testé votre base de données et votre application avec la nouvelle version avant de continuer.
6. Décidez quand planifier la mise à niveau de votre instance de base de données. Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement. La sélection de cette option peut entraîner une interruption de service dans certains cas. Pour plus d'informations, consultez Utilisation du paramètre de modification du calendrier.
7. Sur la page de confirmation, examinez vos modifications. Si elles sont correctes, choisissez Modify DB instance (Modifier l'instance de base de données) pour enregistrer vos modifications.

   Sinon, choisissez Retour pour modifier vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour mettre à niveau la version du moteur d'une instance de base de données RDS pour Oracle, vous pouvez utiliser la modify-db-instancecommande CLI. Spécifiez les paramètres suivants :

- `--db-instance-identifier`: le nom de l'instance de base de données RDS pour Oracle.
- `--engine-version` – numéro de version du moteur de base de données vers lequel effectuer la mise à niveau.

  Pour plus d'informations sur les versions valides du moteur, utilisez la AWS CLI describe-db-engine-versionscommande.
- `--allow-major-version-upgrade`— pour mettre à niveau la version du moteur de base de données.
- `--no-apply-immediately` – pour appliquer les modifications au cours de la fenêtre de maintenance suivante. Pour appliquer les modifications immédiatement, utilisez `--apply-immediately`.

Example

L'exemple suivant met à niveau une instance CDB nommée `myorainst` de sa version actuelle de `19.0.0.0.ru-2024-01.rur-2024-01.r1` vers la version`21.0.0.0.ru-2024-04.rur-2024-04.r1`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier myorainst \
    --engine-version 21.0.0.0.ru-2024-04.rur-2024-04.r1 \
    --allow-major-version-upgrade \
    --no-apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier myorainst ^
    --engine-version 21.0.0.0.ru-2024-04.rur-2024-04.r1 ^
    --allow-major-version-upgrade ^
    --no-apply-immediately
```

API RDS

Pour mettre à niveau une instance de base de données RDS pour Oracle, utilisez l'action ModifyDBInstance. Spécifiez les paramètres suivants :

- `DBInstanceIdentifier` – nom de l'instance de base de données, par exemple *`myorainst`*.
- `EngineVersion` – numéro de version du moteur de base de données vers lequel effectuer la mise à niveau. Pour plus d'informations sur les versions valides du moteur, utilisez l'opération DescribeDB EngineVersions.
- `AllowMajorVersionUpgrade` – pour autoriser une mise à niveau de version majeure. Pour ce faire, définissez la valeur sur `true`.
- `ApplyImmediately` – si des modifications doivent être appliquées immédiatement ou au cours du prochain créneau de maintenance. Pour appliquer les modifications immédiatement, définissez la valeur sur `true`. Pour appliquer les modifications pendant le créneau de maintenance suivant, définissez la valeur sur `false`.

Mise à niveau d'un instantané de base de données Oracle

La mise à niveau de vos instantanés de base de données Oracle sur Amazon RDS garantit que votre base de données reste sécurisée, compatible et entièrement prise en charge. Lorsque les anciennes versions d'Oracle atteignent la fin de la prise en charge des correctifs, vous pouvez mettre à niveau tous les instantanés de base de données manuels liés à ces versions afin d'éviter d'éventuelles vulnérabilités ou limitations de service. Pour de plus amples informations, veuillez consulter Gestion des versions du moteur Oracle.

Amazon RDS prend en charge la mise à niveau des instantanés dans tous les domaines Régions AWS.

Console

Pour mettre à niveau un instantané de base de données Oracle

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Instantanés, puis sélectionnez l'instantané de base de données que vous souhaitez mettre à niveau.
3. Pour Actions, choisissez Upgrade Snapshot (Mettre à niveau l'instantané). La page Upgrade Snapshot (Mettre à niveau l'instantané) s'affiche.

4. Choisissez la nouvelle version du moteur vers laquelle mettre à niveau l'instantané.
5. (Facultatif) Pour Groupe d'options, choisissez le groupe d'options pour l'instantané de base de données mis à niveau. Les mêmes considérations relatives au groupe d'options s'appliquent pour la mise à niveau d'un instantané de base de données et la mise à niveau d'une instance de base de données. Pour plus d'informations, consultez Considérations relatives au groupe d'options.
6. Choisissez Enregistrer les modifications pour enregistrer vos modifications.

   Pendant le processus de mise à niveau, toutes les actions d'instantané sont désactivées pour l'instantané de base de données. De même, le statut de l'instantané de base de données passe de disponible à upgrading (mise à niveau), puis passe à active, une fois la mise à niveau terminée. Si l'instantané de base de données ne peut pas être mis à jour en raison d'un problème d'instantané endommagé, le statut devient indisponible. Vous ne pouvez pas récupérer l'instantané lorsqu'il a ce statut.

   > **Note**
   >
   > Si la mise à niveau de l'instantané de base de données échoue, l'instantané revient à l'état d'origine avec la version originale.

AWS CLI

Pour mettre à niveau un instantané de base de données Oracle à l'aide du AWS CLI, appelez la modify-db-snapshotcommande avec les paramètres suivants :

- `--db-snapshot-identifier` – Nom de l'instantané de base de données.
- `--engine-version` – Version vers laquelle mettre à niveau l'instantané.

Vous pouvez également devoir fournir le paramètre suivant. Les mêmes considérations relatives au groupe d'options s'appliquent pour la mise à niveau d'un instantané de base de données et la mise à niveau d'une instance de base de données. Pour plus d'informations, consultez Considérations relatives au groupe d'options.

- `--option-group-name` – Groupe d'options pour l'instantané de base de données mis à niveau.

Example

L'exemple suivant met à niveau un instantané de base de données.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot \
    --db-snapshot-identifier mydbsnapshot \
    --engine-version 19.0.0.0.ru-2020-10.rur-2020-10.r1 \
    --option-group-name default:oracle-se2-19
```

Dans Windows:

```
aws rds modify-db-snapshot ^
    --db-snapshot-identifier mydbsnapshot ^
    --engine-version 19.0.0.0.ru-2020-10.rur-2020-10.r1 ^
    --option-group-name default:oracle-se2-19
```

RDS API

Pour mettre à niveau un instantané de base de données Oracle à l'aide d'Amazon RDSAPI, appelez l'odifyDBSnapshotopération M avec les paramètres suivants :

- `DBSnapshotIdentifier` – Nom de l'instantané de base de données.
- `EngineVersion` – Version vers laquelle mettre à niveau l'instantané.

Vous devrez peut-être également inclure le paramètre `OptionGroupName`. Les mêmes considérations relatives au groupe d'options s'appliquent pour la mise à niveau d'un instantané de base de données et la mise à niveau d'une instance de base de données. Pour plus d'informations, consultez Considérations relatives au groupe d'options.

Utilisation d'un logiciel tiers avec votre instance de base de données RDS for Oracle

Vous pouvez héberger une instance de base de données RDS pour Oracle qui prend en charge des outils et des logiciels tiers.

Rubriques

- Utilisation d'Oracle GoldenGate avec Amazon RDS for Oracle
- Utilisation de l'utilitaire Oracle Repository Creation Utility sur RDS for Oracle
- Configuration d'Oracle Connection Manager sur une instance Amazon EC2
- Installation d'une base de données Siebel sur Oracle sur Amazon RDS

Utilisation d'Oracle GoldenGate avec Amazon RDS for Oracle

Oracle GoldenGate collecte, réplique et gère les données transactionnelles entre les bases de données. Il s'agit d'un package logiciel de réplication et de capture de données modifiées basé sur les journaux utilisé avec les bases de données pour les systèmes de traitement transactionnel en ligne (OLTP). Oracle GoldenGate crée des fichiers de suivi qui contiennent les données les plus récentes modifiées de la base de données source. Il transmet (push) ensuite ces fichiers au serveur, où un processus convertit le fichier de suivi en code SQL standard à appliquer à la base de données cible.

Oracle GoldenGate with RDS for Oracle prend en charge les fonctionnalités suivantes :

- Réplication de base de données active-active
- Reprise après sinistre
- Protection des données
- Réplication dans la Région et entre Régions
- Migration et mises à niveau sans temps d'arrêt
- Réplication de données entre une instance de base de données RDS for Oracle et une base de données non-Oracle

> **Note**
>
> Pour obtenir la liste des bases de données prises en charge, consultez Oracle Fusion Middleware Supported System Configurations (Configurations système prises en charge par Oracle Fusion Middleware) dans la documentation Oracle.

Vous pouvez utiliser Oracle GoldenGate avec RDS pour Oracle pour effectuer une mise à niveau vers les versions majeures d'Oracle Database. Par exemple, vous pouvez utiliser Oracle pour effectuer une mise GoldenGate à niveau d'une base de données sur site Oracle Database 11g vers une base de données Oracle Database 19c sur une instance de base de données Amazon RDS.

Rubriques

- Versions prises en charge et options de licence pour Oracle GoldenGate
- Exigences et limites relatives à Oracle GoldenGate
- GoldenGate Architecture Oracle

- Configuration d'Oracle GoldenGate
- Utilisation des REPLICAT utilitaires EXTRACT et des utilitaires d'Oracle GoldenGate
- Surveillance d'Oracle GoldenGate
- Dépannage d'Oracle GoldenGate

Versions prises en charge et options de licence pour Oracle GoldenGate

Vous pouvez utiliser l'édition Standard 2 (SE2) ou Enterprise (EE) de RDS pour Oracle avec Oracle GoldenGate version 12c ou ultérieure. Vous pouvez utiliser les GoldenGate fonctionnalités Oracle suivantes :

- Oracle GoldenGate Remote Capture (extrait) est pris en charge.
- La capture (extract) est prise en charge sur les instances de base de données RDS for Oracle qui utilisent l'architecture de base de données traditionnelle non-CDB. La capture Oracle GoldenGate Remote PDB est prise en charge sur Oracle Database 21c ou Oracle Database CDBs 19c version 19.0.0.0.ru-2024-04.rur-2024-04.rur-2024-04.r1 ou supérieure.
- Oracle GoldenGate Remote Delivery (replicat) est pris en charge sur RDS pour les instances de base de données Oracle qui utilisent des architectures non CDB ou CDB. Remote Delivery prend en charge les processus Replicat intégré, Replicat parallèle, Replicat coordonné et Replicat classique.
- RDS for Oracle prend en charge les architectures classiques et microservices d'Oracle. GoldenGate
- La réplication des valeurs GoldenGate DDL et de séquence Oracle est prise en charge lors de l'utilisation du mode de capture intégré.

Vous êtes responsable de la gestion des GoldenGate licences Oracle (BYOL) destinées à être utilisées avec Amazon RDS dans l'ensemble. Régions AWS Pour de plus amples informations, veuillez consulter RDSpour les options de licence Oracle.

Exigences et limites relatives à Oracle GoldenGate

Lorsque vous travaillez avec Oracle GoldenGate et RDS pour Oracle, tenez compte des exigences et limites suivantes :

- Vous êtes responsable de la configuration et de la gestion d'Oracle en GoldenGate vue de son utilisation avec RDS pour Oracle.

- Vous êtes responsable de la configuration d'une GoldenGate version d'Oracle certifiée avec les bases de données source et cible. Pour de plus amples informations, veuillez consulter la page Configurations système prises en charge par Oracle Fusion Middleware dans la documentation Oracle.
- Vous pouvez utiliser Oracle dans GoldenGate de nombreux AWS environnements différents pour de nombreux cas d'utilisation différents. Si vous rencontrez un problème lié au support Oracle GoldenGate, contactez les services de support Oracle.
- Vous pouvez utiliser Oracle GoldenGate on RDS pour les instances de base de données Oracle qui utilisent Oracle Transparent Data Encryption (TDE). Pour préserver l'intégrité des données répliquées, configurez le chiffrement sur le GoldenGate hub Oracle à l'aide des volumes chiffrés Amazon EBS ou du chiffrement des fichiers de suivi. Configurez également le chiffrement des données envoyées entre le GoldenGate hub Oracle et les instances de base de données source et cible. Les instances de base de données RDS for Oracle prennent en charge le chiffrement avec Oracle Secure Sockets Layer (SSL) ou Oracle NNE (Native Network Encryption).

GoldenGate Architecture Oracle

L' GoldenGate architecture Oracle à utiliser avec Amazon RDS comprend les modules découplés suivants :

Base de données source

Votre base de données source peut être une base de données Oracle sur site, une base de données Oracle sur une EC2 instance Amazon ou une base de données Oracle sur une RDS instance de base de données Amazon.

GoldenGate Hub Oracle

Un GoldenGate hub Oracle déplace les informations de transaction de la base de données source vers la base de données cible. Votre hub peut être l'un des suivants :

- Une EC2 instance Amazon sur laquelle Oracle Database et Oracle GoldenGate sont installés
- Une installation Oracle sur site

Vous pouvez avoir plusieurs EC2 hubs Amazon. Nous vous recommandons d'utiliser deux hubs si vous utilisez Oracle GoldenGate pour la réplication entre régions.

Base de données cible

Votre base de données cible peut se trouver soit sur une RDS instance de base de données Amazon, soit sur une EC2 instance Amazon, soit sur site.

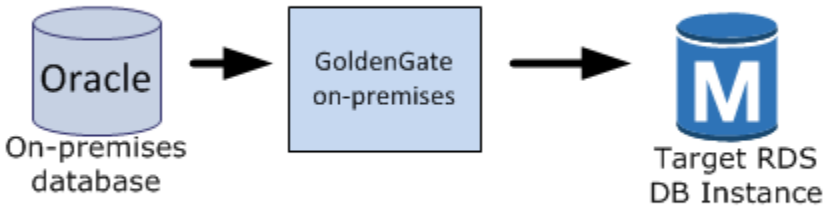
Les sections suivantes décrivent des scénarios courants pour Oracle GoldenGate sur AmazonRDS.

Rubriques

- Base de données source sur site et hub Oracle GoldenGate
- Base de données source sur site et Amazon Hub EC2
- Base de données RDS source Amazon et EC2 hub Amazon
- Base de données EC2 source Amazon et EC2 hub Amazon
- EC2Hubs Amazon dans différentes AWS régions

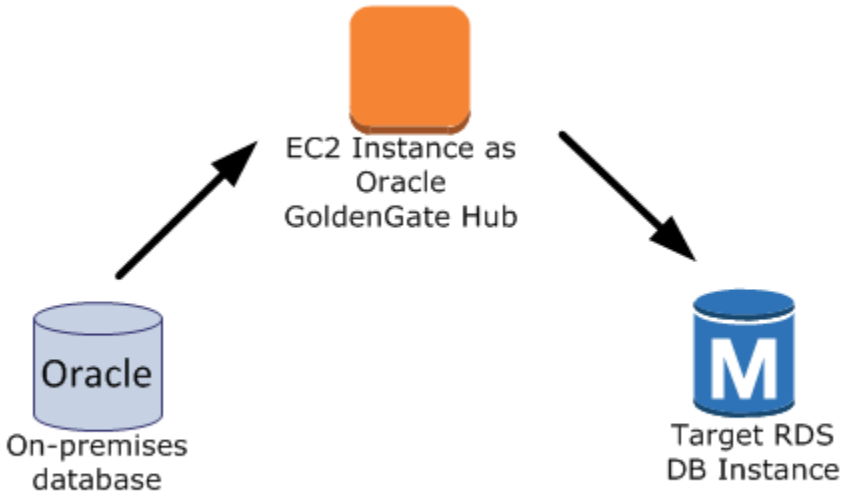
Base de données source sur site et hub Oracle GoldenGate

Dans ce scénario, une base de données source Oracle sur site et un GoldenGate hub Oracle sur site fournissent des données à une RDS instance de base de données Amazon cible.



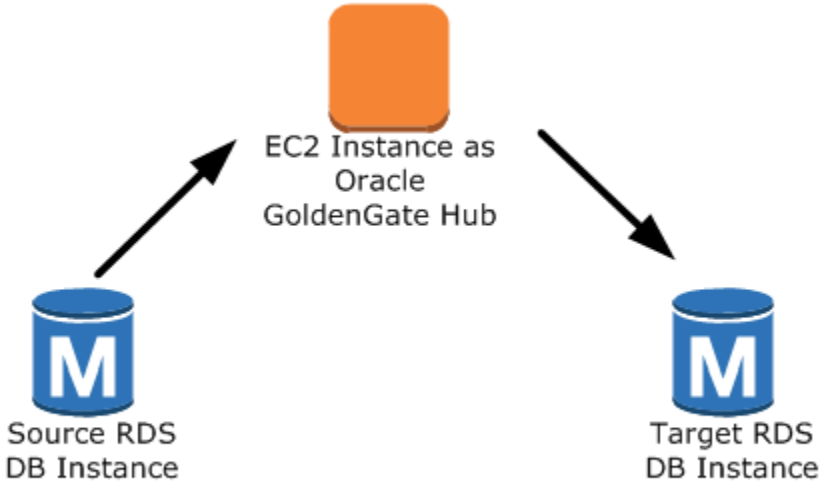
Base de données source sur site et Amazon Hub EC2

Dans ce scénario, une base de données Oracle sur site agit comme base de données source. Il est connecté à un hub d'EC2instance Amazon. Ce hub fournit des données à une cible RDS pour une instance de base de données Oracle.



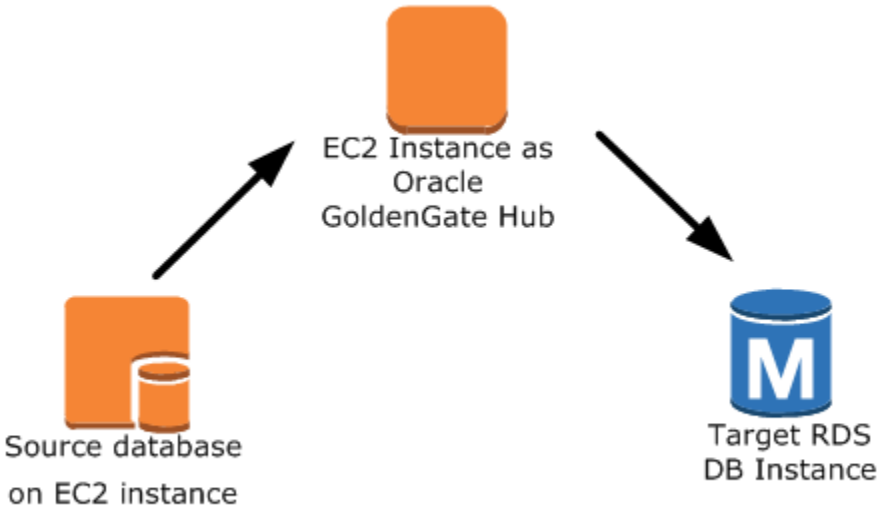
Base de données RDS source Amazon et EC2 hub Amazon

Dans ce scénario, une instance de base de données RDS pour Oracle fait office de base de données source. Il est connecté à un hub d'EC2instance Amazon. Ce hub fournit des données à une cible RDS pour une instance de base de données Oracle.



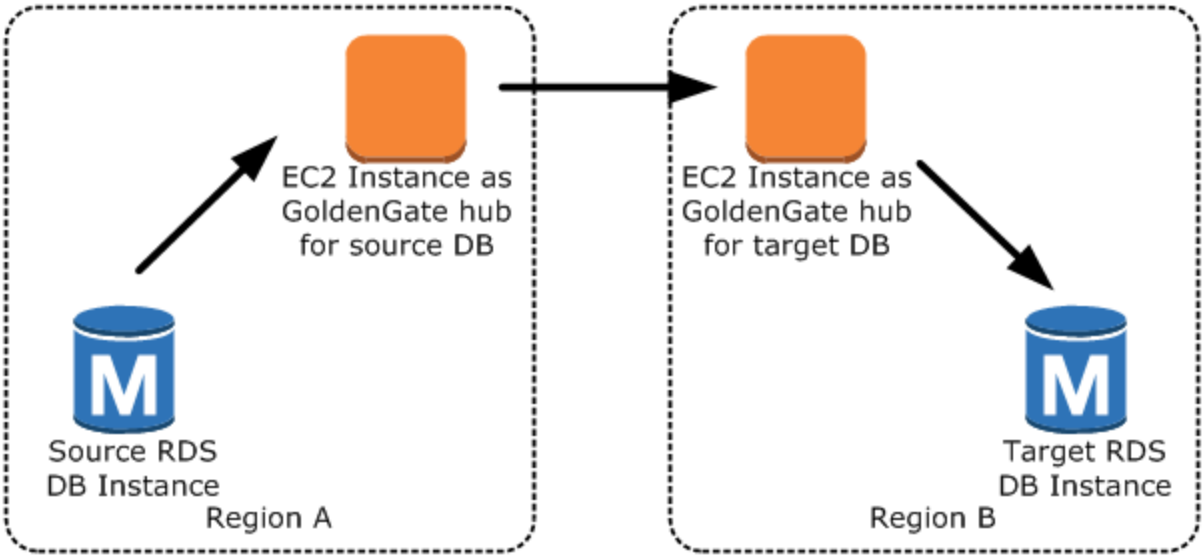
Base de données EC2 source Amazon et EC2 hub Amazon

Dans ce scénario, une base de données Oracle sur une EC2 instance Amazon fait office de base de données source. Il est connecté à un hub d'EC2instance Amazon. Ce hub fournit des données à une cible RDS pour une instance de base de données Oracle.



EC2Hubs Amazon dans différentes AWS régions

Dans ce scénario, une base de données Oracle sur une RDS instance de base de données Amazon est connectée à un hub d'EC2instance Amazon dans la même AWS région. Le hub est connecté à un hub d'EC2instance Amazon dans une autre AWS région. Ce deuxième hub fournit des données à la cible RDS pour l'instance de base de données Oracle dans la même AWS région que le deuxième hub d'EC2instance Amazon.



> **Note**
>
> Tout problème affectant l'exécution d'Oracle dans un environnement GoldenGate sur site a également une incidence sur l'exécution d'Oracle GoldenGate sur AWS. Nous vous recommandons vivement de surveiller le GoldenGate hub Oracle pour vous en assurer `EXTRACT` et `REPLICAT` de le reprendre en cas de basculement. Le GoldenGate hub Oracle étant exécuté sur une EC2 instance Amazon, Amazon RDS ne gère pas le GoldenGate hub Oracle et ne peut pas garantir son fonctionnement.

Configuration d'Oracle GoldenGate

Pour configurer Oracle à GoldenGate l'aide d'AmazonRDS, configurez le hub sur une EC2 instance Amazon, puis configurez les bases de données source et cible. Les sections suivantes fournissent un exemple de configuration d'Oracle GoldenGate pour une utilisation avec Amazon RDS pour Oracle.

Rubriques

- Configuration d'un GoldenGate hub Oracle sur Amazon EC2
- Configuration d'une base de données source à utiliser avec Oracle GoldenGate sur Amazon RDS
- Configuration d'une base de données cible à utiliser avec Oracle GoldenGate sur Amazon RDS

Configuration d'un GoldenGate hub Oracle sur Amazon EC2

Pour créer un GoldenGate hub Oracle sur une EC2 instance Amazon, vous devez d'abord créer une EC2 instance Amazon avec une installation client complète d'OracleRDBMS. Le GoldenGate logiciel Oracle doit également être installé sur l'EC2instance Amazon. Les versions du GoldenGate logiciel Oracle dépendent des versions de base de données source et cible. Pour plus d'informations sur l'installation d'Oracle GoldenGate, consultez la GoldenGatedocumentation Oracle.

L'EC2instance Amazon qui sert de GoldenGate hub Oracle stocke et traite les informations de transaction de la base de données source dans des fichiers de suivi. Pour prendre en charge ce processus, veillez à respecter les exigences suivantes :

- Vous avez alloué suffisamment de stockage aux fichiers de suivi.
- L'EC2instance Amazon dispose d'une puissance de traitement suffisante pour gérer la quantité de données.

- L'EC2instance dispose de suffisamment de mémoire pour stocker les informations de transaction avant qu'elles ne soient écrites dans le fichier de suivi.

Pour configurer un hub d'architecture GoldenGate classique Oracle sur une EC2 instance Amazon

1. Créez des sous-répertoires dans le GoldenGate répertoire Oracle.

   Dans le shell de ligne de EC2 commande Amazon`ggsci`, lancez l'interpréteur de GoldenGate commandes Oracle. La commande `CREATE SUBDIRS` crée des sous-répertoires sous le répertoire `/gg` pour les fichiers de point de vérification, de rapport et de paramètres.

   ```
   prompt$ cd /gg
   prompt$ ./ggsci

   GGSCI> CREATE SUBDIRS
   ```

2. Configurez le fichier `mgr.prm`.

   L'exemple suivant ajoute des lignes dans le fichier `$GGHOME/dirprm/mgr.prm`.

   ```
   PORT 8199
   PurgeOldExtracts ./dirdat/*, UseCheckpoints, MINKEEPDAYS 5
   ```

3. Démarrez le gestionnaire.

   L'exemple suivant lance `ggsci` et exécute la commande `start mgr`.

   ```
   GGSCI> start mgr
   ```

Le GoldenGate hub Oracle est maintenant prêt à être utilisé.

Configuration d'une base de données source à utiliser avec Oracle GoldenGate sur Amazon RDS

Effectuez les tâches suivantes pour configurer une base de données source à utiliser avec Oracle GoldenGate.

Étapes de configuration

- Étape 1 : activer une journalisation supplémentaire sur la base de données source

- Étape 2 : définissez le paramètre d'REPLICATIONinitialisation ENABLE GOLDENGATE _ _ sur true
- Étape 3 : définir la période de conservation des journaux sur la base de données source
- Étape 4 : créer un compte GoldenGate utilisateur Oracle dans la base de données source
- Étape 5 : accorder des privilèges de compte d'utilisateur sur la base de données source
- Étape 6 : ajouter un TNS alias pour la base de données source

Étape 1 : activer une journalisation supplémentaire sur la base de données source

Pour activer la journalisation supplémentaire minimale au niveau de la base de données, exécutez la procédure PL/ suivante : SQL

```
EXEC rdsadmin.rdsadmin_util.alter_supplemental_logging(p_action => 'ADD')
```

Étape 2 : définissez le paramètre d'REPLICATIONinitialisation ENABLE GOLDENGATE _ _ sur true

Lorsque vous définissez le paramètre d'initialisation `ENABLE_GOLDENGATE_REPLICATION` sur `true`, il permet aux services de base de données de prendre en charge la réplication logique. Si votre base de données source se trouve sur une RDS instance de base de données Amazon, assurez-vous qu'un groupe de paramètres est attribué à l'instance de base de données avec le paramètre d'`ENABLE_GOLDENGATE_REPLICATION`initialisation défini sur. `true` Pour plus d'informations sur le paramètre d'initialisation `ENABLE_GOLDENGATE_REPLICATION`, consultez la documentation Oracle Database.

Étape 3 : définir la période de conservation des journaux sur la base de données source

Veillez à configurer la base de données source de manière à conserver les journaux redo archivés. Considérez les directives suivantes :

- Spécifiez la durée de conservation des journaux en heures. La valeur minimale est d'une heure.
- Définissez cette durée de manière à ce qu'elle dépasse tout temps d'arrêt éventuel de l'instance de base de données source ou la durée de tout problème potentiel de communication ou de mise en réseau pour l'instance source. Une telle durée permet à Oracle de GoldenGate récupérer les journaux de l'instance source selon les besoins.
- Assurez-vous de disposer d'un espace de stockage suffisant sur votre instance pour les fichiers.

Par exemple, définissez la période de conservation des journaux redo archivés sur 24 heures.

```
EXEC rdsadmin.rdsadmin_util.set_configuration('archivelog retention hours',24)
```

Si la conservation des journaux n'est pas activée ou si la valeur de conservation est trop faible, vous recevez un message d'erreur similaire au suivant.

```
2022-03-06 06:17:27  ERROR   OGG-00446  error 2 (No such file or directory)
opening redo log /rdsdbdata/db/GGTEST3_A/onlinelog/o1_mf_2_9k4bp1n6_.log for sequence
 1306
Not able to establish initial position for begin time 2022-03-06 06:16:55.
```

Étant donné que votre instance de base de données conserve vos journaux redo archivés, assurez-vous de disposer de suffisamment d'espace pour les fichiers. Pour voir combien d'espace vous avez utilisé au cours de la dernière *num_hours* heures, exécutez la requête suivante en remplaçant *num_hours* avec le nombre d'heures.

```
SELECT SUM(BLOCKS * BLOCK_SIZE) BYTES FROM V$ARCHIVED_LOG
   WHERE NEXT_TIME>=SYSDATE-num_hours/24 AND DEST_ID=1;
```

Étape 4 : créer un compte GoldenGate utilisateur Oracle dans la base de données source

Oracle GoldenGate fonctionne en tant qu'utilisateur de base de données et nécessite les privilèges de base de données appropriés pour accéder aux journaux de restauration et aux journaux de journalisation archivés de la base de données source. Pour cela, créez un compte d'utilisateur sur la base de données source. Pour plus d'informations sur les autorisations associées à un compte GoldenGate utilisateur Oracle, consultez la documentation Oracle.

Les instructions suivantes créent un compte d'utilisateur nommé `oggadm1`.

```
CREATE TABLESPACE administrator;
CREATE USER oggadm1  IDENTIFIED BY "password"
   DEFAULT TABLESPACE ADMINISTRATOR TEMPORARY TABLESPACE TEMP;
ALTER USER oggadm1 QUOTA UNLIMITED ON administrator;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Étape 5 : accorder des privilèges de compte d'utilisateur sur la base de données source

Dans cette tâche, vous accordez les privilèges de compte nécessaires aux utilisateurs de base de données sur votre base de données source.

Pour accorder des privilèges de compte sur la base de données source

1. Accordez les privilèges nécessaires au compte GoldenGate utilisateur Oracle à l'aide de la SQL commande `grant` et de la `rdsadmin.rdsadmin_util` procédure`grant_sys_object`. Les instructions suivantes accordent des privilèges à un utilisateur nommé `oggadm1`.

```
GRANT CREATE SESSION, ALTER SESSION TO oggadm1;
GRANT RESOURCE TO oggadm1;
GRANT SELECT ANY DICTIONARY TO oggadm1;
GRANT FLASHBACK ANY TABLE TO oggadm1;
GRANT SELECT ANY TABLE TO oggadm1;
GRANT SELECT_CATALOG_ROLE TO rds_master_user_name WITH ADMIN OPTION;
EXEC rdsadmin.rdsadmin_util.grant_sys_object ('DBA_CLUSTERS', 'OGGADM1');
GRANT EXECUTE ON DBMS_FLASHBACK TO oggadm1;
GRANT SELECT ON SYS.V_$DATABASE TO oggadm1;
GRANT ALTER ANY TABLE TO oggadm1;
```

2. Accordez les privilèges nécessaires à un compte utilisateur pour être un GoldenGate administrateur Oracle. Exécutez le SQL programme PL/ suivant.

```
EXEC rdsadmin.rdsadmin_dbms_goldengate_auth.grant_admin_privilege (
    grantee                 => 'OGGADM1',
    privilege_type          => 'capture',
    grant_select_privileges => true,
    do_grants               => TRUE);
```

   Pour révoquer des privilèges, utilisez la procédure `revoke_admin_privilege` dans le même package.

Étape 6 : ajouter un TNS alias pour la base de données source

Ajoutez l'entrée suivante à `$ORACLE_HOME/network/admin/tnsnames.ora` dans le répertoire de base de base de données Oracle que doit utiliser le processus `EXTRACT`. Pour de plus amples informations sur le fichier `tnsnames.ora`, veuillez consulter la documentation Oracle.

```
OGGSOURCE=
```

```
   (DESCRIPTION=
        (ENABLE=BROKEN)
        (ADDRESS_LIST=
            (ADDRESS=(PROTOCOL=TCP)(HOST=goldengate-source.abcdef12345.us-
west-2.rds.amazonaws.com)(PORT=8200)))
        (CONNECT_DATA=(SERVICE_NAME=ORCL))
    )
```

Configuration d'une base de données cible à utiliser avec Oracle GoldenGate sur Amazon RDS

Dans cette tâche, vous configurez une instance de base de données cible à utiliser avec Oracle GoldenGate.

Étapes de configuration

- Étape 1 : définissez le paramètre d'REPLICATIONinitialisation ENABLE GOLDENGATE _ _ sur true
- Étape 2 : créer un compte GoldenGate utilisateur Oracle sur la base de données cible
- Étape 3 : accorder des privilèges de compte sur la base de données cible
- Étape 4 : ajouter un TNS alias pour la base de données cible

Étape 1 : définissez le paramètre d'REPLICATIONinitialisation ENABLE GOLDENGATE _ _ sur true

Lorsque vous définissez le paramètre d'initialisation `ENABLE_GOLDENGATE_REPLICATION` sur `true`, il permet aux services de base de données de prendre en charge la réplication logique. Si votre base de données source se trouve sur une RDS instance de base de données Amazon, assurez-vous qu'un groupe de paramètres est attribué à l'instance de base de données avec le paramètre d'`ENABLE_GOLDENGATE_REPLICATIONinitialisation` défini sur. `true` Pour plus d'informations sur le paramètre d'initialisation `ENABLE_GOLDENGATE_REPLICATION`, consultez la documentation Oracle Database.

Étape 2 : créer un compte GoldenGate utilisateur Oracle sur la base de données cible

Oracle GoldenGate fonctionne en tant qu'utilisateur de base de données et nécessite les privilèges de base de données appropriés. Pour vous assurer qu'il possède ces privilèges, créez un compte d'utilisateur sur la base de données cible.

L'instruction suivante crée un utilisateur nommé `oggadm1`.

```
CREATE TABLESPSACE administrator;
CREATE USER oggadm1  IDENTIFIED BY "password"
```

```
   DEFAULT TABLESPACE administrator
   TEMPORARY TABLESPACE temp;
ALTER USER oggadm1 QUOTA UNLIMITED ON administrator;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Étape 3 : accorder des privilèges de compte sur la base de données cible

Dans cette tâche, vous accordez les privilèges de compte nécessaires aux utilisateurs de base de données sur votre base de données cible.

Pour accorder des privilèges de compte sur la base de données cible

1. Accordez les privilèges nécessaires au compte GoldenGate utilisateur Oracle sur la base de données cible. Dans l'exemple suivant, vous accordez des privilèges à `oggadm1`.

```
GRANT CREATE SESSION        TO oggadm1;
GRANT ALTER SESSION         TO oggadm1;
GRANT CREATE CLUSTER        TO oggadm1;
GRANT CREATE INDEXTYPE      TO oggadm1;
GRANT CREATE OPERATOR       TO oggadm1;
GRANT CREATE PROCEDURE      TO oggadm1;
GRANT CREATE SEQUENCE       TO oggadm1;
GRANT CREATE TABLE          TO oggadm1;
GRANT CREATE TRIGGER        TO oggadm1;
GRANT CREATE TYPE           TO oggadm1;
GRANT SELECT ANY DICTIONARY TO oggadm1;
GRANT CREATE ANY TABLE      TO oggadm1;
GRANT ALTER ANY TABLE       TO oggadm1;
GRANT LOCK ANY TABLE        TO oggadm1;
GRANT SELECT ANY TABLE      TO oggadm1;
GRANT INSERT ANY TABLE      TO oggadm1;
GRANT UPDATE ANY TABLE      TO oggadm1;
GRANT DELETE ANY TABLE      TO oggadm1;
```

2. Accordez les privilèges nécessaires à un compte utilisateur pour être un GoldenGate administrateur Oracle. Exécutez le SQL programme PL/ suivant.

```
EXEC rdsadmin.rdsadmin_dbms_goldengate_auth.grant_admin_privilege (
```

```
    grantee                 => 'OGGADM1',
    privilege_type          => 'apply',
    grant_select_privileges => true,
    do_grants               => TRUE);
```

Pour révoquer des privilèges, utilisez la procédure `revoke_admin_privilege` dans le même package.

Étape 4 : ajouter un TNS alias pour la base de données cible

Ajoutez l'entrée suivante à `$ORACLE_HOME/network/admin/tnsnames.ora` dans le répertoire de base de base de données Oracle que doit utiliser le processus `REPLICAT`. Pour les bases de données Oracle Multitenant, assurez-vous que l'TNSalias pointe vers le nom de service duPDB. Pour de plus amples informations sur le fichier `tnsnames.ora`, veuillez consulter la documentation Oracle.

```
OGGTARGET=
    (DESCRIPTION=
        (ENABLE=BROKEN)
        (ADDRESS_LIST=
            (ADDRESS=(PROTOCOL=TCP)(HOST=goldengate-target.abcdef12345.us-
west-2.rds.amazonaws.com)(PORT=8200)))
        (CONNECT_DATA=(SERVICE_NAME=ORCL))
    )
```

Utilisation des REPLICAT utilitaires EXTRACT et des utilitaires d'Oracle GoldenGate

Les GoldenGate utilitaires `EXTRACT` Oracle `REPLICAT` travaillent ensemble pour synchroniser les bases de données source et cible grâce à une réplication incrémentielle des transactions à l'aide de fichiers de suivi. Toutes les modifications apportées à la base de données source sont automatiquement détectées`EXTRACT`, puis formatées et transférées vers des fichiers de suivi sur le hub d'EC2instance Oracle sur GoldenGate site ou Amazon. Après le chargement initial, les données sont lues depuis ces fichiers et répliquées sur la base de données cible par l'utilitaire `REPLICAT`.

Exécution de l' GoldenGate EXTRACTutilitaire Oracle

L'utilitaire `EXTRACT` récupère et convertit des données provenant de la base de données source pour les fournir en sortie dans des fichiers de suivi. La procédure de base est la suivante :

1. EXTRACT met en file d'attente les détails de transaction pour les stocker dans la mémoire ou dans un stockage temporaire sur disque.
2. La base de données source valide la transaction.
3. EXTRACT écrit les détails de la transaction dans un fichier de suivi.
4. Le fichier de suivi achemine ces informations vers le hub d'EC2instance Oracle GoldenGate sur site ou Amazon, puis vers la base de données cible.

Les étapes suivantes démarrent l'utilitaire EXTRACT, capturent les données de EXAMPLE.TABLE dans la base de données source OGGSOURCE et créent les fichiers de suivi.

Pour exécuter l'EXTRACTutilitaire

1. Configurez le fichier de EXTRACT paramètres sur le GoldenGate hub Oracle (sur site ou EC2 instance Amazon). La liste suivante affiche un exemple de fichier de paramètres EXTRACT nommé $GGHOME/dirprm/eabc.prm.

```
EXTRACT EABC

USERID oggadm1@OGGSOURCE, PASSWORD "my-password"
EXTTRAIL /path/to/goldengate/dirdat/ab

IGNOREREPLICATES
GETAPPLOPS
TRANLOGOPTIONS EXCLUDEUSER OGGADM1

TABLE EXAMPLE.TABLE;
```

2. Sur le GoldenGate hub Oracle, connectez-vous à la base de données source et lancez l'interface de ligne de GoldenGate commande Oracleggsci. L'exemple suivant illustre le format pour la connexion.

```
dblogin oggadm1@OGGSOURCE
```

3. Ajoutez des données de transaction pour activer la journalisation supplémentaire pour la table de base de données.

```
add trandata EXAMPLE.TABLE
```

4. En utilisant la ligne de commande `ggsci`, activez l'utilitaire `EXTRACT` en utilisant les commandes suivantes.

```
add extract EABC tranlog, INTEGRATED tranlog, begin now
add exttrail /path/to/goldengate/dirdat/ab
   extract EABC,
   MEGABYTES 100
```

5. Enregistrez l'utilitaire `EXTRACT` avec la base de données afin que les journaux d'archivage ne soient pas supprimés. Cette tâche vous permet de récupérer d'anciennes transactions non validées, si nécessaire. Pour enregistrer l'utilitaire `EXTRACT` avec la base de données, utilisez la commande suivante.

```
register EXTRACT EABC, DATABASE
```

6. Démarrez l'utilitaire `EXTRACT` avec la commande suivante.

```
start EABC
```

Exécution de l' GoldenGate REPLICATutilitaire Oracle

L'utilitaire `REPLICAT` transmet (push) les informations de transaction des fichiers de suivi vers la base de données cible.

Les étapes suivantes permettent d'activer et de démarrer l'utilitaire `REPLICAT` afin qu'il puisse répliquer les données capturées dans la table `EXAMPLE.TABLE` de la base de données cible `OGGTARGET`.

Pour exécuter l'REPLICATEutilitaire

1. Configurez le fichier de `REPLICAT` paramètres sur le GoldenGate hub Oracle (sur site ou sur EC2 instance). La liste suivante affiche un exemple de fichier de paramètres `REPLICAT` nommé `$GGHOME/dirprm/rabc.prm`.

```
REPLICAT RABC

USERID oggadm1@OGGTARGET, password "my-password"

ASSUMETARGETDEFS
```

```
MAP EXAMPLE.TABLE, TARGET EXAMPLE.TABLE;
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

2. Connectez-vous à la base de données cible et lancez l'interface de ligne de GoldenGate commande Oracle (`ggsci`). L'exemple suivant illustre le format pour la connexion.

```
dblogin userid oggadm1@OGGTARGET
```

3. En utilisant la ligne de commande `ggsci`, ajoutez une table de points de vérification. L'utilisateur indiqué doit être le compte GoldenGate utilisateur Oracle, et non le propriétaire du schéma de table cible. L'exemple suivant crée une table de points de vérification nommée `gg_checkpoint`.

```
add checkpointtable oggadm1.oggchkpt
```

4. Pour activer l'utilitaire `REPLICAT`, utilisez la commande suivante :

```
add replicat RABC EXTTRAIL /path/to/goldengate/dirdat/ab CHECKPOINTTABLE
 oggadm1.oggchkpt
```

5. Démarrez l'utilitaire `REPLICAT` en utilisant la commande suivante :

```
start RABC
```

Surveillance d'Oracle GoldenGate

Lorsque vous utilisez Oracle GoldenGate pour la réplication, assurez-vous que le GoldenGate processus Oracle est en cours d'exécution et que les bases de données source et cible sont synchronisées. Vous pouvez utiliser les outils de surveillance suivants :

- Amazon CloudWatch est un service de surveillance utilisé dans ce modèle pour surveiller les journaux GoldenGate d'erreurs.
- Amazon SNS est un service de notification par message utilisé dans ce modèle pour envoyer des notifications par e-mail.

Pour obtenir des instructions détaillées, consultez Surveiller GoldenGate les journaux Oracle à l'aide d'Amazon CloudWatch.

Dépannage d'Oracle GoldenGate

Cette section décrit les problèmes les plus courants liés à l'utilisation d'Oracle GoldenGate avec Amazon RDS for Oracle.

Rubriques

- Erreur lors de l'ouverture d'un journal redo en ligne
- Oracle GoldenGate semble être correctement configuré mais la réplication ne fonctionne pas
- REPLICATLent intégré en raison de l'activation de la requêteSYS. » _ DBA _ _ APPLY _ CDR _ INFO »

Erreur lors de l'ouverture d'un journal redo en ligne

Veillez à configurer vos bases de données pour conserver les journaux redo archivés. Considérez les directives suivantes :

- Spécifiez la durée de conservation des journaux en heures. La valeur minimale est d'une heure.
- Définissez cette durée de manière à ce qu'elle dépasse tout temps d'arrêt éventuel de l'instance de base de données source ou la durée de tout problème potentiel de communication ou de mise en réseau pour l'instance de base de données source. Une telle durée permet à Oracle de GoldenGate récupérer les journaux de l'instance de base de données source selon les besoins.
- Assurez-vous de disposer d'un espace de stockage suffisant sur votre instance pour les fichiers.

Si la conservation des journaux n'est pas activée ou si la valeur de conservation est trop faible, vous recevez un message d'erreur similaire au suivant.

```
2022-03-06 06:17:27  ERROR   OGG-00446  error 2 (No such file or directory)
opening redo log /rdsdbdata/db/GGTEST3_A/onlinelog/o1_mf_2_9k4bp1n6_.log for sequence
 1306
Not able to establish initial position for begin time 2022-03-06 06:16:55.
```

Oracle GoldenGate semble être correctement configuré mais la réplication ne fonctionne pas

Pour les tables préexistantes, vous devez spécifier SCN celles à partir desquelles Oracle GoldenGate travaille.

Pour résoudre ce problème

1. Connectez-vous à la base de données source et lancez l'interface de ligne de GoldenGate commande Oracle (`ggsci`). L'exemple suivant illustre le format pour la connexion.

```
dblogin userid oggadm1@OGGSOURCE
```

2. À l'aide de la ligne de `ggsci` commande, configurez SCN le démarrage du `EXTRACT` processus. L'exemple suivant définit la SCN valeur 223274 pour le`EXTRACT`.

```
ALTER EXTRACT EABC SCN 223274
start EABC
```

3. Connectez-vous à la base de données cible. L'exemple suivant illustre le format pour la connexion.

```
dblogin userid oggadm1@OGGTARGET
```

4. À l'aide de la ligne de `ggsci` commande, configurez SCN le démarrage du `REPLICAT` processus. L'exemple suivant définit la SCN valeur 223274 pour le`REPLICAT`.

```
start RABC atcsn 223274
```

REPLICATLent intégré en raison de l'activation de la requêteSYS. » _ DBA _ _ APPLY _ CDR _ INFO »

Oracle GoldenGate Conflict Detection and Resolution (CDR) fournit des routines de résolution de conflits de base. Par exemple, CDR peut résoudre un conflit unique pour une `INSERT` instruction.

Lorsqu'CDRune collision est résolue, elle peut insérer `_DBA_APPLY_CDR_INFO` temporairement des enregistrements dans la table des exceptions. Le processus `REPLICAT` intégré supprime ces enregistrements par la suite. Il existe un scénario rare dans lequel le processus `REPLICAT` intégré peut traiter un grand nombre de collisions, mais aucun nouveau processus `REPLICAT` intégré ne le remplace. Au lieu d'être supprimées, les lignes existantes de `_DBA_APPLY_CDR_INFO` sont orphelines. Tous les nouveaux processus `REPLICAT` intégrés ralentissent car ils interrogent des lignes orphelines dans `_DBA_APPLY_CDR_INFO`.

Pour supprimer toutes les lignes de`_DBA_APPLY_CDR_INFO`, utilisez la RDS procédure Amazon`rdsadmin.rdsadmin_util.truncate_apply$_cdr_info`. Cette procédure est publiée

dans le cadre de la version et de la mise à jour d'octobre 2020. La procédure est disponible dans les versions suivantes des bases de données :

- Version 21,0.0.0.ru-2022-01.rur-2022-01.r1 et versions ultérieures
- Version 19,0.0.0.ru-2020-10.rur-2020-10.r1 et versions ultérieures

L'exemple suivant tronque la table _DBA_APPLY_CDR_INFO.

```
SET SERVEROUTPUT ON SIZE 2000
EXEC rdsadmin.rdsadmin_util.truncate_apply$_cdr_info;
```

Utilisation de l'utilitaire Oracle Repository Creation Utility sur RDS for Oracle

Vous pouvez utiliser Amazon RDS pour héberger une instance de base de données RDS for Oracle qui contient les schémas pour prendre en charge vos composants Oracle Fusion Middleware. Avant de pouvoir utiliser les composants Fusion Middleware, créez et remplissez les schémas pour eux dans votre base de données. Vous créez et remplissez les schémas à l'aide de l'utilitaire Oracle Repository Creation Utility (RCU).

Versions prises en charge et options de licence pour RCU

Amazon RDS prend en charge l'utilitaire Oracle Repository Creation Utility (RCU) version 12c seulement. Vous pouvez utiliser le RCU dans les configurations suivantes :

- Chiffrement RCU 12c avec Oracle Database 21c
- Chiffrement RCU 12c avec Oracle Database 19c

Avant de pouvoir utiliser le RCU, assurez-vous d'effectuer les opérations suivantes :

- Obtenez une licence pour Oracle Fusion Middleware.
- Suivez les consignes sur les licences d'Oracle pour la base de données Oracle qui héberge le référentiel. Pour de plus amples informations, veuillez consulter le Manuel d'utilisateur sur les informations de licence Oracle Fusion Middleware dans la documentation Oracle.

Fusion MiddleWare prend en charge les référentiels sur Oracle Database Enterprise Edition et Standard Edition 2. Oracle recommande l'Enterprise Edition pour les installations de production nécessitant un partitionnement et les installations nécessitant une reconstruction d'index en ligne.

Avant de créer votre instance RDS for Oracle, confirmez que la version de la base de données Oracle dont vous avez besoin pour prendre en charge les composants que vous souhaitez déployer. Pour connaître les exigences relatives aux composants et aux versions de Fusion Middleware que vous souhaitez déployer, utilisez la matrice de certification. Pour de plus amples informations, veuillez consulter la page Configurations système prises en charge par Oracle Fusion Middleware dans la documentation Oracle.

Amazon RDS prend en charge les mises à niveau de version de la base de données Oracle en fonction des besoins. Pour plus d’informations, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Exigences et restrictions pour RCU

Pour utiliser RCU, vous avez besoin d'un VPC Amazon. Votre instance de base de données Amazon RDS doit être disponible seulement pour vos composants Fusion Middleware et non pour l'Internet public. Par conséquent, hébergez votre instance de base de données Amazon RDS dans un sous-réseau privé, offrant une sécurité supérieure. Vous avez également besoin d'une instance de base de données RDS for Oracle. Pour plus d'informations, consultez Création et connexion à une instance de base de données Oracle.

Vous pouvez stocker les schémas de tous les composants Fusion Middleware dans votre instance de base de données Amazon RDS. Les schémas suivants ont été vérifiés pour une installation correcte :

- Analytics (ACTIVITIES)
- Audit Services (IAU)
- Audit Services Append (IAU_APPEND)
- Audit Services Viewer (IAU_VIEWER)
- Discussions (DISCUSSIONS)
- Metadata Services (MDS)
- Oracle Business Intelligence (BIPLATFORM)
- Oracle Platform Security Services (OPSS)
- Portal and Services (WEBCENTER)
- Portlet Producers (PORTLET)
- Service Table (STB)
- SOA Infrastructure (SOAINFRA)
- User Messaging Service (UCSUMS)
- WebLogic Services (WLS)

Conseils pour l'exécution de RCU

Voici quelques recommandations pour l'utilisation de votre instance de base de données dans ce scénario :

- Nous recommandons d'utiliser Multi-AZ pour les charges de travail de production. Pour plus d'informations sur l'utilisation de plusieurs zones de disponibilité, consultez Régions, zones de disponibilité et zones locales.

- Pour une sécurité optimale, Oracle recommande que vous utilisiez Transparent Data Encryption (TDE) pour chiffrer les données au repos. Si vous avez une licence Enterprise Edition qui inclut l'option de sécurité avancée, vous pouvez activer le chiffrement au repos à l'aide de l'option TDE. Pour plus d'informations, consultez Oracle Transparent Data Encryption.

  Amazon RDS fournit également une option de chiffrement au repos pour toutes les éditions de la base de données. Pour plus d'informations, consultez Chiffrement des ressources Amazon RDS.
- Configurez vos groupes de sécurité VPC pour permettre la communication entre vos serveurs d'applications et votre instance de base de données Amazon RDS. Les serveurs d'applications qui hébergent les composants Fusion Middleware peuvent être sur Amazon EC2 ou sur site.

Exécution de RCU

Utilisez l'utilitaire Oracle Repository Creation Utility (RCU) pour créer et remplir les schémas afin de prendre en charge vos composants Fusion Middleware. Vous pouvez exécuter RCU de différentes façons.

Rubriques

- Exécution de RCU à l'aide de la ligne de commande en une seule étape
- Exécution de RCU dans la ligne de commande en plusieurs étapes
- Exécution de RCU en mode interactif

Exécution de RCU à l'aide de la ligne de commande en une seule étape

Si vous n'avez besoin de modifier aucun de vos schémas avant de les remplir, vous pouvez exécuter RCU en une seule étape. Sinon, consultez la section suivante pour exécuter RCU en plusieurs étapes.

Vous pouvez exécuter le RCU en mode silencieux à l'aide du paramètre de ligne de commande `-silent`. Lorsque vous exécutez RCU en mode silencieux, vous pouvez éviter de saisir des mots de passe sur la ligne de commande en créant un fichier texte contenant les mots de passe. Créez un fichier texte avec le mot de passe pour `dbUser` sur la première ligne et le mot de passe de chaque composant sur les lignes suivantes. Vous spécifiez le nom du fichier mot de passe en tant que dernier paramètre de la commande RCU.

Example

L'exemple suivant permet de créer et remplir les schémas pour le composant d'infrastructure SOA (et ses dépendances) en une seule étape.

Pour LinuxmacOS, ou Unix :

```
export ORACLE_HOME=/u01/app/oracle/product/12.2.1.0/fmw
export JAVA_HOME=/usr/java/jdk1.8.0_65
${ORACLE_HOME}/oracle_common/bin/rcu \
-silent \
-createRepository \
-connectString ${dbhost}:${dbport}:${dbname} \
-dbUser ${dbuser} \
-dbRole Normal \
-honorOMF \
-schemaPrefix ${SCHEMA_PREFIX} \
-component MDS \
-component STB \
-component OPSS \
-component IAU \
-component IAU_APPEND \
-component IAU_VIEWER \
-component UCSUMS \
-component WLS \
-component SOAINFRA \
-f < /tmp/passwordfile.txt
```

Pour de plus amples informations, veuillez consulter la page Exécution de l'utilitaire RCU dans la ligne de commande dans la documentation Oracle.

Exécution de RCU dans la ligne de commande en plusieurs étapes

Afin de modifier manuellement vos scripts de schéma, exécutez RCU en plusieurs étapes :

1. Exécutez RCU en mode Prepare Scripts for System Load (Préparer les scripts pour le chargement système) à l'aide du paramètre de ligne de commande `-generateScript` pour créer les scripts pour vos schémas.
2. Modifiez et exécutez manuellement le script généré `script_systemLoad.sql`.
3. Exécutez RCU à nouveau en mode Perform Product Load (Exécuter la charge de produit) à l'aide du paramètre de ligne de commande `-dataLoad` pour remplir les schémas.

4. Exécutez le script de nettoyage généré `script_postDataLoad.sql`.

Pour exécuter le RCU en mode silencieux, spécifiez le paramètre de ligne de commande `-silent`. Lorsque vous exécutez le RCU en mode silencieux, cela vous évite de taper des mots de passe sur la ligne de commande en créant un fichier texte contenant les mots de passe. Créez un fichier texte avec le mot de passe pour `dbUser` sur la première ligne et le mot de passe de chaque composant sur les lignes suivantes. Spécifiez le nom du fichier mot de passe en tant que dernier paramètre de la commande RCU.

Example

L'exemple suivant permet de créer des scripts de schéma pour le composant d'infrastructure SOA et ses dépendances.

Pour LinuxmacOS, ou Unix :

```
export ORACLE_HOME=/u01/app/oracle/product/12.2.1.0/fmw
export JAVA_HOME=/usr/java/jdk1.8.0_65
${ORACLE_HOME}/oracle_common/bin/rcu \
-silent \
-generateScript \
-connectString ${dbhost}:${dbport}:${dbname} \
-dbUser ${dbuser} \
-dbRole Normal \
-honorOMF \
[-encryptTablespace true] \
-schemaPrefix ${SCHEMA_PREFIX} \
-component MDS \
-component STB \
-component OPSS \
-component IAU \
-component IAU_APPEND \
-component IAU_VIEWER \
-component UCSUMS \
-component WLS \
-component SOAINFRA \
-scriptLocation /tmp/rcuscripts \
-f < /tmp/passwordfile.txt
```

Désormais, vous pouvez modifier le script généré, vous connecter à votre instance de base de données Oracle et exécuter le script. Le script généré est appelé `script_systemLoad.sql`. Pour

plus d'informations sur la connexion à votre instance de base de données Oracle, consultez Étape 3 : Connecter votre client SQL à une instance de base de données Oracle.

L'exemple suivant permet de remplir les schémas pour le composant d'infrastructure SOA (et ses dépendances).

Pour LinuxmacOS, ou Unix :

```
export JAVA_HOME=/usr/java/jdk1.8.0_65
${ORACLE_HOME}/oracle_common/bin/rcu \
-silent \
-dataLoad \
-connectString ${dbhost}:${dbport}:${dbname} \
-dbUser ${dbuser} \
-dbRole Normal \
-honorOMF \
-schemaPrefix ${SCHEMA_PREFIX} \
-component MDS \
-component STB \
-component OPSS \
-component IAU \
-component IAU_APPEND \
-component IAU_VIEWER \
-component UCSUMS \
-component WLS \
-component SOAINFRA \
-f < /tmp/passwordfile.txt
```

Pour finir, vous vous connectez à votre instance de base de données Oracle et vous exécutez le script de nettoyage. Ce script est appelé `script_postDataLoad.sql`.

Pour de plus amples informations, veuillez consulter la page Exécution de l'utilitaire RCU dans la ligne de commande dans la documentation Oracle.

Exécution de RCU en mode interactif

Pour utiliser l'interface utilisateur graphique de RCU, exécutez RCU en mode interactif. Incluez le paramètre `-interactive` et omettez le paramètre `-silent`. Pour de plus amples informations, veuillez consulter la page Présentation des écrans de l'utilitaire RCU dans la documentation Oracle.

Example

L'exemple suivant démarre le RCU en mode interactif et pré-remplit les informations de connexion.

Pour LinuxmacOS, ou Unix :

```
export ORACLE_HOME=/u01/app/oracle/product/12.2.1.0/fmw
export JAVA_HOME=/usr/java/jdk1.8.0_65
${ORACLE_HOME}/oracle_common/bin/rcu \
-interactive \
-createRepository \
-connectString ${dbhost}:${dbport}:${dbname} \
-dbUser ${dbuser} \
-dbRole Normal
```

Résolution des problèmes liés à RCU

Soyez attentifs aux points suivants.

Rubriques

- Fichiers gérés Oracle (OMF)
- Privilèges d'objet
- Service de planification d'entreprise

Fichiers gérés Oracle (OMF)

Amazon RDS utilise des fichiers de données OMF pour simplifier la gestion du stockage. Vous pouvez personnaliser les attributs de table, telles que la taille et la gestion de l'extension. Cependant, si vous spécifiez un nom de fichier de données lorsque vous exécutez la RCU, le code d'espace disque logique échoue en renvoyant l'erreur `ORA-20900`. Vous pouvez utiliser le RCU avec OMF de la manière suivante :

- Dans RCU 12.2.1.0 et version ultérieure, utilisez le paramètre de ligne de commande `-honorOMF`.
- Dans RCU 12.1.0.3 et plus tard, utilisez plusieurs étapes et modifier le script généré. Pour plus d'informations, consultez Exécution de RCU dans la ligne de commande en plusieurs étapes.

Privilèges d'objet

Amazon RDS étant un service géré, vous n'avez pas un accès `SYSDBA` complet à votre instance de base de données RDS for Oracle. Cependant, RCU 12c prend en charge les utilisateurs avec des privilèges inférieurs. Dans la plupart des cas, le privilège de l'utilisateur principal est suffisant pour créer des référentiels.

Le compte maître peut accorder directement les privilèges qui lui ont déjà été accordés par `WITH GRANT OPTION`. Dans certains cas, lorsque vous tentez d'accorder des privilèges d'objet `SYS`, le RCU peut échouer et renvoyer l'erreur `ORA-01031`. Vous pouvez réessayer et exécuter la procédure stockée `rdsadmin_util.grant_sys_object`, comme le montre l'exemple suivant :

```
BEGIN
  rdsadmin.rdsadmin_util.grant_sys_object('GV_$SESSION','MY_DBA','SELECT');
END;
/
```

Si vous tentez d'accorder des privilèges `SYS` sur l'objet `SCHEMA_VERSION_REGISTRY`, l'opération peut échouer et renvoyer l'erreur `ORA-20199: Error in rdsadmin_util.grant_sys_object`. Vous pouvez qualifier la table `SCHEMA_VERSION_REGISTRY$` et la vue `SCHEMA_VERSION_REGISTRY` avec le nom du propriétaire du schéma, qui est `SYSTEM`, puis réessayer l'opération. Ou bien, vous pouvez créer un synonyme. Connectez-vous en tant qu'utilisateur maître et exécutez les instructions suivantes :

```
CREATE OR REPLACE VIEW SYSTEM.SCHEMA_VERSION_REGISTRY
  AS SELECT * FROM SYSTEM.SCHEMA_VERSION_REGISTRY$;
CREATE OR REPLACE PUBLIC SYNONYM SCHEMA_VERSION_REGISTRY FOR
 SYSTEM.SCHEMA_VERSION_REGISTRY;
CREATE OR REPLACE PUBLIC SYNONYM SCHEMA_VERSION_REGISTRY$ FOR SCHEMA_VERSION_REGISTRY;
```

Service de planification d'entreprise

Lorsque vous utilisez le RCU pour transférer un référentiel de service de planificateur d'entreprise, le RCU peut échouer et renvoyer l'erreur `Error: Component drop check failed`.

Configuration d'Oracle Connection Manager sur une instance Amazon EC2

Oracle Connection Manager (CMAN) est un serveur proxy qui transfère les requêtes de connexion à des serveurs de base de données ou à d'autres serveurs proxy. Vous pouvez utiliser CMAN pour configurer les éléments suivants :

Contrôle d'accès

Vous pouvez créer des règles qui filtrent les requêtes client spécifiées par l'utilisateur et en acceptent d'autres.

Multiplexage de session

Vous pouvez acheminer plusieurs sessions client via une connexion réseau vers une destination de serveur partagé.

En règle générale, CMAN réside sur un hôte distinct du serveur de base de données et des hôtes clients. Pour de plus amples informations, veuillez consulter Configuring Oracle Connection Manager dans la documentation Oracle Database.

Rubriques

- Versions prises en charge et options de licence pour CMAN
- Exigences et restrictions pour CMAN
- Configuration de CMAN

Versions prises en charge et options de licence pour CMAN

CMAN prend en charge l'édition Enterprise de toutes les versions d'Oracle Database prises en charge par Amazon RDS. Pour plus d'informations, consultez Versions RDS for Oracle.

Vous pouvez installer Oracle Connection Manager sur un hôte distinct de l'hôte sur lequel Oracle Database est installé. Vous n'avez pas besoin d'une licence distincte pour l'hôte qui exécute CMAN.

Exigences et restrictions pour CMAN

Pour offrir une expérience entièrement gérée, Amazon RDS restreint l'accès au système d'exploitation. Vous ne pouvez pas modifier les paramètres de base de données nécessitant un accès au système d'exploitation. Par conséquent, Amazon RDS ne prend pas en charge les fonctions de CMAN qui exigent que vous vous connectiez au système d'exploitation.

Configuration de CMAN

Lorsque vous configurez CMAN, vous effectuez la majeure partie du travail en dehors de votre base de données RDS for Oracle.

Rubriques

- Étape 1 : configurer CMAN sur une instance Amazon EC2 dans le même VPC que l'instance RDS for Oracle
- Étape 2 : configurer les paramètres de base de données pour CMAN
- Étape 3 : associer le groupe de paramètres à l'instance de base de données

Étape 1 : configurer CMAN sur une instance Amazon EC2 dans le même VPC que l'instance RDS for Oracle

Pour savoir comment configurer CMAN, suivez les instructions détaillées dans l'article de blog Configuring and using Oracle Connection Manager on Amazon EC2 for Amazon RDS for Oracle.

Étape 2 : configurer les paramètres de base de données pour CMAN

Pour les fonctions CMAN telles que le mode Traffic Director et le multiplexage de session, définissezle paramètre `REMOTE_LISTENER` à l'adresse de l'instance CMAN dans un groupe de paramètres de base de données. Réfléchissez au scénario suivant :

- L'instance CMAN réside sur un hôte doté d'une adresse IP `10.0.159.100` et utilise le port `1521`.
- Les bases de données `orcla`, `orclb` et `orclc` se trouvent sur des instances de bases de données RDS for Oracle.

Le tableau suivant montre la façon de définir la valeur `REMOTE_LISTENER`. La valeur `LOCAL_LISTENER` est automatiquement définie par Amazon RDS.

Nom de l'instance de base de données	IP d'instance de base de données	Valeur de l'écouteur local (définie automatiquement)	Valeur de l'écouteur distant (définie par l'utilisateur)
`orcla`	`10.0.159.200`	`( address= (protocol=tcp)`	`10.0.159.100:1521`

Nom de l'instance de base de données	IP d'instance de base de données	Valeur de l'écouteur local (définie automatiquement)	Valeur de l'écouteur distant (définie par l'utilisateur)
		(host=10.0.159.200) (port=1521))	
orclb	10.0.159.300	(address= (protocol=tcp) (host=10.0.159.300) (port=1521))	10.0.159.100:1521
orclc	10.0.159.400	(address= (protocol=tcp) (host=10.0.159.400) (port=1521))	10.0.159.100:1521

Étape 3 : associer le groupe de paramètres à l'instance de base de données

Créez ou modifiez votre instance de base de données pour utiliser le groupe de paramètres que vous avez configuré dans Étape 2 : configurer les paramètres de base de données pour CMAN. Pour plus d'informations, consultez Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon.

Installation d'une base de données Siebel sur Oracle sur Amazon RDS

Vous pouvez utiliser Amazon RDS pour héberger une base de données Siebel sur une instance de base de données Oracle. La base de données Siebel fait partie de l'architecture d'application Siebel Customer Relationship Management (CRM). Pour une illustration, consultez Architecture générique de Siebel Business Application.

Reportez-vous à la rubrique suivante pour configurer une base de données Siebel sur une instance de base de données Oracle sur Amazon RDS. Vous pouvez également trouver comment utiliser Amazon Web Services pour prendre en charge les autres composants requis par l'architecture d'application Siebel CRM.

> **Note**
>
> Pour installer une base de données Siebel sur Oracle sur Amazon RDS, vous devez utiliser le compte utilisateur principal. Vous n'avez pas besoin de privilège `SYSDBA` ; le privilège de l'utilisateur principal est suffisant. Pour plus d'informations, consultez Privilèges du compte utilisateur principal.

Licence et Versions

Pour installer une base de données Siebel sur Amazon RDS, vous devez utiliser votre propre licence d'Oracle Database et votre propre licence Siebel. Vous devez posséder la licence d'Oracle Database appropriée (avec la licence de mise à jour du logiciel et le support) pour la classe d'instance de base de données et l'édition d'Oracle Database que vous souhaitez exécuter. Pour plus d'informations, consultez RDSpour les options de licence Oracle.

Oracle Database Enterprise Edition est la seule édition certifiée par Siebel pour ce scénario. Amazon RDS prend en charge les versions 15.0 ou 16.0 de Siebel CRM.

Amazon RDS prend en charge les mises à niveau de la version de la base de données. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'une instance de base de données.

Avant de commencer

Avant de commencer, vous avez besoin d'une instance Amazon VPC. Étant donné que votre instance de base de données Amazon RDS doit être disponible uniquement pour votre serveur d'entreprise Siebel et pas pour le réseau Internet public, votre instance de base de données Amazon

RDS est hébergée dans un sous-réseau privé, fournissant une plus grande sécurité. Pour plus d'informations sur la création d'une instance Amazon VPC à utiliser avec Siebel CRM, consultez Création et connexion à une instance de base de données Oracle.

Avant de commencer, vous avez également besoin d'une instance de base de données Oracle. Pour plus d'informations sur la création d'une instance de base de données Oracle à utiliser avec Siebel CRM, consultez Création d'une instance de base de données Amazon RDS.

Installation et configuration d'une base de données Siebel

Après avoir créé votre instance de base de données Oracle, vous pouvez installer votre base de données Siebel. Vous installez la base de données en créant des comptes de propriétaire et d'administrateur de table, en installant les fonctions et procédures stockées, puis en exécutant l'Assistant de configuration de base de données Siebel. Pour plus d'informations, consultez la page Installation de la base de données Siebel sur le SGBDR.

Pour exécuter l'Assistant de configuration de base de données Siebel, vous devez utiliser le compte utilisateur principal. Vous n'avez pas besoin de privilège `SYSDBA` ; le privilège de l'utilisateur principal est suffisant. Pour plus d'informations, consultez Privilèges du compte utilisateur principal.

Utilisation d'autres fonctions Amazon RDS avec une base de données Siebel

Après avoir créé votre instance de base de données Oracle, vous pouvez utiliser d'autres fonctions Amazon RDS pour vous aider à personnaliser votre base de données Siebel.

Collecte de statistiques avec l'Option Oracle Statspack

Vous pouvez ajouter des fonctions à votre instance de base de données grâce à l'utilisation d'options dans les groupes d'options de base de données. Lorsque vous avez créé votre instance de base de données Oracle, vous avez utilisé le groupe d'options de base de données par défaut. Si vous souhaitez ajouter des fonctions à votre base de données, vous pouvez créer un nouveau groupe d'options pour votre instance de base de données.

Si vous souhaitez collecter des statistiques de performances sur votre base de données Siebel, vous pouvez ajouter la fonction Oracle Statspack. Pour plus d'informations, consultez Oracle Statspack.

Certaines modifications des options sont appliquées immédiatement, et certaines modifications des options sont appliquées pendant le créneau de maintenance suivant pour l'instance de base de données. Pour plus d'informations, consultez Utilisation de groupes d'options. Une fois que vous

créez un groupe d'options personnalisé, modifiez votre instance de base de données pour l'attacher. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Réglage des performances avec les paramètres

Vous gérez la configuration de votre moteur de base de données à l'aide de paramètres dans un groupe de paramètres de base de données. Lorsque vous avez créé votre instance de base de données Oracle, vous avez utilisé le groupe de paramètres de base de données par défaut. Si vous souhaitez personnaliser votre configuration de base de données, vous pouvez créer un nouveau groupe de paramètres pour votre instance de base de données.

Lorsque vous modifiez un paramètre, en fonction du type du paramètre, les modifications sont appliquées immédiatement ou après le redémarrage manuel de l'instance de base de données. Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS). Une fois que vous créez un groupe de paramètres personnalisé, modifiez votre instance de base de données pour l'attacher. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.

Afin d'optimiser votre instance de base de données Oracle pour Siebel CRM, vous pouvez personnaliser certains paramètres. Le tableau suivant illustre certains des paramètres recommandés. Pour plus d'informations sur le réglage des performances de Siebel CRM, consultez Guide de réglage des performances de Siebel CRM.

Nom du paramètre	Valeur par défaut	Recommandations pour obtenir des performances optimales avec Siebel CRM
`_always_semi_join`	`CHOOSE`	`OFF`
`_b_tree_bitmap_plans`	`TRUE`	`FALSE`
`_like_with_bind_as_equality`	`FALSE`	`TRUE`
`_no_or_expansion`	`FALSE`	`FALSE`

Nom du paramètre	Valeur par défaut	Recommandations pour obtenir des performances optimales avec Siebel CRM
`_optimizer_join_sel_sanity_check`	`TRUE`	`TRUE`
`_optimizer_max_permutations`	2000	100
`_optimizer_sortmerge_join_enabled`	`TRUE`	`FALSE`
`_partition_view_enabled`	`TRUE`	`FALSE`
`open_cursors`	`300`	Au moin **`2000`**.

Création d'instantanés

Une fois que vous créez votre base de données Siebel, vous pouvez copier la base de données en utilisant les fonctions de l'instantané de Amazon RDS. Pour de plus amples informations, veuillez consulter Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS et Restauration vers une instance de base de données.

Prise en charge d'autres composants Siebel CRM

En plus de votre base de données Siebel, vous pouvez également utiliser Amazon Web Services pour prendre en charge les autres composants de votre architecture d'application Siebel CRM. Vous trouverez plus d'informations sur la prise en charge fournie par Amazon AWS pour les composants Siebel CRM supplémentaires dans le tableau suivant.

Composant Siebel CRM	AWS Support Amazon
Siebel Enterprise (avec un ou plusieurs serveurs Siebel)	Vous pouvez héberger vos serveurs Siebel sur les instances Amazon Elastic Compute Cloud (Amazon EC2). Vous pouvez utiliser Amazon EC2 pour lancer autant de serveurs virtuels que vous avez besoin. A l'aide de Amazon EC2, vous pouvez effectuer facilement des mises à l'échelle ascendantes et descendantes pour gérer les modifications apportées aux conditions. Pour de plus amples informations, veuillez consulter Qu'est-ce qu'Amazon EC2 ? Vous pouvez mettre vos serveurs dans le même VPC avec votre instance de base de données et utiliser le groupe de sécurité VPC pour accéder à la base de données. Pour plus d'informations, consultez Utilisation d'un(e) instance de base de données dans un VPC.
Serveurs Web (avec les extensions de serveur Web Siebel)	Vous pouvez installer plusieurs serveurs Web sur plusieurs instances EC2. Vous pouvez ensuite utiliser Elastic Load Balancing pour répartir le trafic entrant entre les instances. Pour de plus amples informations, veuillez consulter la documentation relative à la présentation d'Elastic Load Balancing.
Nom du serveur Siebel Gateway	Vous pouvez héberger votre serveur de nom Siebel Gateway sur une instance EC2. Vous pouvez ensuite mettre votre serveur dans le même VPC avec l'instance de base de données et utiliser le groupe de sécurité VPC pour accéder à la base de données. Pour plus d'informa tions, consultez Utilisation d'un(e) instance de base de données dans un VPC.

Notes de mise à jour pour le moteur de base de données Oracle

Des mises à jour apportées à vos instances de bases de données Amazon RDS for Oracle assurent que les instances restent à jour. Si vous appliquez des mises à jour, vous avez la garantie que votre instance de base de données exécute une version du logiciel de base de données testée par Oracle et par Amazon. Nous ne prenons pas en charge l'application de correctifs ponctuels aux instances de base de données RDS for Oracle individuelles.

Vous pouvez spécifier toute version de base de données Oracle actuellement prise en charge lorsque vous créez une nouvelle instance de base de données. Vous pouvez spécifier la version majeure, comme Oracle Database 19c, puis toute version mineure prise en charge pour la version majeure spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version prise en charge, généralement la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée. Pour afficher la liste des versions prises en charge et des versions par défaut pour des instances de base de données nouvellement créées, utilisez la commande `describe-db-engine-versions` de l'AWS CLI.

Pour obtenir plus de détails sur les versions d'Oracle Database prises en charge par Amazon RDS, consultez les Amazon RDS for Oracle Release Notes (Notes de mise à jour d'Amazon RDS for Oracle).

Amazon RDS for PostgreSQL

Amazon RDS prend en charge les instances de bases de données qui exécutent plusieurs versions de PostgreSQL. Pour obtenir la liste des versions disponibles, veuillez consulter Versions de base de données PostgreSQL disponibles.

Vous pouvez créer des instances de base de données et des instantanés de base de données, des point-in-time restaurations et des sauvegardes. Les instances de base de données qui exécutent PostgreSQL prennent en charge les déploiements Multi-AZ, les réplicas en lecture et les IOPS provisionnés, et peuvent être créées au sein d'un cloud privé virtuel (VPC). Vous pouvez également utiliser le protocole SSL pour vous connecter à une instance de base de données exécutant PostgreSQL.

Avant de créer une instance de base de données, assurez-vous d'avoir effectué les étapes de la section Configuration de votre environnement Amazon RDS.

Vous pouvez utiliser une application cliente SQL standard quelconque pour exécuter les commandes pour l'instance à partir de votre ordinateur client. De telles applications incluent pgAdmin, un outil d'administration et de développement Open Source fréquemment utilisé pour PostgreSQL, ou psql, un utilitaire de ligne de commande inclus dans une installation PostgreSQL. Pour offrir une expérience de service géré, Amazon RDS ne fournit pas l'accès hôte aux instances de base de données. Il restreint également l'accès à certaines procédures système et tables qui requièrent des privilèges avancés. Amazon RDS prend en charge l'accès aux bases de données sur une instance de base de données en utilisant toute application cliente SQL standard. Amazon RDS ne permet pas d'accès de l'hôte direct à une instance de base de données via Telnet ou Secure Shell (SSH).

Amazon RDS for PostgreSQL est conforme à de nombreuses normes du secteur. Par exemple, vous pouvez utiliser des bases de données Amazon RDS for PostgreSQL afin de développer des applications conformes à la loi HIPAA et de stocker les informations relatives aux soins de santé. Cela inclut le stockage des informations de santé protégées (PHI, Protected Health Information) selon les termes d'un Accord d'association commerciale (BAA, Business Associate Agreement) conclu avec AWS. Amazon RDS for PostgreSQL respecte également les exigences de sécurité du Programme fédéral de gestion des risques et des autorisations (FedRAMP) Amazon RDS for PostgreSQL a reçu l'autorisation provisoire d'exploitation (P-ATO) du FedRAMP Joint Authorization Board (JAB) sur la base de référence FedRAMP HIGH au sein des régions. AWS GovCloud (US) Pour plus d'informations sur les normes de conformité prises en charge, veuillez consulter Conformité du Cloud AWS.

Pour importer des données PostgreSQL dans une instance de base de données, suivez les informations fournies dans la section Importer des données dans Postgre SQL sur Amazon RDS.

> **Important**
>
> Si vous rencontrez un problème avec votre instance de base de données RDS pour PostgreSQL, AWS votre agent de support peut avoir besoin de plus d'informations sur l'état de vos bases de données. L'objectif est de faire en sorte que le AWS Support obtienne les informations requises le plus rapidement possible.
> Vous pouvez utiliser PG Collector pour rassembler de précieuses informations de base de données dans un fichier HTML consolidé. Pour plus d'informations sur PG Collector, comment l'exécuter et comment télécharger le rapport HTML, consultez PG Collector.
> Une fois terminé avec succès, et sauf indication contraire, le script renvoie le résultat dans un format HTML lisible. Le script est conçu pour exclure du code HTML toute donnée ou tout détail de sécurité susceptible de compromettre votre activité. De plus, il n'apporte aucune modification à la base de données ou à son environnement. Toutefois, si vous trouvez dans le code HTML des informations que vous n'êtes pas à l'aise de partager, n'hésitez pas à supprimer les informations problématiques avant de télécharger le code HTML. Lorsque le code HTML est acceptable, téléchargez-le à l'aide de la section des pièces jointes dans les détails de votre dossier d'assistance.

Rubriques

- Tâches de gestion courantes pour Amazon RDS pour Postgre SQL
- Utilisation de l'environnement de prévisualisation de base de données
- Versions de base de données PostgreSQL disponibles
- Comprendre le processus de publication incrémentielle de RDS pour PostgreSQL
- Versions d'SQLextension Postgre prises en charge
- Utilisation des SQL fonctionnalités de Postgre prises en charge par Amazon RDS pour Postgre SQL
- Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL
- Sécuriser les connexions RDS à Postgre SQL avec/SSLTLS
- Utilisation de l'authentification Kerberos avec Amazon RDS for PostgreSQL
- Utilisation d'un DNS serveur personnalisé pour l'accès au réseau sortant

- Mises à niveau du moteur de base de données RDS pour PostgreSQL
- Mise à niveau d'une version du moteur de snapshots SQL de base de données Postgre
- Utilisation de réplicas en lecture pour Amazon RDS for PostgreSQL
- Amélioration des performances des requêtes pour RDS for PostgreSQL avec Lectures optimisées pour Amazon RDS
- Importer des données dans Postgre SQL sur Amazon RDS
- Exportation de données à partir d'une instance de base de données RDS for PostgreSQL vers Amazon S3
- Invocation d'une AWS Lambda fonction depuis une instance de base de données
- Tâches courantes d'administration de bases de données pour Amazon RDS for PostgreSQL
- Réglage avec les événements d'attente pour RDS for PostgreSQL
- Optimisation RDSpour Postgre SQL Aurora Postgre d'Amazon DevOps Guru
- Utilisation des extensions PostgreSQL avec Amazon RDS for PostgreSQL
- Utilisation des wrappers de données étrangers pris en charge pour SQL
- Utilisation de Trusted Language Extensions pour PostgreSQL

Tâches de gestion courantes pour Amazon RDS pour Postgre SQL

Vous trouverez ci-dessous les tâches de gestion courantes que vous effectuez avec une SQL instance de base de données Amazon RDS pour Postgre, avec des liens vers la documentation pertinente pour chaque tâche.

Type de tâche	Documentation
Configuration d'Amazon RDS pour la première utilisation Vous devez remplir quelques conditions préalables avant de créer votre instance de base de données. Par exemple, des instances de bases de données sont créées par défaut avec un pare-feu qui empêche d'y accéder. Vous devez créer un groupe de sécurité avec les adresses IP et la configuration réseau voulues pour accéder à l'instance de base de données.	Configuration de votre environnement Amazon RDS

Type de tâche	Documentation
Comprendre les RDS instances de base de données Amazon Si vous créez une instance de base de données à des fins de production, vous devez comprendre comment IOPS fonctionn ent les classes d'instance, les types de stockage et le mode provisionné sur AmazonRDS.	Classes d'instances de base de données Types de stockage Amazon RDS Stockage SSD d'IOPS par seconde provisionnées
Trouver les versions Postgres disponibles SQL Amazon RDS prend en charge plusieurs versions de PostgreSQ L.	Versions de base de données PostgreSQL disponibles
Configuration de la haute disponibilité et de la prise en charge du basculement Une instance de base de données de production doit utiliser des déploiements multi-AZ. Les déploiements Multi-AZ améliorent la disponibilité, la durabilité des donnés et la tolérance aux pannes pour les instances de bases de données.	Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS
Comprendre le réseau Amazon Virtual Private Cloud (VPC) Si votre AWS compte possède une valeur par défautVPC, votre instance de base de données est automatiquement créée dans cette valeur par défautVPC. Dans certains cas, il se peut que votre compte n'ait pas de valeur par défaut VPC et que vous souhaitiez que l'instance de base de données se trouve dans unVPC. Dans ces cas, créez les groupes de sous-réseaux VPC et avant de créer l'instance de base de données.	Utilisation d'un(e) instance de base de données dans un VPC
Importation de données dans Amazon RDS Postgrer SQL Vous pouvez utiliser différents outils pour importer des données dans votre SQL instance de base de données Postgre sur AmazonRDS.	Importer des données dans Postgre SQL sur Amazon RDS

Type de tâche	Documentation
Configuration des réplicas en lecture (principaux et de secours) en lecture seule RDScar Postgre SQL prend en charge les répliques de lecture à la fois dans la même AWS région et dans une AWS région différente de celle de l'instance principale.	Utilisation des réplicas en lecture d'instance de base de données Utilisation de réplicas en lecture pour Amazon RDS for PostgreSQL Création d'un réplica en lecture dans une autre Région AWS
Présentation des groupes de sécurité Par défaut, les instances de bases de données sont créées avec un pare-feu qui empêche d'y accéder. Pour fournir un accès via ce pare-feu, vous modifiez les règles entrantes pour le groupe de VPC sécurité associé à l'VPChébergement de l'instance de base de données.	Contrôle d'accès par groupe de sécurité
Configuration des fonctionnalités et des groupes de paramètres Pour modifier les paramètres par défaut de votre instance de base de données, créez un groupe de paramètres de base de données personnalisé et modifiez les paramètres. Si vous procédez de la sorte avant de créer votre instance de base de données, vous pouvez choisir votre groupe de paramètres de base de données personnalisé lorsque vous créez l'instance.	Groupes de paramètres pour (Amazon RDS)
Connexion à votre instance de SQL base de données Postgre Après avoir créé un groupe de sécurité et l'avoir associé à une instance de base de données, vous pouvez vous connecter à l'instance de base de données à l'aide de n'importe quelle application SQL client standard telle que `psql` ou`pgAdmin`.	Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL Utilisation de SSL avec une instance de base de données PostgreSQL

Type de tâche	Documentation
Sauvegarde et restauration de votre instance de base de données Vous pouvez configurer votre instance de base de données pour que les sauvegardes soient exécutées automatiquement ou que les instantanés soient créés manuellement, puis que les instances soient restaurées à partir des sauvegardes ou des instantanés.	Sauvegarde, restauration et exportation de données
Surveillance de l'activité et des performances de votre instance de base de données Vous pouvez surveiller une SQL instance de base de données Postgre à l'aide RDS des métriques, des événements et de la surveillance améliorée d' CloudWatch Amazon.	Afficher les métriques dans la RDS console Amazon Consulter les RDS événements Amazon
Mise à niveau de la version de base de SQL données Postgre Vous pouvez effectuer des mises à niveau de version majeures et mineures pour votre SQL instance de base de données Postgre.	Mises à niveau du moteur de base de données RDS pour PostgreSQL Choix d'une version majeure pour une mise à niveau de RDS pour PostgreSQL
Utilisation des fichiers journaux Vous pouvez accéder aux fichiers journaux de votre SQL instance de base de données Postgre.	Fichiers journaux de base de données RDS for PostgreSQL
Comprendre les meilleures pratiques pour les instances de SQL base de données Postgre Découvrez certaines des meilleures pratiques pour travailler avec Postgre SQL sur AmazonRDS.	Bonnes pratiques pour utiliser les moteurs de stockage PostgreSQL

Voici une liste d'autres sections de ce guide qui peuvent vous aider à comprendre et à utiliser les fonctionnalités importantes de RDS for Postgre SQL :

- Comprendre les rôles et les autorisations PostgreSQL
- Contrôle de l'accès utilisateur à la base de données PostgreSQL
- Utilisation de paramètres sur votre instance de base de données RDS for PostgreSQL
- Comprendre les mécanismes de journalisation pris en charge par RDS for Postgre SQL
-
- Utilisation d'un DNS serveur personnalisé pour l'accès au réseau sortant

Utilisation de l'environnement de prévisualisation de base de données

La SQL communauté Postgre publie en permanence de nouvelles SQL versions et extensions de Postgre, y compris des versions bêta. Cela donne aux SQL utilisateurs de Postgre la possibilité d'essayer une nouvelle SQL version de Postgre à un stade précoce. Pour en savoir plus sur le processus de publication de la version bêta de SQL la communauté Postgre, consultez les informations sur la version bêta dans la documentation de PostgreSQL. De même, Amazon RDS propose certaines versions SQL bêta de Postgre sous forme de versions préliminaires. Cela vous permet de créer des instances de base de données à l'aide de la préversion et de tester ses fonctionnalités dans l'environnement en préversion de base de données.

RDSpour les SQL instances de base de données Postgre dans l'environnement de prévisualisation de base de données, les fonctionnalités sont similaires RDS aux autres instances SQL Postgre. Toutefois, vous ne pouvez pas utiliser une préversion pour la production.

Retenez bien les limites importantes suivantes :

- Toutes les instances de base de données sont supprimées 60 jours après leur création, en même temps que leurs sauvegardes et leurs instantanés.
- Vous ne pouvez créer une instance de base de données que dans un cloud privé virtuel (VPC) basé sur le VPC service Amazon.
- Vous ne pouvez utiliser que le IOPS SSD stockage à usage général SSD et provisionné.
- Vous ne pouvez pas obtenir d'aide auprès du AWS Support pour les instances de base de données. Vous pouvez plutôt publier vos questions sur la communauté de AWS questions-réponses gérée,AWS Re:post.
- Vous ne pouvez pas copier un instantané d'instance de base de données dans un environnement de production.

Les options suivantes sont prises en charge par la préversion.

- Vous pouvez créer des instances de base de données à l'aide des types d'instance M6i, R6i, M6g, M5, T3, R6g et R5 uniquement. Pour plus d'informations sur les classes d'RDSinstances, consultezClasses d'instances de base de données .
- Vous pouvez utiliser à la fois des déploiements mono-AZ et multi-AZ.

- Vous pouvez utiliser les fonctions standard de SQL vidage et de chargement de Postgre pour exporter des bases de données depuis ou importer des bases de données vers l'environnement de prévisualisation des bases de données.

Rubriques

- Fonctions non prises en charge dans l'environnement de prévisualisation de base de données
- Postgre SQL version 17 dans l'environnement de prévisualisation de la base de données
- Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données

Fonctions non prises en charge dans l'environnement de prévisualisation de base de données

Les fonctions suivantes ne sont pas disponibles dans l'environnement de prévisualisation de base de données :

- Copie d'instantanés entre Régions
- Réplicas en lecture entre Régions

Postgre SQL version 17 dans l'environnement de prévisualisation de la base de données

> **Note**
>
> Il s'agit d'une version préliminaire de la documentation d'Amazon RDS Postgre SQL version 17. Elle est susceptible d’être modifiée.

SQLLa version 17.0 de Postgre est désormais disponible dans l'environnement Amazon RDS Database Preview. SQLLa version 17.0 de Postgre contient plusieurs améliorations qui sont décrites dans la SQL documentation Postgre suivante, SQLPostgre 17 Release !

Pour plus d'informations sur l'environnement en préversion de base de données, consultez the section called “Utilisation de l'environnement de prévisualisation de base de données”.

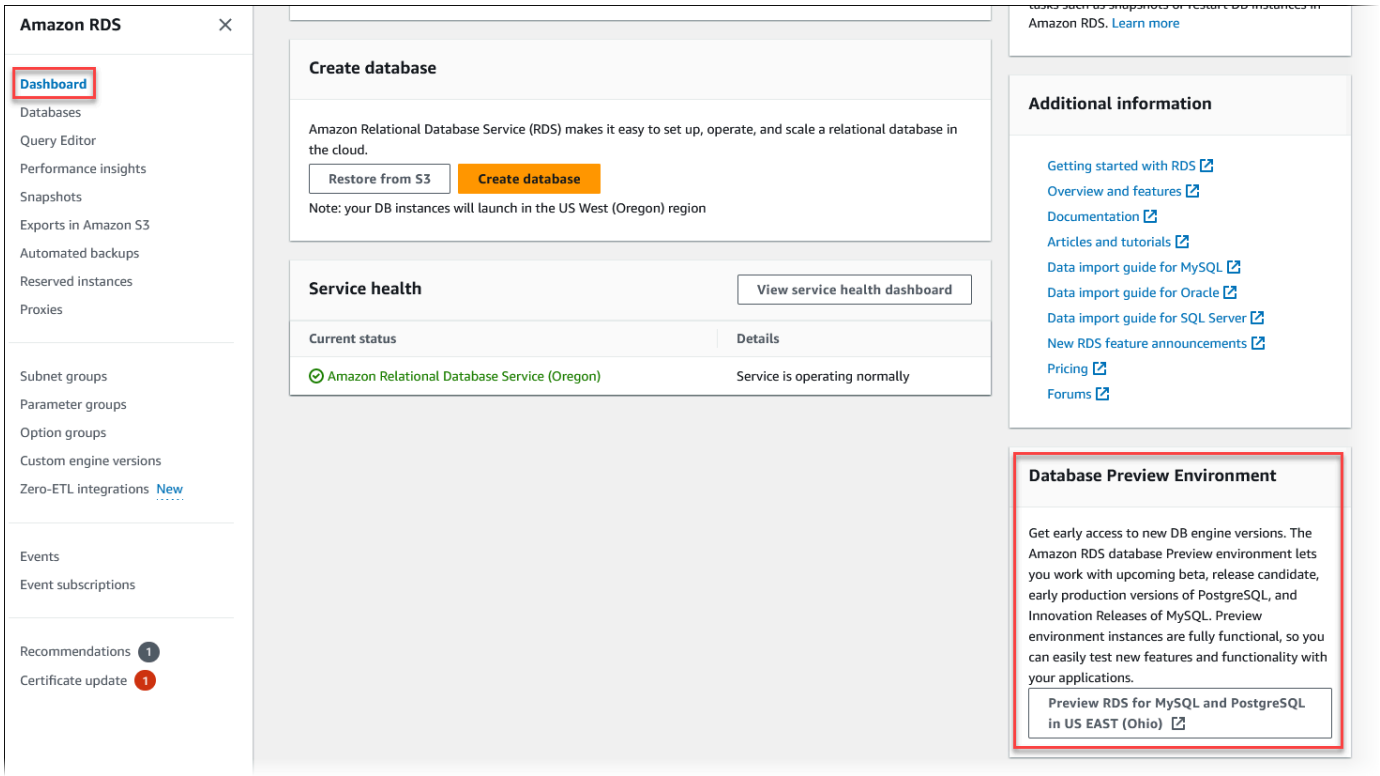
Pour accéder à l'environnement de prévisualisation depuis la console, sélectionnez https://console.aws.amazon.com/rds-preview/.

Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données

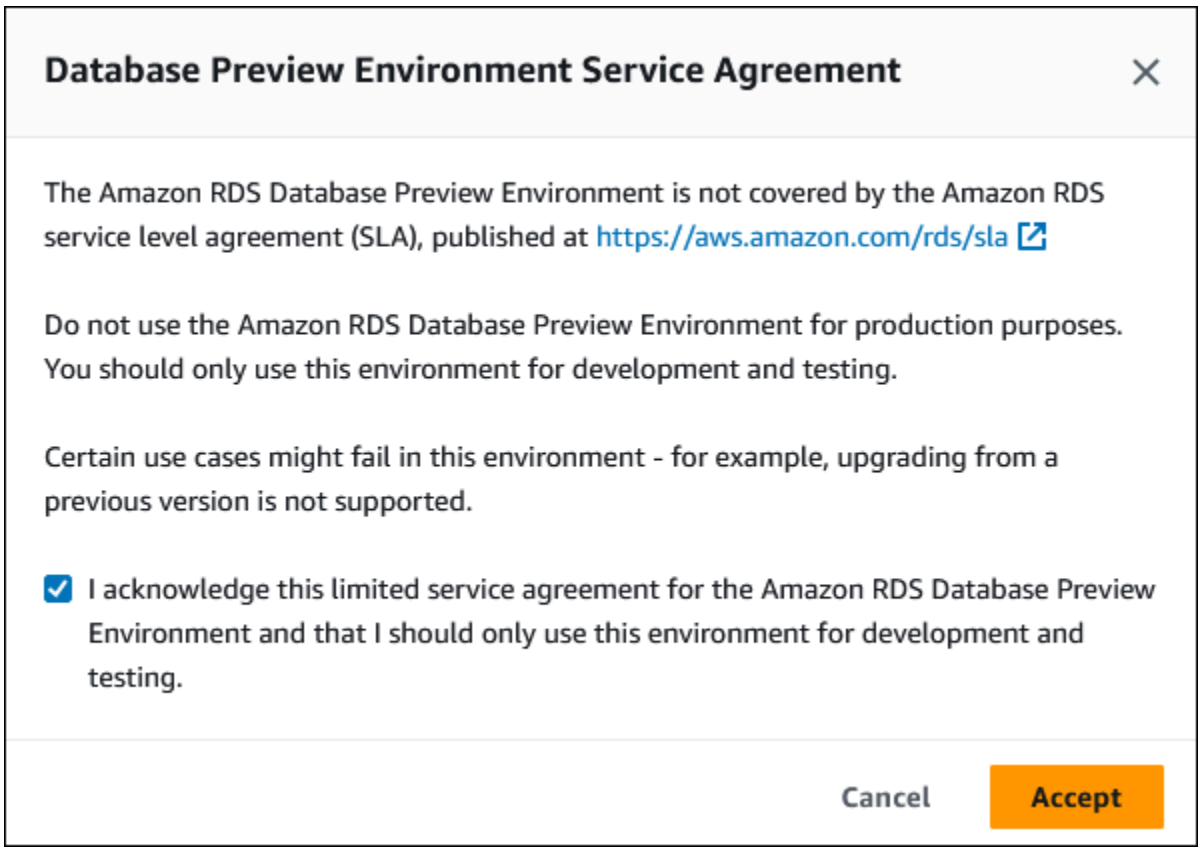
Utilisez la procédure suivante pour créer une instance de base de données dans l'environnement en préversion.

Pour créer une instance de base de données dans l'environnement de prévisualisation de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez Dashboard (Tableau de bord) dans le panneau de navigation.
3. Sur la page Tableau de bord, recherchez la section Database Preview Environment (Environnement en préversion de base de données), comme illustré dans l'image suivante.



Vous pouvez accéder directement à l'environnement de prévisualisation de base de données. Avant de poursuivre, vous devez reconnaître et accepter les limites.



4. Pour créer l'SQLinstance de base de données RDS for Postgre, suivez le même processus que celui de création de n'importe quelle RDS instance de base de données Amazon. Pour plus d'informations, consultez la procédure console dans Création d'une instance de base de données.

Pour créer une instance dans l'environnement de prévisualisation de base de données à l'aide du RDS API ou du AWS CLI, utilisez le point de terminaison suivant.

```
rds-preview.us-east-2.amazonaws.com
```

Versions de base de données PostgreSQL disponibles

Amazon RDS prend en charge les instances de bases de données qui exécutent plusieurs éditions de PostgreSQL. Vous pouvez spécifier toute version de PostgreSQL actuellement disponible lors de la création d'une instance de base de données. Vous pouvez spécifier la version majeure (telle que PostgreSQL 14), et toute version mineure disponible pour la version majeure spécifiée. Si aucune version n'est spécifiée, Amazon RDS utilise par défaut une version disponible, généralement la version la plus récente. Si une version majeure est spécifiée, mais qu'une version mineure ne l'est pas, Amazon RDS utilise par défaut une version récente de la version majeure que vous avez spécifiée.

Pour voir la liste des versions disponibles, ainsi que les valeurs par défaut pour les instances de base de données nouvellement créées, utilisez la `describe-db-engine-versions` AWS CLI commande. Par exemple, pour afficher la version par défaut du moteur PostgreSQL, utilisez la commande suivante :

```
aws rds describe-db-engine-versions --default-only --engine postgres
```

Pour plus d'informations sur les versions PostgreSQL prises en charge sur Amazon RDS, consultez Amazon RDS for PostgreSQL Release Notes (Notes de mise à jour Amazon RDS for PostgreSQL). Vous pouvez également consulter les informations relatives aux dates de support des principales versions du moteur en exécutant la AWS CLI commande describe-db-major-engine-versions ou en utilisant l'opération d'API DBMajorEngineVersionsDescribe RDS.

Si vous n'êtes pas prêt à passer manuellement à une nouvelle version majeure du moteur avant la date de fin du support standard RDS, Amazon RDS inscrira automatiquement vos bases de données au support étendu Amazon RDS après la date de fin du support standard RDS. Vous pouvez ensuite continuer à exécuter RDS pour PostgreSQL version 11 ou ultérieure. Pour plus d'informations, consultez et Tarification d'Amazon RDS.

Versions obsolètes pour Amazon RDS for PostgreSQL

Notez les versions obsolètes suivantes :

- RDS pour PostgreSQL 10 est devenu obsolète en février 2023.
- RDS pour PostgreSQL 9.6 est devenu obsolète en mars 2022.
- RDS pour PostgreSQL 9.5 est devenu obsolète en mars 2021.

Pour en savoir plus sur la politique d'obsolescence de RDS pour PostgreSQL, consultez Amazon RDS. FAQs Pour plus d'informations sur les versions de PostgreSQL, veuillez consulter Versioning Policydans la documentation PostgreSQL.

Comprendre le processus de publication incrémentielle de RDS pour PostgreSQL

RDS pour PostgreSQL fournit des correctifs de sécurité, des améliorations de performances et de nouvelles fonctionnalités par le biais de versions incrémentielles, tout en préservant la compatibilité des versions mineures. Ces versions sont étiquetées R1, R2, R3, etc.

Convention de dénomination des versions

- R1 est la version initiale d'une version mineure. Il inclut parfois de nouvelles fonctionnalités, des extensions ou des mises à niveau d'extensions existantes.
- Les versions ultérieures (R2, R3 et versions ultérieures) incluent :
  - Mises à jour de sécurité
  - Améliorations des performances
  - Correctifs de bogues
  - mises à jour des extensions

Avantages du processus de publication incrémentielle de RDS pour PostgreSQL

Le processus de publication incrémentielle offre les avantages suivants :

- Adoption rapide des nouvelles versions de la communauté PostgreSQL tout en gérant séparément les améliorations spécifiques à RDS dans les versions suivantes. Cela rationalise le processus de publication et garantit une livraison plus rapide des mises à jour critiques.
- Accès aux corrections de bogues, aux nouvelles fonctionnalités, aux mises à jour de sécurité et aux mises à jour des extensions tout en maintenant la compatibilité avec la version mineure de PostgreSQL.

Gestion des mises à jour des versions

Amazon RDS vous informe des nouvelles versions incrémentielles par le biais d'actions de maintenance en attente dans le. AWS Management Console Vous pouvez mettre à jour votre base de données à l'aide de l'une des méthodes suivantes :

- Activez les mises à jour automatiques pendant les fenêtres de maintenance planifiées.
- Appliquez les mises à jour manuellement par le biais d'actions de maintenance en attente.
- Utilisez des déploiements bleu/vert avec réplication physique pour minimiser les temps d'arrêt. Pour plus d'informations, consultez la section Les déploiements bleu/vert prennent en charge la mise à niveau de versions mineures de RDS pour PostgreSQL.

Avant de mettre à jour votre base de données, tenez compte des points essentiels suivants :

- Le redémarrage de la base de données est nécessaire pour les mises à jour, sauf si vous utilisez des déploiements bleu/vert avec réplication physique.
- Certaines versions incrémentielles sont obligatoires, en particulier celles qui incluent des correctifs de sécurité.

Pour plus d'informations sur la mise à jour de votre instance de base de données Amazon RDS, consultez Extensions SQL fiables de Postgre et apply-pending-maintenance-action.

Versions d'SQLextension Postgre prises en charge

RDSfor Postgre SQL supporte de nombreuses extensions PostgreSQL. La SQL communauté Postgre les appelle parfois modules. Les extensions étendent les fonctionnalités fournies par le SQL moteur Postgre. Vous trouverez une liste des extensions prises en charge par Amazon RDS dans le groupe de paramètres de base de données par défaut pour cette SQL version de Postgre. Vous pouvez également consulter la liste actuelle des extensions utilisant `psql` en affichant le paramètre `rds.extensions` comme dans l'exemple suivant.

```
SHOW rds.extensions;
```

> **Note**
>
> Les paramètres ajoutés à une version mineure peuvent s'afficher de manière incorrecte lors de l'utilisation du paramètre `rds.extensions` dans `psql`.

À partir RDS de Postgre SQL 13, certaines extensions peuvent être installées par des utilisateurs de base de données autres que le`rds_superuser`. On les appelle des extensions de confiance. Pour en savoir plus, consultez Extensions SQL fiables de Postgre.

Certaines versions de RDS for Postgre SQL prennent en charge le `rds.allowed_extensions` paramètre. Ce paramètre permet de `rds_superuser` limiter les extensions qui peuvent être installées dans l'SQLinstance de base de données RDS for Postgre. Pour de plus amples informations, veuillez consulter Restreindre l'installation des extensions Postgre SQL.

Pour obtenir la liste des SQL extensions et des versions de Postgre prises en charge par chacune des versions disponibles RDS pour Postgre, consultez la section SQLExtensions Postgre prises en charge sur Amazon dans RDS Amazon RDS pour les notes de SQL version de SQL Postgre.

Restreindre l'installation des extensions Postgre SQL

Vous pouvez restreindre les extensions qui peuvent être installées sur une SQL instance de base de données Postgre. Par défaut, ce paramètre n'est pas défini. Par conséquent, toute extension prise en charge peut être ajoutée si l'utilisateur dispose des autorisations appropriées. Pour ce faire, définissez le paramètre `rds.allowed_extensions` sur une chaîne de noms d'extension séparés par des virgules. En ajoutant une liste d'extensions à ce paramètre, vous identifiez explicitement les

extensions que votre SQL instance de base de données RDS pour Postgre peut utiliser. Seules ces extensions peuvent ensuite être installées dans l'SQLinstance de base de données Postgre.

La chaîne par défaut du paramètre `rds.allowed_extensions` est « * », ce qui signifie que toute extension disponible pour la version du moteur peut être installée. La modification du paramètre `rds.allowed_extensions` ne nécessite pas de redémarrage de la base de données, car il s'agit d'un paramètre dynamique.

Le moteur d'SQLinstance de base de données Postgre doit être l'une des versions suivantes pour que vous puissiez utiliser le `rds.allowed_extensions` paramètre :

- Toutes les versions de Postgre SQL 16
- Postgre SQL 15 et toutes les versions supérieures
- Postgre SQL 14 et toutes les versions supérieures
- Postgre SQL 13.3 et versions mineures supérieures
- Postgre SQL 12.7 et versions mineures supérieures

Pour voir quelles installations d'extension sont autorisées, utilisez la commande psql suivante.

```
postgres=> SHOW rds.allowed_extensions;
 rds.allowed_extensions
------------------------
 *
```

Si une extension a été installée avant d'être exclue de la liste dans le paramètre `rds.allowed_extensions`, l'extension peut toujours être utilisée normalement, et les commandes telles que `ALTER EXTENSION` et `DROP EXTENSION` resteront opérationnelles. Cependant, une fois qu'une extension est restreinte, les commandes `CREATE EXTENSION` de l'extension restreinte échouent.

L'installation des dépendances d'extension avec `CREATE EXTENSION CASCADE` sont également restreintes. L'extension et ses dépendances doivent être spécifiées dans `rds.allowed_extensions`. Si une installation de dépendance d'extension échoue, l'instruction `CREATE EXTENSION CASCADE` échouera dans son intégralité.

Si une extension n'est pas incluse avec le paramètre `rds.allowed_extensions`, vous verrez une erreur telle que la suivante si vous essayez de l'installer.

```
ERROR: permission denied to create extension "extension-name"
HINT: This extension is not specified in "rds.allowed_extensions".
```

Extensions SQL fiables de Postgre

L'installation de la plupart des SQL extensions Postgre nécessite des `rds_superuser` privilèges. Postgre SQL 13 a introduit des extensions fiables, qui réduisent le besoin d'accorder `rds_superuser` des privilèges aux utilisateurs réguliers. Cette fonction permet aux utilisateurs d'installer de nombreuses extensions s'ils disposent du privilège `CREATE` sur la base de données actuelle, sans exiger le rôle `rds_superuser`. Pour plus d'informations, consultez la SQL CREATEEXTENSIONcommande dans la SQL documentation Postgre.

La liste suivante répertorie les extensions qui peuvent être installées par un utilisateur qui possède le privilège `CREATE` sur la base de données actuelle, sans exiger le rôle `rds_superuser` :

- bool_plperl
- btree_gin
- btree_gist
- citext
- cube
- dict_int
- fuzzystrmatch
- hstore
- intarray
- isn
- jsonb_plperl
- ltree
- pg_trgm
- pgcrypto
- plperl
- plpgsql
- pltcl
- tablefunc

- tsm_system_rows
- tsm_system_time
- unaccent
- uuid-ossp

Pour obtenir la liste des SQL extensions et des versions de Postgre prises en charge par chacune des versions disponibles RDS pour Postgre, consultez la section SQLExtensions Postgre prises en charge sur Amazon dans RDS Amazon RDS pour les notes de SQL version de SQL Postgre.

Utilisation des SQL fonctionnalités de Postgre prises en charge par Amazon RDS pour Postgre SQL

Amazon RDS pour Postgre SQL prend en charge la plupart des fonctionnalités Postgre SQL les plus courantes. Par exemple, Postgre SQL dispose d'une fonction d'aspiration automatique qui effectue la maintenance de routine de la base de données. La fonction d'autovacuum est active par défaut. Bien que vous puissiez désactiver cette fonction, nous vous recommandons vivement de la conserver active. Comprendre cette fonctionnalité et ce que vous pouvez faire pour vous assurer qu'elle fonctionne comme il se doit est une tâche fondamentaleDBA. Pour de plus amples informations sur l'autovacuum, veuillez consulter . Pour en savoir plus sur les autres DBA tâches courantes,Tâches courantes d'administration de bases de données pour Amazon RDS for PostgreSQL.

RDSfor Postgre prend SQL également en charge les extensions qui ajoutent des fonctionnalités importantes à l'instance de base de données. Par exemple, vous pouvez utiliser l'GISextension Post pour travailler avec des données spatiales, ou utiliser l'extension pg_cron pour planifier la maintenance depuis l'instance. Pour plus d'informations sur les SQL extensions Postgre, consultezUtilisation des extensions PostgreSQL avec Amazon RDS for PostgreSQL.

Les enveloppeurs de données étrangères sont un type d'extension spécifique conçu pour permettre à votre RDS SQL instance de base de données Postgre de fonctionner avec d'autres bases de données ou types de données commerciaux. Pour plus d'informations sur les enveloppeurs de données étrangers pris en charge par RDS for PostgreSQL, consultez. Utilisation des wrappers de données étrangers pris en charge pour SQL

Vous trouverez ci-dessous des informations sur d'autres fonctionnalités prises en charge par RDS PostgreSQL.

Rubriques

- Types de données et énumérations personnalisés avec RDS pour Postgre SQL
- Déclencheurs d'événements RDS pour Postgre SQL
- De grandes pages RDS pour Postgre SQL
- Réplication logique pour Amazon RDS for PostgreSQL
- RAMdisque pour le répertoire stats_temp_directory
- Tablespaces pour Postgre RDS SQL
- RDSpour les SQL collations Postgre EBCDIC et les autres migrations de mainframe
- Gestion de la synchronisation des emplacements logiques pour RDS pour PostgreSQL

Types de données et énumérations personnalisés avec RDS pour Postgre SQL

Postgre SQL prend en charge la création de types de données personnalisés et l'utilisation d'énumérations. Pour plus d'informations sur la création et l'utilisation d'énumérations et d'autres types de données, consultez la section Types énumérés dans la documentation Postgre. SQL

Voici un exemple de création d'un type en tant qu'énumération, puis d'insertion de valeurs dans une table.

```
CREATE TYPE rainbow AS ENUM ('red', 'orange', 'yellow', 'green', 'blue', 'purple');
CREATE TYPE
CREATE TABLE t1 (colors rainbow);
CREATE TABLE
INSERT INTO t1 VALUES ('red'), ( 'orange');
INSERT 0 2
SELECT * from t1;
colors
--------
red
orange
(2 rows)
postgres=> ALTER TYPE rainbow RENAME VALUE 'red' TO 'crimson';
ALTER TYPE
postgres=> SELECT * from t1;
colors
---------
crimson
orange
(2 rows)
```

Déclencheurs d'événements RDS pour Postgre SQL

Toutes les SQL versions actuelles de Postgre prennent en charge les déclencheurs d'événements, de même que toutes les versions disponibles de RDS for SQL Postgre. Vous pouvez utiliser le compte d'utilisateur principal (par défaut, `postgres`) pour créer, modifier, renommer et supprimer des déclencheurs d'évènements. Les déclencheurs d'évènements sont au niveau de l'instance de base de données, de sorte qu'ils peuvent s'appliquer à toutes les bases de données sur une instance.

Par exemple, le code suivant crée un déclencheur d'événement qui affiche l'utilisateur actuel à la fin de chaque commande du langage de définition des données (DDL).

```
CREATE OR REPLACE FUNCTION raise_notice_func()
    RETURNS event_trigger
    LANGUAGE plpgsql AS
$$
BEGIN
    RAISE NOTICE 'In trigger function: %', current_user;
END;
$$;

CREATE EVENT TRIGGER event_trigger_1
    ON ddl_command_end
EXECUTE PROCEDURE raise_notice_func();
```

Pour plus d'informations sur les déclencheurs d'SQLévénements Postgre, consultez la section Déclencheurs d'événements dans la documentation PostgreSQL.

L'utilisation des déclencheurs d'SQLévénements Postgre sur Amazon RDS comporte plusieurs limites. Tel est le cas des éléments suivants :

- Vous ne pouvez pas créer de déclencheurs d'évènements sur les réplicas en lecture. En revanche, vous pouvez créer de déclencheurs d'évènements sur une source de réplica en lecture. Les déclencheurs d'évènements sont ensuite copiés sur le réplica en lecture. Les déclencheurs d'évènements sur le réplica en lecture ne s'activent pas sur le réplica en lecture lorsque les modifications sont poussées depuis la source. En revanche, si le réplica en lecture est promu, les déclencheurs d'évènements existants s'activent lorsque des opérations de base de données ont lieu.
- Pour effectuer une mise à niveau de version majeure vers une SQL instance de base de données Postgre qui utilise des déclencheurs d'événements, assurez-vous de supprimer les déclencheurs d'événements avant de mettre à niveau l'instance.

De grandes pages RDS pour Postgre SQL

Les Huge pages (Grandes pages) sont une fonction de gestion de la mémoire qui réduit la surcharge lorsqu'une instance de base de données fonctionne avec de gros morceaux de mémoire contigus, tels que ceux utilisés par les tampons partagés. Cette SQL fonctionnalité de Postgre est prise en charge par toutes les versions actuellement disponibles RDS pour PostgreSQL. Vous allouez de grandes pages pour votre application en utilisant des appels à la mémoire partagée `mmap` ou `SYSV`. RDScar Postgre SQL prend en charge les tailles de page de 4 Ko et de 2 Mo.

Vous pouvez activer ou désactiver les Grandes pages en modifiant la valeur du paramètre `huge_pages`. La fonction est activée par défaut pour toutes les classes d'instances de base de données autres que les classes micro, petites et medium.

RDScar Postgre SQL utilise d'énormes pages en fonction de la mémoire partagée disponible. Si l'instance de base de données ne peut pas utiliser de grandes pages en raison de contraintes de mémoire partagée, Amazon RDS empêche le démarrage de l'instance de base de données. Dans ce cas, Amazon RDS définit le statut de l'instance de base de données sur un état de paramètres incompatibles. Dans ce cas, vous pouvez définir le `huge_pages` paramètre sur pour permettre `off` RDS à Amazon de démarrer l'instance de base de données.

Le paramètre `shared_buffers` est essentiel pour définir le pool de mémoire partagée requis pour utiliser les grandes pages. La valeur par défaut du paramètre `shared_buffers` utilise une macro de paramètres de base de données. Cette macro définit un pourcentage du total des 8 Ko pages disponibles pour la mémoire de l'instance de base de données. Lorsque vous utilisez des grandes pages, celles-ci se trouvent avec les grandes pages. Amazon RDS place une instance de base de données dans un état de paramètres incompatibles si les paramètres de mémoire partagée sont définis pour nécessiter plus de 90 % de la mémoire de l'instance de base de données.

Pour en savoir plus sur la gestion de SQL la mémoire Postgre, consultez la section Consommation de ressources dans la documentation PostgreSQL.

Réplication logique pour Amazon RDS for PostgreSQL

Dès la version 10.4, RDS for PostgreSQL prend en charge la syntaxe SQL de publication et d'abonnement qui a été introduite dans PostgreSQL 10. Pour en savoir plus, veuillez consulter Réplication logique dans la documentation PostgreSQL.

> **Note**
>
> Outre la fonctionnalité de réplication logique native de PostgreSQL introduite dans PostgreSQL 10, RDS for PostgreSQL prend également en charge l'extension `pglogical`. Pour de plus amples informations, veuillez consulter Utilisation de pglogical pour synchroniser les données entre les instances.

Ci-dessous, vous trouverez des informations sur la configuration de la réplication logique pour une instance de base de données RDS for PostgreSQL.

Rubriques

- Compréhension de la réplication logique et du décodage logique
- Utilisation des emplacements de réplication logique
- Réplication des données au niveau des tables à l'aide de la réplication logique

Compréhension de la réplication logique et du décodage logique

RDS for PostgreSQL prend en charge le streaming des modifications WAL (write-ahead log) à l'aide d'emplacements de réplication logique PostgreSQL. Il prend également en charge l'utilisation du décodage logique. Vous pouvez configurer des emplacements logiques de réplication sur votre instance et diffuser les modifications de base de données à travers ces emplacements pour un client comme `pg_recvlogical`. Les emplacements logiques de réplication sont créés au niveau de la base de données et prennent en charge les connexions de réplication avec une base de données unique.

Les clients les plus courants pour la réplication logique PostgreSQL AWS Database Migration Service sont les hôtes gérés de manière personnalisée sur une instance Amazon. EC2 L'emplacement de réplication logique ne contient aucune information sur le récepteur du flux. De plus, il n'est pas nécessaire que la cible soit une base de données de réplica. Si vous configurez un emplacement de réplication logique et que vous ne lisez pas à partir de l'emplacement, les données peuvent être écrites dans le stockage de votre instance de base de données et le remplir rapidement.

La réplication logique et le décodage logique PostgreSQL sur Amazon RDS sont activés avec un paramètre, un type de connexion de réplication et un rôle de sécurité. Le client pour le décodage logique peut être n'importe quel client capable d'établir une connexion de réplication avec une base de données sur une instance de base de données PostgreSQL.

Pour activer le décodage logique pour une instance de base de données RDS for PostgreSQL

1. Assurez-vous que le compte utilisateur que vous utilisez possède les rôles suivants :
   - Le rôle `rds_superuser` de manière à ce que vous puissiez activer la réplication logique
   - Le rôle `rds_replication` pour accorder les autorisations de gérer des emplacements logiques et de diffuser les données à l'aide d'emplacements logiques
2. Définissez le paramètre statique `rds.logical_replication` sur 1. Dans le cadre de l'application de ce paramètre, configurez également les paramètres `wal_level`, `max_wal_senders`, `max_replication_slots` et `max_connections`. Ces modifications de

paramètres peuvent accroître la génération WAL de sorte que vous ne devez définir le paramètre `rds.logical_replication` que lorsque vous utilisez des emplacements logiques.

3. Redémarrez l'instance de base de données pour que le paramètre statique `rds.logical_replication` prenne effet.
4. Créez un emplacement de réplication logique comme expliqué dans la section suivante. Cela nécessite que vous précisiez un plugin de décodage. Actuellement, RDS for PostgreSQL prend en charge les plugins de sortie test_decoding et wal2json fournis avec PostgreSQL.

Pour plus d'informations sur le décodage logique PostgreSQL, consultez la documentation PostgreSQL.

Utilisation des emplacements de réplication logique

Vous pouvez utiliser les commandes SQL pour utiliser les emplacements logiques. Par exemple, la commande suivante crée un emplacement logique nommé `test_slot` à l'aide du plugin de sortie PostgreSQL par défaut `test_decoding`.

```
SELECT * FROM pg_create_logical_replication_slot('test_slot', 'test_decoding');
slot_name    | xlog_position
-----------------+---------------
regression_slot | 0/16B1970
(1 row)
```

Pour lister les emplacements logiques, utilisez la commande suivante.

```
SELECT * FROM pg_replication_slots;
```

Pour supprimer un emplacement logique, utilisez la commande suivante.

```
SELECT pg_drop_replication_slot('test_slot');
pg_drop_replication_slot
-----------------------
(1 row)
```

Pour plus d'exemples sur l'utilisation des emplacements de réplication logique, consultez Exemples de décodage logique dans la documentation PostgreSQL.

Après avoir créé l'emplacement de réplication logique, vous pouvez commencer le streaming. L'exemple suivant montre comment le décodage logique est contrôlé par le protocole de streaming.

Celui-ci utilise le programme pg_recvlogical, qui est inclus dans la distribution PostgreSQL. Cela nécessite que l'authentification du client soit configurée pour autoriser les connexions de réplication.

```
pg_recvlogical -d postgres --slot test_slot -U postgres
    --host -instance-name.111122223333.aws-region.rds.amazonaws.com
    -f -  --start
```

Pour afficher le contenu de la vue `pg_replication_origin_status`, interrogez la fonction `pg_show_replication_origin_status`.

```
SELECT * FROM pg_show_replication_origin_status();
local_id | external_id | remote_lsn | local_lsn
----------+-------------+------------+-----------
(0 rows)
```

Réplication des données au niveau des tables à l'aide de la réplication logique

Vous pouvez utiliser la réplication logique pour répliquer les données des tables sources vers les tables cibles dans RDS pour PostgreSQL. La réplication logique effectue d'abord un chargement initial des données existantes à partir des tables sources, puis continue à répliquer les modifications en cours.

1. Création des tables sources

   Connectez-vous à la base de données source dans votre instance de base de données RDS pour PostgreSQL :

   ```
   source=> CREATE TABLE testtab (slno int primary key);
   CREATE TABLE
   ```

2. Insérer des données dans les tables sources

   ```
   source=> INSERT INTO testtab VALUES (generate_series(1,1000));
   INSERT 0 1000
   ```

3. Création d'une publication pour les tables sources

   - Créez une publication pour les tables sources :

```
source=> CREATE PUBLICATION testpub FOR TABLE testtab;
CREATE PUBLICATION
```

   - Utilisez une requête SELECT pour vérifier les détails de la publication créée :

```
source=> SELECT * FROM pg_publication;
  oid   | pubname | pubowner | puballtables | pubinsert | pubupdate | pubdelete |
 pubtruncate | pubviaroot
--------+---------+----------+--------------+-----------+-----------+-----------
+-------------+------------
 115069 | testpub |    16395 | f            | t         | t         | t         |
 t           | f
(1 row)
```

   - Vérifiez que les tables sources sont ajoutées à la publication :

```
source=> SELECT * FROM pg_publication_tables;
pubname | schemaname | tablename
---------+------------+-----------
 testpub | public     | testtab
(1 rows)
```

   - Pour répliquer toutes les tables d'une base de données, utilisez :

```
CREATE PUBLICATION testpub FOR ALL TABLES;
```

   - Si la publication est déjà créée pour une table individuelle et que vous devez ajouter une nouvelle table, vous pouvez exécuter la requête ci-dessous pour ajouter de nouvelles tables dans la publication existante :

```
ALTER PUBLICATION <publication_name> add table <new_table_name>;
```

4. Connectez-vous à la base de données cible et créez des tables cibles

   - Connectez-vous à la base de données cible dans l'instance de base de données cible. Créez les tables cibles portant les mêmes noms que les tables sources :

```
target=> CREATE TABLE testtab (slno int primary key);
CREATE TABLE
```

   - Assurez-vous qu'aucune donnée n'est présente dans les tables cibles en exécutant une requête SELECT sur les tables cibles :

```
target=> SELECT count(*) FROM testtab;
 count
-------
     0
(1 row)
```

5. Création et vérification de l'abonnement dans la base de données cible

   - Créez l'abonnement dans la base de données cible :

```
target=> CREATE SUBSCRIPTION testsub
CONNECTION 'host=<source RDS/host endpoint> port=5432 dbname=<source_db_name>
 user=<user> password=<password>'
PUBLICATION testpub;
NOTICE:  Created replication slot "testsub" on publisher
CREATE SUBSCRIPTION
```

   - Utilisez une requête SELECT pour vérifier que l'abonnement est activé :

```
target=> SELECT oid, subname, subenabled, subslotname, subpublications FROM
 pg_subscription;
  oid  | subname | subenabled | subslotname | subpublications
-------+---------+------------+-------------+-----------------
 16434 | testsub | t          | testsub     | {testpub}
(1 row)
```

- Lorsque l'abonnement est créé, il charge toutes les données des tables sources vers les tables cibles. Exécutez une requête SELECT sur les tables cibles pour vérifier que les données initiales sont chargées :

```
target=> SELECT count(*) FROM testtab;
 count
-------
  1000
(1 row)
```

6. Vérifier le slot de réplication dans la base de données source

   La création d'un abonnement dans la base de données cible crée un emplacement de réplication dans la base de données source. Vérifiez les détails du slot de réplication en exécutant la requête SELECT suivante sur la base de données source :

```
source=> SELECT * FROM pg_replication_slots;

slot_name |  plugin  | slot_type | datoid | database | temporary | active |
 active_pid | xmin | catalog_xmin | restart_lsn | confirmed_flush_lsn | wal_status
 | safe_wal_size
----------+----------+-----------+--------+----------+-----------+--------
+------------+------+--------------+-------------+---------------------
+------------+---------------
testsub   | pgoutput | logical   | 115048 | source   | f         | t      |
 846 |      |         6945 | 58/B4000568 | 58/B40005A0         | reserved   |
(1 row)
```

7. Tester la réplication

   - Vérifiez si les modifications apportées aux données dans les tables sources sont répliquées dans les tables cibles en insérant des lignes dans les tables sources :

```
source=> INSERT INTO testtab VALUES(generate_series(1001,2000));
INSERT 0 1000

source=> SELECT count(*) FROM testtab;
 count
-------
  2000
(1 row)
```

   - Vérifiez le nombre de lignes dans les tables cibles pour confirmer que les nouveaux inserts sont répliqués :

```
target=> SELECT count(*) FROM testtab;
 count
-------
  2000
(1 row)
```

8. Actualisation de l'abonnement après avoir ajouté des tables

   - Lorsque vous ajoutez de nouveaux tableaux à une publication existante, il est obligatoire d'actualiser l'abonnement pour que les modifications prennent effet :

```
ALTER SUBSCRIPTION <subscription_name> REFRESH PUBLICATION;
```

   - Cette commande récupère les informations de table manquantes auprès de l'éditeur et lance la réplication des tables qui ont été ajoutées aux publications abonnées depuis la création de l'abonnement ou la dernière actualisation de l'abonnement.

RAMdisque pour le répertoire stats_temp_directory

Vous pouvez utiliser le SQL paramètre RDS for Postgre `rds.pg_stat_ramdisk_size` pour spécifier la mémoire système allouée à un RAM disque pour stocker le SQL `stats_temp_directory` Postgre. Le paramètre de RAM disque n'est disponible que RDS pour les SQL versions 14 et antérieures de Postgre.

Avec certaines charges de travail, ce paramètre peut améliorer les performances et réduire les exigences relatives aux I/O. Pour plus d'informations à ce sujet`stats_temp_directory`, consultez la SQL documentation Postgre. .

Pour configurer un RAM disque pour votre`stats_temp_directory`, définissez le `rds.pg_stat_ramdisk_size` paramètre sur une valeur littérale entière dans le groupe de paramètres utilisé par votre instance de base de données. Ce paramètre est indiqué en Mo, vous devez donc utiliser une valeur entière. Les expressions, formules et fonctions ne sont pas valides pour le paramètre `rds.pg_stat_ramdisk_size`. Veillez à réinitialiser l'instance de base de données pour que la modification prenne effet. Pour plus d'informations sur la définition des paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Par exemple, la AWS CLI commande suivante définit le paramètre du RAM disque sur 256 Mo.

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name pg-95-ramdisk-testing \
    --parameters "ParameterName=rds.pg_stat_ramdisk_size, ParameterValue=256,
 ApplyMethod=pending-reboot"
```

Après le redémarrage, exécutez la commande suivante pour afficher le statut de `stats_temp_directory`.

```
postgres=> SHOW stats_temp_directory;
```

La commande devrait renvoyer le résultat suivant.

```
stats_temp_directory
---------------------------
/rdsdbramdisk/pg_stat_tmp
(1 row)
```

Tablespaces pour Postgre RDS SQL

RDScar Postgre SQL prend en charge les tablespaces pour des raisons de compatibilité. Étant donné que tout le stockage se trouve sur un seul volume logique, vous ne pouvez pas utiliser d'espaces de table pour le fractionnement ou l'isolement des I/O. Nos évaluations et notre expérience indiquent qu'un seul volume logique constitue la meilleure configuration dans la plupart des cas d'utilisation.

Le rôle est requis pour créer et utiliser des tablespaces avec votre SQL instance RDS de base de données Postgre. `rds_superuser` Le compte utilisateur principal de votre SQL instance de base de données RDS for Postgre (nom par défaut`postgres`) est membre de ce rôle. Pour de plus amples informations, veuillez consulter Comprendre les rôles et les autorisations PostgreSQL.

Si vous spécifiez un nom de fichier lors de la création d'un espace de table, le préfixe du chemin est `/rdsdbdata/db/base/tablespace`. L'exemple suivant montre comment placer les fichiers d'espace de table dans `/rdsdbdata/db/base/tablespace/data`. Cet exemple suppose qu'un utilisateur `dbadmin` (rôle) existe et qu'il se soit vu accorder le rôle `rds_superuser` nécessaire à l'utilisateur des espaces de table.

```
postgres=> CREATE TABLESPACE act_data
  OWNER dbadmin
  LOCATION '/data';
CREATE TABLESPACE
```

Pour en savoir plus sur les tablespaces Postgre, consultez SQL Tablespaces dans la documentation Postgre. SQL

RDSpour les SQL collations Postgre EBCDIC et les autres migrations de mainframe

RDSpour les SQL versions 10 et supérieures de Postgre, incluez ICU la version 60.2, qui est basée sur Unicode 10.0 et inclut des collations provenant du référentiel de données locales communes Unicode, 32. CLDR Ces bibliothèques d'internationalisation logicielle garantissent que les codages de caractères sont présentés de manière cohérente, quel que soit le système d'exploitation ou la plateforme. Pour plus d'informations sur Unicode CLDR -32, consultez la note de mise à jour CLDR 32 sur le CLDR site Web d'Unicode. Vous pouvez en savoir plus sur les composants d'internationalisation pour Unicode (ICU) sur le site Web du Comité ICU technique (ICU-TC). Pour plus d'informations sur ICU -60, voir Download ICU 60.

À partir de la version 14.3, RDS Postgre inclut SQL également des classements qui facilitent l'intégration et la conversion des données à partir de systèmes EBCDIC basés. Le code d'échange ou EBCDICcodage décimal codé binaire étendu est couramment utilisé par les systèmes d'exploitation mainframe. Ces classements RDS fournis par Amazon sont définis de manière étroite pour ne trier que les caractères Unicode qui correspondent directement aux EBCDIC pages de code. Les caractères sont triés par ordre de EBCDIC point de code pour permettre la validation des données après la conversion. Ces classements n'incluent pas les formes dénormalisées, pas plus qu'ils n'incluent les caractères Unicode qui ne correspondent pas directement à un caractère de la page de code sourceEBCDIC.

Les mappages de caractères entre les pages de EBCDIC code et les points de code Unicode sont basés sur des tables publiées parIBM. L'ensemble complet est disponible IBM sous forme de fichier compressé à télécharger. RDSfor Postgre a SQL utilisé ces mappages avec les outils fournis par le ICU pour créer les classements répertoriés dans les tableaux de cette section. Les noms des classements incluent une langue et un pays, comme l'exige leICU. Toutefois, les pages de EBCDIC code ne spécifient pas les langues, et certaines pages de EBCDIC code couvrent plusieurs pays. Cela signifie que les parties langue et pays des noms de classement dans la table sont arbitraires et qu'elles ne doivent pas nécessairement correspondre à la locale actuelle. En d'autres termes, le numéro de page de code est la partie la plus importante du nom du classement dans ce tableau. Vous pouvez utiliser n'importe laquelle des collations répertoriées dans les tableaux suivants dans n'importe quelle SQL base RDS de données Postgre.

- Unicode to EBCDIC collations table— Certains outils de migration des données du mainframe utilisent LATIN1 ou permettent d'encoder et LATIN9 de traiter des données en interne. Ces outils utilisent des schémas aller-retour pour préserver l'intégrité des données et prendre en charge la conversion inverse. Les classements de ce tableau peuvent être utilisés par des outils qui traitent les données à l'aide d'un LATIN1 codage, ce qui ne nécessite pas de manipulation particulière.
- Unicode to LATIN9 collations table— Vous pouvez utiliser ces classements dans n'importe quelle base RDS de SQL données Postgre.

Dans le tableau suivant, vous trouverez des classements disponibles dans Postgre SQL qui RDS font correspondre des pages de EBCDIC code à des points de code Unicode. Nous vous recommandons d'utiliser les classements de ce tableau pour le développement d'applications nécessitant un tri basé sur l'ordre des pages de IBM code.

Nom de la SQL collation Postgrer	Description du mappage code-page et de l'ordre de tri
da-DK-cp277-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 277 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 277
de-DE-cp273-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 273 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 273
en-GB-cp285-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 285 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 285
en-US-cp037-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 037 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 37
es-ES-cp284-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 284 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 284
fi-FI-cp278-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 278 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 278
fr-FR-cp297-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 297 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 297

Nom de la SQL collation Postgrer	Description du mappage code-page et de l'ordre de tri
it-IT-cp280-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 280 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 280
nl-BE-cp500-x-icu	Les caractères Unicode qui correspondent directement à la page de IBM EBCDIC code 500 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 500

Amazon RDS fournit un ensemble de classements supplémentaires qui trient les points de code Unicode qui correspondent aux LATIN9 caractères à l'aide des tables publiées parIBM, dans l'ordre des points de code d'origine conformément à la page de EBCDIC code des données source.

Nom de la SQL collation Postgrer	Description du mappage code-page et de l'ordre de tri
DA-DK-CP1142 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1142 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1142
DE-DE-CP1141 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1141 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1141
FR-GB-CP1146 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1146

Nom de la SQL collation Postgrer	Description du mappage code-page et de l'ordre de tri
	(selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1146
EN-US-CP1140 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1140 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1140
ES-ES-CP1145 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1145 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1145
fi-Fi-CP1143 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1143 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1143
FR-FR-CP1147 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1147 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1147
IT-IT-CP1144 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1144 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1144

Nom de la SQL collation Postgrer	Description du mappage code-page et de l'ordre de tri
NL-BE-CP1148 m-x-icu	Les caractères Unicode qui correspondent aux LATIN9 caractères initialement convertis à partir de la page de IBM EBCDIC code 1148 (selon les tables de conversion) sont triés dans l'ordre des points de code IBM CP 1148

Vous trouverez ci-dessous un exemple d'utilisation d'un SQL classement RDS for Postgre.

```
db1=> SELECT pg_import_system_collations('pg_catalog');
 pg_import_system_collations
----------------------------
                          36
db1=> SELECT '¤' < 'a' col1;
 col1
------
 t
db1=> SELECT '¤' < 'a' COLLATE "da-DK-cp277-x-icu" col1;
 col1
------
 f
```

Nous vous recommandons d'utiliser les classements dans Unicode to EBCDIC collations table et dans le Unicode to LATIN9 collations table pour le développement d'applications qui nécessitent un tri basé sur l'ordre des pages de IBM code. Les classements suivants (suffixés par la lettre « b ») sont également visibles dans les outils d'intégration et de migration de données du mainframe`pg_collation`, mais ils sont destinés à être utilisés par ces outils pour cartographier des pages de code AWS présentant des décalages de points de code spécifiques et nécessitant un traitement spécial lors du classement. En d'autres termes, l'utilisation des classements suivants n'est pas recommandée.

- DA-DK-277 b-x-icu
- DA-DK-1142 b-x-icu
- DE-DE-CP273 b-x-icu
- DE-DE-CP1141 b-x-icu

- FR-GB-CP1146 b-x-icu
- FR-GB-CP285 b-x-icu
- FR-US-CP037 b-x-icu
- EN-US-CP1140 b-x-icu
- ES-ES-CP1145 b-x-icu
- ES-ES-CP284 b-x-icu
- fi-Fi-CP1143 b-x-icu
- FR-FR-CP1147 b-x-icu
- FR-FR-CP297 b-x-icu
- IT-IT-CP1144 b-x-icu
- IT-IT-CP280 b-x-icu
- NL-BE-CP1148 b-x-icu
- NL-BE-CP500 b-x-icu

Pour en savoir plus sur la migration d'applications depuis des environnements mainframe vers des environnements mainframe AWS, voir Qu'est-ce que la modernisation des AWS mainframes ? .

Pour plus d'informations sur la gestion des classements dans PostgreSQL, consultez Support des classements dans la documentation PostgreSQL.

Gestion de la synchronisation des emplacements logiques pour RDS pour PostgreSQL

Depuis la version communautaire de PostgreSQL 17, une nouvelle fonctionnalité permettant de synchroniser automatiquement les emplacements de réplication logique entre les serveurs principaux et les serveurs de secours a été introduite via le `sync_replication_slots` paramètre ou la `pg_sync_replication_slots()` fonction associée, qui synchronise manuellement les emplacements lors de l'exécution.

Ces fonctionnalités sont disponibles à partir de RDS pour PostgreSQL 17. Une configuration typique comporte une instance principale et sa réplique en lecture, ainsi qu'un abonné de réplication logique à l'instance principale.

Assurez-vous que l'abonnement est créé avec l'option failover définie sur true :

```
CREATE SUBSCRIPTION subname CONNECTION 'host=...' PUBLICATION pubname WITH (failover =
 true);
```

Cela crée un emplacement logique sur l'éditeur avec le basculement activé.

```
postgres=> SELECT slot_name, slot_type, failover FROM pg_catalog.pg_replication_slots;
 slot_name | slot_type | failover
-----------+-----------+----------
 subname   | logical   | t
(1 row)
```

En activant la synchronisation des emplacements, tous les emplacements de réplication logique de basculement sur le serveur principal sont automatiquement créés sur les emplacements de secours physiques et sont synchronisés périodiquement. Assurez-vous que les valeurs suivantes ont été définies par le biais de groupes de paramètres :

- `rds.logical_replication`doit être 1 pour permettre la réplication logique
- `hot_standby_feedback`doit être 1 en veille
- `rds.logical_slot_sync_dbname`en veille doit être défini sur un nom de base de données valide

  La valeur par défaut du paramètre est`postgres`. Si l'instance de publication logique possède la `postgres` base de données, il n'est pas nécessaire de modifier le paramètre par défaut.
- `synchronized_standby_slots`sur le serveur principal doit être réglé sur le slot de réplication physique du serveur de secours destiné à être synchronisé
- `sync_replication_slots`doit être 1 pour activer la synchronisation automatique

Avec un emplacement d'abonnement activé pour le basculement et les valeurs de paramètres ci-dessus, lorsqu'une mise en veille est promue, l'abonné peut modifier son abonnement à cette instance nouvellement promue et poursuivre la réplication logique de manière fluide.

Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL

Après qu'Amazon RDS a provisionné votre instance de base de données, vous pouvez utiliser n'importe quelle application cliente SQL standard pour vous connecter à l'instance. Avant de pouvoir vous connecter, l'instance de base de données doit être disponible et accessible. La méthode utilisée pour créer l'instance de base de données Amazon RDS détermine s'il est possible ou non de vous y connecter depuis l'extérieur du VPC :

- Si vous avez créé votre instance de base de données en tant que public, les appareils et EC2 les instances Amazon extérieurs au VPC peuvent se connecter à votre base de données.
- Si vous avez créé votre instance de base de données en tant que privée, seuls EC2 les instances et appareils Amazon au sein d'Amazon VPC peuvent se connecter à votre base de données.

Pour vérifier si votre instance de base de données est publique ou privée, utilisez l' AWS Management Console onglet Connectivité et sécurité pour votre instance. Sous Security (Sécurité), vous pouvez trouver la valeur « Publicly accessible (Accessible publiquement) », avec No (Non) pour privé, Yes (Oui) pour public.

Pour en savoir plus sur les différentes configurations Amazon RDS et Amazon VPC, et comment elles affectent l'accessibilité, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.

Table des matières

- Installation du client PSQL
- Recherche des informations de connexion pour une instance de base de données RDS pour PostgreSQL
- Utilisation pgAdmin pour se connecter à une instance RDS de SQL base de données Postgre
- Utilisation de psql pour vous connecter à votre instance de base RDS de données Postgre SQL
- Connexion RDS à Postgre SQL avec le pilote Amazon Web Services (AWS) JDBC
- Connexion RDS à Postgre SQL avec le pilote Python Amazon Web Services (AWS)
- Résolution des problèmes de connexion à votre instance RDS for PostgreSQL
  - Erreur – IRRÉCUPÉRABLE : le nom de la base de données name n'existe pas
  - Erreur – Impossible de se connecter au serveur : la connexion a expiré

- Erreurs liées aux règles d'accès du groupe de sécurité

Installation du client PSQL

Pour vous connecter à votre instance de base de données à partir d'une EC2 instance, vous pouvez installer un client PostgreSQL sur l'instance. EC2 Pour installer la dernière version du client psql sur Amazon Linux 2023, exécutez la commande suivante :

```
sudo dnf install postgresql<version number>
```

Pour installer la dernière version du client psql sur Amazon Linux 2, exécutez la commande suivante :

```
sudo yum install -y postgresql
```

Pour installer la dernière version du client psql sur Ubuntu, exécutez la commande suivante :

```
sudo apt install -y postgresql-client
```

Recherche des informations de connexion pour une instance de base de données RDS pour PostgreSQL

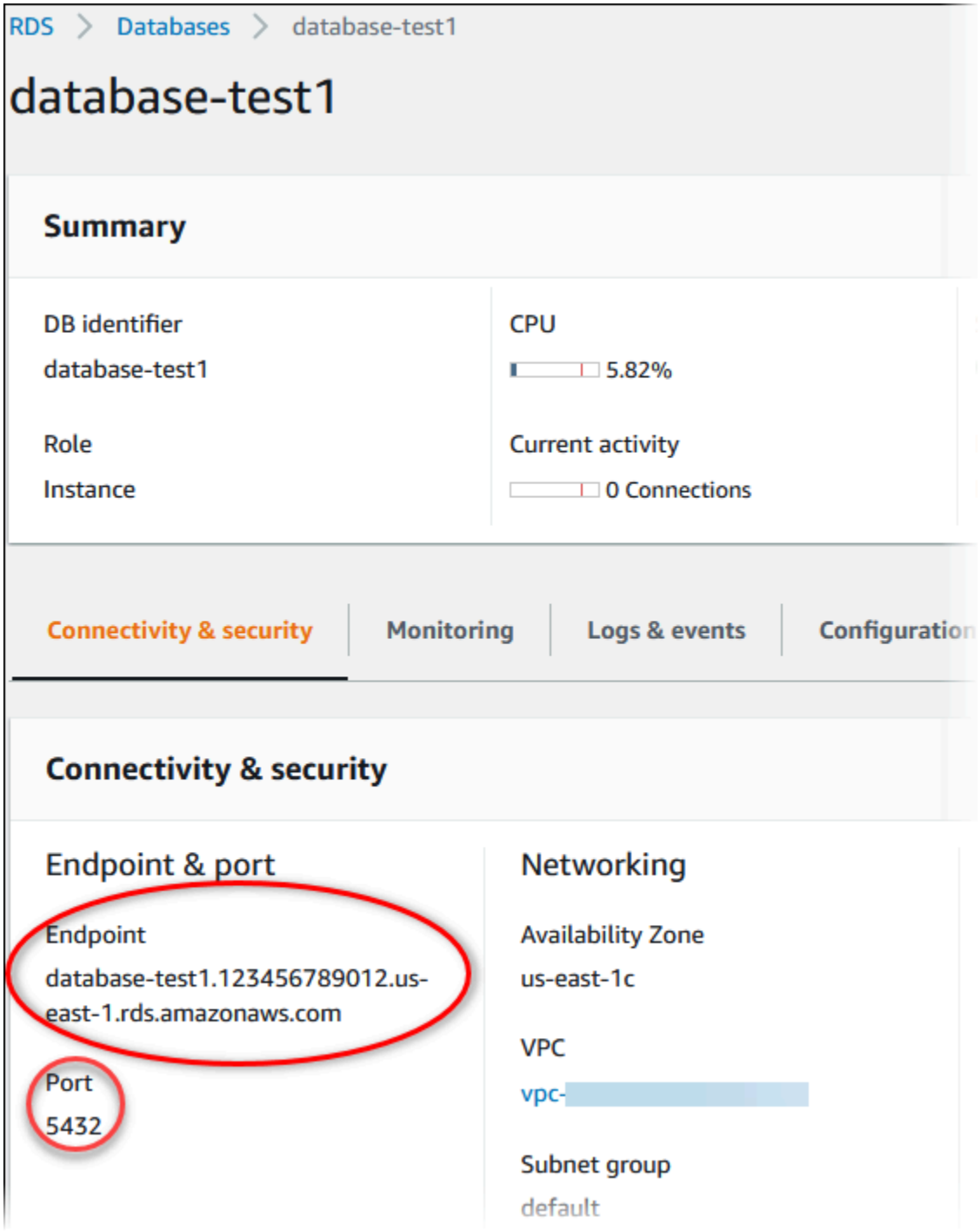
Si l'instance de base de données est disponible et accessible, vous pouvez vous connecter en fournissant les informations suivantes à l'application cliente SQL :

- Point de terminaison de l'instance de base de données servant de nom d'hôte (nom DNS) pour l'instance.
- Le port au niveau duquel l'instance de base de données écoute. Le port par défaut pour PostgreSQL est 5432.
- Le nom d'utilisateur et le mot de passe de l'instance de base de données. Le « nom d'utilisateur principal » par défaut pour PostgreSQL est `postgres`.
- Nom et mot de passe de la base de données (Nom de base de données).

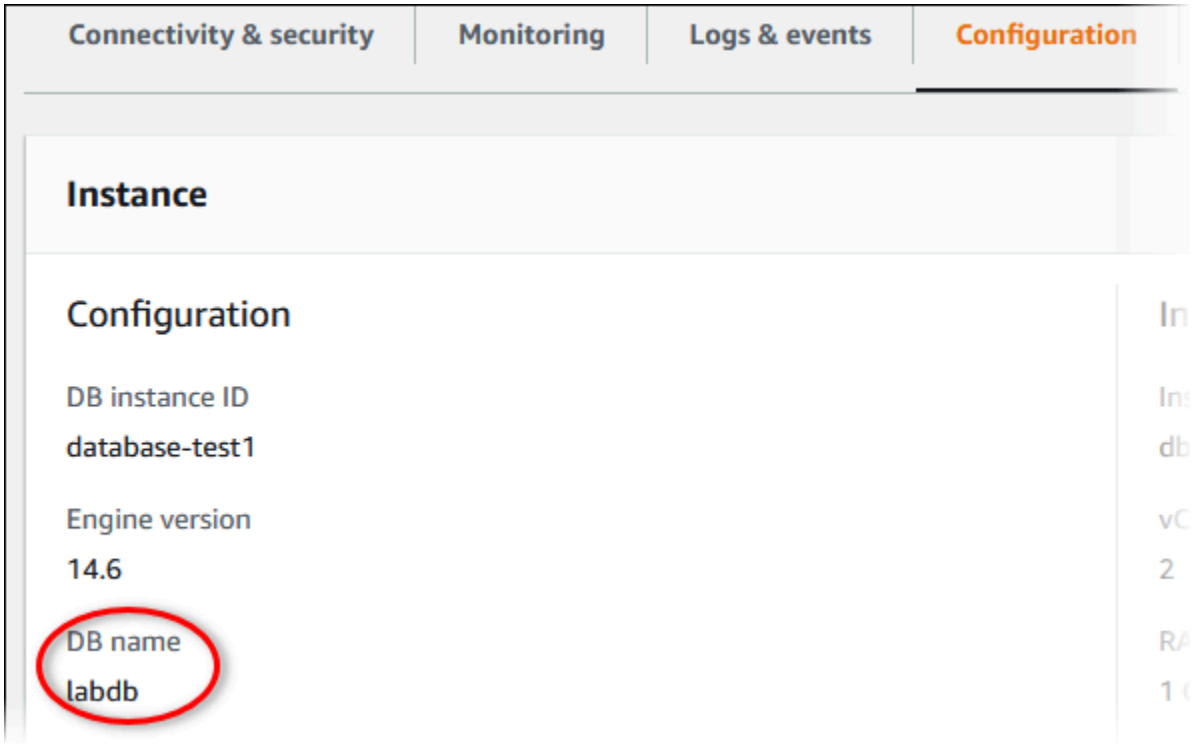
Vous pouvez obtenir ces informations à l'aide de la AWS Management Console AWS CLI describe-db-instancescommande ou de l'DBInstancesopération Amazon RDS API Describe.

Pour trouver le point de terminaison, le numéro de port et le nom de la base de données à l'aide du AWS Management Console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Ouvrez la console RDS et choisissez Bases de données pour afficher une liste de vos instances de bases de données.
3. Choisissez le nom de l'instance de base de données PostgreSQL pour afficher ses détails.
4. Dans l'onglet Connectivity & security (Connectivité et sécurité), copiez le point de terminaison. Notez également le numéro du port. Vous avez besoin du point de terminaison et du numéro de port pour vous connecter à l'instance de base de données.



5. Sous l'onglet Configuration, indiquez le nom de la base de données. Si vous avez créé une base de données lors de la création de l'instance RDS for PostgreSQL, le nom apparaît sous « Nom de base de données ». Si vous n'avez pas créé de base de données, le nom de la base de données est remplacé par un tiret (-).



Voici deux façons de se connecter à une instance de base de données PostgreSQL. Le premier exemple utilise pgAdmin, célèbre outil open source d'administration et de développement pour PostgreSQL. Le second exemple utilise psql, utilitaire de ligne de commande qui fait partie d'une installation PostgreSQL.

Utilisation pgAdmin pour se connecter à une instance RDS de SQL base de données Postgre

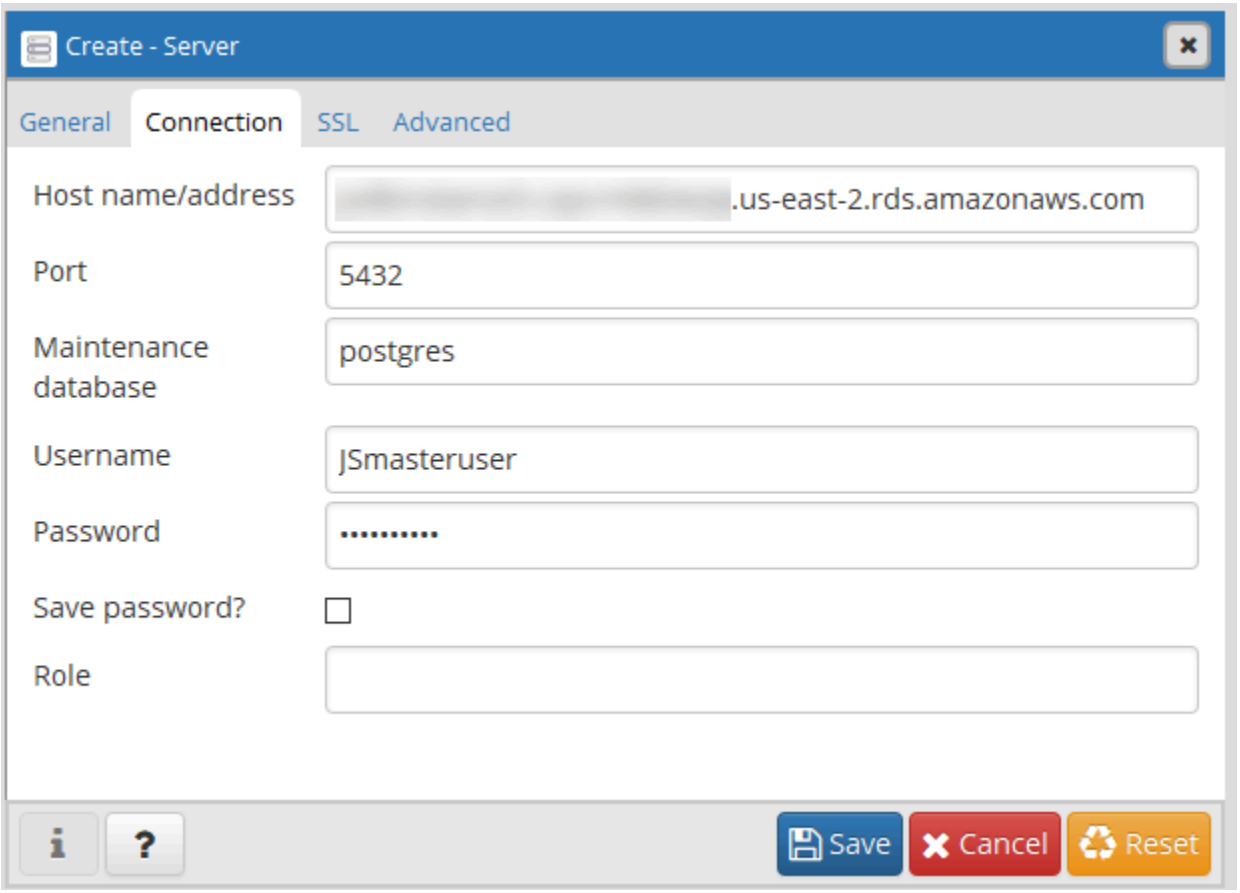
Vous pouvez utiliser l'outil open source pour vous connecter pgAdmin à votre instance de base RDS de données PostgreSQL. Vous pouvez télécharger et installer pgAdmin depuis http://www.pgadmin.org/ sans avoir d'instance locale de Postgre SQL sur votre ordinateur client.

Pour vous connecter à votre SQL instance de base RDS de données Postgre à l'aide de pgAdmin

1. Lancez l' pgAdmin application sur votre ordinateur client.
2. Dans l'onglet Dashboard (Tableau de bord), choisissez Add New Server (Ajouter un nouveau serveur).
3. Dans la boîte de dialogue Créer - Serveur, tapez un nom dans l'onglet Général pour identifier le serveur danspgAdmin.

4. Dans l'onglet Connexion, tapez les informations suivantes depuis votre instance de base de données :

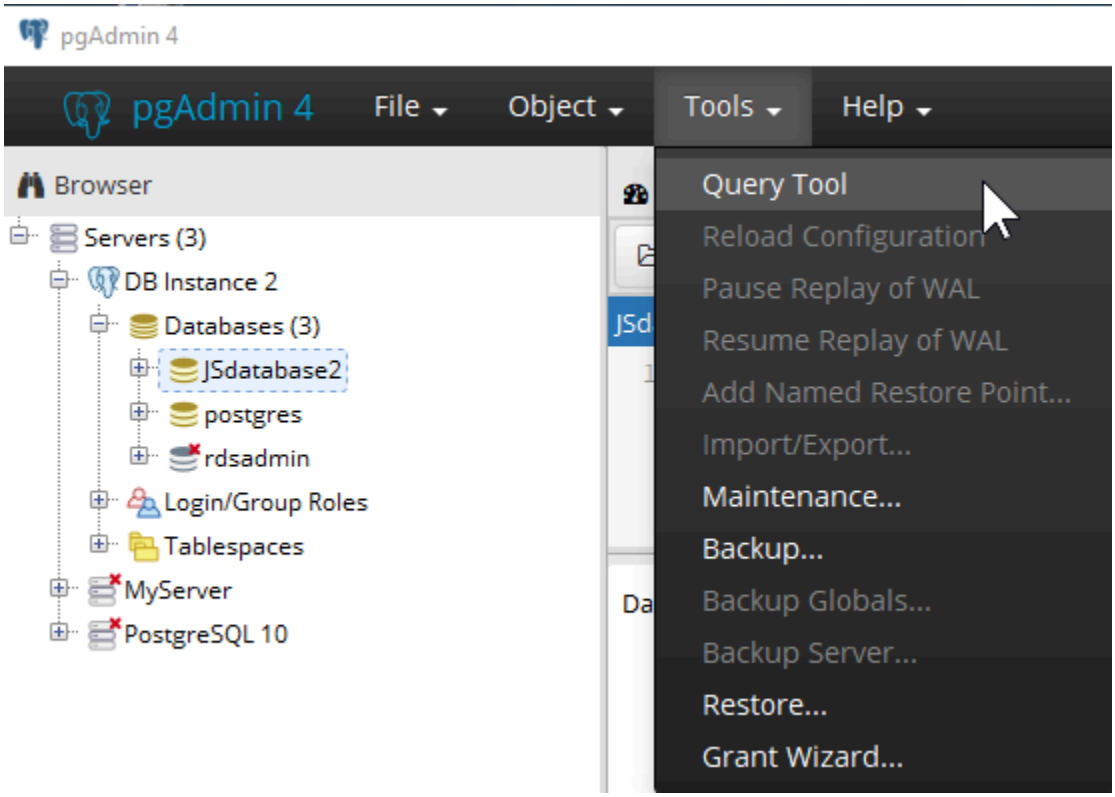
   - Pour Hôte, tapez le point de terminaison, par exemple `mypostgresql.c6c8dntfzzhgv0.us-east-2.rds.amazonaws.com`.
   - Pour Port, tapez le port attribué.
   - Dans le champ Nom d'utilisateur, saisissez le nom d'utilisateur indiqué lors de la création de l'instance de base de données (si vous avez modifié le « nom d'utilisateur principal » par défaut, `postgres`).
   - Pour Mot de passe, tapez le mot de passe entré lors de la création de l'instance de base de données.



5. Choisissez Save (Enregistrer).

   Si vous rencontrez des problèmes de connexion, consultez Résolution des problèmes de connexion à votre instance RDS for PostgreSQL.

6. Pour accéder à une base de données dans le pgAdmin navigateur, développez les serveurs, l'instance de base de données et les bases de données. Choisissez le nom de la base de données de l'instance de base de données.



7. Pour ouvrir un panneau dans lequel vous pouvez entrer SQL des commandes, choisissez Outils, Outil de requête.

Utilisation de psql pour vous connecter à votre instance de base RDS de données Postgre SQL

Vous pouvez utiliser une instance locale de l'utilitaire de ligne de commande psql RDS pour vous connecter à une instance de SQL base de données Postgre. Vous devez installer Postgre SQL ou le client psql sur votre ordinateur client.

Vous pouvez télécharger le SQL client Postgre depuis le site Web de Postgre SQL. Pour installer psql, suivez les instructions spécifiques à votre version du système d'exploitation.

RDSPour vous connecter à votre SQL instance de base de données Postgre à l'aide de psql, vous devez fournir les informations d'hôte (DNS), les informations d'identification d'accès et le nom de la base de données.

Utilisez l'un des formats suivants pour vous connecter à votre RDS SQL instance de base de données Postgre. Lorsque vous vous connectez, vous êtes invité à entrer un mot de passe. Pour les tâches de traitement par lots ou les scripts, utilisez l'option `--no-password`. Cette option est définie pour l'ensemble de la session.

> **Note**
>
> Une tentative de connexion avec `--no-password` échoue lorsque le serveur exige une authentification par mot de passe et qu'aucun mot de passe n'est disponible auprès d'autres sources. Pour plus d'informations, consultez la documentation de psql.

Si c'est la première fois que vous vous connectez à cette instance de base de données, ou si vous n'avez pas encore créé de base de données RDS pour cette SQL instance Postgre, vous pouvez vous connecter à la base de données postgres à l'aide du « nom d'utilisateur principal » et du mot de passe.

Pour Unix, utilisez le format suivant.

```
psql \
   --host=<DB instance endpoint> \
   --port=<port> \
   --username=<master username> \
   --password \
   --dbname=<database name>
```

Pour Windows, utilisez le format suivant.

```
psql ^
   --host=<DB instance endpoint> ^
   --port=<port> ^
   --username=<master username> ^
   --password ^
   --dbname=<database name>
```

Par exemple, la commande suivante se connecte à une base de données appelée `mypgdb` sur une SQL instance de base de données Postgre appelée à `mypostgresql` l'aide d'informations d'identification fictives.

```
psql --host=mypostgresql.c6c8mwvfdgv0.us-west-2.rds.amazonaws.com --port=5432 --
username=awsuser --password --dbname=mypgdb
```

Connexion RDS à Postgre SQL avec le pilote Amazon Web Services (AWS) JDBC

Le JDBC pilote Amazon Web Services (AWS) est conçu comme un JDBC wrapper avancé. Ce wrapper complète et étend les fonctionnalités d'un JDBC pilote existant. Le pilote est compatible directement avec le JDBC pilote Community PG.

Pour installer le AWS JDBC pilote, ajoutez le fichier .jar du AWS JDBC pilote (situé dans l'application`CLASSPATH`) et conservez les références au pilote communautaire correspondant. Mettez à jour le URL préfixe de connexion correspondant comme suit :

- `jdbc:postgresql://` sur `jdbc:aws-wrapper:postgresql://`

Pour plus d'informations sur le AWS JDBC pilote et des instructions complètes pour son utilisation, consultez le GitHub référentiel de JDBC pilotes Amazon Web Services (AWS).

Connexion RDS à Postgre SQL avec le pilote Python Amazon Web Services (AWS)

Le pilote Python Amazon Web Services (AWS) est conçu comme un wrapper Python avancé. Ce wrapper complète et étend les fonctionnalités du pilote open source Psycopg. Le pilote AWS Python prend en charge les versions 3.8 et supérieures de Python. Vous pouvez installer le `aws-advanced-python-wrapper` package à l'aide de la `pip` commande, en même temps que les packages `psycopg` open source.

Pour plus d'informations sur le pilote AWS Python et des instructions complètes pour son utilisation, consultez le GitHub référentiel de pilotes Python Amazon Web Services (AWS).

Résolution des problèmes de connexion à votre instance RDS for PostgreSQL

Rubriques

- Erreur – IRRÉCUPÉRABLE : le nom de la base de données name n'existe pas
- Erreur – Impossible de se connecter au serveur : la connexion a expiré

- Erreurs liées aux règles d'accès du groupe de sécurité

Erreur – IRRÉCUPÉRABLE : le nom de la base de données *name* n'existe pas

Si vous recevez une erreur telle que `FATAL: database name does not exist` lorsque vous tentez de vous connecter, essayez d'utiliser le nom par défaut de la base de données postgres pour l'option `--dbname`.

Erreur – Impossible de se connecter au serveur : la connexion a expiré

Si vous ne parvenez pas à vous connecter à l'instance de base de données, l'erreur la plus courante est `Could not connect to server: Connection timed out.` Si vous recevez cette erreur, procédez comme suit :

- Vérifiez que le nom d'hôte utilisé est le point de terminaison de l'instance de base de données et que le numéro de port utilisé est correct.
- Assurez-vous que l'accessibilité publique de l'instance de base de données est définie sur Oui pour autoriser les connexions externes. Pour modifier le paramètre Accès public, consultez Modification d'une RDS instance de base de données Amazon.
- Assurez-vous que l'utilisateur qui se connecte à la base de données dispose d'un accès CONNECT à celle-ci. Vous pouvez utiliser la requête suivante pour fournir un accès de connexion à la base de données.

  ```
  GRANT CONNECT ON DATABASE database name TO username;
  ```

- Vérifiez que le groupe de sécurité affecté à l'instance de base de données possède les règles pour autoriser l'accès via tout pare-feu que votre connexion peut traverser. Par exemple, si l'instance de base de données a été créée à l'aide du port par défaut 5432, votre entreprise peut disposer de règles de pare-feu bloquant les connexions à ce port depuis les appareils externes à l'entreprise.

  Pour résoudre ce problème, modifiez l'instance de base de données afin qu'elle utilise un autre port. De plus, assurez-vous que les groupes de sécurité appliqués à l'instance de base de données autorisent les connexions au nouveau port. Pour modifier le paramètre Port de la base de données, consultez Modification d'une RDS instance de base de données Amazon.
- Vérifiez si le port que vous essayez d'utiliser est déjà occupé par une instance locale de PostgreSQL ou par un autre service exécuté sur votre ordinateur. Par exemple, si une base de données PostgreSQL locale s'exécute sur le même port (la valeur par défaut est 5432), cela peut empêcher une connexion réussie à l'instance de base de données RDS pour PostgreSQL.

Assurez-vous que le port est libre ou essayez de vous connecter avec un autre numéro de port si possible.

- Voir aussi Erreurs liées aux règles d'accès du groupe de sécurité.

Erreurs liées aux règles d'accès du groupe de sécurité

Le problème de connexion qui intervient le plus fréquemment concerne les règles d'accès du groupe de sécurité assigné à l'instance de base de données. Si vous avez utilisé le groupe de sécurité par défaut lorsque vous avez créé l'instance de base de données, ce groupe de sécurité ne dispose vraisemblablement pas de règles d'accès vous autorisant à accéder à l'instance.

Pour que la connexion s'établisse, le groupe de sécurité que vous avez assigné à l'instance de base de données à sa création doit autoriser l'accès à l'instance de base de données. Par exemple, si l'instance de base de données a été créée à l'intérieur d'un VPC, elle doit avoir un groupe de sécurité VPC qui autorise les connexions. Vérifiez si l'instance de base de données a été créée à l'aide d'un groupe de sécurité qui n'autorise pas les connexions depuis l'appareil ou l' EC2 instance Amazon sur laquelle l'application est exécutée.

Vous pouvez ajouter ou modifier une règle entrante dans le groupe de sécurité. La sélection de Mon IP comme Source permet d'accéder à l'instance de base de données à partir de l'adresse IP détectée dans votre navigateur. Pour plus d'informations, consultez Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Sinon, si l'instance de base de données a été créée à l'extérieur d'un VPC, elle doit avoir un groupe de sécurité de bases de données qui autorise ces connexions.

Pour de plus amples informations sur les groupes de sécurité Amazon RDS, veuillez consulter Contrôle d'accès par groupe de sécurité.

Sécuriser les connexions RDS à Postgre SQL avec/SSLTLS

RDSfor Postgre SQL prend en charge le chiffrement Secure Socket Layer (SSL) pour les instances de base de données PostgreSQL. À l'aide deSSL, vous pouvez chiffrer une SQL connexion Postgre entre vos applications et vos instances de base de données PostgreSQL. Vous pouvez également forcer l'utilisation SSL de toutes les connexions à votre SQL instance de base de données Postgre. RDSfor Postgre supporte SQL également Transport Layer Security (TLS), le protocole successeur deSSL.

Pour en savoir plus sur Amazon RDS et la protection des données, notamment le chiffrement des connexions à l'aide deSSL/TLS, consultezProtection des données dans Amazon RDS.

Rubriques

- Utilisation de SSL avec une instance de base de données PostgreSQL
- Mise à jour des applications pour se connecter aux SQL instances de base de données Postgre à l'aide de nouveaux certificatsSSL/TLS

Utilisation de SSL avec une instance de base de données PostgreSQL

Amazon RDS prend en charge le chiffrement SSL pour les instances de bases de données PostgreSQL. En utilisant SSL, vous pouvez chiffrer une connexion PostgreSQL entre vos applications et vos instances de bases de données PostgreSQL. Par défaut, RDS for PostgreSQL utilise et attend de tous les clients qu'ils se connectent en utilisant SSL/TLS, mais vous pouvez également l'exiger. RDS pour PostgreSQL prend en charge les versions 1.1, 1.2 et 1.3 de Transport Layer Security (TLS).

Pour plus d'informations sur la prise en charge de SSL et les bases de données PostgreSQL, veuillez consulter SSL Support dans la documentation PostgreSQL. Pour plus d'informations sur l'utilisation d'une connexion SSL sur JDBC, veuillez consulter Configuring the Client dans la documentation PostgreSQL.

Le support SSL est disponible dans toutes les AWS régions pour PostgreSQL. Amazon RDS crée un certificat SSL pour votre instance de base de données PostgreSQL lors de la création de celle-ci. Si vous activez la vérification du certificat SSL, ce dernier inclut le point de terminaison de l'instance de base de données en tant que nom commun du certificat SSL pour assurer une protection contre les attaques par usurpation.

Rubriques

- Connexion à une instance de base de données PostgreSQL via SSL
- Exiger une connexion SSL à une instance de base de données PostgreSQL
- Détermination du statut de la connexion SSL
- Suites de chiffrement SSL dans RDS for PostgreSQL

Connexion à une instance de base de données PostgreSQL via SSL

Pour se connecter à une instance de base de données PostgreSQL via SSL

1. Téléchargez le certificat.

   Pour plus d'informations sur le téléchargement de certificats, veuillez consulter .

2. Connectez-vous à votre instance de base de données PostgreSQL sur SSL.

   Lors de la connexion avec SSL, votre client peut choisir de vérifier ou pas la chaîne du certificat. Si vos paramètres de connexion spécifient `sslmode=verify-ca` ou `sslmode=verify-full`, votre client nécessite que les certificats de l'autorité de certification RDS soient dans leur magasin d'approbations ou référencés dans l'URL de connexion. L'exigence nécessite de vérifier la chaîne du certificat qui signe le certificat de votre base de données.

   Quand un client, tel que psql ou JDBC, est configuré avec la prise en charge du protocole SSL, le comportement par défaut est le suivant : le client essaie d'abord de se connecter à la base de données avec le protocole SSL. En cas d'impossibilité, le client revient à la connexion sans protocole SSL. Le mode `sslmode` utilisé par défaut diffère selon qu'il s'agit de clients libpq (comme psql) ou de clients JDBC. Les clients basés sur libpq et JDBC sont par défaut sur. `prefer`

   Utilisez le paramètre `sslrootcert` pour référencer le certificat, par exemple, `sslrootcert=rds-ssl-ca-cert.pem`.

L'exemple suivant montre comment utiliser `psql` pour se connecter à une instance de base de données PostgreSQL en utilisant SSL avec vérification du certificat.

```
$ psql "host=db-name.555555555555.ap-southeast-1.rds.amazonaws.com
    port=5432 dbname=testDB user=testuser sslrootcert=rds-ca-rsa2048-g1.pem
 sslmode=verify-full"
```

Exiger une connexion SSL à une instance de base de données PostgreSQL

Vous pouvez exiger que les connexions à votre instance de base de données PostgreSQL utilisent SSL en utilisant le paramètre `rds.force_ssl`. La valeur par défaut du `rds.force_ssl` paramètre est 1 (activé) pour RDS pour PostgreSQL version 15 et versions ultérieures. Pour toutes les autres versions majeures de RDS pour PostgreSQL 14 et antérieures, la valeur par défaut de ce paramètre est 0 (off). Vous pouvez affecter au paramètre `rds.force_ssl` la valeur 1 (activé) pour exiger SSL/TLS pour les connexions à votre cluster de base de données. Vous pouvez affecter au paramètre `rds.force_ssl` la valeur 1 (activé) pour exiger SSL pour les connexions à votre instance de base de données.

Pour modifier la valeur de ce paramètre, vous devez créer un groupe de paramètres de base de données personnalisé. Vous modifiez ensuite la valeur de `rds.force_ssl` dans votre groupe de paramètres de base de données personnalisé sur 1 pour activer cette fonction. Si vous préparez le groupe de paramètres de base de données personnalisé avant de créer votre instance de base de données RDS for PostgreSQL, vous pouvez le choisir (au lieu d'un groupe de paramètres par défaut) pendant le processus de création. Si vous effectuez cette opération alors que votre instance de base de données RDS for PostgreSQL est déjà en cours d'exécution, vous devez redémarrer l'instance pour que celle-ci utilise le groupe de paramètres personnalisés. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).

Lorsque la fonction `rds.force_ssl` est active sur votre instance de base de données, les tentatives de connexion qui n'utilisent pas SSL sont rejetées avec le message suivant :

```
$ psql -h db-name.555555555555.ap-southeast-1.rds.amazonaws.com port=5432 dbname=testDB
 user=testuser
psql: error: FATAL: no pg_hba.conf entry for host "w.x.y.z", user "testuser", database
 "testDB", SSL off
```

Détermination du statut de la connexion SSL

Le statut chiffré de votre connexion est affiché dans la page d'accueil d'ouverture de session lorsque vous vous connectez à l'instance de base de données :

```
Password for user master:
psql (10.3)
SSL connection (cipher: DHE-RSA-AES256-SHA, bits: 256)
Type "help" for help.
postgres=>
```

Vous pouvez également charger l'extension `sslinfo`, puis appeler la fonction `ssl_is_used()` pour déterminer si SSL est utilisé. La fonction renvoie `t` si la connexion utilise SSL ; sinon, elle renvoie `f`.

```
postgres=> CREATE EXTENSION sslinfo;
CREATE EXTENSION
postgres=> SELECT ssl_is_used();
ssl_is_used
---------
t
(1 row)
```

Pour obtenir des informations plus détaillées, vous pouvez utiliser la requête suivante pour obtenir des informations via `pg_settings` :

```
SELECT name as "Parameter name", setting as value, short_desc FROM pg_settings WHERE
 name LIKE '%ssl%';
             Parameter name             |                  value                  |
                  short_desc
----------------------------------------+-----------------------------------------
+-----------------------------------------------------
 ssl                                    | on                                      |
 Enables SSL connections.
 ssl_ca_file                            | /rdsdbdata/rds-metadata/ca-cert.pem     |
 Location of the SSL certificate authority file.
 ssl_cert_file                          | /rdsdbdata/rds-metadata/server-cert.pem |
 Location of the SSL server certificate file.
 ssl_ciphers                            | HIGH:!aNULL:!3DES                       |
 Sets the list of allowed SSL ciphers.
 ssl_crl_file                           |                                         |
 Location of the SSL certificate revocation list file.
 ssl_dh_params_file                     |                                         |
 Location of the SSL DH parameters file.
 ssl_ecdh_curve                         | prime256v1                              |
 Sets the curve to use for ECDH.
 ssl_key_file                           | /rdsdbdata/rds-metadata/server-key.pem  |
 Location of the SSL server private key file.
 ssl_library                            | OpenSSL                                 |
 Name of the SSL library.
 ssl_max_protocol_version               |                                         |
 Sets the maximum SSL/TLS protocol version to use.
```

```
 ssl_min_protocol_version               | TLSv1.2                                  |
 Sets the minimum SSL/TLS protocol version to use.
 ssl_passphrase_command                 |                                          |
 Command to obtain passphrases for SSL.
 ssl_passphrase_command_supports_reload | off                                      |
 Also use ssl_passphrase_command during server reload.
 ssl_prefer_server_ciphers              | on                                       |
 Give priority to server ciphersuite order.
(14 rows)
```

Vous pouvez également recueillir toutes les informations sur l'utilisation SSL de votre instance de base de données RDS for PostgreSQL par processus, client et application en utilisant la requête suivante :

```
SELECT datname as "Database name", usename as "User name", ssl, client_addr,
 application_name, backend_type
   FROM pg_stat_ssl
   JOIN pg_stat_activity
   ON pg_stat_ssl.pid = pg_stat_activity.pid
   ORDER BY ssl;
 Database name | User name | ssl |  client_addr   |    application_name    |
 backend_type
---------------+-----------+-----+----------------+------------------------
+------------------------------
               |           | f   |                |                        | autovacuum
 launcher
               | rdsadmin  | f   |                |                        | logical
 replication launcher
               |           | f   |                |                        | background
 writer
               |           | f   |                |                        |
 checkpointer
               |           | f   |                |                        | walwriter
 rdsadmin      | rdsadmin  | t   | 127.0.0.1      |                        | client
 backend
 rdsadmin      | rdsadmin  | t   | 127.0.0.1      | PostgreSQL JDBC Driver | client
 backend
 postgres      | postgres  | t   | 204.246.162.36 | psql                   | client
 backend
(8 rows)
```

Pour identifier le chiffrage utilisé pour votre connexion SSL, vous pouvez utiliser la requête suivante :

```
postgres=> SELECT ssl_cipher();
ssl_cipher
--------------------
DHE-RSA-AES256-SHA
(1 row)
```

Pour en savoir plus sur l'option `sslmode`, consultez Database connection control functions (Fonctions de contrôle des connexions aux bases de données) dans la documentation de PostgreSQL.

Suites de chiffrement SSL dans RDS for PostgreSQL

Le paramètre de configuration ssl_ciphers de PostgreSQL indique les catégories de suites de chiffrement autorisées pour les connexions SSL à la base de données lors de l'utilisation de TLS 1.2 ou d'une version antérieure.

Dans RDS pour PostgreSQL 16 et versions ultérieures, vous pouvez modifier `ssl_ciphers` le paramètre pour utiliser des valeurs spécifiques issues des suites de chiffrement autorisées. Il s'agit d'un paramètre dynamique qui ne nécessite pas le redémarrage de l'instance de base de données. Pour consulter les suites de chiffrement autorisées, utilisez la console Amazon RDS ou la commande CLI AWS suivante :

```
aws rds describe-db-parameters --db-parameter-group-name <your-parameter-group> --
region <region> --endpoint-url <endpoint-url> --output json | jq '.Parameters[] |
 select(.ParameterName == "ssl_ciphers")'
```

Le tableau suivant répertorie à la fois les suites de chiffrement par défaut et les suites de chiffrement autorisées pour les versions qui prennent en charge les configurations personnalisées.

Version du moteur PostgreSQL	Valeurs par défaut de la suite ssl_cipher	Autoriser les valeurs de la suite ssl_cipher personnalisée répertoriées
17	`HIGH:!aNULL:!3DES`	`TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384` `TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256`

Version du moteur PostgreSQL	Valeurs par défaut de la suite ssl_cipher	Autoriser les valeurs de la suite ssl_cipher personnalisée répertoriées

		`TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256`
		`TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384`
		`TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`

16	`HIGH:!aNULL:!3DES`	`TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384`
		`TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256`
		`TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256`
		`TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384`
		`TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`

15	`HIGH:!aNULL:!3DES`	Le ssl_ciphers personnalisé n'est pas pris en charge
14	`HIGH:!aNULL:!3DES`	Le ssl_ciphers personnalisé n'est pas pris en charge

Version du moteur PostgreSQL	Valeurs par défaut de la suite ssl_cipher	Autoriser les valeurs de la suite ssl_cipher personnalisée répertoriées
13	`HIGH:!aNULL:!3DES`	Le ssl_ciphers personnalisé n'est pas pris en charge
12	`HIGH:!aNULL:!3DES`	Le ssl_ciphers personnalisé n'est pas pris en charge
11.4 et versions mineures ultérieures	`HIGH:MEDIUM:+3DES:!aNULL:!RC4`	Le ssl_ciphers personnalisé n'est pas pris en charge
11.1, 11.2	`HIGH:MEDIUM:+3DES:!aNULL`	Le ssl_ciphers personnalisé n'est pas pris en charge
10.9 et versions mineures ultérieures	`HIGH:MEDIUM:+3DES:!aNULL:!RC4`	Le ssl_ciphers personnalisé n'est pas pris en charge
10.7 et versions mineures inférieures	`HIGH:MEDIUM:+3DES:!aNULL`	Le ssl_ciphers personnalisé n'est pas pris en charge

Pour configurer toutes les connexions d'instance afin d'utiliser la suite de `TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384` chiffrement, modifiez votre groupe de paramètres comme indiqué dans l'exemple suivant :

```
aws rds modify-db-parameter-group --db-parameter-group-name <your-parameter-group> --
parameters
 "ParameterName='ssl_ciphers',ParameterValue='TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384',ApplyMet
```

Cet exemple utilise un chiffrement ECDSA, qui oblige votre instance à utiliser une autorité de certification avec cryptographie à courbe elliptique (ECC) pour établir une connexion. Pour plus d'informations sur les autorités de certification fournies par Amazon RDS, consultez la section Autorités de certification.

Vous pouvez vérifier les chiffrements utilisés à l'aide des méthodes décrites dans la section Détermination de l'état de la connexion SSL.

Les chiffrements peuvent avoir des noms différents selon le contexte :

- Les chiffrements autorisés que vous pouvez configurer dans votre groupe de paramètres sont désignés par leur nom IANA.
- La bannière `sslinfo` et la bannière `psql` d'ouverture de session font référence aux chiffrements utilisant leur nom OpenSSL.

Par défaut, la valeur de `ssl_max_protocol_version` dans RDS pour PostgreSQL 16 et versions ultérieures est TLS v1.3. Vous devez définir la valeur de ce paramètre sur TLS v1.2 car TLS v1.3 n'utilise pas les configurations de chiffrement spécifiées dans le paramètre. `ssl_ciphers` Lorsque vous définissez la valeur comme TLS v1.2, les connexions utilisent uniquement les chiffrements que vous définissez dans. `ssl_ciphers`

```
aws rds modify-db-parameter-group --db-parameter-group-name <your-parameter-group> --
parameters
 "ParameterName='ssl_max_protocol_version',ParameterValue='TLSv1.2',ApplyMethod=immediate"
```

Pour garantir que les connexions à la base de données utilisent le protocole SSL, définissez la `rds.force_ssl parameter` valeur 1 dans votre groupe de paramètres. Pour plus d'informations sur les paramètres et les groupes de paramètres, consultez la section Groupes de paramètres pour Amazon RDS.

Mise à jour des applications pour se connecter aux SQL instances de base de données Postgre à l'aide de nouveaux certificatsSSL/TLS

Les certificats utilisés pour Secure Socket Layer ou Transport Layer Security (SSL/TLS) ont généralement une durée de vie définie. Lorsque les fournisseurs de services mettent à jour leurs certificats d'autorité de certification (CA), les clients doivent mettre à jour leurs applications pour utiliser les nouveaux certificats. Vous trouverez ci-dessous des informations sur la manière de déterminer si vos applications clientes utilisentSSL/TLSpour se connecter à votre SQL instance de base de données Amazon RDS for Postgre. Vous trouverez également des informations sur la façon de vérifier si ces applications vérifient le certificat du serveur lorsqu'elles se connectent.

> **Note**
>
> Une application cliente configurée pour vérifier le certificat du serveur avant la TLS connexionSSL/doit disposer d'un certificat CA valide dans le trust store du client. Mettez à jour le magasin de confiance du client lorsque cela est nécessaire pour de nouveaux certificats.

Une fois que vous avez mis à jour les certificats de l'autorité de certification dans les magasins d'approbations des applications clientes, vous pouvez soumettre les certificats de vos instances de bases de données à une rotation. Nous vous recommandons vivement de tester ces procédures dans un environnement de développement ou intermédiaire avant de les implémenter dans vos environnements de production.

Pour de plus amples informations sur la rotation de certificats, veuillez consulter Rotation de votre certificat SSL/TLS. Pour en savoir plus sur le téléchargement de certificats, consultez . Pour plus d'informations sur l'utilisation deSSL/TLSavec les SQL instances de base de données Postgre, consultezUtilisation de SSL avec une instance de base de données PostgreSQL.

Rubriques

- Déterminer si les applications se connectent aux SQL instances de base de données Postgre à l'aide de SSL
- Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter
- Mise à jour du magasin d'approbations de votre application
- Utilisation TLS des connexionsSSL/pour différents types d'applications

Déterminer si les applications se connectent aux SQL instances de base de données Postgre à l'aide de SSL

Dans la configuration de l'instance de base de données, vérifiez la valeur du paramètre `rds.force_ssl`. Par défaut, le `rds.force_ssl` paramètre est défini sur `0` (off) pour les instances de base de données utilisant des SQL versions de Postgre antérieures à la version 15. Par défaut, `rds.force_ssl` il est défini sur 1 (activé) pour les instances de base de données utilisant Postgre SQL version 15 et versions majeures ultérieures. Si le `rds.force_ssl` paramètre est défini sur 1 (on), les clients doivent utiliserSSL/TLSpour les connexions. Pour plus d'informations sur les groupes de paramètres, consultez Groupes de paramètres pour (Amazon RDS).

Si vous utilisez RDS Postgre SQL version 9.5 ou version majeure ultérieure et que ce paramètre n'`rds.force_ssl`est pas défini sur 1 (activé), interrogez la `pg_stat_ssl` vue pour vérifier les connexions à l'aide SSL de. Par exemple, la requête suivante renvoie uniquement les SSL connexions et les informations sur les clients qui les utilisentSSL.

```
SELECT datname, usename, ssl, client_addr
  FROM pg_stat_ssl INNER JOIN pg_stat_activity ON pg_stat_ssl.pid =
 pg_stat_activity.pid
```

```
  WHERE ssl is true and usename<>'rdsadmin';
```

Seules les lignes utilisant TLS les connexionsSSL/sont affichées avec des informations sur la connexion. Voici un exemple de sortie.

```
 datname  | usename | ssl | client_addr
----------+---------+-----+-------------
 benchdb  | pgadmin | t   | 53.95.6.13
 postgres | pgadmin | t   | 53.95.6.13
(2 rows)
```

La requête n'affiche que les connexions actives au moment de la requête. L'absence de résultats n'indique pas qu'aucune application n'utilise de SSL connexions. D'autres SSL connexions peuvent être établies à un moment différent.

Contrôle de la nécessité d'une vérification du certificat du client pour qu'il puisse se connecter

Lorsqu'un client, tel que psql ouJDBC, est configuré avec le SSL support, le client essaie d'abord de se connecter à la base de données SSL par défaut. Si le client ne parvient pas à se connecter avecSSL, il revient à se connecter sansSSL. Le `sslmode` mode par défaut est utilisé pour les deux clients basés sur libpq (tels que psql) et JDBC est défini sur. `prefer` Le certificat sur le serveur n'est vérifié que s'il `sslrootcert` est doté de la `sslmode` valeur set to `verify-ca` ou`verify-full`. En cas de non-validité du certificat, une erreur est déclenchée.

`PGSSLROOTCERT`À utiliser pour vérifier le certificat à l'aide de la variable d'`PGSSLMODE`environnement, avec la valeur `PGSSLMODE` définie sur `verify-ca` ou`verify-full`.

```
PGSSLMODE=verify-full PGSSLROOTCERT=/fullpath/ssl-cert.pem psql -h
 pgdbidentifier.cxxxxxxxx.us-east-2.rds.amazonaws.com -U masteruser -d postgres
```

Utilisez l'`sslrootcert`argument pour vérifier le certificat au format chaîne de connexion, avec `sslmode` défini sur `verify-ca` ou `verify-full` pour vérifier le certificat. `sslmode`

```
psql "host=pgdbidentifier.cxxxxxxxx.us-east-2.rds.amazonaws.com sslmode=verify-full
 sslrootcert=/full/path/ssl-cert.pem user=masteruser dbname=postgres"
```

Par exemple, dans le cas précédent, si vous utilisez un certificat racine non valide, une erreur similaire à celle qui suit s'affiche sur votre client.

```
psql: SSL error: certificate verify failed
```

Mise à jour du magasin d'approbations de votre application

Pour plus d'informations sur la mise à jour du trust store pour les SQL applications Postgre, consultez la section Connexions TCP /IP sécurisées avec SSL dans la documentation SQL Postgre.

Pour plus d'informations sur le téléchargement du certificat racine, consultez .

Pour obtenir des exemples de scripts qui importent des certificats, consultez Exemple de script pour importer les certificats dans votre magasin d'approbations.

> **Note**
>
> Lors de la mise à jour du magasin d'approbations, vous pouvez conserver les certificats plus anciens en complément de l'ajout des nouveaux certificats.

Utilisation TLS des connexionsSSL/pour différents types d'applications

Vous trouverez ci-dessous des informations sur l'utilisation TLS des connexionsSSL/pour différents types d'applications :

- psql

  Le client est appelé depuis la ligne de commande en spécifiant des options comme chaîne de connexion ou comme variables d'environnement. Pour les TLS connexionsSSL/, les options pertinentes sont `sslmode` (variable d'environnement`PGSSLMODE`), `sslrootcert` (variable d'environnement`PGSSLROOTCERT`).

  Pour la liste complète des options, consultez la section Mots clés des paramètres dans la SQL documentation Postgre. Pour la liste complète des variables d'environnement, consultez la section Variables d'environnement dans la SQL documentation Postgre.
- pgAdmin

  Ce client basé sur un navigateur est une interface plus conviviale pour se connecter à une base de données SQL Postgre.

  Pour plus d'informations sur la configuration des connexions, consultez la pgAdmin documentation.

- JDBC

  JDBCpermet de connecter des bases de données à des applications Java.

  Pour des informations générales sur la connexion à une SQL base de données Postgre avecJDBC, consultez la section Connexion à la base de données dans la documentation du SQL JDBC pilote Postgre. Pour plus d'informations sur la connexion avecSSL/TLS, consultez la section Configuration du client dans la documentation du SQL JDBC pilote Postgre.

- Python

  Une bibliothèque Python populaire pour se connecter aux SQL bases de données Postgre est`psycopg2`.

  Pour plus d'informations sur l'utilisation de `psycopg2`, veuillez consulter la documentation psycopg2. Pour un court didacticiel sur la façon de se connecter à une SQL base de données Postgre, consultez le didacticiel Psycopg2. Vous trouverez des informations sur les options acceptées par la commande de connexion dans The psycopg2 module content.

> **Important**
>
> Une fois que vous avez déterminé que vos connexions à la base de données utilisentSSL/TLSet que vous avez mis à jour le magasin de confiance des applications, vous pouvez mettre à jour votre base de données pour utiliser les certificats rds-ca-rsa 2048-g1. Pour obtenir des instructions, veuillez consulter l'étape 3 dans Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données.

Utilisation de l'authentification Kerberos avec Amazon RDS for PostgreSQL

Vous pouvez utiliser Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de bases de données qui exécute PostgreSQL. Pour ce faire, configurez votre instance de de base de données à utiliser AWS Directory Service for Microsoft Active Directory pour l'authentification Kerberos. AWS Directory Service for Microsoft Active Directory est également appelé AWS Managed Microsoft AD. C'est une fonctionnalité disponible avec AWS Directory Service. Pour en savoir plus, consultez Qu'est-ce que c'est AWS Directory Service ? dans le Guide AWS Directory Service d'administration.

Pour commencer, créez un AWS Managed Microsoft AD répertoire pour stocker les informations d'identification des utilisateurs. Vous fournissez ensuite à votre instance de bases de données PostgreSQL le domaine d'Active Directory ainsi que d'autres informations. Lorsque les utilisateurs s'authentifient auprès de l'instance de de bases de données PostgreSQL, les demandes d'authentification sont transférées vers l'annuaire AWS Managed Microsoft AD .

Vous pouvez gagner du temps et de l'argent en conservant toutes les informations d'identification dans le même annuaire. Vous avez un endroit centralisé de stockage et de gestion des informations d'identification pour plusieurs instances de base de données. L'utilisation d'un annuaire peut également améliorer votre profil de sécurité global.

Vous pouvez également accéder aux informations d'identification à partir de votre propre annuaire Microsoft Active Directory sur site. Pour ce faire, vous créez une relation de domaine d'approbation afin que l'annuaire AWS Managed Microsoft AD approuve votre annuaire Microsoft Active Directory sur site. De cette façon, vos utilisateurs peuvent accéder à vos instances PostgreSQL avec la même expérience d'authentification unique (SSO) Windows que lorsqu'ils accèdent aux charges de travail de votre réseau sur site.

Une base de données peut utiliser l'authentification par mot de passe ou l'authentification par mot de passe avec l'authentification Kerberos ou AWS Identity and Access Management (IAM). Pour plus d'informations sur l'authentification IAM, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

Rubriques

- Disponibilité des régions et des versions
- Présentation de l'authentification Kerberos pour les instances de base de données PostgreSQL

- Configuration de l'authentification Kerberos pour les instances de base de données PostgreSQL
- Gestion d'une RDS pour PostgreSQL dans un domaine Active Directory
- Connexion à PostgreSQL avec l'authentification Kerberos

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour obtenir plus d'informations sur la disponibilité des versions et des régions de RDS pour PostgreSQL avec authentification Kerberos, consultez Régions et moteurs de base de données pris en charge pour l'authentification Kerberos dans Amazon RDS.

Présentation de l'authentification Kerberos pour les instances de base de données PostgreSQL

Pour configurer l'authentification Kerberos pour une instance de base de données PostgreSQL, exécutez les étapes suivantes, décrites plus en détails par la suite :

1. AWS Managed Microsoft AD À utiliser pour créer un AWS Managed Microsoft AD répertoire. Vous pouvez utiliser le AWS Management Console AWS CLI, le ou l' AWS Directory Service API pour créer le répertoire. Veillez à ouvrir les ports sortants appropriés sur le groupe de sécurité de répertoire afin que le répertoire puisse communiquer avec l'instance.
2. Créez un rôle qui fournit à un accès Amazon RDS pour passer des appels vers votre AWS Managed Microsoft AD annuaire. Pour ce faire, créez un rôle AWS Identity and Access Management (IAM) qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess`

   Pour que le rôle IAM autorise l'accès, le point de terminaison AWS Security Token Service (AWS STS) doit être activé dans la AWS région appropriée pour votre AWS compte. AWS STS les points de terminaison sont actifs par défaut dans tous les cas Régions AWS, et vous pouvez les utiliser sans autre action. Pour plus d'informations, consultez la section Activation et désactivation AWS STS dans une AWS région dans le guide de l'utilisateur IAM.
3. Créez et configurez des utilisateurs dans l' AWS Managed Microsoft AD annuaire à l'aide des outils Microsoft Active Directory. Pour plus d'informations sur la création d'utilisateurs dans votre Active Directory, voir Gérer les utilisateurs et les groupes dans AWS Managed Microsoft AD dans le Guide d'AWS Directory Service administration.

4. Si vous prévoyez de localiser le répertoire et l'instance de base de données dans différents AWS comptes ou clouds privés virtuels (VPCs), configurez le peering VPC. Pour plus d'informations, consultez Qu'est-ce que l'appairage de VPC ? dans le Guide d'appairage de VPC Amazon.
5. Créez ou modifiez une instance de base de données PostgreSQL depuis la console, la CLI ou l'API RDS à l'aide de l'une des méthodes suivantes :
   - Création d'une instance de base de données Amazon RDS
   - Modification d'une RDS instance de base de données Amazon
   - Restauration vers une instance de base de données
   - Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS

   Vous pouvez localiser l'instance de dans le même Amazon Virtual Private Cloud (VPC) que le répertoire ou dans un autre compte AWS ou VPC. Lors de la création ou de la modification de l'instance de base de données PostgreSQL, procédez comme suit :
   - Fournissez l'identifiant du domaine (identifiant `d-*`) qui a été généré lors de la création de votre annuaire.
   - Fournissez le nom du rôle IAM que vous avez créé.
   - Veillez à ce que le groupe de sécurité de l'instance de base de données puisse recevoir le trafic entrant depuis le groupe de sécurité de l'annuaire.
6. Utilisez les informations d'identification de l'utilisateur principal RDS pour vous connecter à l'instance de base de données PostgreSQL. Créez l'utilisateur dans PostgreSQL en vue de son identification externe. Les utilisateurs identifiés en externe peuvent se connecter à l'instance de base de données PostgreSQL à l'aide de l'authentification Kerberos.

Configuration de l'authentification Kerberos pour les instances de base de données PostgreSQL

Pour configurer l'authentification Kerberos, procédez comme suit :

Rubriques

- Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD
- Étape 2 : (Facultatif) Créer une relation de confiance entre votre annuaire Active Directory sur site et AWS Directory Service
- Étape 3 : créer un rôle IAM pour RDS) afin d'accéder au AWS Directory Service
- Étape 4 : Créer et configurer des utilisateurs

- Étape 5 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données
- Étape 6 : Créer ou modifier une de cluster de bases de données PostgreSQL
- Étape 7 : Créer des utilisateurs PostgreSQL pour vos principaux Kerberos
- Étape 8 : Configurer un client PostgreSQL

Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD

AWS Directory Service crée un Active Directory entièrement géré dans le AWS cloud. Lorsque vous créez un AWS Managed Microsoft AD annuaire, il AWS Directory Service crée deux contrôleurs de domaine et deux serveurs DNS pour vous. Les serveurs de répertoire sont créés dans des sous-réseaux différents d'un VPC. Cette redondance permet de s'assurer que votre annuaire reste accessible, y compris en cas de défaillance.

Lorsque vous créez un AWS Managed Microsoft AD annuaire, AWS Directory Service exécute les tâches suivantes en votre nom :

- Configuration d'un annuaire Active Directory dans votre VPC.
- Création d'un compte d'administrateur d'annuaire avec le nom d'utilisateur `Admin` et le mot de passe spécifié. Ce compte est utilisé pour gérer votre annuaire.

  > **Important**
  >
  > Assurez-vous d'enregistrer ce mot de passe. AWS Directory Service ne stocke pas ce mot de passe et il ne peut pas être récupéré ou réinitialisé.

- Création d'un groupe de sécurité pour les contrôleurs de l'annuaire. Le groupe de sécurité doit autoriser la communication avec l'instance de base de données PostgreSQL.

Lorsque vous lancez AWS Directory Service for Microsoft Active Directory, AWS crée une unité organisationnelle (UO) qui contient tous les objets de votre répertoire. Cette unité d'organisation, qui porte le nom NetBIOS que vous avez entré lorsque vous avez créé votre annuaire, est située dans la racine du domaine. La racine du domaine est détenue et gérée par AWS.

Le `Admin` compte créé avec votre AWS Managed Microsoft AD annuaire dispose d'autorisations pour les activités administratives les plus courantes de votre unité d'organisation :

- Création, mise à jour et suppression des utilisateurs

- Ajouter des ressources à votre domaine, comme des serveurs de fichiers ou d'impression, puis attribuer des autorisations pour ces ressources aux utilisateurs dans votre unité d'organisation
- Créez des conteneurs OUs et des conteneurs supplémentaires
- Déléguer des autorités
- Restaurer des objets supprimés de la corbeille Active Directory
- Exécuter les modules Active Directory et DNS (Domain Name Service) pour Windows PowerShell sur le service Web Active Directory

Le compte `Admin` dispose également de droits pour exécuter les activités suivantes au niveau du domaine :

- Gérer les configurations DNS (ajouter, supprimer ou mettre à jour des enregistrements, des zones et des redirecteurs)
- Afficher les journaux d'évènements DNS
- Afficher les journaux d'évènements de sécurité

Pour créer un répertoire avec AWS Managed Microsoft AD

1. Dans le panneau de navigation de la console AWS Directory Service, choisissez Directories (Répertoires), puis Set up directory (Configurer le répertoire).
2. Choisissez AWS Managed Microsoft AD. AWS Managed Microsoft AD est la seule option actuellement prise en charge pour une utilisation avec Amazon RDS.
3. Choisissez Suivant.
4. Sur la page Enter directory information (Saisir les détails du répertoire), renseignez les informations suivantes :

   Edition

   Choisissez l'édition qui correspond à vos besoins.

   Nom de DNS de l'annuaire

   Nom complet de l'annuaire, par exemple **`corp.example.com`**.

   Nom NetBIOS de l'annuaire

   Nom court facultatif pour l'annuaire, par exemple `CORP`.

Description de l'annuaire

Description facultative de l'annuaire.

Mot de passe administrateur

Mot de passe de l'administrateur de l'annuaire. Le processus de création d'un annuaire crée un compte d'administrateur avec le nom utilisateur `Admin` et ce mot de passe.

Le mot de passe de l'administrateur de l'annuaire ne peut pas contenir le terme « admin ». Le mot de passe est sensible à la casse et doit comporter entre 8 et 64 caractères. Il doit également contenir au moins un caractère de trois des quatre catégories suivantes :

- Lettres minuscules (a–z)
- Lettres majuscules (A–Z)
- Chiffres (0–9)
- Caractères non alphanumériques (~!@#$%^&*_-+=`|\(){}[]:;"'<>,.?/)

Confirmer le mot de passe

Saisissez à nouveau le mot de passe de l'administrateur.

> **Important**
>
> Assurez-vous d'enregistrer ce mot de passe. AWS Directory Service ne stocke pas ce mot de passe et il ne peut pas être récupéré ou réinitialisé.

5. Choisissez Suivant.
6. Sur la page Choose VPC and subnets (Choisir un VPC et des sous-réseaux), indiquez les informations suivantes :

   VPC

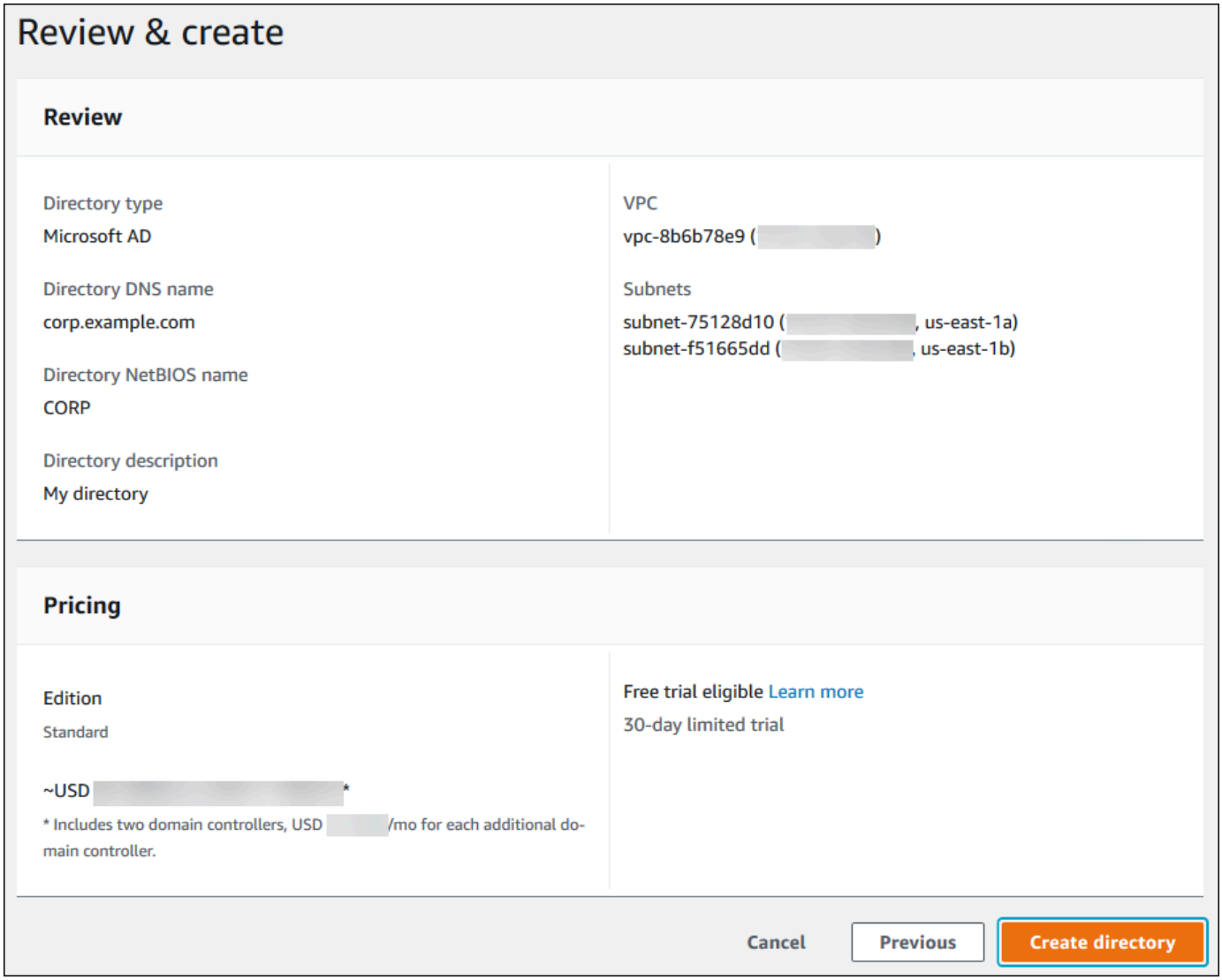
   Sélectionnez le VPC pour l'annuaire. Vous pouvez créer l'instance de base de données PostgreSQL dans ce même VPC ou dans un autre VPC.

   Sous-réseaux

   Choisissez les sous-réseaux pour les serveurs d'annuaires. Les deux sous-réseaux doivent être dans des zones de disponibilité différentes.

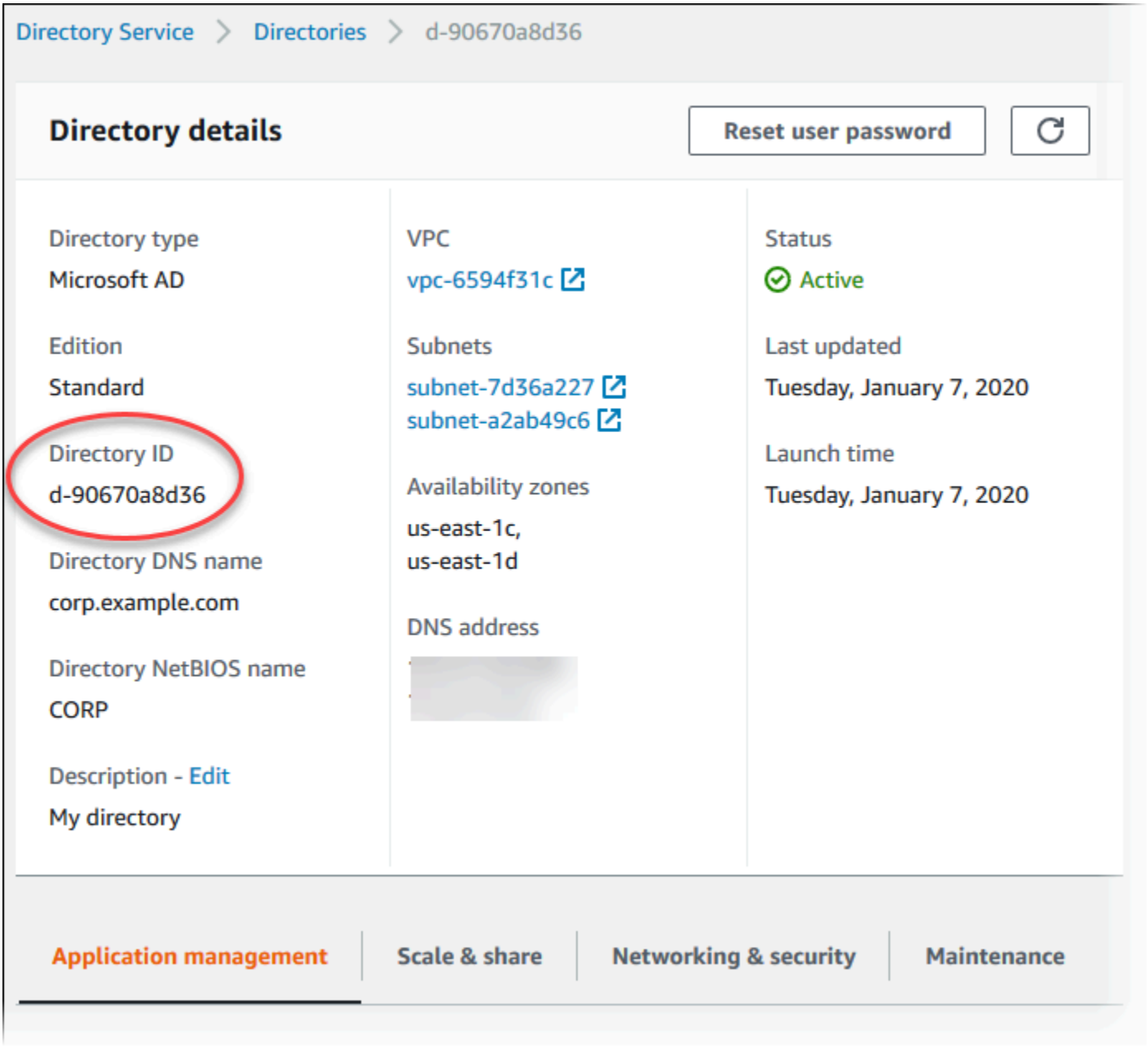
7. Choisissez Suivant.

8. Vérifiez les informations du répertoire. Si vous devez apporter des modifications, choisissez Previous (Précédent) et entrez ces modifications. Lorsque les informations sont correctes, choisissez Create directory (Créer l'annuaire).



La création de l'annuaire prend plusieurs minutes. Lorsqu'il est créé, la valeur du champ Statut devient Actif.

Pour consulter les informations relatives à votre annuaire, choisissez l'ID de l'annuaire dans la liste. Notez la valeur de Directory ID (ID du répertoire). Vous en aurez besoin pour créer ou modifier votre instance de base de données PostgreSQL.



Étape 2 : (Facultatif) Créer une relation de confiance entre votre annuaire Active Directory sur site et AWS Directory Service

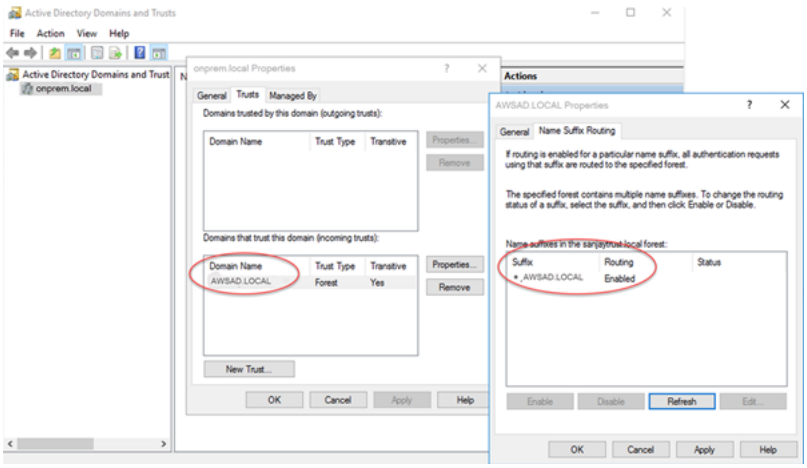
Si vous ne prévoyez pas d'utiliser votre propre Microsoft Active Directory sur site, passez à Étape 3 : créer un rôle IAM pour RDS) afin d'accéder au AWS Directory Service.

Pour obtenir l'authentification Kerberos à l'aide de votre Active Directory local, vous devez créer une relation de domaine de confiance à l'aide d'une approbation forestière entre votre Microsoft Active Directory local et l' AWS Managed Microsoft AD annuaire (créé dans). Étape 1 : créer un répertoire à l'aide de AWS Managed Microsoft AD La confiance peut être unidirectionnelle, lorsque l' AWS

Managed Microsoft AD annuaire fait confiance à Microsoft Active Directory local. L'approbation peut également être bidirectionnelle. Dans ce cas, les deux Active Directory s'approuvent mutuellement. Pour plus d'informations sur la configuration des approbations à l'aide AWS Directory Service de la section Quand créer une relation de confiance dans le Guide d'AWS Directory Service administration.

> **Note**
>
> Si vous utilisez un Microsoft Active Directory local, les clients Windows se connectent en utilisant le nom de domaine du AWS Directory Service point de terminaison plutôt que rds.amazonaws.com. Pour en savoir plus, veuillez consulter la section Connexion à PostgreSQL avec l'authentification Kerberos.

Assurez-vous que le nom de domaine de votre Microsoft Active Directory sur site inclut un routage de suffixe DNS correspondant à la relation d'approbation nouvellement créée. La capture d'écran suivante présente un exemple.



Étape 3 : créer un rôle IAM pour RDS) afin d'accéder au AWS Directory Service

Pour qu' Amazon RDS puisse vous appeler AWS Directory Service , votre AWS compte a besoin d'un rôle IAM qui utilise la politique IAM gérée. `AmazonRDSDirectoryServiceAccess` Ce rôle permet à Amazon RDS d'appeler AWS Directory Service.

Lorsque vous créez une instance de base de données à l'aide de AWS Management Console et que votre compte utilisateur de console dispose de l'`iam:CreateRole`autorisation, la console crée automatiquement le rôle IAM nécessaire. Dans ce cas, le nom du rôle est `rds-directoryservice-kerberos-access-role`. Sinon, vous devez créer le rôle IAM

manuellement. Lorsque vous créez ce rôle IAM`Directory Service`, choisissez et associez la politique AWS gérée `AmazonRDSDirectoryServiceAccess` à celui-ci.

Pour plus d'informations sur la création de rôles IAM pour un service, consultez la section Création d'un rôle pour déléguer des autorisations à un AWS service dans le Guide de l'utilisateur IAM.

> **Note**
>
> Le rôle IAM utilisé pour l'authentification Windows pour RDS for Microsoft SQL Server ne peut pas être utilisé pour Amazon RDS for PostgreSQL.

Vous pouvez également créer des stratégies avec les autorisations obligatoires au lieu d'utiliser la politique gérée `AmazonRDSDirectoryServiceAccess`. Dans ce cas, le rôle IAM doit avoir la politique d'approbation IAM suivante :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.amazonaws.com",
          "rds.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

Le rôle doit également avoir la politique de rôle IAM suivante.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "ds:DescribeDirectories",
        "ds:AuthorizeApplication",
        "ds:UnauthorizeApplication",
        "ds:GetAuthorizedApplicationDetails"
      ],
    "Effect": "Allow",
    "Resource": "*"
    }
  ]
}
```

Pour l'opt-in Régions AWS, utilisez les principes de service spécifiques à la région dans les politiques de confiance des rôles IAM. Lorsque vous créez une politique de confiance pour les services dans ces régions, spécifiez le code de région dans le principal du service.

L'exemple suivant montre une politique de confiance qui inclut des principes de service spécifiques à la région :

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "directoryservice.rds.REGION-CODE.amazonaws.com",
          "rds.REGION-CODE.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
```

```
    }
  ]
}
```

Remplacez REGION-CODE par le code de votre région spécifique. Par exemple, utilisez les principes de service suivants pour la région Asie-Pacifique (Melbourne) :

```
"Service": [
  "directoryservice.rds.ap-southeast-4.amazonaws.com",
  "rds.ap-southeast-4.amazonaws.com"
]
```

Étape 4 : Créer et configurer des utilisateurs

Vous pouvez créer des utilisateurs à l'aide de l'outil Active Directory Users and Computers (Utilisateurs et ordinateurs Active Directory). Cet outil fait partie des outils Services AD DS (Active Directory Domain Services) et Services AD LDS (Active Directory Lightweight Directory Services). Pour plus d'informations, consultez Ajouter des utilisateurs et des ordinateurs au domaine Active Directory dans la documentation Microsoft. Dans ce cas, les utilisateurs sont des individus ou d'autres entités, tels que leurs ordinateurs, qui font partie du domaine et dont les identités sont conservées dans l'annuaire.

Pour créer des utilisateurs dans un AWS Directory Service annuaire, vous devez être connecté à une EC2 instance Amazon basée sur Windows qui est membre de l' AWS Directory Service annuaire. Parallèlement, vous devez être connecté en tant qu'utilisateur disposant de privilèges pour créer des utilisateurs. Pour plus d'informations, consultez Créer un utilisateur dans le Guide d'administration AWS Directory Service .

Étape 5 : Activer le trafic entre VPC entre le répertoire et l'instance de base de données

Si vous avez l'intention de rechercher le répertoire et l'instance de base de données dans le même VPC, ignorez cette étape et passez à Étape 6 : Créer ou modifier une de cluster de bases de données PostgreSQL.

Si vous prévoyez de localiser le répertoire et l'instance de base de données différemment VPCs, configurez le trafic inter-VPC à l'aide du peering VPC ou de Transit Gateway.AWS

La procédure suivante active le trafic entre les utilisateurs de VPCs l'appairage VPC.
Suivez les instructions de Qu'est-ce que l'appairage de VPC ? dans le Guide de l'appairage Amazon Virtual Private Cloud.

Pour activer le trafic entre VPC à l'aide de l'appairage de VPC

1. Configurez les règles de routage de VPC appropriées afin de veiller à ce que le trafic réseau puisse être acheminé dans les deux sens.
2. Veillez à ce que le groupe de sécurité de l'instance de base de données puisse recevoir le trafic entrant depuis le groupe de sécurité de l'annuaire.
3. Assurez-vous qu'il n'existe aucune règle de liste de contrôle d'accès (ACL) pour bloquer le trafic.

Si le répertoire appartient à un autre AWS compte, vous devez le partager.

Pour partager le répertoire entre AWS comptes

1. Commencez à partager le répertoire avec le AWS compte dans lequel l'instance de base de données sera créée en suivant les instructions du Tutoriel : Partage de votre répertoire Microsoft AD AWS géré pour une connexion de EC2 domaine fluide dans le Guide d'AWS Directory Service administration.
2. Connectez-vous à la AWS Directory Service console à l'aide du compte de l'instance de base de données et assurez-vous que le domaine possède le `SHARED` statut requis avant de continuer.
3. Lorsque vous êtes connecté à la AWS Directory Service console à l'aide du compte de l'instance de base de données, notez la valeur de l'ID du répertoire. Vous utilisez cet ID pour joindre l'instance de base de données au domaine.

Étape 6 : Créer ou modifier une de cluster de bases de données PostgreSQL

Créez ou modifiez une instance de base de données PostgreSQL en vue de son utilisation avec votre répertoire. Vous pouvez utiliser la console, la CLI ou l'API RDS pour associer une instance de base de données à un répertoire. Vous pouvez effectuer cette opération de différentes manières :

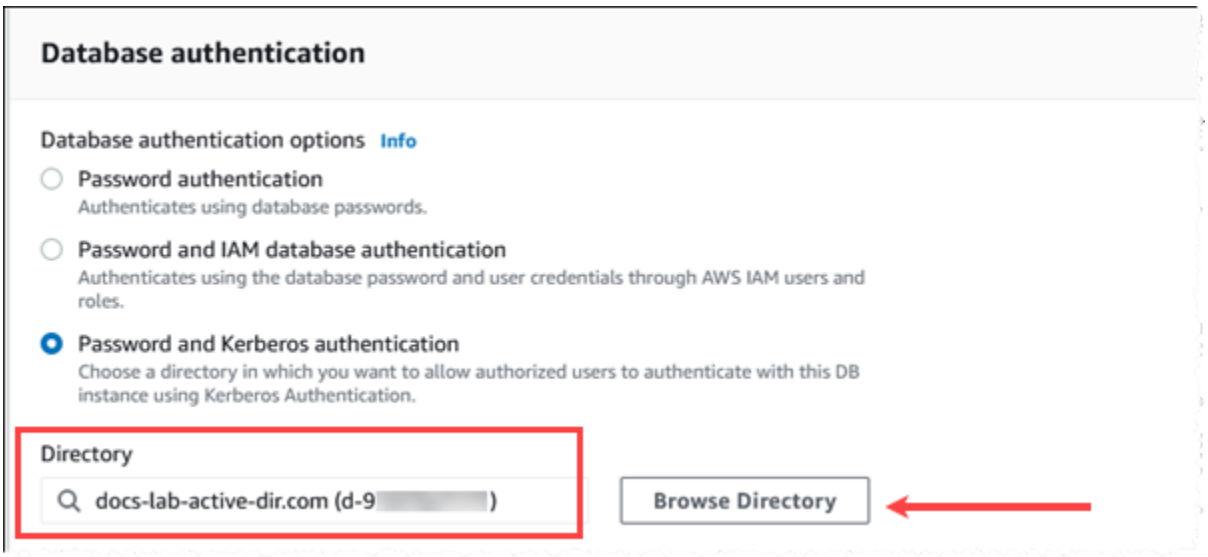
- Créez une nouvelle instance de base de données PostgreSQL à l'aide de la console, de la commande create-db-instanceCLI ou de l'opération DBInstance Create RDS API. Pour obtenir des instructions, veuillez consulter Création d'une instance de base de données Amazon RDS.

- Modifiez une instance de base de données PostgreSQL existante à l'aide de la console, de la commande modify-db-instanceCLI ou de l'opération DBInstance Modify RDS API. Pour obtenir des instructions, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Restaurez une instance de base de données PostgreSQL à partir d'un instantané de base de données à l'aide de la console, de la commande CLI restore-db-instance-from-db-snapshot ou de l'opération Restore From RDS API. DBInstance DBSnapshot Pour obtenir des instructions, veuillez consulter Restauration vers une instance de base de données.
- Restaurez une instance de base de données PostgreSQL à point-in-time l'aide de la console, de la commande restore-db-instance-to- point-in-time CLI ou de l'opération DBInstance ToPointInTime Restore RDS API. Pour obtenir des instructions, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

L'authentification Kerberos est uniquement prise en charge pour les instances de base de données PostgreSQL dans un VPC. L'instance de base de données peut se trouver dans le même VPC que le répertoire ou dans un autre VPC. L'instance de base de données doit utiliser un groupe de sécurité qui accepte les entrées et les sorties dans le VPC du répertoire pour permettre à l'instance de base de données de communiquer avec le répertoire.

console

Lorsque vous utilisez la console pour créer, modifier ou restaurer une instance de bases de données, choisissez Password and Kerberos authentication (Mot de passe et authentification Kerberos) dans la section Database authentication (Authentification de base de données). Ensuite, choisissez Parcourir les annuaires. Sélectionnez le répertoire ou choisissez Create a new directory (Créer un nouveau répertoire) pour utiliser Directory Service.



AWS CLI

Lorsque vous utilisez le AWS CLI, les paramètres suivants sont requis pour que l'instance de de base de données puisse utiliser le répertoire que vous avez créé :

- Pour le paramètre `--domain`, vous devez indiquer l'identifiant du domaine (identifiant « d-* ») généré lors de la création de l'annuaire.
- Pour le paramètre `--domain-iam-role-name`, utilisez le rôle que vous avez créé qui utilise la politique IAM gérée `AmazonRDSDirectoryServiceAccess`.

Par exemple, la commande de CLI suivante modifie une instance de base de données de façon à utiliser un répertoire.

```
aws rds modify-db-instance --db-instance-identifier mydbinstance --domain d-Directory-ID --domain-iam-role-name role-name
```

> **Important**
>
> Si vous modifiez une instance de base de données de façon à activer l'authentification Kerberos, redémarrez l'instance de base de données après avoir effectué la modification.

Étape 7 : Créer des utilisateurs PostgreSQL pour vos principaux Kerberos

À ce stade, votre instance de base de données RDS for PostgreSQL est jointe au domaine AWS Managed Microsoft AD . Les utilisateurs que vous avez créés dans l'annuaire dans Étape 4 : Créer et configurer des utilisateurs doivent être configurés en tant qu'utilisateurs de base de données PostgreSQL et bénéficier de privilèges leur permettant de se connecter à la base de données. Pour ce faire, vous devez vous connecter en tant qu'utilisateur de base de données doté de privilèges `rds_superuser`. Par exemple, si vous avez accepté les valeurs par défaut lors de la création de votre instance de base de données RDS for PostgreSQL, vous utilisez `postgres`, comme indiqué dans les étapes suivantes.

Pour créer des utilisateurs de base de données PostgreSQL pour les principaux Kerberos

1. Utilisez `psql` pour vous connecter à votre point de terminaison d'instance de base de données RDS for PostgreSQL à l'aide de `psql`. L'exemple suivant utilise le compte `postgres` par défaut pour le rôle `rds_superuser`.

```
psql --host=cluster-instance-1.111122223333.aws-region.rds.amazonaws.com --
port=5432 --username=postgres --password
```

2. Créez un nom d'utilisateur de base de données pour chaque principal Kerberos (nom d'utilisateur Active Directory) auquel vous souhaitez accorder l'accès à la base de données. Utilisez le nom d'utilisateur canonique (identité) tel que défini dans l'instance Active Directory, c'est-à-dire un `alias` en minuscule (nom d'utilisateur dans Active Directory) et le nom en majuscule du domaine Active Directory pour ce nom d'utilisateur. Le nom d'utilisateur Active Directory est un utilisateur authentifié de manière externe. Utilisez donc des guillemets autour du nom, comme indiqué ci-dessous.

```
postgres=> CREATE USER "username@CORP.EXAMPLE.COM" WITH LOGIN;
CREATE ROLE
```

3. Accordez le rôle `rds_ad` à l'utilisateur de la base de données.

```
postgres=> GRANT rds_ad TO "username@CORP.EXAMPLE.COM";
GRANT ROLE
```

Une fois que vous avez fini de créer tous les utilisateurs PostgreSQL pour vos identités utilisateur Active Directory, les utilisateurs peuvent accéder à l'instance de base de données RDS for PostgreSQL à l'aide de leurs informations d'identification Kerberos.

Les utilisateurs de base de données qui s'authentifient à l'aide de Kerberos doivent le faire à partir de machines clientes membres du domaine Active Directory.

Les utilisateurs de base de données auxquels le rôle `rds_ad` a été attribué ne peuvent pas disposer également du rôle `rds_iam`. Cela s'applique également aux adhésions imbriquées. Pour de plus amples informations, veuillez consulter Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

Étape 8 : Configurer un client PostgreSQL

Pour configurer un client PostgreSQL, procédez comme suit :

- Créez un fichier krb5.conf (ou équivalent) pointant vers le domaine.
- Vérifiez que le trafic peut circuler entre l'hôte client et AWS Directory Service. Utilisez un utilitaire réseau tel que Netcat pour les opérations suivantes :

  - Vérifiez le trafic via DNS pour le port 53.
  - Vérifiez le trafic dépassé TCP/UDP pour le port 53 et pour Kerberos, qui inclut les ports 88 et 464 pour. AWS Directory Service
- Vérifiez que le trafic peut circuler entre l'hôte du client et l'instance de base de données via le port de la base de données. Par exemple, utilisez psql pour vous connecter à la base de données et y accéder.

Voici un exemple de contenu du fichier krb5.conf pour. AWS Managed Microsoft AD

```
[libdefaults]
 default_realm = EXAMPLE.COM
[realms]
 EXAMPLE.COM = {
  kdc = example.com
  admin_server = example.com
 }
[domain_realm]
 .example.com = EXAMPLE.COM
 example.com = EXAMPLE.COM
```

Vous trouverez ci-après un exemple de contenu krb5.conf pour un Microsoft Active Directory sur site.

```
[libdefaults]
 default_realm = EXAMPLE.COM
[realms]
 EXAMPLE.COM = {
  kdc = example.com
  admin_server = example.com
 }
 ONPREM.COM = {
  kdc = onprem.com
  admin_server = onprem.com
 }
[domain_realm]
 .example.com = EXAMPLE.COM
 example.com = EXAMPLE.COM
 .onprem.com = ONPREM.COM
 onprem.com = ONPREM.COM
 .rds.amazonaws.com = EXAMPLE.COM
 .amazonaws.com.cn = EXAMPLE.COM
```

```
.amazon.com = EXAMPLE.COM
```

Gestion d'une RDS pour PostgreSQL dans un domaine Active Directory

Vous pouvez utiliser la console, la CLI ou l'API RDS pour gérer votre instance de base de données de de bases de données et sa relation avec votre Microsoft Active Directory. Par exemple, vous pouvez associer un annuaire Active Directory de façon à activer l'authentification Kerberos. Vous pouvez également supprimer l'association d'un annuaire Active Directory pour désactiver l'authentification Kerberos. Vous pouvez également transférer une instance de base de données vers un autre élément de même type afin de subir une authentification en externe par un annuaire Microsoft Active Directory.

Par exemple, la CLI vous permet d'effectuer les actions suivantes :

- Pour réessayer d'activer l'authentification Kerberos en cas d'échec d'adhésion, utilisez la commande CLI modify-db-instance. Spécifiez l'ID d'annuaire du membre actuel pour l'option `--domain`.
- Pour désactiver l'authentification Kerberos sur une instance de base de données, utilisez la commande modify-db-instanceCLI. Spécifiez `none` pour l'option `--domain`.
- Pour déplacer une instance de base de données d'un domaine à un autre, utilisez la commande modify-db-instanceCLI. Spécifiez l'identifiant du nouveau domaine pour l'option `--domain`.

Présentation de l'appartenance au domaine

Une fois que vous avez créé ou modifié votre instance de base de données, il devient membre du domaine. Vous pouvez consulter l'état de l'appartenance au domaine dans la console ou en exécutant la commande describe-db-instancesCLI. Le statut de l'instance de base de données peut avoir les valeurs suivantes :

- `kerberos-enabled` – L'instance de base de données a l'authentification Kerberos activée.
- `enabling-kerberos`— AWS est en train d'activer l'authentification Kerberos sur cette instance de base de données.
- `pending-enable-kerberos` – L'activation de l'authentification Kerberos est en attente sur cette instance de base de données.
- `pending-maintenance-enable-kerberos`— AWS tentera d'activer l'authentification Kerberos sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.

- `pending-disable-kerberos` – La désactivation de l'authentification Kerberos est en attente sur cette instance de base de données.
- `pending-maintenance-disable-kerberos`— AWS tentera de désactiver l'authentification Kerberos sur l'instance de base de données lors de la prochaine fenêtre de maintenance planifiée.
- `enable-kerberos-failed`— Un problème de configuration a AWS empêché l'activation de l'authentification Kerberos sur l'instance de base de données. Corrigez le problème de configuration avant de réémettre la commande de modification de l'instance de base de données.
- `disabling-kerberos`— AWS est en train de désactiver l'authentification Kerberos sur cette instance de base de données.

Une demande d'activation de l'authentification Kerberos peut échouer à cause d'un problème de connectivité réseau ou d'un rôle IAM incorrect. Dans certains cas, la tentative d'activation de l'authentification Kerberos peut échouer lorsque vous créez ou modifiez une instance de base de données. Si tel est le cas, vérifiez que vous utilisez le rôle IAM correct, puis modifiez l'instance de base de données afin d'effectuer son rattachement au domaine.

> **Note**
>
> Seule l'authentification Kerberos avec RDS for PostgreSQL envoie le trafic aux serveurs DNS du domaine. Toutes les autres demandes DNS sont traitées comme un accès réseau sortant sur vos instances de bases de données exécutant PostgreSQL. Pour plus d'informations sur l'accès réseau sortant avec RDS for PostgreSQL, veuillez consulter Utilisation d'un DNS serveur personnalisé pour l'accès au réseau sortant.

Connexion à PostgreSQL avec l'authentification Kerberos

Vous pouvez vous connecter à PostgreSQL via l'authentification Kerberos avec l'interface pgAdmin ou avec une CLI telle que psql. Pour plus d'informations sur la connexion, consultez Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL . Pour plus d'informations sur l'obtention du point de terminaison, du numéro de port et d'autres détails nécessaires à la connexion, consultez Se connecter à une instance DB PostgreSQL.

> **Note**
>
> L'authentification et le chiffrement GSSAPI dans PostgreSQL sont implémentés par la bibliothèque Kerberos. `libkrb5.so` Des fonctionnalités telles que `postgres_fdw` et

s'appuient `dblink` également sur cette même bibliothèque pour les connexions sortantes avec authentification ou cryptage Kerberos.

pgAdmin

Pour vous connecter à PostgreSQL avec l'authentification Kerberos en utilisant pgAdmin, procédez comme suit :

1. Lancez l'application pgAdmin sur votre ordinateur client.
2. Dans l'onglet Dashboard (Tableau de bord), choisissez Add New Server (Ajouter un nouveau serveur).
3. Dans la boîte de dialogue Create - Server (Créer – Serveur), entrez un nom sur l'onglet General (Général) pour identifier le serveur dans pgAdmin.
4. Dans l'onglet Connection (Connexion), entrez les informations suivantes à partir de votre base de données RDS for PostgreSQL.
   - Pour Host (Hôte), entrez le point de terminaison de . Instance de base de données RDS for PostgreSQL. Un point de terminaison ressemble à ce qui suit :

     ```
     RDS-DB-instance.111122223333.aws-region.rds.amazonaws.com
     ```

     Pour vous connecter à un Microsoft Active Directory local à partir d'un client Windows, vous devez utiliser le nom de domaine du AWS Managed Active Directory plutôt que celui du point `rds.amazonaws.com` de terminaison hôte. Supposons, par exemple, que le nom de domaine de AWS Managed Active Directory soit`corp.example.com`. Puis, pour Host (Hôte), le point de terminaison serait spécifié comme suit :

     ```
     RDS-DB-instance.111122223333.aws-region.corp.example.com
     ```

   - Pour Port, entrez le port attribué.
   - Pour Maintenance database (Base de données de maintenance), entrez le nom de la base de données initiale à laquelle le client se connectera.
   - Pour Username (Nom d'utilisateur), saisissez le nom d'utilisateur que vous avez entré pour l'authentification Kerberos dans Étape 7 : Créer des utilisateurs PostgreSQL pour vos principaux Kerberos .
5. Choisissez Enregistrer.

Psql

Pour vous connecter à PostgreSQL avec l'authentification Kerberos en utilisant psql, procédez comme suit :

1. A partir d'une invite de commande, exécutez la commande suivante.

   ```
   kinit username
   ```

   Remplacez *username* par le nom de l'utilisateur. À l'invite, entrez le mot de passe stocké dans le Microsoft Active Directory pour l'utilisateur.

2. Si l'instance de cluster de base de PostgreSQL utilise un VPC accessible au public, saisissez l'adresse IP du point de terminaison de votre instance de de base de données dans `/etc/hosts` votre fichier sur le client. EC2 Par exemple, les commandes suivantes permettent d'obtenir l'adresse IP et de la placer dans le fichier `/etc/hosts`.

   ```
   % dig +short PostgreSQL-endpoint.AWS-Region.rds.amazonaws.com
   ;; Truncated, retrying in TCP mode.
   ec2-34-210-197-118.AWS-Region.compute.amazonaws.com.
   34.210.197.118

   % echo " 34.210.197.118  PostgreSQL-endpoint.AWS-Region.rds.amazonaws.com" >> /etc/
   hosts
   ```

   Si vous utilisez une instance Microsoft Active Directory sur site à partir d'un client Windows, vous devez vous connecter à l'aide d'un point de terminaison spécifique. Au lieu d'utiliser le domaine Amazon `rds.amazonaws.com` dans le point de terminaison hôte, utilisez le nom de domaine du AWS Managed Active Directory.

   Supposons, par exemple, que le nom de domaine de votre annuaire Active Directory AWS géré soit`corp.example.com`. Alors, utilisez le format *PostgreSQL-endpoint.AWS-Region.corp.example.com* pour le point de terminaison et placez-le dans le fichier `/etc/hosts`.

   ```
   % echo " 34.210.197.118  PostgreSQL-endpoint.AWS-Region.corp.example.com" >> /etc/
   hosts
   ```

3. Utilisez la commande psql suivante pour vous connecter à une instance de base de données PostgreSQL intégré(e) à Active Directory.

```
psql -U username@CORP.EXAMPLE.COM -p 5432 -h PostgreSQL-endpoint.AWS-
Region.rds.amazonaws.com postgres
```

Pour vous connecter au cluster de base de données PostgreSQL à partir d'un client Windows à l'aide d'un Active Directory sur site, utilisez la commande psql suivante avec le nom de domaine de l'étape précédente (`corp.example.com`):

```
psql -U username@CORP.EXAMPLE.COM -p 5432 -h PostgreSQL-endpoint.AWS-
Region.corp.example.com postgres
```

Utilisation d'un DNS serveur personnalisé pour l'accès au réseau sortant

RDSfor Postgre SQL prend en charge l'accès réseau sortant sur vos instances de base de données et permet la résolution du Domain Name Service (DNS) à partir d'un DNS serveur personnalisé appartenant au client. Vous ne pouvez résoudre que des noms de domaine complets à partir de votre SQL instance RDS de base de données Postgre via votre DNS serveur personnalisé.

Rubriques

- Activer la DNS résolution personnalisée
- Désactiver la DNS résolution personnalisée
- Configuration d'un DNS serveur personnalisé

Activer la DNS résolution personnalisée

Pour activer la DNS résolution chez votre clientVPC, associez d'abord un groupe de paramètres de base de données personnalisé à votre SQL instance RDS for Postgre. Activez ensuite le paramètre `rds.custom_dns_resolution` en le définissant à la valeur 1, puis redémarrez l'instance de base de données pour que les modifications soient prises en compte.

Désactiver la DNS résolution personnalisée

Pour désactiver la DNS résolution chez votre clientVPC, désactivez d'abord le `rds.custom_dns_resolution` paramètre de votre groupe de paramètres de base de données personnalisé en le définissant sur 0. Redémarrez ensuite l'instance de base de données pour que les modifications soient prises en compte.

Configuration d'un DNS serveur personnalisé

Après avoir configuré votre serveur de DNS noms personnalisé, la propagation des modifications sur votre instance de base de données prend jusqu'à 30 minutes. Une fois les modifications propagées à votre instance de base de données, tout le trafic réseau sortant nécessitant une DNS recherche interroge votre DNS serveur via le port 53.

> **Note**
>
> Si vous ne configurez pas de DNS serveur personnalisé et que la valeur `rds.custom_dns_resolution` est définie sur 1, les hôtes sont résolus à l'aide d'une zone privée Amazon Route 53. Pour plus d'informations, veuillez onsulter Utilisation des zones hébergées privées.

Pour configurer un DNS serveur personnalisé RDS pour votre instance de SQL base de données Postgre

1. À partir des options du protocole Dynamic Host Configuration Protocol (DHCP) associées à votreVPC, définissez l'`domain-name-servers`option sur l'adresse IP de votre serveur de DNS noms. Pour plus d'informations, consultez la section Ensembles DHCP d'options.

   > **Note**
   >
   > L'`domain-name-servers`option accepte jusqu'à quatre valeurs, mais votre RDS instance de base de données Amazon n'utilise que la première valeur.

2. Assurez-vous que votre DNS serveur peut résoudre toutes les requêtes de recherche, y compris les DNS noms publics, les DNS noms EC2 privés Amazon et les noms spécifiques aux clientsDNS. Si le trafic réseau sortant contient des DNS recherches que votre DNS serveur ne peut pas gérer, les DNS fournisseurs en amont appropriés doivent être configurés sur votre DNS serveur.
3. Configurez votre DNS serveur pour produire des réponses User Datagram Protocol (UDP) de 512 octets ou moins.
4. Configurez votre DNS serveur pour produire des réponses du protocole de contrôle de transmission (TCP) de 1 024 octets ou moins.
5. Configurez votre DNS serveur pour autoriser le trafic entrant depuis vos RDS instances de base de données Amazon via le port 53. Si votre DNS serveur se trouve dans un AmazonVPC, VPC il doit disposer d'un groupe de sécurité contenant des règles entrantes autorisant le TCP trafic sur le port 53. UDP Si votre DNS serveur ne se trouve pas dans un AmazonVPC, il doit disposer des paramètres de pare-feu appropriés pour autoriser UDP le trafic TCP entrant sur le port 53.

   Pour plus d'informations, consultez les sections Groupes de sécurité pour vous VPC et Ajout et suppression de règles.

6. Configurez votre RDS instance VPC de base de données Amazon pour autoriser le trafic sortant via le port 53. Vous VPC devez disposer d'un groupe de sécurité contenant des règles sortantes autorisant le TCP trafic sur le port 53. UDP

   Pour plus d'informations, consultez les sections Groupes de sécurité pour vous VPC et Ajout et suppression de règles dans le guide de VPC l'utilisateur Amazon.
7. Assurez-vous que le chemin de routage entre l'RDSinstance de base de données Amazon et le DNS serveur est correctement configuré pour autoriser DNS le trafic.

   De plus, si l'RDSinstance de base de données Amazon et le DNS serveur ne se trouvent pas dans le même emplacementVPC, assurez-vous qu'une connexion d'appairage est établie entre eux. Pour plus d'informations, consultez Qu'est-ce que le VPC peering ? dans Amazon VPC Peering Guide.

Mises à niveau du moteur de base de données RDS pour PostgreSQL

Il existe deux types de mises à niveau que vous pouvez gérer pour votre base de données PostgreSQL :

- Mises à jour du système d'exploitation : il peut arriver qu'Amazon RDS doive mettre à jour le système d'exploitation sous-jacent de votre base de données afin d'appliquer des correctifs de sécurité ou des modifications du système d'exploitation. Vous pouvez décider quand Amazon RDS applique les mises à jour du système d'exploitation à l'aide de la console RDS, AWS Command Line Interface (AWS CLI) ou de l'API RDS. Pour plus d'informations sur les mises à jour de SE, consultez Appliquer des mises à jour à un d'instances de base de données.
- Mises à niveau du moteur de base de données : quand Amazon RDS prend en charge une nouvelle version d'un moteur de base de données, vous pouvez mettre à niveau vos bases de données vers cette nouvelle version.

Une base de données dans ce contexte est une instance de base de données RDS for PostgreSQL ou un cluster de bases de données multi-AZ.

Il existe deux types de mises à niveau du moteur pour les bases de données PostgreSQL : les mises à niveau des versions majeures et les mises à niveau des versions mineures.

Mises à niveau de version majeure.

Les mises à niveau de version majeure peuvent contenir des modifications de base de données qui ne sont pas rétrocompatibles avec les applications existantes. Par conséquent, vous devez effectuer manuellement les mises à niveau de version majeure de vos bases de données. Vous pouvez lancer une mise à niveau de version majeure en modifiant votre instance de base de données ou votre cluster de bases de données multi-AZ. Avant d'effectuer une mise à niveau de version majeure, nous vous recommandons de suivre les étapes décrites dans Choix d'une version majeure pour une mise à niveau de RDS pour PostgreSQL.

Amazon RDS gère les mises à niveau des versions majeures multi-AZ de la manière suivante :

- Déploiement d'instances de base de données multi-AZ : Amazon RDS met à niveau simultanément l'instance principale et toutes les instances de secours. Il se peut que votre base de données ne soit pas disponible pendant plusieurs minutes, le temps que la mise à niveau soit terminée.

- Déploiement de clusters de bases de données multi-AZ : Amazon RDS met à niveau simultanément les instances de lecteur et d'enregistreur. Il se peut que votre base de données ne soit pas disponible pendant plusieurs minutes, le temps que la mise à niveau soit terminée.

Si vous mettez à niveau une instance de base de données qui possède des répliques de lecture locales, Amazon RDS met à niveau les répliques en même temps que l'instance de base de données principale.

Amazon RDS ne met pas à niveau les réplicas en lecture d'un cluster de bases de données multi-AZ. Si vous effectuez une mise à niveau de version majeure d'un cluster de base de données multi-AZ, l'état de réplication de ses répliques de lecture devient terminé. Vous devez supprimer et recréer manuellement les réplicas en lecture une fois la mise à niveau terminée.

> **Tip**
>
> Vous pouvez minimiser le temps d'arrêt requis pour une mise à niveau de version majeure en utilisant un déploiement bleu/vert. Pour de plus amples informations, veuillez consulter Utilisation des déploiements bleu/vert pour les mises à jour de base de données.

Mises à niveau de version mineure.

En revanche, une mise à niveau de version mineure contient uniquement des modifications rétrocompatibles avec les applications existantes. Vous pouvez lancer manuellement une mise à niveau de version mineure en modifiant votre base de données. Vous pouvez également activer l'option de mise à niveau automatique des versions mineures lorsque vous créez ou modifiez une base de données. Cela signifie qu'Amazon RDS met automatiquement à niveau votre base de données après avoir testé et approuvé la nouvelle version.

Amazon RDS gère les mises à niveau des versions mineures multi-AZ de la manière suivante :

- Déploiement d'instances de base de données multi-AZ : Amazon RDS met à niveau simultanément l'instance principale et toutes les instances de secours. Il se peut que votre base de données ne soit pas disponible pendant plusieurs minutes, le temps que la mise à niveau soit terminée.
- Déploiement de clusters de bases de données multi-AZ : Amazon RDS met à niveau les instances de base de données du lecteur une par une. Ensuite, l'une des instances de base de données du lecteur devient la nouvelle instance de base de données du rédacteur. Amazon RDS met ensuite à niveau l'ancienne instance d'écriture (qui est désormais une instance de

lecteur). Les clusters de bases de données multi-AZ réduisent généralement le temps d'arrêt des mises à niveau de versions mineures à environ 35 secondes. Lorsqu'ils sont utilisés avec RDS Proxy, ils peuvent encore réduire les temps d'arrêt à une seconde ou moins. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS. Vous pouvez également utiliser un proxy de base de données open source tel que ProxySQL ou le pilote AWS Advanced JDBC Wrapper. PgBouncer

Si votre base de données possède des répliques en lecture, vous devez d'abord mettre à niveau toutes les répliques en lecture avant de mettre à niveau l'instance ou le cluster source.

Pour de plus amples informations, veuillez consulter Mises à niveau automatiques des versions mineures RDS pour Postgre SQL. Pour de plus amples informations sur l'exécution manuelle d'une mise à niveau de version mineure, veuillez consulter Mise à niveau manuelle de la version du moteur.

Pour plus d'informations sur les versions du moteur de base de données et la politique de dépréciation des versions du moteur de base de données, consultez la section Versions du moteur de base de données sur Amazon RDS. FAQs

Rubriques

- Considérations relatives aux mises à niveau de PostgreSQL
- Trouver des cibles de mise à niveau valides
- Numéros de SQL version de Postgre
- RDSnuméros de version RDS pour Postgre SQL
- Choix d'une version majeure pour une mise à niveau de RDS pour PostgreSQL
- Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL
- Mises à niveau automatiques des versions mineures RDS pour Postgre SQL
- Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre

Considérations relatives aux mises à niveau de PostgreSQL

Pour mettre à niveau vos bases de données en toute sécurité, Amazon RDS utilise l'`pg_upgrade`utilitaire décrit dans la documentation PostgreSQL

Si votre période de rétention des sauvegardes est supérieure à 0, Amazon RDS crée deux instantanés de base de données pendant la mise à niveau. Le premier instantané de base de

données porte sur la base de données avant que toute modification de mise à niveau soit apportée. Si la mise à niveau échoue pour vos bases de données, vous pouvez restaurer cet instantané afin de créer une base de données exécutant l'ancienne version. Le second instantané de base de données est pris à la fin de la mise à niveau. Ces instantanés de base de données sont automatiquement supprimés à l'expiration de la période de conservation des sauvegardes.

> **Note**
>
> Amazon RDS ne prend des instantanés de base de données pendant le processus de mise à niveau que si vous avez défini la période de conservation des sauvegardes de votre base de données sur un nombre supérieur à 0. Pour modifier la période de conservation des sauvegardes pour une instance de base de données, consultez the section called “Modification d'une instance de base de données”. Vous ne pouvez pas configurer une période de conservation personnalisée des sauvegardes pour un cluster de bases de données multi-AZ.

Quand vous effectuez une mise à niveau de version majeure d'une instance de base de donnée, tous les réplicas en lecture dans la région sont également automatiquement mis à niveau. Une fois que le flux de mise à niveau a démarré, les réplicas en lecture attendent que `pg_upgrade` se termine correctement sur l'instance de base de données principale. Ensuite, la mise à niveau de l'instance de base de données principale attend que les mises à niveau du réplica en lecture se terminent. Vous faites face à une panne tant que la mise à niveau n'est pas terminée. Quand vous effectuez une mise à niveau de version majeure d'un cluster de bases de données multi-AZ, l'état de réplication de ses réplicas en lecture devient résilié.

Une fois qu'une mise à niveau est terminée, vous ne pouvez pas rétablir la version précédente du moteur de base de données. Si vous souhaitez revenir à la version précédente, restaurez l'instantané de base de données pris avant la mise à niveau pour créer une nouvelle base de données.

Trouver des cibles de mise à niveau valides

Lorsque vous utilisez le AWS Management Console pour mettre à niveau une base de données, il indique les cibles de mise à niveau valides pour la base de données. Vous pouvez également utiliser la AWS CLI commande suivante pour identifier les cibles de mise à niveau valides pour une base de données :

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine postgres \
  --engine-version version-number \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine postgres ^
  --engine-version version-number ^
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Par exemple, pour identifier les cibles de mise à niveau valides pour une base de données PostgreSQL version 16.1, exécutez la commande suivante : AWS CLI

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
  --engine postgres \
  --engine-version 16.1 \
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
  --engine postgres ^
  --engine-version 16.1 ^
  --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Numéros de SQL version de Postgre

La séquence de numérotation des versions du moteur de SQL base de données Postgre est la suivante :

- Pour les SQL versions 10 et supérieures de Postgre, le numéro de version du moteur est au format major.minor. Le numéro de version majeure est la partie entière du numéro de version. Le numéro de version mineure est la partie fractionnaire du numéro de version.

Une mise à niveau de version majeure augmente la partie entière du numéro de version, par exemple la mise à niveau de 10.mineure à 11.mineure.

- Pour les SQL versions de Postgre inférieures à 10, le numéro de version du moteur est au format major.major.minor. Le numéro de version majeure du moteur est à la fois l'entier et la première partie fractionnaire du numéro de version. Par exemple, 9.6 est une version majeure. Le numéro de version mineure est la troisième partie du numéro de version. Par exemple, pour la version 9.6.12, le numéro 12 est le numéro de version mineure.

  Une mise à niveau majeure augmente la partie majeure du numéro de version. Par exemple, une mise à niveau de la version 9.6.12 vers la version 11.14 est une mise à niveau de version majeure, 9.6 et 11 étant les numéros des versions majeures.

Pour plus d'informations sur la numérotation des versions du Support RDS étendu, consultezDénomination de la version d'Amazon RDS Extended Support.

RDSnuméros de version RDS pour Postgre SQL

RDSles numéros de version utilisent le schéma de *major*.*minor*.*patch* dénomination. Une version de RDS correctif inclut des corrections de bogues importantes ajoutées à une version mineure après sa sortie. Pour plus d'informations sur la numérotation des versions du Support RDS étendu, consultezDénomination de la version d'Amazon RDS Extended Support.

Pour identifier le numéro de RDS version Amazon de votre base de données, vous devez d'abord créer l'`rds_tools`extension à l'aide de la commande suivante :

```
CREATE EXTENSION rds_tools;
```

À partir de la sortie de la SQL version 15.2-R2 de Postgre, vous pouvez connaître le numéro de RDS version de votre SQL base de données Postgre RDS à l'aide de la requête suivante : SQL

```
postgres=> SELECT rds_tools.rds_version();
```

Par exemple, l'interrogation d'une base RDS de données Postgre SQL 15.2 renvoie ce qui suit :

```
rds_version
----------------
 15.2.R2
(1 row)
```

Choix d'une version majeure pour une mise à niveau de RDS pour PostgreSQL

Les mises à niveau de versions majeures peuvent contenir des modifications qui ne sont pas rétrocompatibles avec les versions précédentes de la base de données. Les nouvelles fonctionnalités peuvent empêcher vos applications existantes de fonctionner correctement. Pour cette raison, Amazon RDS n'applique pas automatiquement les mises à niveau des versions majeures. Pour effectuer une mise à niveau de version majeure, vous modifiez manuellement votre base de données. Veillez à tester soigneusement toute mise à niveau pour vérifier que vos applications fonctionnent correctement avant d'appliquer la mise à niveau à vos bases de données de production. Lorsque vous effectuez une mise à niveau de version majeure de PostgreSQL, nous vous recommandons de suivre les étapes décrites dans Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL.

Lorsque vous mettez à niveau une instance de base de données mono-AZ PostgreSQL ou un déploiement d'instance de base de données multi-AZ vers sa prochaine version majeure, tous les réplicas en lecture associés à la base de données sont également mis à niveau vers cette prochaine version majeure. Dans certains cas, vous pouvez passer à une version majeure ultérieure lors de la mise à niveau. Si votre mise à niveau ignore une version majeure, les réplicas en lecture sont également mis à niveau vers cette version majeure cible. Les mises à niveau vers la version 11 qui sautent d'autres versions majeures présentent certaines limitations. Vous trouverez les détails dans les étapes décrites dans la section Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL.

La plupart des extensions PostgreSQL ne sont pas mises à jour lors d'une mise à niveau du moteur PostgreSQL. Elles doivent être mises à niveau séparément. Pour de plus amples informations, veuillez consulter Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre.

Vous pouvez savoir quelles versions majeures sont disponibles pour votre base de données RDS pour PostgreSQL en exécutant la requête suivante : AWS CLI

```
aws rds describe-db-engine-versions --engine postgres  --engine-version your-version
 --query "DBEngineVersions[*].ValidUpgradeTarget[*].{EngineVersion:EngineVersion}" --
output text
```

Le tableau suivant récapitule les résultats de cette requête pour toutes les versions disponibles. Un astérisque (*) sur le numéro de version signifie que la version n'est plus prise en charge. Si votre

version actuelle n'est pas prise en charge, nous vous recommandons de passer à la dernière cible de mise à niveau de la version mineure ou à l'une des autres cibles de mise à niveau disponibles pour cette version.

Version source actuelle	Cibles de mise à niveau
17,6	Aucun
17,5	17,6
17,4	17,6, 17,5
17,3*, 17,2	17,6, 17,5, 17,4
17,1*	17,6, 17,5, 17,4, 17,2
16,10	17,6
16,9	17,6, 17,5 16,10
16,8	17,6, 17,5, 17,4 16,10, 16,9
16,7*	17,6, 17,5, 17,4 16,10, 16,9, 16,8
16,7	17,6, 17,5, 17,4 16,10, 16,9, 16,8
16,6	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8
16,5*, 16,4	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6

Version source actuelle	Cibles de mise à niveau
16,3	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4
16,2*, 16,1*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3
15,14	17,6 16,10
15,13	17,6, 17,5 16,10, 16,9 15,14
15,12, 15,11*	17,6, 17,5, 17,4 16,10, 16,9, 16,8 15,14, 15,13
15,10	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12
15,9*	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12, 15,10
15,8	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4 15,14, 15,13, 15,12, 15,10

Version source actuelle	Cibles de mise à niveau
15,7	16,9 , 16,8, 16,7, 16,6, 16,5, 16,4, 16,3 15,13 , 15,12, 15,11, 15,10, 15,9, 15,8
15,6*, 15,5*, 15,4*, 15,3*, 15,2*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7
14,19	17,6 16,10 15,14
14,18	17,6, 17,5 16,10, 16,9 15,14, 15,13 14,19
14,17, 14,16*	17,6, 17,5, 17,4 16,10, 16,9, 16,8 15,14, 15,13, 15,12 14,19, 14,18
14,15	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12, 15,10 14,19, 14,18, 14,17

Version source actuelle	Cibles de mise à niveau
14,14*	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12, 15,10 14,19, 14,18, 14,17, 14,15
14,13	16,4 15,13 , 15,12, 15,11, 15,10, 15,9, 15,8 14,18 , 14,17, 14,16, 14,15, 14,14
14,12	16,3 15,13 , 15,12, 15,11, 15,10, 15,9, 15,8, 15,7 14,18 , 14,17, 14,16, 14,15, 14,13
14,10*, 14,10*, 14,9*, 14,8*, 14,7*, 14,6*, 14,5*, 14,4*, 14,3*, 14,2*, 14,1*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7 14,19 , 14,18, 14,17, 14,15, 14,13, 14,12
13,22	17,6 16,10 15,14 14,19

Version source actuelle	Cibles de mise à niveau
13,21	17,6, 17,5 16,10, 16,9 15,14, 15,13 14,19, 14,18 13,22
13,20, 13,19*	17,6, 17,5, 17,4 16,10, 16,9, 16,8 15,14, 15,13, 15,12 14,19, 14,18, 14,17 13,22, 13,21
13,18, 13,17*	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12, 15,10 14,19, 14,18, 14,17, 14,15 13,22, 13,21, 13,20
13,16	16,4 15,8 14,18 , 14,17, 14,15, 14,14, 14,13 13,21 , 13,20, 13,19, 13,18, 13,17

Version source actuelle	Cibles de mise à niveau
13,15	16,3 15,8, 15,7 14,18 , 14,17, 14,15, 14,14, 14,13, 14,12 13,21 , 13,20, 13,19, 13,18, 13,17, 13,16
13,14*, 13,13*, 13,12*, 13,10*, 13,9*, 13,8*, 13,7*, 13,6*, 13,5*, 13,5*, 13,4*, 13,3*, 13,1*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7 14,19 , 14,18, 14,17, 14,15, 14,13, 14,12 13,22 , 13,21, 13,20, 13,18, 13,16, 13,15
12.22-rds.20250508	17,6, 17,5 16,10, 16,9 15,14, 15,13 14,19, 14,18 13,22, 13,21
12.22-rds.20250220	17,6, 17,5, 17,4 16,10, 16,9, 16,8 15,14, 15,13, 15,12 14,19, 14,18, 14,17 13,22, 13,21, 13,20 12.22-rds.20250508

Version source actuelle	Cibles de mise à niveau
12,22, 12,21*	17,6, 17,5, 17,4, 17,2 16,10, 16,9, 16,8, 16,6 15,14, 15,13, 15,12, 15,10 14,19, 14,18, 14,17, 14,15 13,22, 13,21, 13,20, 13,18 12.22-rds.20250220, 12.22-rds.20250508
12,20*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4 15,14 , 15,13, 15,12, 15,10, 15,8 14,19 , 14,18, 14,17, 14,15, 14,13 13,22 , 13,21, 13,20, 13,18, 13,16 12.22, 12.22-rds.20250220, 12.22-rds.20250508
12,18*, 12,18*, 12,16*, 12,15*, 12,14*, 12,13*, 12,12*, 12,11*, 12,10*, 12,9*, 12,8*, 12,7*, 12,6*, 12,5*, 12,5*, 12,2*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7 14,19 , 14,18, 14,17, 14,15, 14,13, 14,12 13,22 , 13,21, 13,20, 13,18, 13,16, 13,15 12.22, 12.22-rds.20250220, 12.22-rds.20250508

Version source actuelle	Cibles de mise à niveau
11.22-rds.20250508	17,6, 17,5 16,10, 16,9 15,14, 15,13 14,19, 14,18 13,22, 13,21 12.22-rds.20250508
11.22-rds.20250220	17,6, 17,5, 17,4 16,10, 16,9, 16,8 15,14, 15,13, 15,12 14,19, 14,18, 14,17 13,22, 13,21, 13,20 12.22-rds.20250220, 12.22-rds.20250508 11.22-rds.20250508

Version source actuelle	Cibles de mise à niveau
11.22-rds.20240509	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7 14,19 , 14,18, 14,17, 14,15, 14,13, 14,12 13,22 , 13,21, 13,20, 13,18, 13,16, 13,15 12.22, 12.22-rds.20250220, 12.22-rds.20250508 11.22-rds.20240808, 11.22-rds.20241121, 11.22-rds.20250220 , 11.22-rds.20250508
11,22, 11,21*, 11,20*, 11,19*, 11,18*, 11,16*, 11,15*, 11,14*, 11,13*, 11,12*, 11,10*, 11,9*, 11,8*, 11,7*, 11,6*, 11,5*, 11,4*, 11,2*, 11,1*	17,6, 17,5, 17,4, 17,2 16,10 , 16,9, 16,8, 16,6, 16,4, 16,3 15,14 , 15,13, 15,12, 15,10, 15,8, 15,7 14,19 , 14,18, 14,17, 14,15, 14,13, 14,12 13,22 , 13,21, 13,20, 13,18, 13,16, 13,15 12.22, 12.22-rds.20250220, 12.22-rds.20250508 11.22 rds.20240418, 11.22 rds.20240509, 11.22 rds.20240808, 11.22 rds.20241121 , 11.22 rds.20250220, 11.22-rds.20250508

* Cette version n'est plus prise en charge.

Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL

Nous vous recommandons le processus suivant lors d'une mise à niveau de version majeure sur une base de données Amazon RDS for PostgreSQL :

1. Préparez un groupe de paramètres compatible avec la version – Si vous utilisez un groupe de paramètres personnalisé, vous avez deux options. Vous pouvez spécifier un groupe de paramètres par défaut pour la nouvelle version du moteur de base de données. Ou vous pouvez créer votre propre groupe de paramètres personnalisé pour la nouvelle version du moteur de base de données Pour plus d'informations, consultez the section called “Groupes de paramètres” et the section called “Groupes de paramètres de cluster DB”.
2. Vérifiez les classes de bases de données non prises en charge : vérifiez que la classe d'instances de votre base de données est compatible avec la version PostgreSQL vers laquelle vous effectuez la mise à niveau. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour les classes d'instance de base de données.
3. Recherchez une utilisation non prise en charge :
   - Transactions préparées – Validez ou restaurez toutes les transactions préparées ouvertes avant d'essayer d'effectuer une mise à niveau.

     Vous pouvez utiliser la requête suivante pour vérifier qu'aucune transaction préparée ouverte ne figure sur votre base de données.

     ```
     SELECT count(*) FROM pg_catalog.pg_prepared_xacts;
     ```

   - Types de données Reg* – Supprimez toutes les utilisations des types de données reg* avant d'essayer d'effectuer une mise à niveau. À l'exception de `regtype` et `regclass`, vous ne pouvez pas mettre à niveau les types de données reg*. L'utilitaire `pg_upgrade` ne peut pas conserver ce type de données, qui est utilisé par Amazon RDS pour effectuer la mise à niveau.

     Pour vérifier l'absence de toute utilisation des types de données reg* non pris en charge, utilisez la requête suivante pour chaque base de données.

     ```
     SELECT count(*) FROM pg_catalog.pg_class c, pg_catalog.pg_namespace n,
      pg_catalog.pg_attribute a
       WHERE c.oid = a.attrelid
           AND NOT a.attisdropped
     ```

```
      AND a.atttypid IN ('pg_catalog.regproc'::pg_catalog.regtype,
                         'pg_catalog.regprocedure'::pg_catalog.regtype,
                         'pg_catalog.regoper'::pg_catalog.regtype,
                         'pg_catalog.regoperator'::pg_catalog.regtype,
                         'pg_catalog.regconfig'::pg_catalog.regtype,
                         'pg_catalog.regdictionary'::pg_catalog.regtype)
      AND c.relnamespace = n.oid
      AND n.nspname NOT IN ('pg_catalog', 'information_schema');
```

4. Vérifiez la présence de bases de données non valides :

   - Assurez-vous qu'aucune base de données n'est invalide. La `datconnlimit` colonne du `pg_database` catalogue inclut une valeur de `-2` pour marquer comme non valides les bases de données qui ont été interrompues au cours d'une `DROP DATABASE` opération.

     Utilisez la requête suivante pour vérifier la présence de bases de données non valides :

     ```
     SELECT datname FROM pg_database WHERE datconnlimit = - 2;
     ```

   - La requête précédente renvoie des noms de base de données non valides. Vous pouvez l'utiliser `DROP DATABASE` *`invalid_db_name`*`;` pour supprimer les bases de données non valides. Vous pouvez également utiliser la commande suivante pour supprimer les bases de données non valides :

     ```
     SELECT 'DROP DATABASE ' || quote_ident(datname) || ';' FROM pg_database WHERE
      datconnlimit = -2 \gexec
     ```

   Pour plus d'informations sur les bases de données non valides, voir Comprendre le comportement d'Autovacuum avec des bases de données non valides.

5. Traitez les emplacements de réplication logique : une mise à niveau ne peut pas se produire si la base de données possède des emplacements de réplication logique. Les emplacements de réplication logique sont généralement utilisés pour la migration AWS DMS et la réplication de tables de la base de données vers des lacs de données, des outils BI et d'autres cibles. Avant de procéder à la mise à niveau, assurez-vous de connaître l'objectif des emplacements de réplication logique utilisés et confirmez qu'il est possible de les supprimer. Si les emplacements de réplication logique sont toujours utilisés, vous ne devez pas les supprimer et vous ne pouvez pas procéder à la mise à niveau.

Si les emplacements de réplication logique ne sont pas nécessaires, vous pouvez les supprimer à l'aide du code SQL suivant :

```
SELECT * FROM pg_replication_slots WHERE slot_type NOT LIKE 'physical';
SELECT pg_drop_replication_slot(slot_name);
```

Les installations de réplication logique qui utilisent l'extension `pglogical` doivent également avoir des emplacements supprimés pour une mise à niveau de version majeure réussie. Pour obtenir des informations sur la façon d'identifier et de supprimer les emplacements créés à l'aide de l'extension `pglogical`, consultez Gestion des emplacements de réplication logiques pour SQL.

6. Traitez les réplicas en lecture : une mise à niveau d'une instance de base de données mono-AZ ou d'un déploiement d'instance de base de données multi-AZ met également à niveau les réplicas en lecture dans la région ainsi que l'instance de base de données principale. Amazon RDS ne met pas à niveau les réplicas en lecture d'un cluster de bases de données multi-AZ.

   Vous ne pouvez pas mettre à niveau les réplicas en lecture séparément. Si vous le pouviez, cela pourrait conduire à des situations où les bases de données principale et de réplica auraient des versions majeures différentes de PostgreSQL. Toutefois, les mises à niveau de réplica en lecture peuvent augmenter les temps d'arrêt sur l'instance de base de données principale. Pour éviter la mise à niveau d'un réplica en lecture, promouvez le réplica en instance autonome ou supprimez-le avant de démarrer le processus de mise à niveau.

   Le processus de mise à niveau recrée le groupe de paramètres du réplica en lecture en fonction du groupe de paramètres actuel du réplica en lecture. Vous pouvez appliquer un groupe de paramètres personnalisé à un réplica en lecture uniquement une fois la mise à niveau terminée en modifiant le réplica en lecture. Pour plus d'informations sur les réplicas en lecture, consultez Utilisation de réplicas en lecture pour Amazon RDS for PostgreSQL.

7. Gérez les intégrations sans ETL : si vous avez une intégration zéro ETL existante, supprimez-la avant d'effectuer une mise à niveau de version majeure. Ensuite, une fois la mise à niveau terminée, recréez l'intégration.

8. Effectuez une sauvegarde – Nous vous recommandons d'effectuer une sauvegarde avant d'effectuer la mise à niveau de version majeure, afin de disposer d'un point de restauration connu pour votre base de données. Si la période de conservation des sauvegardes est supérieure à 0, le processus de mise à niveau crée des instantanés de base de données de votre base de données avant et après la mise à niveau. Pour modifier la période de rétention des sauvegardes, consultez

Modification d'une RDS instance de base de données Amazon et the section called “Modification d'un cluster de base de données multi-AZ”.

Pour effectuer une sauvegarde manuellement, consultez the section called “Création d'un instantané de base de données pour une instance de base de données mono-AZ” et the section called “Création d'un instantané de cluster de bases de données multi-AZ”.

9. Mise à niveau de certaines extensions avant une mise à niveau de version majeure – Si vous envisagez d'omettre une version majeure avec la mise à niveau, vous devez mettre à jour certaines extensions avant d'effectuer la mise à niveau de version majeure. Par exemple, la mise à niveau des versions 9.5.x ou 9.6.x vers une version 11.x entraîne une omission de la version majeure. Les extensions à mettre à jour comprennent PostGIS et les extensions connexes pour le traitement des données spatiales.

   - `address_standardizer`
   - `address_standardizer_data_us`
   - `postgis_raster`
   - `postgis_tiger_geocoder`
   - `postgis_topology`

   Vous ne pouvez pas passer directement à la version 17 de PostgreSQL si vous `rdkit` utilisez la version 4.6.0 ou inférieure, et la version 16 ou inférieure de PostgreSQL, en raison d'une incompatibilité. `rdkit` Vous trouverez ci-dessous les options de mise à niveau :

   - Si vous utilisez PostgreSQL version 13 ou inférieure, vous devez d'abord effectuer une mise à niveau majeure vers la version 14.14 et supérieure 14, les versions 15.9 et supérieures 15, ou les versions 16.5 et supérieures 16, puis effectuer la mise à niveau de version vers PostgreSQL 17.
   - Si vous utilisez PostgreSQL version 14, 15 ou 16, vous devez effectuer une mise à niveau de la version mineure vers les versions 14.14 et supérieures 14, 15.9 et supérieures 15, ou 16.5 et versions 16 supérieures, puis passer à PostgreSQL version 17.

   Exécutez la commande suivante pour chaque extension que vous utilisez :

   ```
   ALTER EXTENSION PostgreSQL-extension UPDATE TO 'new-version';
   ```

   Pour de plus amples informations, veuillez consulter Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre. Pour en savoir plus sur la mise à niveau de PostGIS, consultez Étape 6 : mise à niveau de l'GISextension Post.

10. Supprimez certaines extensions avant une mise à niveau de version majeure – Une mise à niveau qui omet une version majeure pour passer directement à la version 11.x ne prend pas en charge la mise à jour de l'extension `pgRouting`. La mise à niveau des versions 9.4.x, 9.5.x ou 9.6.x vers les versions 11.x entraîne l'omission d'une version majeure. Il est plus sûr de supprimer l'extension `pgRouting` puis de la réinstaller sur une version compatible une fois la mise à niveau effectuée. Pour connaître les versions d'extension vers lesquelles vous pouvez effectuer une mise à jour, veuillez consulter Versions d'SQLextension Postgre prises en charge.

    Les extensions `tsearch2` et `chkpass` ne sont plus prises en charge pour PostgreSQL versions 11 ou ultérieures. Si vous effectuez une mise à niveau vers la version 11.x, supprimez les extensions `tsearch2` et `chkpass` avant la mise à niveau.

11. Supprimer les types de données inconnus – Supprimez les types de données `unknown` en fonction de la version cible.

    PostgreSQL version 10 a cessé de prendre en charge le type de données `unknown`. Si une base de données version 9.6 utilise le type de données `unknown`, une mise à niveau vers une version 10 affiche un message d'erreur tel que :

    ```
    Database instance is in a state that cannot be upgraded: PreUpgrade checks failed:
    The instance could not be upgraded because the 'unknown' data type is used in user
     tables.
    Please remove all usages of the 'unknown' data type and try again."
    ```

    Pour rechercher le type de données `unknown` dans votre base de données afin de pouvoir supprimer la colonne incriminée ou la remplacer par un type de données pris en charge, utilisez le code SQL suivant :

    ```
    SELECT DISTINCT data_type FROM information_schema.columns WHERE data_type ILIKE
     'unknown';
    ```

12. Réalisez un essai de mise à niveau – Nous vous recommandons fortement de tester la mise à niveau de version majeure sur un doublon de votre base de données de production avant d'essayer d'effectuer la mise à niveau sur votre base de données de production. Vous pouvez surveiller les plans d'exécution sur la base de données de test dupliquée pour détecter d'éventuelles régressions du plan d'exécution et évaluer ses performances. Pour créer une instance de test dupliquée, vous pouvez soit restaurer votre base de données à partir d'un instantané récent, soit point-in-time restaurer votre base de données à sa date de restauration la plus récente.

Pour plus d'informations, consultez the section called "Restaurer à partir d'un instantané" ou the section called "Point-in-time rétablissement". Pour les clusters de bases de données multi-AZ, consultez the section called "Restauration d'un instantané dans un cluster de base de données multi-AZ" ou the section called "Restauration d'un cluster de base de données multi-AZ à une date définie".

Pour en savoir plus sur la procédure de mise à niveau, consultez the section called "Mise à niveau manuelle de la version du moteur".

Lors de la mise à niveau d'une base de données de version 9.6 vers la version 10, sachez que PostgreSQL 10 active les requêtes parallèles par défaut. Vous pouvez tester l'impact du parallélisme avant la mise à niveau en modifiant le paramètre `max_parallel_workers_per_gather` sur votre base de données de test et en spécifiant 2.

> **Note**
>
> La valeur par défaut du paramètre `max_parallel_workers_per_gather` dans le groupe de paramètres de base de données `default.postgresql10` est 2.

Pour de plus amples informations, veuillez consulter Parallel Query dans la documentation PostgreSQL. Pour désactiver le parallélisme sur la version 10, définissez le paramètre `max_parallel_workers_per_gather` sur 0.

Durant la mise à niveau de version majeure, les bases de données `public` et `template1`, ainsi que le schéma `public` figurant dans chaque base de données, sont renommés temporairement. Ces objets apparaissent dans les journaux avec leur nom d'origine et une chaîne aléatoire ajoutée. Cette chaîne est ajoutée afin que les paramètres personnalisés tels que `locale` et `owner` soient conservés au cours de la mise à niveau de version majeure. À la fin de la mise à niveau, les objets reprennent leurs noms d'origine.

> **Note**
>
> Pendant le processus de mise à niveau de la version majeure, vous ne pouvez pas point-in-time restaurer votre instance de base de données ou votre cluster de base de données multi-AZ. Une fois qu'Amazon RDS a effectué la mise à niveau, le service réalise une sauvegarde automatique de la base de données. Vous pouvez effectuer une point-in-time

restauration avant le début de la mise à niveau et une fois la sauvegarde automatique de votre base de données terminée.

13 Résolvez les problèmes si une mise à niveau échoue avec des erreurs de procédure de pré-vérification – Au cours du processus de mise à niveau de version majeure, Amazon RDS for PostgreSQL exécute d'abord une procédure de pré-vérification pour identifier les éventuels problèmes pouvant provoquer un échec de la mise à niveau. Le procédure de pré-vérification recherche les éventuelles conditions d'incompatibilité entre toutes les bases de données de l'instance.

Si la pré-vérification détecte un problème, elle crée un événement de journal indiquant l'échec de la pré-vérification de mise à niveau. Les détails de la procédure de pré-vérification se trouvent dans un journal de mise à niveau nommé `pg_upgrade_precheck.log` pour toutes les bases de données d'une base de données. Amazon RDS ajoute un horodatage au nom du fichier. Pour de plus amples informations sur l'affichage des journaux, veuillez consulter Surveillance des fichiers journaux RDSAmazon.

Si la mise à niveau d'un réplica en lecture échoue au stade de la vérification préalable, la réplication sur le réplica en lecture défaillant est interrompue et le réplica en lecture est mis à l'état résilié. Supprimez le réplica en lecture et recréez un nouveau réplica en lecture basé sur l'instance de base de données principale mise à niveau.

Résolvez tous les problèmes identifiés dans le journal de pré-vérification puis relancez la mise à niveau de version majeure. Voici un exemple de journal de pré-vérification.

```
-----------------------------------------------------------------------
Upgrade could not be run on Wed Apr 4 18:30:52 2018
-----------------------------------------------------------------------
The instance could not be upgraded from 9.6.11 to 10.6 for the following reasons.
Please take appropriate action on databases that have usage incompatible with the
 requested major engine version upgrade and try the upgrade again.

* There are uncommitted prepared transactions. Please commit or rollback all prepared
 transactions.* One or more role names start with 'pg_'. Rename all role names that
 start with 'pg_'.

* The following issues in the database 'my"million$"db' need to be corrected before
 upgrading:** The ["line","reg*"] data types are used in user tables. Remove all
 usage of these data types.
```

```
** The database name contains characters that are not supported by RDS for
 PostgreSQL. Rename the database.
** The database has extensions installed that are not supported on the target
 database version. Drop the following extensions from your database: ["tsearch2"].

* The following issues in the database 'mydb' need to be corrected before
 upgrading:** The database has views or materialized views that depend on
 'pg_stat_activity'. Drop the views.
```

14. Si une mise à niveau de réplica en lecture échoue lors de la mise à niveau de la base de données, résolvez le problème : un réplica en lecture ayant échoué obtient le statut `incompatible-restore` et la réplication est arrêtée sur la base de données. Supprimez le réplica en lecture et recréez un nouveau réplica en lecture basé sur l'instance de base de données principale mise à niveau.

> **Note**
>
> Amazon RDS ne met pas à niveau les réplicas en lecture pour les clusters de bases de données multi-AZ. Si vous effectuez une mise à niveau de version majeure sur un cluster de base de données multi-AZ, l'état de réplication de ses répliques de lecture devient terminé.

Une mise à niveau de réplica en lecture peut échouer pour les raisons suivantes :

- Elle n'a pas pu s'aligner sur l'instance de base de données principale même après un temps d'attente.
- Elle était dans un état de mise hors service ou de cycle de vie incompatible, tel que storage-full, incompatible-restore, etc.
- Lorsque la mise à niveau de l'instance de base de données principale a démarré, une mise à niveau de version mineure distincte était en cours d'exécution sur le réplica en lecture.
- Le réplica en lecture utilisait des paramètres incompatibles.
- Le réplica en lecture n'a pas pu communiquer avec l'instance de base de données principale pour synchroniser le dossier de données.

15. Mettez à niveau votre base de données de production : quand l'essai de mise à niveau de version majeure est un succès, vous devez être en mesure de mettre à niveau en toute confiance votre base de données de production. Pour de plus amples informations, veuillez consulter Mise à niveau manuelle de la version du moteur.

16 Exécutez l'opération ANALYZE pour actualiser la table `pg_statistic`. Vous devez le faire pour chaque base de données de toutes vos bases de données PostgreSQL. Les statistiques de l'optimiseur ne sont pas transférées lors d'une mise à niveau majeure de la version. Vous devez donc régénérer toutes les statistiques pour éviter les problèmes de performances. Exécutez la commande sans paramètres pour générer des statistiques pour toutes les tables normales de la base de données actuelle, comme suit :

```
ANALYZE VERBOSE;
```

L'indicateur VERBOSE est facultatif, mais son utilisation vous montre la progression. Pour en savoir plus, veuillez consulter ANALYZE dans la documentation PostgreSQL.

Lorsque vous analysez des tables spécifiques au lieu d'utiliser ANALYZE VERBOSE, exécutez la commande ANALYZE pour chaque table comme suit :

```
ANALYZE table_name;
```

Pour les tables partitionnées, analysez toujours la table parent. Ce processus :

- Échantillonne automatiquement les lignes de toutes les partitions
- Met à jour les statistiques de chaque partition de manière récursive
- Maintient les statistiques de planification essentielles au niveau des parents

Bien que les tables parents ne stockent aucune donnée réelle, leur analyse est essentielle pour l'optimisation des requêtes. L'exécution d'ANALYZE uniquement sur des partitions individuelles peut nuire aux performances des requêtes, car l'optimiseur ne disposera pas des statistiques complètes nécessaires à une planification efficace entre partitions.

> **Note**
>
> Exécutez ANALYZE sur votre système après la mise à niveau pour éviter les problèmes de performances.

Une fois la mise à niveau de version majeure effectuée, nous vous recommandons d'effectuer les tâches suivantes :

- Une mise à niveau de PostgreSQL ne met à niveau aucune extension PostgreSQL. Pour mettre à niveau les extensions, veuillez consulter Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre.
- Vous pouvez utiliser Amazon RDS pour consulter deux journaux générés par l'utilitaire `pg_upgrade`. Il s'agit des journaux `pg_upgrade_internal.log` et `pg_upgrade_server.log`. Amazon RDS ajoute un horodatage au nom de fichier de ces journaux. Vous pouvez consultez ces journaux comme tout autre journal. Pour de plus amples informations, veuillez consulter Surveillance des fichiers journaux RDSAmazon.

  Vous pouvez également télécharger les journaux de mise à niveau sur Amazon CloudWatch Logs. Pour de plus amples informations, veuillez consulter Publication de journaux PostgreSQL sur Amazon Logs CloudWatch .
- Pour vérifier si tout fonctionne comme prévu, testez votre application sur la base de données mise à niveau avec une charge de travail similaire. Une fois la mise à niveau vérifiée, vous pouvez supprimer l'instance de test.

Mises à niveau automatiques des versions mineures RDS pour Postgre SQL

Si vous activez l'option Mise à niveau automatique des versions mineures au moment de créer ou modifier une instance de base de données ou un cluster de bases de données multi-AZ, votre base de données peut être mise à niveau automatiquement.

RDSPour chaque version SQL majeure de Postgre, une version mineure est désignée RDS comme version de mise à niveau automatique. Une fois qu'une version mineure a été testée et approuvée par AmazonRDS, la mise à niveau de la version mineure s'effectue automatiquement pendant votre période de maintenance. RDSne définit pas automatiquement les nouvelles versions mineures publiées comme version de mise à niveau automatique. Avant qu'RDS ne désigne une version de mise à niveau automatique plus récente, plusieurs critères sont pris en compte, à savoir :

- Problèmes de sécurité connus
- Bugs dans la version SQL communautaire de Postgre
- Stabilité globale du parc depuis la publication de la version mineure

Vous pouvez utiliser la AWS CLI commande suivante pour déterminer la version cible de mise à niveau mineure automatique actuelle pour une version SQL mineure de Postgre spécifiée dans un domaine spécifique Région AWS.

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine postgres \
--engine-version minor-version \
--region region \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output text
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
--engine postgres ^
--engine-version minor-version ^
--region region ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output text
```

Par exemple, la AWS CLI commande suivante détermine la cible de mise à niveau mineure automatique pour la version SQL mineure 16.1 de Postgre dans l'est des États-Unis (Ohio) (Région AWS us-east-2).

Dans Linux, macOS, ou Unix:

```
aws rds describe-db-engine-versions \
--engine postgres \
--engine-version 16.1 \
--region us-east-2 \
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" \
--output table
```

Dans Windows:

```
aws rds describe-db-engine-versions ^
```

```
--engine postgres ^
--engine-version 16.1 ^
--region us-east-2 ^
--query "DBEngineVersions[*].ValidUpgradeTarget[*].
{AutoUpgrade:AutoUpgrade,EngineVersion:EngineVersion}" ^
--output table
```

Votre sortie est similaire à ce qui suit.

```
---------------------------------
|    DescribeDBEngineVersions    |
+--------------+-----------------+
|  AutoUpgrade |  EngineVersion  |
+--------------+-----------------+
|  False       |  16.2           |
|  True        |  16.3           |
|  False       |  16.4           |
|  False       |  16.5           |
|  False       |  16.6           |
|  False       |  17.1           |
|  False       |  17.2           |
+--------------+-----------------+
```

Dans cet exemple, la `AutoUpgrade` valeur correspond `True` à la SQL version 16.3 de Postgre. La cible de mise à niveau mineure automatique est donc la SQL version 16.3 de Postgre, qui est surlignée dans le résultat.

Une SQL base de données Postgre est automatiquement mise à niveau pendant votre période de maintenance si les critères suivants sont remplis :

- L'option Mise à niveau automatique des versions mineures est activée pour la base de données.
- La base de données exécute une version mineure du moteur de base de données qui est inférieure à la version mineure de la mise à niveau automatique actuelle.

Pour de plus amples informations, veuillez consulter Mise à niveau automatique de la version mineure du moteur.

> **Note**
>
> Une SQL mise à niveau de Postgre ne met pas à jour les extensions PostgreSQL. Pour mettre à niveau les extensions, veuillez consulter Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre.

Mise à niveau des SQL extensions Postgre RDS pour les bases de données SQL Postgre

Une mise à niveau SQL du moteur Postgre ne met pas à niveau la plupart des extensions PostgreSQL. Pour mettre à jour une extension après une mise à niveau de version, utilisez la commande `ALTER EXTENSION UPDATE`.

> **Note**
>
> Pour plus d'informations sur la mise à jour de GIS l'extension Post, consultez Gestion des données spatiales avec l'GISextension Post (Étape 6 : mise à niveau de l'GISextension Post). Pour mettre à jour l'extension `pg_repack`, supprimez l'extension, puis créez la nouvelle version dans la base de données mise à niveau. Pour plus d'informations, veuillez consulter pg_repack installation dans la documentation `pg_repack`.

Pour mettre à niveau une extension, utilisez la commande suivante.

```
ALTER EXTENSION extension_name UPDATE TO 'new_version';
```

Pour obtenir la liste des versions prises en charge des SQL extensions Postgre, consultezVersions d'SQLextension Postgre prises en charge.

Pour répertorier les extensions actuellement installées, utilisez le catalogue Postgre SQL pg_extension dans la commande suivante.

```
SELECT * FROM pg_extension;
```

Pour afficher la liste des versions d'extension spécifiques disponibles pour votre installation, utilisez la vue Postgre SQL pg_available_extension_versions dans la commande suivante.

```
SELECT * FROM pg_available_extension_versions;
```

Mise à niveau d'une version du moteur de snapshots SQL de base de données Postgre

Avec AmazonRDS, vous pouvez créer un instantané de base de données de volume de stockage de votre SQL instance de base de données Postgre. Lorsque vous créez un instantané de base de données, celui-ci est basé sur la version du moteur utilisée par votre RDS instance Amazon. Vous pouvez mettre à niveau la version du moteur pour vos instantanés de base de données.

Après avoir restauré un instantané de base de données mis à niveau vers une nouvelle version de moteur, veillez à vérifier que la mise à jour est réussie. Pour de plus amples informations sur une mise à niveau des versions majeures, veuillez consulter Mises à niveau du moteur de base de données RDS pour PostgreSQL. Pour savoir comment restaurer un instantané de base de données, consultez Restauration vers une instance de base de données.

Vous pouvez mettre à niveau des instantanés de base de données manuels chiffrés ou non chiffrés.

Pour afficher les versions de moteur disponibles RDS pour votre SQL instantané de base de données Postgre, utilisez l' AWS CLI exemple suivant.

```
aws rds describe-db-engine-versions --engine postgres  --engine-version example-
engine-version --query "DBEngineVersions[*].ValidUpgradeTarget[*].
{EngineVersion:EngineVersion}" --output text --include-all
```

Pour plus d'informations sur les versions de moteur disponibles RDS pour les instantanés de base de SQL données Postgre, consultez. Choix d'une version majeure pour une mise à niveau de RDS pour PostgreSQL

> **Note**
>
> Vous ne pouvez pas mettre à niveau des instantanés DB automatisés qui sont créés lors du processus de sauvegarde automatique.

Console

Pour mettre à niveau un instantané de base de données

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.

2. Dans le panneau de navigation, choisissez Snapshots.
3. Choisissez l'instantané que vous souhaitez mettre à niveau.
4. Pour Actions, choisissez Upgrade Snapshot (Mettre à niveau l'instantané). La page Upgrade Snapshot (Mettre à niveau l'instantané) s'affiche.
5. Choisissez la New engine version (Version du nouveau moteur) vers laquelle mettre à niveau.
6. Choisissez Save changes (Enregistrer les changements) pour mettre à niveau l'instantané.

   Pendant le processus de mise à niveau, toutes les actions d'instantané sont désactivées pour l'instantané de base de données. De même, le statut de l'instantané de base de données passe de disponible à upgrading (mise à niveau), puis passe à active, une fois la mise à niveau terminée. Si l'instantané de base de données ne peut pas être mis à jour en raison d'un problème d'instantané endommagé, le statut devient indisponible. Vous ne pouvez pas récupérer l'instantané lorsqu'il a ce statut.

   > **Note**
   >
   > Si la mise à niveau de l'instantané de base de données échoue, l'instantané revient à l'état d'origine avec la version originale.

AWS CLI

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, utilisez la AWS CLI modify-db-snapshotcommande.

Paramètres

- `--db-snapshot-identifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être un nom de ressource Amazon unique (ARN). Pour de plus amples informations, veuillez consulter Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `--engine-version` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Example

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-snapshot \
```

```
    --db-snapshot-identifier my_db_snapshot \
    --engine-version new_version
```

Dans Windows:

```
aws rds modify-db-snapshot ^
    --db-snapshot-identifier my_db_snapshot ^
    --engine-version new_version
```

RDS API

Pour mettre à niveau un instantané de base de données vers une nouvelle version du moteur de base de données, appelez l'odifyDBSnapshotopération Amazon RDS API M.

- `DBSnapshotIdentifier` – L'identifiant de l'instantané de base de données à mettre à niveau. L'identifiant doit être un nom de ressource Amazon unique (ARN). Pour de plus amples informations, veuillez consulter Noms des ressources Amazon (ARNs) dans Amazon RDS.
- `EngineVersion` – La version du moteur vers laquelle la mise à niveau de l'instantané de base de données doit être effectuée.

Utilisation de réplicas en lecture pour Amazon RDS for PostgreSQL

Vous pouvez dimensionner les lectures pour vos instances de base de données Amazon RDS for PostgreSQL en ajoutant des répliques de lecture aux instances. Comme les autres moteurs de base de données Amazon RDS, RDS pour PostgreSQL utilise les mécanismes de réplication natifs de PostgreSQL pour maintenir les répliques en lecture à jour en fonction des modifications apportées à la base de données source. Pour obtenir des informations générales sur les réplicas en lecture et Amazon RDS, consultez Utilisation des réplicas en lecture d'instance de base de données.

Ensuite, vous trouverez des informations propres à l'utilisation des réplicas en lecture avec RDS for PostgreSQL.

Décodage logique sur une réplique lue

RDS pour PostgreSQL prend en charge la réplication logique en mode veille avec PostgreSQL 16.1. Cela vous permet de créer un décodage logique à partir d'un mode veille en lecture seule qui réduit la charge sur l'instance de base de données principale. Vous pouvez améliorer la disponibilité de vos applications qui doivent synchroniser les données entre plusieurs systèmes. Cette fonctionnalité améliore les performances de votre entrepôt de données et de vos analyses de données.

De plus, les emplacements de réplication d'un serveur de secours donné permettent de continuer à transformer ce serveur de secours en serveur principal. Cela signifie qu'en cas de basculement d'une instance de base de données principale ou de promotion d'une instance de secours en tant que nouvelle instance principale, les emplacements de réplication seront conservés et les anciens abonnés de secours ne seront pas affectés.

Pour créer un décodage logique sur une réplique lue

1. Activer la réplication logique : pour créer un décodage logique en mode veille, vous devez activer la réplication logique sur votre instance de base de données source et sa réplique physique. Pour de plus amples informations, veuillez consulter Configuration de réplicas en lecture avec PostgreSQL.

   - Pour activer la réplication logique pour une instance de base de données RDS pour PostgreSQL nouvellement créée, créez un nouveau groupe de paramètres personnalisés de base de données et définissez le paramètre statique sur. `rds.logical_replication` 1 Associez ensuite ce groupe de paramètres de base de données à l'instance de base de données source et à sa réplique de lecture physique. Pour de plus amples informations,

veuillez consulter Associer un groupe de paramètres de base de données à une instance de base de données dans RDSAmazon.

- Pour activer la réplication logique pour une instance de base de données RDS pour PostgreSQL existante, modifiez le groupe de paramètres personnalisés de base de données de l'instance de base de données source et sa réplique physique en lecture pour définir le paramètre statique sur. `rds.logical_replication` 1 Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

> **Note**
>
> Vous devez redémarrer l'instance de base de données pour appliquer ces modifications de paramètres.

Vous pouvez utiliser la requête suivante pour vérifier les valeurs pour `wal_level` et `rds.logical_replication` sur l'instance de base de données source et sa réplique de lecture physique.

```
Postgres=>SELECT name,setting FROM pg_settings WHERE name IN
 ('wal_level','rds.logical_replication');

 name                    | setting
-------------------------+---------
 rds.logical_replication | on
 wal_level               | logical
(2 rows)
```

2. Créez une table dans la base de données source : connectez-vous à la base de données dans votre instance de base de données source. Pour de plus amples informations, veuillez consulter Connexion à une instance de base de données exécutant le moteur de base de données PostgreSQL.

   Utilisez les requêtes suivantes pour créer une table dans votre base de données source et pour insérer des valeurs :

```
Postgres=>CREATE TABLE LR_test (a int PRIMARY KEY);
```

```
CREATE TABLE
```

```
Postgres=>INSERT INTO LR_test VALUES (generate_series(1,10000));
INSERT 0 10000
```

3. Créer une publication pour la table source : utilisez la requête suivante pour créer une publication pour la table sur l'instance de base de données source.

```
Postgres=>CREATE PUBLICATION testpub FOR TABLE LR_test;
CREATE PUBLICATION
```

Utilisez une requête SELECT pour vérifier les détails de la publication créée à la fois sur l'instance de base de données source et sur l'instance physique de réplication en lecture.

```
Postgres=>SELECT * from pg_publication;

oid    | pubname | pubowner | puballtables | pubinsert | pubupdate | pubdelete |
 pubtruncate | pubviaroot
-------+---------+----------+--------------+-----------+-----------+-----------
+-------------+------------
 16429 | testpub |    16413 | f            | t         | t         | t         | t
          | f
(1 row)
```

4. Créer un abonnement à partir d'une instance de réplique logique : créez une autre instance de base de données RDS pour PostgreSQL en tant qu'instance de réplique logique. Assurez-vous que le VPC est correctement configuré pour que cette instance de réplique logique puisse accéder à l'instance de réplique physique en lecture. Pour de plus amples informations, veuillez consulter Amazon VPC et Amazon RDS. Si votre instance de base de données source est inactive, des problèmes de connectivité peuvent survenir et le serveur principal n'envoie pas les données en veille.

```
Postgres=>CREATE SUBSCRIPTION testsub CONNECTION 'host=Physical replica host name
 port=port
               dbname=source_db_name user=user password=password'
 PUBLICATION testpub;
NOTICE:  created replication slot "testsub" on publisher
CREATE SUBSCRIPTION
```

```
Postgres=>CREATE TABLE LR_test (a int PRIMARY KEY);
```

```
CREATE TABLE
```

Utilisez une requête SELECT pour vérifier les détails de l'abonnement sur l'instance de réplique logique.

```
Postgres=>SELECT oid,subname,subenabled,subslotname,subpublications FROM
 pg_subscription;

oid    | subname | subenabled | subslotname | subpublications
-------+---------+------------+-------------+-----------------
 16429 | testsub | t          | testsub     | {testpub}
(1 row)
postgres=> select count(*) from LR_test;
 count
-------
 10000
(1 row)
```

5. Inspectez l'état du slot de réplication logique : vous ne pouvez voir que le slot de réplication physique sur votre instance de base de données source.

```
Postgres=>select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;

slot_name                                   | slot_type | confirmed_flush_lsn
--------------------------------------------+-----------+---------------------
 rds_us_west_2_db_dhqfsmo5wbbjqrn3m6b6ivdhu4 | physical  |
(1 row)
```

Toutefois, sur votre instance de réplication en lecture, vous pouvez voir le slot de réplication logique et la `confirmed_flush_lsn` valeur changer au fur et à mesure que l'application consomme activement des modifications logiques.

```
Postgres=>select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;

slot_name | slot_type | confirmed_flush_lsn
----------+-----------+---------------------
 testsub   | logical   | 0/500002F0
(1 row)
```

```
Postgres=>select slot_name, slot_type, confirmed_flush_lsn from
 pg_replication_slots;

slot_name | slot_type | confirmed_flush_lsn
-----------+-----------+---------------------
 testsub   | logical   | 0/5413F5C0
(1 row)
```

Limites des réplicas en lecture avec PostgreSQL

Les limites des réplicas en lecture pour PostgreSQL sont les suivantes :

- Les réplicas en lecture PostgreSQL sont en lecture seule. Bien qu'un réplica en lecture ne soit pas une instance de base de données accessible en écriture, vous pouvez le promouvoir en instance de base de données RDS for PostgreSQL autonome. Toutefois, le processus n'est pas réversible.
- Vous ne pouvez pas créer de réplica en lecture à partir d'un autre réplica en lecture si votre instance de base de données RDS for PostgreSQL exécute une version de PostgreSQL antérieure à 14.1. RDS for PostgreSQL prend en charge les réplicas en lecture en cascade sur RDS for PostgreSQL version 14.1 et versions ultérieures uniquement. Pour de plus amples informations, veuillez consulter Utilisation de réplicas en lecture en cascade avec RDS for PostgreSQL.
- Si vous promouvez un réplica en lecture PostgreSQL, il devient une instance de base de données accessible en écriture. Il arrête de recevoir les fichiers WAL (write-ahead log) d'une instance de base de données source, et n'est plus une instance en lecture seule. Vous pouvez créer des réplicas en lecture à partir de l'instance de base de données promue comme pour n'importe quelle instance de base de données RDS for PostgreSQL. Pour de plus amples informations, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.
- Si vous promouvez une réplique de lecture PostgreSQL à partir d'une chaîne de réplication (une série de répliques de lecture en cascade), toutes les répliques de lecture en aval existantes continuent de recevoir automatiquement les fichiers WAL de l'instance promue. Pour de plus amples informations, veuillez consulter Utilisation de réplicas en lecture en cascade avec RDS for PostgreSQL.
- Si aucune transaction utilisateur n'est en cours d'exécution sur l'instance de base de données source, le réplica en lecture PostgreSQL associé signale un retard de réplication pouvant atteindre cinq minutes. Le décalage de réplication est calculé comme `currentTime - lastCommitedTransactionTimestamp`. Cela signifie que lorsqu'aucune transaction n'est

traitée, la valeur du décalage de réplication augmente pendant un certain temps jusqu'à ce que le segment WAL (journal d'écriture anticipée) bascule. Par défaut, RDS for PostgreSQL change le segment WAL toutes les cinq minutes, ce qui entraîne l'enregistrement d'une transaction et une diminution du décalage signalé.

- Vous ne pouvez pas activer les sauvegardes automatiques pour les réplicas en lecture PostgreSQL pour les versions de RDS for PostgreSQL antérieures à 14.1. Les sauvegardes automatiques pour les réplicas en lecture sont prises en charge pour RDS for PostgreSQL 14.1 et versions ultérieures uniquement. Pour RDS for PostgreSQL 13 et versions antérieures, créez un instantané à partir d'un réplica en lecture si vous souhaitez en obtenir une sauvegarde.
- Point-in-time la restauration (PITR) n'est pas prise en charge pour les répliques en lecture. Vous pouvez utiliser PITR avec une instance principale (enregistreur) uniquement, pas avec un réplica en lecture. Pour en savoir plus, veuillez consulter la section Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.
- Les répliques de lecture pour les versions 12 et antérieures de PostgreSQL redémarrent automatiquement pendant la période de maintenance de 60 à 90 jours pour appliquer la rotation des mots de passe. Si le réplica perd la connexion à la source avant le redémarrage planifié, il redémarre tout de même pour reprendre la réplication. Pour les versions 13 et supérieures de PostgreSQL, les répliques en lecture peuvent subir de brèves déconnexions et reconnexions pendant le processus de rotation des mots de passe.

Configuration de réplicas en lecture avec PostgreSQL

RDS for PostgreSQL utilise la réplication en continu native de PostgreSQL pour créer une copie en lecture seule d'une instance de base de données source. Cette instance de base de données de réplica en lecture est un réplica physique créé de façon asynchrone de l'instance de base de données source. Elle est créée par une connexion spéciale qui transmet les données WAL (write-ahead log) entre l'instance de base de données source et le réplica en lecture. Pour plus d'informations, consultez Streaming Replication dans la documentation PostgreSQL.

PostgreSQL diffuse de manière asynchrone les modifications de base de données sur cette connexion sécurisée telles qu'elles sont effectuées sur l'instance de base de données source. Vous pouvez chiffrer les communications entre vos applications clientes et l'instance de base de données source ou tout autre réplica en lecture en définissant le paramètre `ssl` sur 1. Pour plus d'informations, consultez Utilisation de SSL avec une instance de base de données PostgreSQL.

PostgreSQL utilise un rôle de réplication pour effectuer la réplication en continu. Ce rôle dispose de privilèges, mais vous ne pouvez pas l'utiliser pour modifier des données. PostgreSQL utilise un processus unique pour traiter la réplication.

Vous pouvez créer un réplica en lecture PostgreSQL sans affecter les opérations ou les utilisateurs de l'instance de base de données source. Amazon RDS définit pour vous les paramètres et autorisations nécessaires, sur l'instance de base de données source et le réplica en lecture, sans impacter le service. Un instantané de l'instance de base de données source est pris, et est utilisé pour créer le réplica en lecture. Si vous supprimez le réplica en lecture à un moment donné dans le futur, aucune panne ne se produit.

Vous pouvez créer jusqu'à 15 réplicas en lecture à partir d'une seule instance de base de données source au sein de la même région. Depuis RDS for PostgreSQL 14.1, vous pouvez également créer jusqu'à trois niveaux de réplica en lecture dans une chaîne (cascade) à partir d'une instance de base de données source. Pour de plus amples informations, veuillez consulter Utilisation de réplicas en lecture en cascade avec RDS for PostgreSQL. Dans tous les cas, les sauvegardes automatisées doivent être configurées sur l'instance de base de données source. Pour cela, vous devez définir la période de conservation des sauvegardes sur votre instance de base de données sur une valeur autre que 0. Pour de plus amples informations, veuillez consulter Création d'un réplica en lecture.

Vous pouvez créer des réplicas en lecture pour votre instance de base de données RDS for PostgreSQL dans la même Région AWS que votre instance de base de données source. Cette procédure se nomme « réplication dans la région ». Vous pouvez également créer des répliques de lecture dans une instance différente Régions AWS de celle de la base de données source. Cette procédure se nomme « réplication entre régions ». Pour plus d'informations sur la configuration des réplicas en lecture entre régions, veuillez consulter Création d'un réplica en lecture dans une autre Région AWS. Les différents mécanismes prenant en charge le processus de réplication dans la région et entre régions diffèrent légèrement en fonction de la version de RDS for PostgreSQL, comme expliqué dans Fonctionnement de la réplication en continu pour différentes versions de RDS for PostgreSQL.

Pour que la réplication fonctionne de façon efficace, chaque réplica en lecture doit avoir la même quantité de ressources de calcul et de stockage que l'instance de base de données source. Si vous mettez à l'échelle l'instance de base de données source, veillez à le faire également pour les réplicas en lecture.

Amazon RDS remplace tout paramètre incompatible sur un réplica en lecture s'il empêche ce dernier de démarrer. Par exemple, supposons que la valeur du paramètre `max_connections` est

supérieure dans l'instance de base de données source à celle du réplica en lecture. Dans ce cas, Amazon RDS met à jour le paramètre dans le réplica en lecture sur la même valeur que celle de l'instance de base de données source.

Les répliques de lecture RDS pour PostgreSQL ont accès à des bases de données externes disponibles via des wrappers de données étrangers FDWs () sur l'instance de base de données source. Par exemple, supposons que votre instance de base de données RDS for PostgreSQL utilise l'encapsuleur `mysql_fdw` pour accéder aux données de RDS for MySQL. Si tel est le cas, vos réplicas en lecture peuvent également accéder à ces données. Les autres supports pris en charge FDWs incluent `oracle_fdw`postgres_fdw, ettds_fdw. Pour de plus amples informations, veuillez consulter Utilisation des wrappers de données étrangers pris en charge pour SQL.

Utilisation des réplicas en lecture RDS for PostgreSQL avec des configurations multi-AZ

Vous pouvez créer un réplica en lecture à partir de déploiements d'instance de base de données mono-AZ ou Multi-AZ. Vous pouvez utiliser des déploiements multi-AZ pour améliorer la durabilité et la disponibilité des données critiques, avec un réplica de secours. Un réplica de secours est un réplica en lecture dédié qui peut assumer la charge de travail en cas de défaillance de la base de données source. Vous ne pouvez pas utiliser votre réplica de secours pour traiter le trafic en lecture. Vous pouvez toutefois créer des réplicas en lecture à partir d'instances de base de données multi-AZ à trafic élevé pour décharger les requêtes en lecture seule. Pour plus d'informations sur les déploiements Multi-AZ, veuillez consulter Déploiements d'instances de base de données multi-AZ pour Amazon RDS.

Si l'instance de base de données source d'un déploiement multi-AZ bascule vers une instance de secours, les réplicas en lecture associés se mettent automatiquement à utiliser l'instance de secours (désormais principale) comme source de réplication. Les réplicas en lecture ont peut-être besoin d'être redémarrés, en fonction de la version de RDS for PostgreSQL, comme suit :

- PostgreSQL 13 et versions ultérieures – Le redémarrage n'est pas obligatoire. Les réplicas en lecture sont automatiquement synchronisés avec la nouvelle instance principale. Toutefois, dans certains cas, votre application cliente risque de mettre en cache les détails du service DNS (Domain Name Service) pour vos réplicas en lecture. Si tel est le cas, définissez la valeur time-to-live (TTL) sur moins de 30 secondes. Cela empêche le réplica en lecture de conserver une adresse IP obsolète (et l'empêche ainsi de se synchroniser avec la nouvelle instance principale). Pour en savoir plus et pour obtenir d'autres bonnes pratiques, consultez Directives opérationnelles de base Amazon RDS.

- PostgreSQL 12 et toutes les versions antérieures – Les réplicas en lecture redémarrent automatiquement après un basculement vers le réplica de secours car ce dernier (désormais principal) a une adresse IP différente et un nom d'instance différent. Le redémarrage synchronise le réplica en lecture avec la nouvelle instance principale.

Pour en savoir plus sur le basculement, consultez Défaillance d'une instance de base de données multi-AZ pour Amazon RDS. Pour plus d'informations sur le fonctionnement des réplicas en lecture dans un déploiement multi-AZ, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

Pour prendre en charge le basculement d'un réplica en lecture, vous pouvez créer le réplica en lecture en tant qu'instance de base de données multi-AZ pour que Amazon RDS crée une instance de secours de votre réplica dans une autre zone de disponibilité. La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ.

Utilisation de réplicas en lecture en cascade avec RDS for PostgreSQL

À partir de la version 14.1, RDS for PostgreSQL prend en charge les réplicas en lecture en cascade. Les réplicas en lecture en cascade vous permettent de mettre à l'échelle les lectures sans surcharger votre instance de base de données RDS for PostgreSQL source. Les mises à jour du journal WAL ne sont pas envoyées par l'instance de base de données source à chaque réplica en lecture. Au lieu de cela, chaque réplica en lecture d'une série en cascade envoie les mises à jour du journal WAL au réplica en lecture suivant de la série. Cela réduit la charge pesant sur l'instance de base de données source.

Avec les réplicas en lecture en cascade, votre instance de base de données RDS for PostgreSQL envoie des données WAL au premier réplica en lecture de la chaîne. Ce réplica en lecture envoie ensuite des données WAL au deuxième réplica de la chaîne, etc. Au final, tous les réplicas en lecture de la chaîne ont reçu les modifications de l'instance de base de données RDS for PostgreSQL, sans surcharger uniquement l'instance de base de données source.

Vous pouvez créer une série comportant jusqu'à trois réplicas en lecture dans une chaîne à partir d'une instance de base de données RDS for PostgreSQL source. Par exemple, supposons que vous disposez d'une instance de base de données RDS for PostgreSQL 14.1, `rpg-db-main`. Vous pouvez effectuer les actions suivantes :

- À partir de `rpg-db-main`, créez le premier réplica en lecture de la chaîne, `read-replica-1`.

- Ensuite, à partir de `read-replica-1`, créez le réplica en lecture suivant dans la chaîne, `read-replica-2`.
- Enfin, à partir de `read-replica-2`, créez le troisième réplica en lecture de la chaîne, `read-replica-3`.

Vous ne pouvez pas créer un autre réplica en lecture au-delà de ce troisième réplica en lecture en cascade dans la série pour `rpg-db-main`. Une série complète d'instances allant d'une instance de base de données source RDS for PostgreSQL à la fin d'une série de réplicas en lecture en cascade peut comporter jusqu'à quatre instances de bases de données.

Pour que les réplicas en lecture en cascade fonctionnent, activez les sauvegardes automatiques sur RDS for PostgreSQL. Commencez par créer le réplica en lecture, puis activez les sauvegardes automatiques sur l'instance de base de données RDS for PostgreSQL. Le processus est le même que pour les autres moteurs de base de données Amazon RDS. Pour de plus amples informations, veuillez consulter Création d'un réplica en lecture.

Comme pour tout réplica en lecture, vous pouvez promouvoir un réplica en lecture faisant partie d'une cascade. La promotion d'un réplica en lecture depuis une chaîne de réplicas en lecture retire ce réplica de la chaîne. Par exemple, supposons que vous souhaitez déplacer une partie de la charge de travail de votre instance de base de données `rpg-db-main` vers une nouvelle instance destinée uniquement au service comptable. En prenant pour hypothèse la chaîne de trois réplicas en lecture de l'exemple, vous décidez de promouvoir `read-replica-2`. La chaîne est affectée comme suit :

- La promotion de `read-replica-2` le retire de la chaîne de réplication.
  - Il s'agit désormais d'une instance de base de données en lecture/écriture complète.
  - La réplication continue sur `read-replica-3`, tout comme avant la promotion.
- Votre `rpg-db-main` continue la réplication sur `read-replica-1`.

Pour plus d'informations sur la promotion des réplicas en lecture, consultez Promotion d'un réplica en lecture en instance de bases de données autonome.

> **Note**
>
> - RDS pour PostgreSQL ne prend pas en charge les mises à niveau de version majeures pour les répliques en cascade. Avant d'effectuer une mise à niveau de version majeure, vous devez supprimer les répliques en cascade. Vous pouvez les recréer après avoir

terminé la mise à niveau sur votre instance de base de données source et sur les répliques de premier niveau.

- Pour les réplicas en lecture en cascade, RDS for PostgreSQL prend en charge 15 réplicas en lecture pour chaque instance de base de données source au premier niveau de réplication, et 5 réplicas en lecture pour chaque instance de base de données source aux deuxième et troisième niveaux de réplication.

Création de répliques de lecture en cascade entre régions avec RDS pour PostgreSQL

RDS pour PostgreSQL prend en charge les répliques de lecture en cascade entre régions. Vous pouvez créer une réplique entre régions à partir de l'instance de base de données source, puis créer des répliques de même région à partir de celle-ci. Vous pouvez également créer une réplique de même région à partir de l'instance de base de données source, puis créer des répliques entre régions à partir de celle-ci.

Créez une réplique entre régions, puis créez des répliques correspondant à la même région

Vous pouvez utiliser une instance de base de données RDS pour PostgreSQL avec la version 14.1 ou supérieure `rpg-db-main` pour effectuer les opérations suivantes :

1. Commencez par `rpg-db-main` (US-EAST-1), créez la première réplique de lecture interrégionale de la chaîne (US-WEST-2). `read-replica-1`
2. À l'aide de la première inter-région `read-replica-1` (US-WEST-2), créez la deuxième réplique de lecture de la chaîne (US-WEST-2). `read-replica-2`
3. À l'aide de`read-replica-2`, créez la troisième réplique de lecture de la chaîne `read-replica-3` (US-WEST-2).

Créez une réplique dans la même région, puis créez des répliques entre régions

Vous pouvez utiliser une instance de base de données RDS pour PostgreSQL avec la version 14.1 ou supérieure `rpg-db-main` pour effectuer les opérations suivantes :

1. En commençant par `rpg-db-main` (US-EAST-1), créez la première réplique de lecture de la chaîne `read-replica-1` (US-EAST-1).

2. À l'aide de `read-replica-1` (US-EAST-1), créez la première réplique de lecture interrégionale de la chaîne (US-WEST-2). `read-replica-2`
3. À l'aide de `read-replica-2` (US-WEST-2), créez la troisième réplique de lecture de la chaîne `read-replica-3` (US-WEST-2).

Limitations liées à la création de répliques de lecture entre régions

- Une chaîne en cascade interrégionale de répliques de bases de données peut couvrir au maximum deux régions, avec un maximum de quatre niveaux. Les quatre niveaux incluent la source de base de données et trois répliques de lecture.

Avantages de l'utilisation de répliques de lecture en cascade

- Évolutivité de lecture améliorée : en répartissant les requêtes de lecture sur plusieurs répliques, la réplication en cascade permet d'équilibrer la charge. Cela améliore les performances, en particulier dans les applications nécessitant beaucoup de lecture, en réduisant la charge de travail de la base de données d'écriture.
- Distribution géographique — Les répliques en cascade peuvent être situées dans différents emplacements géographiques. Cela réduit le temps de latence pour les utilisateurs éloignés de la base de données principale et fournit une réplique en lecture locale, améliorant ainsi les performances et l'expérience utilisateur.
- Haute disponibilité et reprise après sinistre : en cas de panne d'un serveur principal, les répliques peuvent être promues au rang de serveur principal, ce qui garantit la continuité. La réplication en cascade améliore encore cela en proposant plusieurs niveaux d'options de basculement, améliorant ainsi la résilience globale du système.
- Flexibilité et croissance modulaire — À mesure que le système se développe, de nouvelles répliques peuvent être ajoutées à différents niveaux sans reconfiguration majeure de la base de données principale. Cette approche modulaire permet une croissance évolutive et gérable de la configuration de réplication.

Bonnes pratiques pour l'utilisation de répliques de lecture entre régions

- Avant de promouvoir une réplique, créez des répliques supplémentaires. Cela permettra de gagner du temps et de gérer efficacement la charge de travail.

Fonctionnement de la réplication en continu pour différentes versions de RDS for PostgreSQL

Comme indiqué dans Configuration de réplicas en lecture avec PostgreSQL, RDS for PostgreSQL utilise le protocole de réplication en continu natif de PostgreSQL pour envoyer des données WAL à partir de l'instance de base de données source. Il envoie les données WAL source aux réplicas en lecture pour les réplicas en lecture dans la région et entre régions. Avec la version 9.4, PostgreSQL a introduit des emplacements de réplication physiques comme mécanisme de prise en charge du processus de réplication.

Un emplacement de réplication physique empêche une instance de base de données source de supprimer les données WAL avant qu'elles ne soient utilisées par tous les réplicas en lecture. Chaque réplica en lecture possède son propre emplacement physique sur l'instance de base de données source. L'emplacement permet de suivre le WAL le plus ancien (par numéro de séquence logique, LSN) qui pourrait être requis par le réplica. Une fois que tous les emplacements et les connexions à la base de données ont dépassé un WAL (LSN) donné, le LSN devient candidat à la suppression au point de contrôle suivant.

Amazon RDS utilise Amazon S3 pour archiver les données WAL. Pour les réplicas en lecture dans la région, vous pouvez utiliser ces données archivées pour restaurer le réplica en lecture si nécessaire. Par exemple, vous pouvez le faire si la connexion entre la base de données source et le réplica en lecture est interrompue pour quelque raison que ce soit.

Le tableau suivant résume les différences entre les versions de PostgreSQL et les mécanismes de prise en charge pour la réplication dans la région et entre régions utilisée par RDS for PostgreSQL.

Version	Dans la région	Entre régions
PostgreSQL 14.1 et versions supérieures	• Emplacements de réplication • Archive Amazon S3	• Emplacements de réplication
PostgreSQL 13 et versions antérieures	• Archive Amazon S3	• Emplacements de réplication

Pour de plus amples informations, veuillez consulter Surveillance et réglage du processus de réplication.

Analyse des paramètres qui contrôlent la réplication PostgreSQL

Les paramètres suivants affectent le processus de réplication et déterminent dans quelle mesure les réplicas en lecture restent à jour avec l'instance de base de données source :

max_wal_senders

Le paramètre `max_wal_senders` spécifie le nombre maximal de connexions que l'instance de base de données source peut prendre en charge en même temps via le protocole de réplication en continu.

La valeur par défaut varie pour les versions de RDS pour PostgreSQL :

- Pour les versions 13, 14 et 15, la valeur par défaut est 20.
- Pour les versions 16 et supérieures, la valeur par défaut est 35.

Ce paramètre doit être défini sur une valeur légèrement supérieure au nombre réel de réplicas en lecture. Si la valeur est trop faible pour le nombre de réplicas en lecture, la réplication s'arrête.

Pour plus d'informations, consultez max_wal_senders dans la documentation PostgreSQL.

> **Note**
>
> `max_wal_sendersest` un paramètre statique qui nécessite le redémarrage d'une instance de base de données pour que les modifications prennent effet.

wal_keep_segments

Le paramètre `wal_keep_segments` spécifie le nombre de fichiers WAL (write-ahead log) conservés par l'instance de base de données source dans le répertoire `pg_wal`. La valeur par défaut est 32.

Si la valeur de `wal_keep_segments` n'est pas suffisante pour votre déploiement, un réplica en lecture peut prendre un retard tel que la réplication en continu s'arrête. Dans ce cas, Amazon RDS génère une erreur de réplication et commence la récupération des données sur le réplica en lecture. Pour ce faire, il relit les données WAL archivées de l'instance de base de données source à partir d'Amazon S3. Ce processus de récupération se poursuit jusqu'à ce que le réplica en lecture ait rattrapé suffisamment de retard pour continuer la réplication en continu. Pour voir

ce processus en action tel qu'il est capturé par le journal PostgreSQL, consultez Exemple : Récupération d'un réplica en lecture à la suite d'interruptions de réplication.

> **Note**
>
> Dans PostgreSQL version 13, le paramètre `wal_keep_segments` est nommé `wal_keep_size`. Il a le même objectif que `wal_keep_segments`, mais sa valeur par défaut est exprimée en mégaoctets (Mo) (2 048 Mo) plutôt qu'en nombre de fichiers. Pour de plus amples informations, veuillez consulter wal_keep_segments et wal_keep_size dans la documentation PostgreSQL.

max_slot_wal_keep_size

Le paramètre `max_slot_wal_keep_size` contrôle la quantité de données WAL conservée par l'instance de base de données RDS for PostgreSQL dans le répertoire `pg_wal` pour remplir les emplacements. Ce paramètre est utilisé pour les configurations utilisant des emplacements de réplication. La valeur par défaut de ce paramètre est `-1`, ce qui signifie que la quantité de données WAL conservées sur l'instance de base de données source n'est pas limitée. Pour en savoir plus sur la surveillance de vos emplacements de réplication, consultez Surveillance des emplacements de réplication pour votre instance de base de données RDS for PostgreSQL.

Pour de plus amples informations sur ce paramètre, consultez max_slot_wal_keep_size dans la documentation PostgreSQL.

Chaque fois que le flux qui fournit les données WAL à un réplica en lecture est interrompu, PostgreSQL bascule en mode de récupération. Il restaure la réplique lue en utilisant les données WAL archivées d'Amazon S3 ou en utilisant les données WAL associées au slot de réplication. Une fois ce processus terminé, PostgreSQL rétablit la réplication en continu.

Exemple : Récupération d'un réplica en lecture à la suite d'interruptions de réplication

L'exemple suivant fournit les détails du journal qui illustrent le processus de récupération d'un réplica en lecture. L'exemple provient d'une instance de base de données RDS pour PostgreSQL exécutant PostgreSQL version 12.9 dans la Région AWS même base de données que la base de données source, de sorte que les emplacements de réplication ne sont pas utilisés. Le processus de récupération est le même pour les autres instances de base de données RDS for PostgreSQL exécutant des versions de PostgreSQL antérieures à la version 14.1 avec des réplicas en lecture dans la région.

Lorsque le réplica en lecture a perdu le contact avec l'instance de base de données source, Amazon RDS enregistre le problème dans le journal sous la forme d'un message `FATAL: could not receive data from WAL stream`, ainsi que `ERROR: requested WAL segment ... has already been removed`. Comme indiqué sur la ligne en gras, Amazon RDS récupère le réplica en relisant un fichier WAL archivé.

```
2014-11-07 19:01:10 UTC::@:[23180]:DEBUG:  switched WAL source from archive to stream
 after failure
2014-11-07 19:01:10 UTC::@:[11575]:LOG: started streaming WAL from primary at 1A/
D3000000 on timeline 1
2014-11-07 19:01:10 UTC::@:[11575]:FATAL: could not receive data from WAL stream:
ERROR:  requested WAL segment 00000001000000 1A000000D3 has already been removed
2014-11-07 19:01:10 UTC::@:[23180]:DEBUG: could not restore file "00000002.history"
 from archive: return code 0
2014-11-07 19:01:15 UTC::@:[23180]:DEBUG: switched WAL source from stream to archive
 after failure recovering 00000001000000 1A000000D3
2014-11-07 19:01:16 UTC::@:[23180]:LOG:  restored log file "00000001000000 1A000000D3"
 from archive
```

Lorsqu'Amazon RDS relit un nombre suffisant de données WAL archivées sur le réplica pour qu'il rattrape son retard, le streaming vers le réplica en lecture reprend. Lorsque le streaming reprend, Amazon RDS écrit une entrée dans le fichier journal, semblable à ce qui suit.

```
2014-11-07 19:41:36 UTC::@:[24714]:LOG:started streaming WAL from primary at 1B/
B6000000 on timeline 1
```

Définition des paramètres qui contrôlent la mémoire partagée

Les paramètres que vous définissez déterminent la taille de la mémoire partagée pour le suivi des transactions IDs, des verrous et des transactions préparées. La structure de mémoire partagée d'une instance de secours doit être égale ou supérieure à celle d'une instance principale. Cela garantit que la première ne viendra pas à manquer de mémoire partagée lors de la récupération. Si les valeurs des paramètres sur le réplica sont inférieures aux valeurs des paramètres sur l'instance principale, Amazon RDS ajuste automatiquement les paramètres de réplica et redémarre le moteur.

Les paramètres concernés sont les suivants :

- max_connections
- max_worker_processes
- max_wal_senders

- max_prepared_transactions
- max_locks_per_transaction

Pour éviter le redémarrage des réplicas par RDS en raison d'un manque de mémoire, nous recommandons d'appliquer les modifications de paramètres à chaque réplica sous la forme d'un redémarrage progressif. Vous devez appliquer les règles suivantes lorsque vous définissez les paramètres :

- Augmentation des valeurs des paramètres :
  - Vous devez toujours d'abord augmenter les valeurs des paramètres de tous les réplicas lus, puis effectuer un redémarrage progressif de tous les réplicas. Appliquez ensuite les modifications de paramètres sur l'instance principale et redémarrez.
- Diminution des valeurs des paramètres :
  - Vous devez d'abord diminuer les valeurs des paramètres de l'instance principale, puis effectuer un redémarrage. Appliquez ensuite les modifications des paramètres à tous les réplicas de lecture associés et effectuez un redémarrage progressif.

Surveillance et réglage du processus de réplication

Nous vous recommandons fortement de surveiller régulièrement votre instance de base de données RDS for PostgreSQL et vos réplicas en lecture. Vous devez vous assurer que vos réplicas en lecture suivent le rythme des modifications apportées à l'instance de base de données source. Amazon RDS récupère de manière transparente vos réplicas en lecture lorsque le processus de réplication est interrompu. Cependant, il est préférable d'éviter toute récupération. La récupération à l'aide d'emplacements de réplication est plus rapide que l'utilisation de l'archive Amazon S3, mais tout processus de récupération peut affecter les performances de lecture.

Pour déterminer dans quelle mesure vos réplicas en lecture suivent le rythme de l'instance de base de données source, vous pouvez effectuer les opérations suivantes :

- Vérifiez la valeur de **`ReplicaLag`** entre l'instance de base de données source et les réplicas. Le retard du réplica correspond au retard en secondes qu'accuse un réplica en lecture par rapport à son instance de base de données source. Cette métrique renvoie le résultat de la requête suivante.

```
SELECT extract(epoch from now() - pg_last_xact_replay_timestamp()) AS "ReplicaLag";
```

Le délai de réplication indique dans quelle mesure un réplica en lecture suit le rythme de l'instance de base de données source. Il s'agit de la latence entre l'instance de base de données source et une instance de lecture spécifique. Une valeur de délai de réplication élevée peut indiquer une non-correspondance entre les classes d'instance de base de données et/ou les types de stockages utilisés par l'instance de base de données source et ses réplicas en lecture. La classe d'instance de base de données et les types de stockages pour l'instance de base de données source et tous les réplicas en lecture doivent être identiques.

Le délai de réplication peut également être issu de problèmes de connexion intermittents. Vous pouvez surveiller le délai de réplication dans Amazon CloudWatch en consultant la `ReplicaLag` métrique Amazon RDS. Pour en savoir plus sur `ReplicaLag` et d'autres métriques pour Amazon RDS, veuillez consulter CloudWatch Métriques Amazon pour Amazon RDS.

- Consultez le journal PostgreSQL pour obtenir des informations que vous pouvez utiliser pour ajuster vos paramètres. À chaque point de contrôle, le journal PostgreSQL capture le nombre de fichiers journaux de transactions recyclés, comme illustré dans l'exemple suivant.

```
2014-11-07 19:59:35 UTC::@:[26820]:LOG:  checkpoint complete: wrote 376 buffers
 (0.2%);
0 transaction log file(s) added, 0 removed, 1 recycled; write=35.681 s, sync=0.013 s,
 total=35.703 s;
sync files=10, longest=0.013 s, average=0.001 s
```

  Vous pouvez utiliser ces informations pour déterminer le nombre de fichiers de transactions qui sont recyclés au cours d'une période donnée. Vous pouvez ensuite modifier la valeur de `wal_keep_segments` si nécessaire. Par exemple, supposons que le journal PostgreSQL à `checkpoint complete` montre `35 recycled` pour une intervalle de 5 minutes. Dans ce cas, la valeur par défaut de `wal_keep_segments` (32) n'est pas suffisante pour suivre le rythme de l'activité de streaming. Nous vous recommandons ainsi d'augmenter la valeur de ce paramètre.

- Utilisez Amazon CloudWatch pour surveiller les indicateurs permettant de prévoir les problèmes de réplication. Plutôt que d'analyser directement le journal PostgreSQL, vous pouvez utiliser CloudWatch Amazon pour vérifier les métriques collectées. Par exemple, vous pouvez consulter la valeur de la métrique `TransactionLogsGeneration` pour obtenir la quantité de données WAL générées par l'instance de base de données source. Dans certains cas, la charge de travail de votre instance de base de données risque de générer une grande quantité de données WAL. Si tel est le cas, vous devrez peut-être modifier la classe de votre instance de base de données source

et de vos réplicas en lecture. L'utilisation d'une classe d'instance avec des performances réseau élevées (10 Gbit/s) peut réduire le délai de réplication.

Surveillance des emplacements de réplication pour votre instance de base de données RDS for PostgreSQL

Toutes les versions de RDS for PostgreSQL utilisent des emplacements de réplication pour les réplicas en lecture entre régions. RDS for PostgreSQL 14.1 et versions ultérieures utilisent des emplacements de réplication pour les réplicas en lecture dans la région. Les réplicas en lecture dans la région utilisent également Amazon S3 pour archiver les données WAL. En d'autres termes, si votre instance de base de données et vos réplicas en lecture exécutent PostgreSQL 14.1 ou versions ultérieures, les emplacements de réplication et les archives Amazon S3 sont disponibles pour récupérer le réplica en lecture. La récupération d'un réplica en lecture à l'aide de son emplacement de réplication est plus rapide que la récupération à partir d'une archive Amazon S3. Nous vous recommandons donc de surveiller les emplacements de réplication et les métriques associées.

Vous pouvez afficher les emplacements de réplication sur vos instances de base de données RDS for PostgreSQL en interrogeant la vue `pg_replication_slots`, comme suit.

```
postgres=> SELECT * FROM pg_replication_slots;
slot_name                  | plugin | slot_type | datoid | database | temporary |
 active | active_pid | xmin | catalog_xmin | restart_lsn | confirmed_flush_lsn |
 wal_status | safe_wal_size | two_phase
---------------------------+--------+-----------+--------+----------+-----------
+--------+------------+------+--------------+-------------+---------------------
+------------+---------------+-----------
rds_us_west_1_db_555555555 |        | physical  |        |          | f         | t
  |      13194 |      |              | 23/D8000060 |                     | reserved   |
               | f
(1 row)
```

La valeur `reserved` de `wal_status` signifie que la quantité de données WAL détenue par l'emplacement se situe dans les limites du paramètre `max_wal_size`. En d'autres termes, la taille de l'emplacement de réplication est correcte. Les autres valeurs de statut possibles sont les suivantes :

- `extended` – L'emplacement dépasse la valeur de `max_wal_size`, mais les données WAL sont conservées.
- `unreserved` – L'emplacement ne contient plus toutes les données WAL requises. Une partie sera supprimée au prochain point de contrôle.

- `lost` – Certaines données WAL requises ont été supprimées. L'emplacement n'est plus utilisable.

Les `lost` états `unreserved` et ne `wal_status` sont visibles que lorsqu'ils ne `max_slot_wal_keep_size` sont pas négatifs.

La vue `pg_replication_slots` affiche l'état actuel de vos emplacements de réplication. Pour évaluer les performances de vos emplacements de réplication, vous pouvez utiliser Amazon CloudWatch et surveiller les indicateurs suivants :

- **`OldestReplicationSlotLag`**— Indique la quantité de données du journal d'écriture anticipée (WAL) de la source qui n'a pas été consommée par la réplique la plus retardée.
- **`TransactionLogsDiskUsage`** : indique la quantité de stockage utilisée pour les données WAL. Lorsqu'un réplica en lecture est significativement en retard, la valeur de cette métrique peut augmenter considérablement.

Pour en savoir plus sur l'utilisation d'Amazon CloudWatch et de ses métriques pour RDS pour PostgreSQL, consultez. Surveillance des métriques Amazon RDS () avec Amazon CloudWatch Pour de plus amples informations sur la surveillance de la réplication en continu sur vos instances de base de données RDS for PostgreSQL, consultez Best practices for Amazon RDS PostgreSQL replication sur AWS Database Blog.

Résolution des problèmes liés à la réplication en lecture de RDS pour PostgreSQL

Vous trouverez ci-dessous des idées pour résoudre certains problèmes courants liés à la réplication en lecture de RDS pour PostgreSQL.

Mettre fin à la requête à l'origine du décalage de lecture de la réplique

Les transactions actives ou inactives en état de transaction qui s'exécutent pendant une longue période dans la base de données peuvent interférer avec le processus de réplication WAL, augmentant ainsi le délai de réplication. Veillez donc à surveiller le temps d'exécution de ces transactions avec la vue `pg_stat_activity` PostgreSQL.

Exécutez une requête similaire à la suivante sur l'instance principale pour trouver l'ID de processus (PID) de la requête exécutée depuis longtemps :

```
SELECT datname, pid,usename, client_addr, backend_start,
xact_start, current_timestamp - xact_start AS xact_runtime, state,
backend_xmin FROM pg_stat_activity WHERE state='active';
```

```
SELECT now() - state_change as idle_in_transaction_duration, now() - xact_start as
 xact_duration,*
FROM  pg_stat_activity
WHERE state  = 'idle in transaction'
AND   xact_start is not null
ORDER BY 1 DESC;
```

Après avoir identifié le PID de la requête, vous pouvez choisir de mettre fin à la requête.

Exécutez une requête sur l'instance principale similaire à la suivante pour mettre fin à la requête en cours d'exécution depuis longtemps :

```
SELECT pg_terminate_backend(PID);
```

Amélioration des performances des requêtes pour RDS for PostgreSQL avec Lectures optimisées pour Amazon RDS

Vous pouvez accélérer le traitement des requêtes pour RDS for PostgreSQL avec Lectures optimisées pour Amazon RDS. Une instance de base de données ou un cluster de bases de données multi-AZ RDS for PostgreSQL qui utilise la fonctionnalité Lectures optimisées pour RDS peut traiter les requêtes jusqu'à 50 % plus rapidement qu'une instance ou un cluster qui ne l'utilise pas.

Rubriques

- Présentation de Lectures optimisées pour RDS dans PostgreSQL
- Cas d'utilisation pour RDS Optimized Reads
- Bonnes pratiques relatives à RDS Optimized Reads
- Utilisation de RDS Optimized Reads
- Surveillance des instances de base de données qui utilisent RDS Optimized Reads
- Limites pour Lectures optimisées pour RDS dans PostgreSQL

Présentation de Lectures optimisées pour RDS dans PostgreSQL

Les lectures optimisées sont disponibles par défaut sur RDS pour PostgreSQL versions 15.2 et supérieures, 14.7 et supérieures, et 13.10 et supérieures lors de l'utilisation de classes d'instance de base de données basées sur des classes d'instance de base de données. NVMe Pour les spécifications matérielles indiquant quelles instances sont utilisées NVMe, consultezSpécifications matérielles pour les classes d'instance de base de données .

Lorsque vous utilisez une instance de base de données RDS pour PostgreSQL ou un cluster de base de données multi-AZ sur lequel les lectures optimisées RDS sont activées, les performances de requête sont jusqu'à 50 % plus rapides grâce au stockage au niveau des blocs sur disque SSD NVMe (Non-Volatile Memory Express) local. Vous pouvez accélérer le traitement des requêtes en plaçant les tables temporaires générées par PostgreSQL sur le stockage local, ce qui réduit le trafic vers Elastic Block Storage (EBS) sur le réseau.

Dans PostgreSQL, les objets temporaires sont affectés à un espace de noms temporaire qui est automatiquement supprimé à la fin de la session. Lors de la suppression, l'espace de noms temporaire supprime tous les objets qui dépendent de la session, y compris les objets qualifiés selon le schéma, tels que les tables, les fonctions, les opérateurs ou même les extensions.

Dans RDS for PostgreSQL, le paramètre `temp_tablespaces` est configuré pour cette zone de travail temporaire dans laquelle les objets temporaires sont stockés.

Les requêtes suivantes renvoient le nom de l'espace de table et son emplacement.

```
postgres=> show temp_tablespaces;
temp_tablespaces
--------------------
rds_temp_tablespace
(1 row)
```

`rds_temp_tablespace`Il s'agit d'un tablespace configuré par RDS qui pointe vers le NVMe stockage local. Vous pouvez toujours revenir au stockage Amazon EBS en modifiant ce paramètre en `Parameter group` utilisant le AWS Management Console pour pointer vers un tablespace autre que. `rds_temp_tablespace` Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora). Vous pouvez également utiliser la commande SET pour modifier la valeur du paramètre `temp_tablespaces` sur `pg_default` au niveau de la session à l'aide de la commande SET. La modification du paramètre redirige la zone de travail temporaire vers Amazon EBS. Le retour à Amazon EBS est utile lorsque le stockage local de votre instance ou cluster RDS n'est pas suffisant pour effectuer une opération SQL spécifique.

```
postgres=> SET temp_tablespaces TO 'pg_default';
SET
```

```
postgres=> show temp_tablespaces;

 temp_tablespaces
------------------
 pg_default
```

Cas d'utilisation pour RDS Optimized Reads

Voici quelques cas d'utilisation pour lesquels Lectures optimisées est utilisé :

- Requêtes analytiques qui incluent des expressions de table communes (CTEs), des tables dérivées et des opérations de regroupement.
- Réplicas en lecture qui gèrent les requêtes non optimisées pour une application.

- Requêtes de création de rapports dynamiques ou à la demande avec des opérations complexes telles que GROUP BY et ORDER BY qui ne peuvent pas toujours utiliser les index appropriés.
- Autres charges de travail utilisant des tables temporaires internes.
- `CREATE INDEX`ou `REINDEX` des opérations de tri.

Bonnes pratiques relatives à RDS Optimized Reads

Utilisez les bonnes pratiques suivantes pour RDS Optimized Reads :

- Ajoutez une logique de nouvelle tentative pour les requêtes en lecture seule au cas où elles échoueraient en raison d'un stockage d'instances complet pendant l'exécution.
- Surveillez l'espace de stockage disponible sur le magasin d'instances à l'aide de la CloudWatch métrique`FreeLocalStorage`. Si le stockage d'instances atteint sa limite en raison de la charge de travail sur l'instance de base de données ou le cluster de bases de données multi-AZ, modifiez l'instance ou le cluster pour utiliser une plus grande classe d'instances de base de données.

Utilisation de RDS Optimized Reads

Lorsque vous provisionnez une instance de base de données RDS pour PostgreSQL avec l'une des classes d'instance de base de données basées dans un déploiement d' NVMe instance de base de données mono-AZ, un déploiement d'instance de base de données multi-AZ ou un déploiement de cluster de bases de données multi-AZ, l'instance de base de données utilise automatiquement les lectures optimisées RDS.

Pour plus d'informations sur le déploiement multi-AZ, consultezConfiguration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Pour activer RDS Optimized Reads, effectuez l'une des actions suivantes :

- Créez une instance de base de données RDS pour PostgreSQL ou un cluster de base de données multi-AZ à l'aide de l'une des classes d'instance de base de NVMe données basées. Pour de plus amples informations, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Modifiez une instance de base de données RDS pour PostgreSQL ou un cluster de base de données multi-AZ existant pour utiliser l'une des classes d'instance de base de NVMe données basées. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Les lectures optimisées RDS sont disponibles partout Régions AWS où une ou plusieurs classes d'instances de base de données avec stockage NVMe SSD local sont prises en charge. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .

Pour revenir à une instance RDS aux lectures non optimisées, modifiez la classe d'instances de base de données de votre instance ou cluster RDS pour spécifier la classe d'instances similaire qui prend en charge uniquement le stockage EBS pour vos charges de travail de base de données. Par exemple, si la classe d'instance de base de données actuelle est db.r6gd.4xlarge, choisissez db.r6g.4xlarge pour revenir en arrière. Pour de plus amples informations, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Surveillance des instances de base de données qui utilisent RDS Optimized Reads

Vous pouvez surveiller les instances de base de données qui utilisent des lectures optimisées RDS à l'aide des CloudWatch métriques suivantes :

- `FreeLocalStorage`
- `ReadIOPSLocalStorage`
- `ReadLatencyLocalStorage`
- `ReadThroughputLocalStorage`
- `WriteIOPSLocalStorage`
- `WriteLatencyLocalStorage`
- `WriteThroughputLocalStorage`

Ces métriques fournissent des données sur le stockage disponible dans le stockage d'instances, les IOPS et le débit. Pour plus d'informations sur ces métriques, consultez Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS.

Pour surveiller l'utilisation actuelle de votre stockage local, connectez-vous à votre base de données et exécutez la requête suivante :

```
SELECT
    spcname AS "Name",
    pg_catalog.pg_size_pretty(pg_catalog.pg_tablespace_size(oid)) AS "size"
FROM
    pg_catalog.pg_tablespace
```

```
WHERE
    spcname IN ('rds_temp_tablespace');
```

Pour plus d'informations sur les fichiers temporaires et leur utilisation, consultezGestion des fichiers temporaires avec PostgreSQL.

Limites pour Lectures optimisées pour RDS dans PostgreSQL

La limitation suivante s'applique à la fonctionnalité Lectures optimisées pour RDS dans PostgreSQL :

- Les transactions peuvent échouer lorsque le stockage d'instances est plein.

Importer des données dans Postgre SQL sur Amazon RDS

Supposons que vous disposiez d'un SQL déploiement Postgre existant que vous souhaitez transférer vers AmazonRDS. La complexité de votre tâche dépend de la taille de votre base de données et des types d'objets de base de données que vous transférez. Prenons l'exemple d'une base de données qui contient des jeux de données se mesurant en gigaoctets, ainsi que des déclencheurs et des procédures stockés. Une telle base de données va être plus compliquée qu'une base de données simple avec seulement quelques mégaoctets de données de test et pas de déclencheurs, ni de procédures stockés.

Nous vous recommandons d'utiliser les outils natifs de migration SQL de base de données Postgre dans les conditions suivantes :

- Vous avez une migration homogène, dans le sens où vous migrez depuis une base de données avec le même moteur de base de données que la base de données cible.
- Vous migrez une base de données entière.
- Les outils natifs vous permettent de migrer votre système avec une interruption minimale.

Dans la plupart des autres cas, la meilleure approche consiste à effectuer une migration AWS de base de données à l'aide de Database Migration Service (AWS DMS). AWS DMSpeut migrer des bases de données sans interruption de service et, pour de nombreux moteurs de base de données, poursuivre la réplication continue jusqu'à ce que vous soyez prêt à passer à la base de données cible. Vous pouvez migrer vers le même moteur de base de données ou vers un autre moteur de base de données à l'aide de AWS DMS. Si vous migrez vers un moteur de base de données différent de celui de votre base de données source, vous pouvez utiliser le AWS Schema Conversion Tool (AWS SCT). Vous pouvez l'utiliser AWS SCT pour migrer des objets de schéma qui ne sont pas migrés par AWS DMS. Pour plus d'informations AWS DMS, voir Qu'est-ce que c'est AWS Database Migration Service ?

Modifiez votre groupe de paramètres de base de données pour inclure les paramètres suivants pour votre importation uniquement. Vous devez tester les réglages des paramètres pour déterminer les réglages les plus efficaces pour la taille de votre instance de base de données. Vous devez également revenir aux valeurs de production pour ces paramètres une fois votre importation terminée.

Modifiez les paramètres de l'instance de base de données comme suit :

- Désactivez les sauvegardes de l'instance de base de données (affectez la valeur 0 à backup_retention).
- Désactivez le mode multi-AZ.

Modifiez votre groupe de paramètres DB pour inclure les paramètres suivants. Vous devez utiliser ces paramètres uniquement lors de l'importation des données. Vous devez tester les réglages des paramètres pour déterminer les réglages les plus efficaces pour la taille de votre instance de base de données. Vous devez également revenir aux valeurs de production pour ces paramètres une fois votre importation terminée.

Paramètre	Valeur recommandée lors de l'importation	Description
`maintenance_work_mem`	524288, 1048576, 2097152 ou 4194304 (en Ko). Ces paramètres sont comparables à 512 Mo, 1 Go, 2 Go et 4 Go.	La valeur de ce paramètre dépend de la taille de votre hôte. Ce paramètre est utilisé lors CREATE INDEX des instructions et chaque commande parallel peut utiliser cette quantité de mémoire. Calculez la meilleure valeur afin de ne pas définir de valeur si élevée et risquer de manquer de mémoire.
`max_wal_size`	256 (pour la version 9.6), 4096 (pour les versions 10 et ultérieures)	Taille maximale pour les laisser WAL pousser lors des points de contrôle automatiques. L'augmentation de ce paramètre peut augmenter le temps nécessaire à la reprise sur incident. Ce paramètre remplace `checkpoint_segments` Postgre SQL 9.6 et versions ultérieures. Pour Postgre SQL version 9.6, cette valeur est exprimée en unités de 16 Mo. Pour les versions ultérieures, la valeur est exprimée en unités de 1 Mo. Par exemple, dans la version 9.6, 128 signifie 128 fragments d'une taille de 16 Mo chacun. Dans la version 12.4, 2048 signifie 2048 fragments de 1 Mo chacun.

Paramètre	Valeur recommandée lors de l'importation	Description
`checkpoint_timeout`	1800	La valeur de ce paramètre permet une WAL rotation moins fréquente.
`synchronous_commit`	Désactivé	Désactivez ce paramètre pour accélérer les écritures. La désactivation de ce paramètre peut augmenter le risque de perte de données en cas de panne du serveur (ne le désactivez pasFSYNC).
`wal_buffers`	8192	Cette valeur est en unités de 8 Ko. Cela contribue encore une fois à votre vitesse WAL de génération
`autovacuum`	0	Désactivez le paramètre d'aspiration SQL automatique Postgre pendant que vous chargez des données afin qu'il n'utilise pas de ressources

Utilisez les commandes `pg_dump -Fc` (compressé) ou `pg_restore -j` (parallèle) avec ces paramètres.

> **Note**
>
> La SQL commande Postgre `pg_dumpall` nécessite des autorisations super_user qui ne sont pas accordées lorsque vous créez une instance de base de données. Elle ne peut donc pas être utilisée pour importer des données.

Rubriques

- Importation d'une SQL base de données Postgre depuis une instance Amazon EC2
- Utilisation de la commande \ copy pour importer des données dans une table sur une instance de SQL base de données Postgre
- Importation de données Amazon S3 dans une instance de base de données RDS for PostgreSQL d'un

- Transport de bases de données Postgre entre de bases de données

Importation d'une SQL base de données Postgre depuis une instance Amazon EC2

Si vous avez des données sur un SQL serveur Postgre sur une EC2 instance Amazon et que vous souhaitez les déplacer vers une SQL instance de base de données Postgre, vous pouvez suivre ce processus pour migrer les données.

1. Créez un fichier contenant les données à charger à l'aide de pg_dump
2. Créez l'instance de base de données cible
3. Utilisez psql pour créer la base de données sur l'instance de base de données et pour charger les données
4. Créez un instantané de base de données de l'instance de base de données

Les sections suivantes fournissent plus de détails sur chacune des étapes répertoriées ci-dessus.

Étape 1 : Créer un fichier contenant les données à charger à l'aide de pg_dump

L'`pg_dump`utilitaire utilise la COPY commande pour créer un schéma et un vidage des données d'une SQL base de données Postgre. Le script de vidage généré par `pg_dump` charge les données dans une base de données dotée du même nom et recrée les tables, les index et les clés étrangères. Vous pouvez utiliser la commande `pg_restore` et le paramètre `-d` pour restaurer les données dans une base de données dotée d'un nom différent.

Avant de créer le vidage des données, vous devez interroger les tables à vider pour obtenir le nombre de lignes afin de pouvoir confirmer ce nombre sur l'instance de base de données cible.

La commande suivante crée un fichier de vidage mydb2dump.sql pour une base de données nommée mydb2.

```
prompt>pg_dump dbname=mydb2 -f mydb2dump.sql
```

Étape 2 : Créer l'instance de bases de données cible

Créez l'SQLinstance de base de données Postgre cible à l'aide de la RDS console Amazon AWS CLI, ouAPI. Créez l'instance avec le paramètre de rétention des sauvegardes défini sur 0 et

désactivez le mode multi-AZ. Cela vous permet d'effectuer une importation plus rapide des données. Vous devez créer une base de données sur l'instance avant de pouvoir vider les données. La base de données peut avoir le même nom que celle qui contenait les données vidées. Sinon, vous pouvez créer une base de données avec un autre nom. Dans ce cas, vous pouvez utiliser la commande `pg_restore` et le paramètre `-d` pour restaurer les données dans une base de données dotée d'un nouveau nom.

Par exemple, les commandes suivantes permettent de vider, de restaurer et de renommer une base de données.

```
pg_dump -Fc -v -h [endpoint of instance] -U [master username] [database]
 > [database].dump
createdb [new database name]
pg_restore -v -h [endpoint of instance] -U [master username] -d [new database
 name] [database].dump
```

Étape 3 : Utiliser psql pour créer la base de données sur l'instance de base de données et charger les données

Vous pouvez utiliser la même connexion que vous avez utilisée pour exécuter la commande pg_dump pour vous connecter à l'instance de base de données cible et recréer la base de données. Grâce à psql, vous pouvez utiliser l'identifiant principal et le mot de passe principal pour créer la base de données sur l'instance de base de données.

L'exemple suivant utilise psql et un fichier de vidage nommé mydb2dump.sql pour créer une base de données appelée mydb2 sur une instance de base de données Postgre appelée mypginstance SQL :

Dans Linux, macOS, ou Unix:

```
psql \
   -f mydb2dump.sql \
   --host mypginstance.555555555555.aws-region.rds.amazonaws.com \
   --port 8199 \
   --username myawsuser \
   --password password \
   --dbname mydb2
```

Dans Windows:

```
psql ^
```

```
    -f mydb2dump.sql ^
    --host mypginstance.555555555555.aws-region.rds.amazonaws.com ^
    --port 8199 ^
    --username myawsuser ^
    --password password ^
    --dbname mydb2
```

> **Note**
>
> Spécifiez un mot de passe autre que celui indiqué ici, en tant que bonne pratique de sécurité.

Étape 4 : Créer un instantané de base de données de l'instance de bases de données

Une fois que vous avez vérifié que les données ont été chargées dans votre instance de base de données, nous vous recommandons de créer un instantané de base de données de l'SQLinstance de base de données Postgre cible. Les snapshots DB sont des sauvegardes complètes de votre instance de base de données qui peuvent être utilisées pour restaurer l'instance de base de données à un état connu. Un instantané de base de données pris immédiatement après le chargement vous évite de devoir charger les données à nouveau en cas d'incident. Vous pouvez également l'utiliser pour créer de nouvelles instances de base de données. Pour plus d'informations sur la création d'un instantané de base de données, consultez Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

Utilisation de la commande \ copy pour importer des données dans une table sur une instance de SQL base de données Postgre

La SQL `\copy` commande Postgre est une méta-commande disponible dans l'outil client `psql` interactif. Vous pouvez l'utiliser `\copy` pour importer des données dans une table sur votre SQL instance RDS de base de données Postgre. Pour utiliser la commande `\copy`, vous devez d'abord créer la structure de la table sur l'instance de base de données cible, afin que `\copy` dispose d'une destination pour les données copiées.

Vous pouvez l'utiliser `\copy` pour charger des données à partir d'un fichier de valeurs séparées par des virgules (CSV), tel qu'un fichier exporté et enregistré sur votre poste de travail client.

Pour importer les CSV données vers la cible RDS pour l'instance de base de données Postgre, connectez-vous d'abord à l'SQLinstance de base de données cible à l'aide `psql` de.

```
psql --host=db-instance.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=target-db
```

Exécutez ensuite la commande \copy avec les paramètres suivants afin d'identifier la cible pour les données et son format.

- target_table— Nom de la table qui doit recevoir les données copiées depuis le CSV fichier.
- column_list : spécifications des colonnes pour la table.
- 'filename'— Le chemin complet du CSV fichier sur votre poste de travail local.

```
 \copy target_table from '/path/to/local/filename.csv' WITH DELIMITER ',' CSV;
```

Si votre CSV fichier contient des informations d'en-tête de colonne, vous pouvez utiliser cette version de la commande et des paramètres.

```
\copy target_table (column-1, column-2, column-3, ...)
    from '/path/to/local/filename.csv' WITH DELIMITER ',' CSV HEADER;
```

Si la \copy commande échoue, Postgre SQL affiche des messages d'erreur.

Création d'une nouvelle instance de base de données dans l'environnement de prévisualisation de base de données à l'aide de la psql commande associée à la \copy méta-commande, comme indiqué dans les exemples suivants. Cet exemple utilise source-table comme nom de tableau source, source-table.csv comme fichier .csv et target-db comme base de données cible :

Dans Linux, macOS, ou Unix:

```
$psql target-db \
    -U <admin user> \
    -p <port> \
    -h <DB instance name> \
    -c "\copy source-table from 'source-table.csv' with DELIMITER ','"
```

Dans Windows:

```
$psql target-db ^
    -U <admin user> ^
    -p <port> ^
    -h <DB instance name> ^
```

```
    -c "\copy source-table from 'source-table.csv' with DELIMITER ','"
```

Pour plus de détails sur la `\copy` commande, consultez la page psql de la SQL documentation de Postgre, dans la section Méta-commandes.

Importation de données Amazon S3 dans une instance de base de données RDS for PostgreSQL d'un

Vous pouvez importer des données qui ont été stockées à l'aide d'Amazon Simple Storage Service dans une table sur une instance de base de données RDS for PostgreSQL. Pour ce faire, vous devez d'abord installer l'extension `aws_s3` RDS for PostgreSQL. Cette extension fournit les fonctions que vous utilisez pour importer des données à partir d'un compartiment Amazon S3. Un compartiment est un conteneur Amazon S3 pour les objets et les fichiers. Les données peuvent résider dans un fichier CSV (valeur séparée par des virgules), un fichier texte ou un fichier compressé (gzip). Vous apprendrez ensuite comment installer l'extension et comment importer des données d'Amazon S3 dans un tableau.

Votre base de données doit exécuter PostgreSQL version 10.7 ou supérieure pour importer depuis Simple Storage Service (Amazon S3) vers RDS for PostgreSQL.

Si vous n'avez pas de données stockées sur Amazon S3, vous devez d'abord créer un compartiment et y stocker les données. Pour en savoir plus, consulter les rubriques suivantes dans le Guide de l'utilisateur d'Amazon Simple Storage Service.

- Créez un compartiment
- Ajout d'un objet dans un compartiment

L'importation entre comptes depuis Amazon S3 est prise en charge. Pour plus d’informations, consultez Octroi d'autorisations entre comptes dans le Guide de l'utilisateur Amazon Simple Storage Service.

Vous pouvez utiliser la clé gérée par le client pour le chiffrement lors de l'importation de données depuis S3. Pour plus d'informations, consultez Clés KMS stockées dans AWS KMS dans le Guide de l'utilisateur Amazon Simple Storage Service.

Rubriques

- Installation de l'extension aws_s3
- Présentation de l'importation de données à partir de données Amazon S3

- Configuration de l'accès à un compartiment Amazon S3
- Importation de données d'Amazon S3 vers votre instance de base de données RDS for PostgreSQL
- Références de fonctions

Installation de l'extension aws_s3

Avant de pouvoir utiliser Amazon S3 avec votre pour l'SQLinstance de base de données Postgre, vous devez installer l'`aws_s3`extension. Cette extension fournit des fonctions pour importer des données depuis un compartiment Amazon S3. Il fournit également des fonctions d'exportation de données d' et d'une SQL instance RDS de base de données Postgre vers un compartiment Amazon S3. Pour de plus amples informations, veuillez consulter Exportation de données à partir d'une instance de base de données RDS for PostgreSQL vers Amazon S3. L'extension `aws_s3` dépend de certaines des fonctions d'aide de l'extension `aws_commons`, qui est installée automatiquement lorsque cela est nécessaire.

Pour installer l'extension **`aws_s3`**

1. Utilisez psql (oupgAdmin) pour vous connecter à l' pour SQL instance de base de données Postgre en tant qu'utilisateur disposant de privilèges. `rds_superuser` Si vous avez conservé le nom par défaut pendant le processus d'installation, vous vous connectez en tant que `postgres`.

   ```
   psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password
   ```

2. Pour installer l'extension, exécutez la commande suivante.

   ```
   postgres=> CREATE EXTENSION aws_s3 CASCADE;
   NOTICE: installing required extension "aws_commons"
   CREATE EXTENSION
   ```

3. Pour vérifier que l'extension est installée, vous pouvez utiliser la métacommande psql `\dx`.

   ```
   postgres=> \dx
          List of installed extensions
       Name     | Version |   Schema   |                 Description
   -------------+---------+------------+---------------------------------------------
    aws_commons | 1.2     | public     | Common data types across AWS services
    aws_s3      | 1.1     | public     | AWS S3 extension for importing data from S3
   ```

```
 plpgsql     | 1.0     | pg_catalog | PL/pgSQL procedural language
(3 rows)
```

Les fonctions d'importation de données depuis Amazon S3 et d'exportation de données vers Amazon S3 sont désormais disponibles.

Présentation de l'importation de données à partir de données Amazon S3

Pour importer des données S3 dans Amazon RDS

Tout d'abord, rassemblez les informations que vous devez fournir à la fonction. Il s'agit notamment du nom de la table sur l'instance de base de données RDS pour PostgreSQL, ainsi que du nom du compartiment, du chemin du fichier, du type de fichier et de l'endroit où Région AWS les données Amazon S3 sont stockées. Pour de plus amples informations, veuillez consulter View an object (Afficher un objet) dans le Guide de l'utilisateur Amazon Simple Storage Service.

> **Note**
>
> L'importation de données partitionnées depuis Amazon S3 n'est pas prise en charge actuellement.

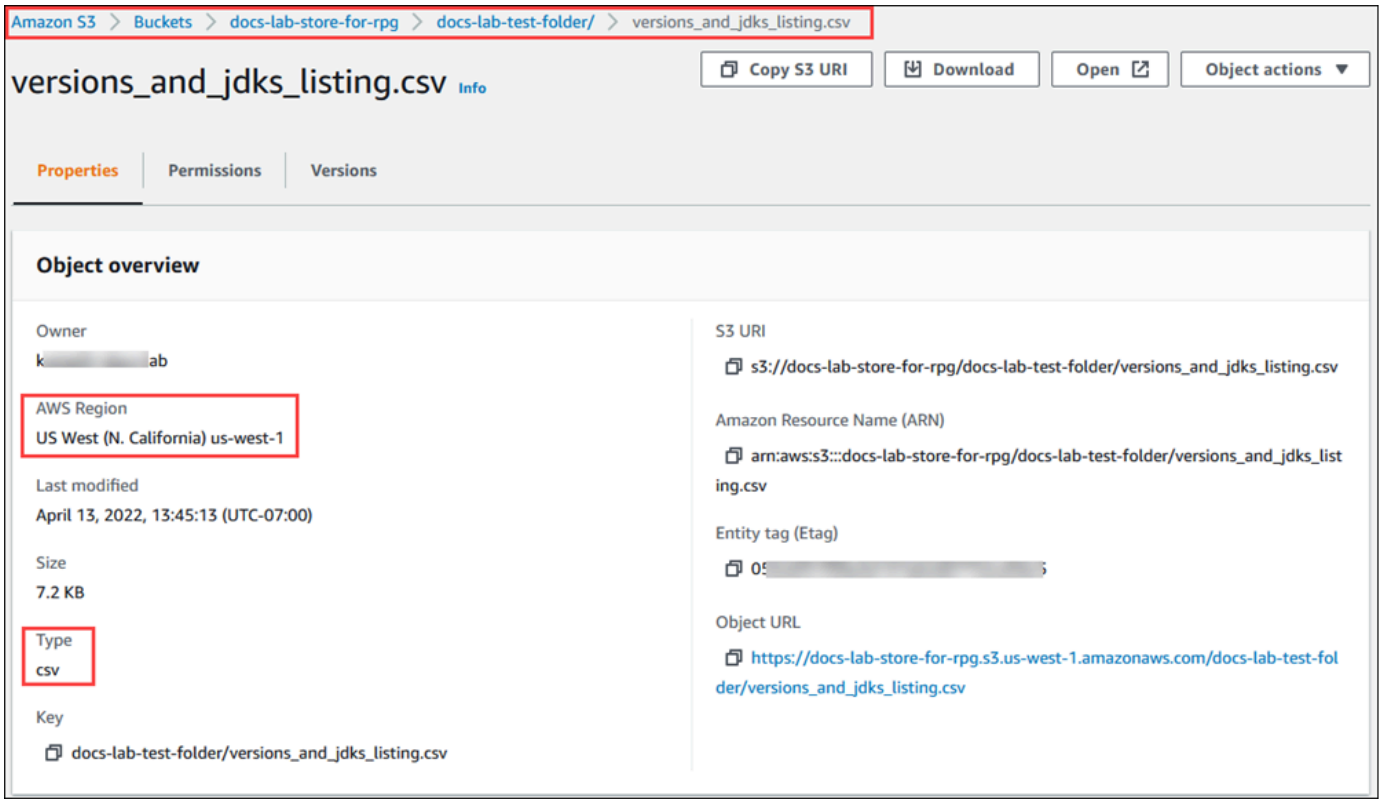
1. Obtenez le nom de la table dans laquelle la fonction `aws_s3.table_import_from_s3` doit importer les données. À titre d'exemple, la commande suivante crée une table `t1` qui peut être utilisée dans les étapes suivantes.

   ```
   postgres=> CREATE TABLE t1
       (col1 varchar(80),
       col2 varchar(80),
       col3 varchar(80));
   ```

2. Obtenez les détails sur le compartiment Amazon S3 et les données à importer. Pour ce faire, ouvrez la console Amazon S3 à l'adresse https://console.aws.amazon.com/s3/, puis choisissez Buckets. Trouvez le compartiment contenant vos données dans la liste. Sélectionnez le compartiment, ouvrez sa page Object overview (Présentation des objets), puis choisissez Properties (Propriétés).

   Notez le nom du compartiment, le chemin Région AWS, le et le type de fichier. Vous aurez besoin du nom Amazon Resource Name (ARN) pour configurer l'accès à Amazon S3 via un rôle

IAM. Pour obtenir plus d'informations, consultez Configuration de l'accès à un compartiment Amazon S3. L'image suivante montre un exemple.



3. Vous pouvez vérifier le chemin d'accès aux données du compartiment Amazon S3 à l'aide de la AWS CLI commande`aws s3 cp`. Si les informations sont correctes, cette commande télécharge une copie du fichier Amazon S3.

```
aws s3 cp s3://amzn-s3-demo-bucket/sample_file_path ./
```

4. Configurez les autorisations sur votre instance de base de données RDS for PostgreSQL pour permettre l'accès au fichier sur le compartiment Amazon S3. Pour ce faire, vous devez utiliser un rôle AWS Identity and Access Management (IAM) ou des informations d'identification de sécurité. Pour de plus amples informations, veuillez consulter Configuration de l'accès à un compartiment Amazon S3.

5. Fournissez le chemin et les autres détails de l'objet Amazon S3 recueillis (voir l'étape 2) à la fonction `create_s3_uri` pour construire un objet URI Amazon S3. Pour en savoir plus sur cette fonction, consultez aws_commons.create_s3_uri. Voici un exemple de construction de cet objet pendant une session psql.

```
postgres=> SELECT aws_commons.create_s3_uri(
   'docs-lab-store-for-rpg',
```

```
   'versions_and_jdks_listing.csv',
   'us-west-1'
) AS s3_uri \gset
```

Dans l'étape suivante, vous transmettez cet objet (`aws_commons._s3_uri_1`) à la fonction `aws_s3.table_import_from_s3` pour importer les données dans la table.

6. Appelez la fonction `aws_s3.table_import_from_s3` pour importer les données d'Amazon S3 dans votre table. Pour obtenir des informations de référence, consultez aws_s3.table_import_from_s3. Pour obtenir des exemples, consultez Importation de données d'Amazon S3 vers votre instance de base de données RDS for PostgreSQL.

Configuration de l'accès à un compartiment Amazon S3

Pour importer des données à partir d'un fichier Amazon S3, vous devez accorder à l'instance de base de données RDS for PostgreSQL une autorisation d'accès au compartiment Amazon S3 contenant le fichier. Pour accorder l'accès à un compartiment Amazon S3, vous pouvez employer une des deux méthodes décrites dans les rubriques suivantes.

Rubriques

- Utilisation d'un rôle IAM pour accéder à un compartiment Amazon S3
- Utilisation d'informations d'identification de sécurité pour accéder à un compartiment Amazon S3
- Résolution des problèmes d'accès à Amazon S3

Utilisation d'un rôle IAM pour accéder à un compartiment Amazon S3

Avant de charger des données à partir d'un fichier Amazon S3, accordez à votre instance de base de données RDS for PostgreSQL l'autorisation d'accéder au compartiment Amazon S3 dans lequel se trouve le fichier. De cette façon, vous n'avez pas à gérer d'informations d'identification supplémentaires ni à les fournir dans l'appel de fonction aws_s3.table_import_from_s3.

Pour ce faire, créez une politique IAM qui donne accès au compartiment Amazon S3. Créez un rôle IAM et attachez la politique à ce rôle. Attribuez ensuite le rôle IAM à votre instance de base de données.

Pour permettre à une instance de base de données RDS for PostgreSQL d'accéder à Amazon S3 via un rôle IAM

1. Créez une politique IAM.

Celle-ci fournit au compartiment et à l'objet les autorisations permettant à votre instance de base de données RDS for PostgreSQL d'accéder à Amazon S3.

Incluez à la politique les actions obligatoires suivantes pour permettre le transfert de fichiers d'un compartiment Amazon S3 vers Amazon RDS :

- `s3:GetObject`
- `s3:ListBucket`

Incluez à la politique les ressources suivantes pour identifier le compartiment Amazon S3 et les objets qu'il contient. Voici le format Amazon Resource Name (ARN) permettant d'accéder à Amazon S3 :

- arn:aws:s3 : *`amzn-s3-demo-bucket`*
- arn:aws:s3 : :1 /* *`amzn-s3-demo-bucket`*

Pour obtenir plus d'informations sur la création d'une politique IAM pour RDS for PostgreSQL, consultez Création et utilisation d'une politique IAM pour l'accès à une base de données IAM. Consultez également Didacticiel : création et attachement de votre première politique gérée par le client dans le Guide de l'utilisateur IAM.

La AWS CLI commande suivante crée une politique IAM nommée `rds-s3-import-policy` avec ces options. Elle accorde un accès à un compartiment nommé *`amzn-s3-demo-bucket`*.

> **Note**
>
> Notez le Amazon Resource Name (ARN) de la politique renvoyée par cette commande. Vous en aurez besoin par la suite pour attacher la politique à un rôle IAM.

Example

Pour LinuxmacOS, ou Unix :

```
aws iam create-policy \
   --policy-name rds-s3-import-policy \
   --policy-document '{
```

```
    "Version": "2012-10-17",
    "Statement": [
      {
        "Sid": "s3import",
        "Action": [
          "s3:GetObject",
          "s3:ListBucket"
        ],
        "Effect": "Allow",
        "Resource": [
          "arn:aws:s3:::amzn-s3-demo-bucket",
          "arn:aws:s3:::amzn-s3-demo-bucket/*"
        ]
      }
    ]
  }'
```

Dans Windows :

```
aws iam create-policy ^
   --policy-name rds-s3-import-policy ^
   --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "s3import",
         "Action": [
           "s3:GetObject",
           "s3:ListBucket"
         ],
         "Effect": "Allow",
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket",
           "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
       }
     ]
   }'
```

2. Créez un rôle IAM.

L'objectif est ici de permettre à Amazon RDS d'endosser ce rôle IAM pour accéder à vos compartiments Amazon S3. Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des politiques basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

Si vous utilisez les deux clés de contexte de condition globale et que la valeur de `aws:SourceArn` contient l'ID de compte, la valeur de `aws:SourceAccount` et le compte indiqué dans la valeur de `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de politique.

- Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
- Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

Dans la politique, veillez à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. L'exemple suivant montre comment procéder à l'aide de la AWS CLI commande pour créer un rôle nommé`rds-s3-import-role`.

Example

Pour LinuxmacOS, ou Unix :

```
aws iam create-role \
   --role-name rds-s3-import-role \
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
```

```
                "aws:SourceAccount": "111122223333",
                "aws:SourceArn": "arn:aws:rds:us-east-1:111122223333:db:dbname"
                }
            }
        }
      ]
   }'
```

Dans Windows :

```
aws iam create-role ^
   --role-name rds-s3-import-role ^
   --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
                "aws:SourceAccount": "111122223333",
                "aws:SourceArn": "arn:aws:rds:us-east-1:111122223333:db:dbname"
                }
            }
       }
     ]
   }'
```

3. Attachez la politique IAM que vous avez créée au rôle IAM que vous venez de créer.

   La AWS CLI commande suivante associe la politique créée à l'étape précédente au rôle nommé `rds-s3-import-role` Remplacer *`your-policy-arn`* par l'ARN de stratégie que vous avez noté à l'étape précédente.

   Example

   Pour LinuxmacOS, ou Unix :

```
aws iam attach-role-policy \
```

```
    --policy-arn your-policy-arn \
    --role-name rds-s3-import-role
```

Dans Windows :

```
aws iam attach-role-policy ^
    --policy-arn your-policy-arn ^
    --role-name rds-s3-import-role
```

4. Ajoutez le rôle IAM à l'instance de base de données.

   Pour ce faire, utilisez le AWS Management Console ou AWS CLI, comme décrit ci-dessous.

console

Pour ajouter un rôle IAM à l'instance de base de données PostgreSQL à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez le nom de l'instance de base de données PostgreSQL pour afficher ses détails.
3. Sous l'onglet Connectivity & security (Connectivité et sécurité), accédez à la section Manage IAM roles (Gérer les rôles IAM) et choisissez le rôle à ajouter sous Add IAM roles to this instance (Ajouter des rôles IAM à ce cluster/cette instance).
4. Sous Feature (Fonction), choisissez s3Import.
5. Choisissez Add role (Ajouter un rôle).

AWS CLI

Pour ajouter un rôle IAM à une instance de base de données PostgreSQL à l'aide de la CLI

- Utilisez la commande suivante pour ajouter le rôle à l'instance de base de données PostgreSQL nommée `my-db-instance`. Remplacez *`your-role-arn`* par l'ARN de rôle que vous avez noté lors d'une étape précédente. Utilisez `s3Import` comme valeur de l'option `--feature-name`.

  Example

  Pour LinuxmacOS, ou Unix :

```
aws rds add-role-to-db-instance \
```

```
   --db-instance-identifier my-db-instance \
   --feature-name s3Import \
   --role-arn your-role-arn   \
   --region your-region
```

Dans Windows :

```
aws rds add-role-to-db-instance ^
   --db-instance-identifier my-db-instance ^
   --feature-name s3Import ^
   --role-arn your-role-arn ^
   --region your-region
```

API RDS

Pour ajouter un rôle IAM pour une instance de cluster de base de données à l'aide de l'API Amazon RDS, appelez l'opération.

Utilisation d'informations d'identification de sécurité pour accéder à un compartiment Amazon S3

Si vous préférez, au lieu de donner à accès un compartiment Amazon S3 avec un rôle IAM, vous pouvez utiliser des informations d'identification de sécurité. Pour ce faire, spécifiez le paramètre `credentials` dans l'appel de fonction aws_s3.table_import_from_s3.

Le `credentials` paramètre est une structure de type contenant `aws_commons._aws_credentials_1` des AWS informations d'identification. Utilisez la fonction aws_commons.create_aws_credentials pour définir la clé d'accès et la clé secrète dans une structure `aws_commons._aws_credentials_1`, comme indiqué ci-après.

```
postgres=> SELECT aws_commons.create_aws_credentials(
   'sample_access_key', 'sample_secret_key', '')
AS creds \gset
```

Après avoir créé la structure `aws_commons._aws_credentials_1` , utilisez la fonction aws_s3.table_import_from_s3 avec le paramètre `credentials` pour importer les données, comme indiqué ci-après.

```
postgres=> SELECT aws_s3.table_import_from_s3(
   't', '', '(format csv)',
   :'s3_uri',
```

```
   :'creds'
);
```

Vous pouvez également inclure l'appel de fonction aws_commons.create_aws_credentials en ligne au sein de l'appel de fonction `aws_s3.table_import_from_s3`.

```
postgres=> SELECT aws_s3.table_import_from_s3(
   't', '', '(format csv)',
   :'s3_uri',
   aws_commons.create_aws_credentials('sample_access_key', 'sample_secret_key', '')
);
```

Résolution des problèmes d'accès à Amazon S3

Si vous rencontrez des problèmes de connexion lorsque vous tentez d'importer des données depuis Amazon S3, consultez les recommandations suivantes :

- Résolution des problèmes liés à Identity and Access Amazon RDS
- Dépannage d'Amazon S3 dans le Guide de l'utilisateur Amazon Simple Storage Service.
- Dépannage d'Amazon S3 et IAM dans le Guide de l'utilisateur IAM

Importation de données d'Amazon S3 vers votre instance de base de données RDS for PostgreSQL

Vous importez des données depuis votre compartiment Amazon S3 en utilisant la fonction `table_import_from_s3` de l'extension aws_s3. Pour obtenir des informations de référence, consultez aws_s3.table_import_from_s3.

> **Note**
>
> Les exemples suivants utilisent la méthode du rôle IAM pour donner accès au compartiment Amazon S3. Les appels de fonction `aws_s3.table_import_from_s3` n'incluent donc aucun paramètre d'informations d'identification.

L'exemple suivant montre un exemple typique.

```
postgres=> SELECT aws_s3.table_import_from_s3(
```

```
   't1',
   '',
   '(format csv)',
   :'s3_uri'
);
```

Les paramètres sont les suivants :

- `t1` – Nom de la table de l'instance de base de données PostgreSQL dans laquelle copier les données.
- `''` – Liste facultative des colonnes de la table de base de données. Vous pouvez utiliser ce paramètre pour indiquer quelles colonnes des données S3 sont copiées dans quelles colonnes de table. Si aucune colonne n'est spécifiée, toutes les colonnes sont copiées dans la table. Pour obtenir un exemple d'utilisation d'une liste de colonnes, veuillez consulter Importation d'un fichier Amazon S3 qui utilise un délimiteur personnalisé.
- `(format csv)` – Arguments PostgreSQL COPY. Le processus de copie utilise les arguments et le format de la commande PostgreSQL COPY pour importer les données. Les choix de format comprennent les valeurs séparées par des virgules (CSV) comme dans cet exemple, le texte et les données binaires. Par défaut, il s'agit de texte.
- `s3_uri` – Structure contenant les informations d'identification du fichier Amazon S3. Pour obtenir un exemple d'utilisation de la fonction aws_commons.create_s3_uri pour créer une structure `s3_uri`, consultez Présentation de l'importation de données à partir de données Amazon S3.

Pour de plus amples informations sur cette fonction, veuillez consulter aws_s3.table_import_from_s3.

La fonction `aws_s3.table_import_from_s3` retourne du texte. Pour spécifier d'autres types de fichiers à importer à partir d'un compartiment Amazon S3, consultez l'un des exemples suivants.

> **Note**
>
> L'importation d'un fichier de 0 octet entraîne une erreur.

Rubriques

- Importation d'un fichier Amazon S3 qui utilise un délimiteur personnalisé
- Importation d'un fichier compressé Amazon S3 (gzip)
- Importation d'un fichier codé Amazon S3

Importation d'un fichier Amazon S3 qui utilise un délimiteur personnalisé

L'exemple suivant montre comment importer un fichier qui utilise un délimiteur personnalisé. Il montre également comment définir l'emplacement de destination des données dans la table de base de données à l'aide du paramètre `column_list` de la fonction aws_s3.table_import_from_s3.

Pour cet exemple, supposons que les informations suivantes sont organisées en colonnes délimitées par une barre verticale dans le fichier Amazon S3.

```
1|foo1|bar1|elephant1
2|foo2|bar2|elephant2
3|foo3|bar3|elephant3
4|foo4|bar4|elephant4
...
```

Pour importer un fichier qui utilise un délimiteur personnalisé

1. Créez une table dans la base de données pour les données importées.

   ```
   postgres=> CREATE TABLE test (a text, b text, c text, d text, e text);
   ```

2. Utilisez le format suivant de la fonction aws_s3.table_import_from_s3 pour importer des données à partir du fichier Amazon S3.

   Vous pouvez inclure l'appel de fonction aws_commons.create_s3_uri en ligne au sein de l'appel de fonction `aws_s3.table_import_from_s3` pour spécifier le fichier.

   ```
   postgres=> SELECT aws_s3.table_import_from_s3(
      'test',
      'a,b,d,e',
      'DELIMITER ''|''',
      aws_commons.create_s3_uri('amzn-s3-demo-bucket', 'pipeDelimitedSampleFile', 'us-
   east-2')
   );
   ```

Les données se retrouvent désormais dans la table dans les colonnes suivantes.

```
postgres=> SELECT * FROM test;
a | b | c | d | e
---+------+---+---+------+-----------
```

```
1 | foo1 | | bar1 | elephant1
2 | foo2 | | bar2 | elephant2
3 | foo3 | | bar3 | elephant3
4 | foo4 | | bar4 | elephant4
```

Importation d'un fichier compressé Amazon S3 (gzip)

L'exemple suivant montre comment importer un fichier compressé avec gzip à partir d'Amazon S3. Le fichier que vous importez doit comporter les métadonnées Amazon S3 suivantes :

- Clé : `Content-Encoding`
- Valeur : `gzip`

Si vous chargez le fichier à l'aide du AWS Management Console, les métadonnées sont généralement appliquées par le système. Pour plus d'informations sur le chargement de fichiers vers Amazon S3 à l'aide de, de AWS Management Console AWS CLI, ou de l'API, consultez la section Chargement d'objets dans le guide de l'utilisateur d'Amazon Simple Storage Service.

Pour de plus amples informations sur les métadonnées Amazon S3 et les métadonnées fournies par le système, veuillez consulter Editing object metadata in the Amazon S3 console dans le Guide de l'utilisateur Amazon Simple Storage Service.

Importez le fichier gzip dans votreinstance de base de données RDS for PostgreSQL comme décrit ci-après.

```
postgres=> CREATE TABLE test_gzip(id int, a text, b text, c text, d text);
postgres=> SELECT aws_s3.table_import_from_s3(
 'test_gzip', '', '(format csv)',
 'amzn-s3-demo-bucket', 'test-data.gz', 'us-east-2'
);
```

Importation d'un fichier codé Amazon S3

L'exemple suivant montre comment importer un fichier codé en Windows-1252 à partir d'Amazon S3.

```
postgres=> SELECT aws_s3.table_import_from_s3(
 'test_table', '', 'encoding ''WIN1252''',
 aws_commons.create_s3_uri('amzn-s3-demo-bucket', 'SampleFile', 'us-east-2')
);
```

Références de fonctions

Fonctions

- aws_s3.table_import_from_s3
- aws_commons.create_s3_uri
- aws_commons.create_aws_credentials

aws_s3.table_import_from_s3

Importe les données Amazon S3 dans une RDS table SQL Amazon . L'extension `aws_s3` fournit la fonction `aws_s3.table_import_from_s3`. La valeur renvoyée est du texte.

Syntaxe

Les paramètres requis sont `table_name`, `column_list` et `options`. Ils identifient la table de base de données et spécifient la façon dont les données sont copiées dans la table.

Vous pouvez également utiliser les paramètres suivants :

- Le paramètre `s3_info` spécifie le fichier Amazon S3 à importer. Lorsque vous utilisez ce paramètre, l'accès à Amazon S3 est fourni par un IAM rôle pour l'instance de SQL de base de données Postgre.

```
aws_s3.table_import_from_s3 (
   table_name text,
   column_list text,
   options text,
   s3_info aws_commons._s3_uri_1
)
```

- Le paramètre `credentials` spécifie les informations d'identification permettant d'accéder à Amazon S3. Lorsque vous utilisez ce paramètre, vous n'utilisez aucun IAM rôle.

```
aws_s3.table_import_from_s3 (
   table_name text,
   column_list text,
   options text,
   s3_info aws_commons._s3_uri_1,
   credentials aws_commons._aws_credentials_1
)
```

Paramètres

table_name

Chaîne de texte obligatoire contenant le nom de la table de SQL base de données Postgre dans laquelle les données doivent être importées.

column_list

Chaîne de texte obligatoire contenant une liste facultative des colonnes de la table de SQL base de données Postgre dans lesquelles copier les données. Si la chaîne est vide, toutes les colonnes de la table sont utilisées. Pour obtenir un exemple, veuillez consulter Importation d'un fichier Amazon S3 qui utilise un délimiteur personnalisé.

options

Chaîne de texte obligatoire contenant les arguments de la SQL `COPY` commande Postgre. Ces arguments indiquent comment les données doivent être copiées dans la SQL table Postgre. Pour plus de détails, consultez la SQLCOPYdocumentation Postgre.

s3_info

Type composite `aws_commons._s3_uri_1` contenant les informations suivantes sur l'objet S3 :

- `bucket` – Nom du compartiment Amazon S3 contenant le fichier.
- `file_path` – Nom du fichier Amazon S3, avec le chemin d'accès à celui-ci.
- `region`— Le AWS Région dans laquelle se trouve le fichier. Pour une liste des AWS Noms de régions et valeurs associées, voirRégions, zones de disponibilité et zones locales.

credentials

Type composite `aws_commons._aws_credentials_1` contenant les informations d'identification suivantes à utiliser pour l'opération d'importation :

- Clé d'accès
- Clé secrète
- Jeton de session

Pour plus d'informations sur la création d'une structure composite `aws_commons._aws_credentials_1`, veuillez consulter aws_commons.create_aws_credentials.

Syntaxe alternative

Pour faciliter le test, vous pouvez utiliser un ensemble étendu de paramètres au lieu des paramètres `s3_info` et `credentials`. Plusieurs variations de syntaxe supplémentaires pour la fonction `aws_s3.table_import_from_s3` sont fournies ci-dessous.

- Au lieu d'utiliser le paramètre `s3_info` pour identifier un fichier Amazon S3, utilisez la combinaison des paramètres `bucket`, `file_path` et `region`. Avec cette forme de fonction, l'accès à Amazon S3 est fourni par un IAM rôle sur l'SQLinstance de base de données Postgre.

```
aws_s3.table_import_from_s3 (
   table_name text,
   column_list text,
   options text,
   bucket text,
   file_path text,
   region text
)
```

- Au lieu d'utiliser le paramètre `credentials` pour spécifier l'accès à Amazon S3, utilisez la combinaison des paramètres `access_key`, `session_key` et `session_token`.

```
aws_s3.table_import_from_s3 (
   table_name text,
   column_list text,
   options text,
   bucket text,
   file_path text,
   region text,
   access_key text,
   secret_key text,
   session_token text
)
```

Autres paramètres

bucket

Chaîne de texte comportant le nom du compartiment Amazon S3 qui contient le fichier.

file_path

Chaîne de texte contenant le nom du fichier Amazon S3, avec le chemin d'accès à celui-ci.

region

Une chaîne de texte identifiant le Région AWS emplacement du fichier. Pour une liste des Région AWS noms et valeurs associées, voirRégions, zones de disponibilité et zones locales.

access_key

Chaîne de texte contenant la clé d'accès à utiliser pour l'opération d'importation. La valeur par défaut estNULL.

secret_key

Chaîne de texte contenant la clé secrète à utiliser pour l'opération d'importation. La valeur par défaut estNULL.

session_token

(Facultatif) Chaîne de texte contenant la clé de session à utiliser pour l'opération d'importation. La valeur par défaut estNULL.

aws_commons.create_s3_uri

Crée une structure `aws_commons._s3_uri_1` pour contenir les informations relatives au fichier Amazon S3. Utilisez les résultats de la fonction `aws_commons.create_s3_uri` dans le paramètre `s3_info` de la fonction aws_s3.table_import_from_s3.

Syntaxe

```
aws_commons.create_s3_uri(
   bucket text,
   file_path text,
   region text
)
```

Paramètres

bucket

Chaîne de texte obligatoire contenant le nom du compartiment Amazon S3 pour le fichier.

file_path

Chaîne de texte obligatoire contenant le nom du fichier Amazon S3, avec le chemin d'accès à celui-ci.

region

Une chaîne de texte obligatoire contenant Région AWS dans lequel se trouve le fichier. Pour une liste des Région AWS noms et valeurs associées, voirRégions, zones de disponibilité et zones locales.

aws_commons.create_aws_credentials

Définit une clé d'accès et une clé secrète dans une structure `aws_commons._aws_credentials_1`. Utilisez les résultats de la fonction `aws_commons.create_aws_credentials` dans le paramètre `credentials` de la fonction aws_s3.table_import_from_s3.

Syntaxe

```
aws_commons.create_aws_credentials(
   access_key text,
   secret_key text,
   session_token text
)
```

Paramètres

access_key

Chaîne de texte obligatoire contenant la clé d'accès à utiliser pour l'importation d'un fichier Amazon S3. La valeur par défaut estNULL.

secret_key

Chaîne de texte obligatoire contenant la clé secrète à utiliser pour l'importation d'un fichier Amazon S3. La valeur par défaut estNULL.

session_token

Chaîne de texte facultative contenant le jeton de session à utiliser pour l'importation d'un fichier Amazon S3. La valeur par défaut estNULL. Si vous saisissez le paramètre `session_token` facultatif, vous pouvez utiliser les informations d'identification temporaires.

Transport de bases de données Postgre entre de bases de données

En utilisant les bases de données SQL transportables Postgre pour AmazonRDS, vous pouvez déplacer une base de données Postgre entre deux instances SQL de base de données. Il s'agit d'un moyen très rapide de migrer de grandes bases de données entre différentes instances de base de données. Pour utiliser cette approche, vos instances de base de données doivent toutes deux exécuter la même version majeure de PostgreSQL.

Cette fonctionnalité nécessite que vous installiez l'extension `pg_transport` sur les instances de base de données source et de destination. L'extension `pg_transport` fournit un mécanisme de transport physique qui déplace les fichiers de base de données avec un traitement minimal. Ce mécanisme déplace les données beaucoup plus rapidement que les processus traditionnels de vidage et de chargement, avec moins de temps d'arrêt.

> **Note**
>
> Les bases de données SQL transportables Postgre sont disponibles RDS pour Postgre SQL 11.5 et versions ultérieures, et RDS pour Postgre SQL version 10.10 et supérieures.

Pour transporter une instance de SQL base de données Postgre d'une RDS instance de SQL base de données Postgre à une autre, vous devez d'abord configurer les instances source et de destination comme indiqué dans. Configuration d'instances de base de données pour leur transport Vous pouvez ensuite transporter la base de données à l'aide de la fonction décrite dans Transporter une base de données Postgre SQL.

Rubriques

- Que se passe-t-il durant le transport d'une base de données ?
- Limitations liées à l'utilisation des bases de données SQL transportables Postgre
- Configuration du transport d'une base de données Postgre SQL
- Transporter une SQL base de données Postgre vers la destination depuis la source
- Référence des fonctions des base de données transportables
- Référence des paramètres des bases de données transportables

Que se passe-t-il durant le transport d'une base de données ?

La fonctionnalité de bases de données SQL transportables Postgre utilise un modèle d'extraction pour importer la base de données de l'instance de base de données source vers la destination. La fonction `transport.import_from_server` crée la base de données en transit sur l'instance de base de données de destination. La base de données en transit est inaccessible sur l'instance de base de données de destination pendant toute la durée du transport.

Lorsque le transport commence, toutes les sessions en cours sur la base de données source cessent. Les bases de données autres que la base de données source sur l'instance de base de données source ne sont pas affectées par le transport.

La base de données source est placée dans un mode lecture seule spécial. Lorsqu'elle est dans ce mode, vous pouvez vous connecter à la base de données source et exécuter des requêtes de lecture seule. Par contre, les requêtes d'écriture et certains autres types de commandes sont bloqués. Seule la base de données source qui fait l'objet du transport est affectée par ces restrictions.

Durant le transport, vous ne pouvez pas restaurer l'instance de base de données de destination à un instant dans le passé. Cela est dû au fait que le transport n'est pas transactionnel et n'utilise pas le journal d'SQLécriture anticipée de Postgre pour enregistrer les modifications. Si les sauvegardes automatiques sont activées pour l'instance de base de données de destination, une sauvegarde est automatiquement effectuée une fois le transport terminé. Point-in-timeles restaurations sont disponibles pendant un certain temps après la fin de la sauvegarde.

En cas d'échec du transport, l'extension `pg_transport` tente d'annuler toutes les modifications apportées aux instances de base de données source et de destination. Cela inclut la suppression de la base de données partiellement transportée sur la destination. Selon le type de défaillance, la base de données source peut continuer à rejeter les requêtes d'écriture. Si tel est le cas, utilisez la commande suivante pour autoriser les requêtes d'écriture.

```
ALTER DATABASE db-name SET default_transaction_read_only = false;
```

Limitations liées à l'utilisation des bases de données SQL transportables Postgre

Les bases de données transportables présentent les limites suivantes :

- Réplicas en lecture – Vous ne pouvez pas utiliser des bases de données transportables sur des réplicas en lecture ou des instances parentes de réplicas en lecture.

- Types de colonne non pris en charge – Vous ne pouvez pas utiliser les types de données `reg` dans des tables de bases de données que vous souhaitez transporter avec cette méthode. Ces types dépendent de l'objet du catalogue système IDs (OIDs), qui change souvent pendant le transport.
- Espaces de tables – Tous les objets de base de données sources doivent se trouver dans l'espace de table `pg_default` par défaut .
- Compatibilité — Les instances de base de données source et de destination doivent exécuter la même version majeure de PostgreSQL.
- Extensions : l'instance de base de données source ne peut avoir que `pg_transport` installé.
- Rôles et ACLs — Les privilèges d'accès et les informations de propriété de la base de données source ne sont pas transférés vers la base de données de destination. Tous les objets de base de données sont créés par l'utilisateur de destination locale du transport et sont sa propriété.
- Transports simultanés : une seule instance de base de données peut prendre en charge jusqu'à 32 transports simultanés, y compris les importations et les exportations, si les processus de travail ont été correctement configurés.
- RDSpour les SQL instances de base de données Postgre uniquement — Les bases de données SQL transportables Postgre ne sont prises en charge que RDS pour les instances de base de données PostgreSQL. Vous ne pouvez pas l'utiliser avec des bases de données sur site ou des bases de données exécutées sur AmazonEC2.

Configuration du transport d'une base de données Postgre SQL

Avant de commencer, assurez-vous que vos SQL instances de base de données RDS pour Postgre répondent aux exigences suivantes :

- Les SQL instances RDS de base de données Postgre pour la source et la destination doivent exécuter la même version de SQL Postgre.
- La base de données de destination ne peut pas avoir de base de données portant le même nom que la base de données source que vous souhaitez transporter.
- Le compte que vous utilisez pour exécuter le transport doit avoir les privilèges `rds_superuser` sur les bases de données source et de destination.
- Le groupe de sécurité de l'instance de base de données source doit autoriser l'accès entrant depuis l'instance de base de données de destination. C'est peut-être déjà le cas si vos instances de base de données source et de destination se trouvent dans leVPC. Pour plus d’informations sur les groupes de sécurité, consultez Contrôle d'accès par groupe de sécurité.

Le transport de bases de données d'une instance de base de données source vers une instance de base de données de destination nécessite plusieurs modifications apportées au groupe de paramètres de base de données associé à chaque instance. Cela signifie que vous devez créer un groupe de paramètres de base de données personnalisé pour l'instance de base de données source et créer un groupe de paramètres de base de données personnalisé pour l'instance de base de données de destination.

> **Note**
>
> Si vos instances de base de données sont déjà configurées à l'aide de groupes de paramètres de base de données personnalisés, vous pouvez commencer par l'étape 2 de la procédure suivante.

Pour configurer les paramètres de groupe de base de données personnalisés pour le transport de bases de données

Pour les étapes suivantes, utilisez un compte doté des privilèges `rds_superuser`.

1. Si les instances de base de données source et de destination utilisent un groupe de paramètres de base de données par défaut, vous devez créer un groupe de paramètres de base de données personnalisé en utilisant la version appropriée pour vos instances. Vous pouvez ainsi modifier les valeurs de plusieurs paramètres. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).
2. Dans le groupe de paramètres de base de données personnalisé, modifiez les valeurs des paramètres suivants :

   - `shared_preload_libraries` : ajoutez `pg_transport` à la liste des bibliothèques.
   - `pg_transport.num_workers` : la valeur par défaut est 3. Augmentez ou réduisez cette valeur au besoin pour votre base de données. Pour une base de données de 200 Go, nous recommandons de ne pas dépasser 8. N'oubliez pas que si vous augmentez la valeur par défaut de ce paramètre, vous devez également augmenter la valeur de `max_worker_processes`.
   - `pg_transport.work_mem`— La valeur par défaut est de 128 Mo ou 256 Mo, selon la SQL version de Postgre. Le paramètre par défaut peut généralement rester inchangé.
   - `max_worker_processes` : la valeur de ce paramètre doit être définie à l'aide du calcul suivant :

```
(3 * pg_transport.num_workers) + 9
```

Cette valeur est nécessaire au niveau de la destination pour gérer les divers processus employés en arrière-plan impliqués dans le transport. Pour en savoir plus sur la consommation de ressources, `max_worker_processes,` consultez la SQL documentation Postgre.

Pour de plus amples informations sur les paramètres `pg_transport`, veuillez consulter Référence des paramètres des bases de données transportables .

3. Redémarrez la source RDS pour l'SQLinstance de base de données Postgre et l'instance de destination afin que les paramètres prennent effet.
4. Connectez-vous à votre instance de base de données SQL source RDS pour Postgre.

```
psql --host=source-instance.111122223333.aws-region.rds.amazonaws.com --port=5432
 --username=postgres --password
```

5. Supprimez les extensions externes du schéma public de l'instance de base de données. Seule l'extension `pg_transport` est autorisée pendant l'opération de transport réelle.
6. Installez l'extension `pg_transport` comme suit :

```
postgres=> CREATE EXTENSION pg_transport;
CREATE EXTENSION
```

7. Connectez-vous à votre instance RDS de base de données de SQL destination Postgre. Supprimez toutes les extensions externes, puis installez l'extension `pg_transport`.

```
postgres=> CREATE EXTENSION pg_transport;
CREATE EXTENSION
```

Transporter une SQL base de données Postgre vers la destination depuis la source

Une fois terminé le processus décrit dans Configuration du transport d'une base de données Postgre SQL, vous pouvez démarrer le transport. Pour cela, exécutez la fonction `transport.import_from_server` sur l'instance de base de données de destination. Dans la syntaxe suivante, vous trouverez les paramètres de la fonction.

```
SELECT transport.import_from_server(
```

```
   'source-db-instance-endpoint',
    source-db-instance-port,
   'source-db-instance-user',
   'source-user-password',
   'source-database-name',
   'destination-user-password',
   false);
```

Le valeur `false` illustrée dans l'exemple indique à la fonction qu'il ne s'agit pas d'un test. Pour tester la configuration de votre transport, vous pouvez spécifier `true` pour `dry_run` lorsque vous appelez la fonction, comme illustré ci-après :

```
postgres=> SELECT transport.import_from_server(
    'docs-lab-source-db.666666666666aws-region.rds.amazonaws.com', 5432,
    'postgres', '********', 'labdb', '******', true);
INFO:  Starting dry-run of import of database "labdb".
INFO:  Created connections to remote database        (took 0.03 seconds).
INFO:  Checked remote cluster compatibility          (took 0.05 seconds).
INFO:  Dry-run complete                          (took 0.08 seconds total).
 import_from_server
--------------------

(1 row)
```

Les INFO lignes sont sorties car le pg_`transport.timing` paramètre est défini sur sa valeur par défaut,`true`. Définissez `dry_run` à la valeur `false` lorsque vous exécutez la commande et que la base de données source est importée vers la destination, comme indiqué ci-dessous :

```
INFO:  Starting import of database "labdb".
INFO:  Created connections to remote database        (took 0.02 seconds).
INFO:  Marked remote database as read only           (took 0.13 seconds).
INFO:  Checked remote cluster compatibility          (took 0.03 seconds).
INFO:  Signaled creation of PITR blackout window     (took 2.01 seconds).
INFO:  Applied remote database schema pre-data       (took 0.50 seconds).
INFO:  Created connections to local cluster          (took 0.01 seconds).
INFO:  Locked down destination database              (took 0.00 seconds).
INFO:  Completed transfer of database files          (took 0.24 seconds).
INFO:  Completed clean up                            (took 1.02 seconds).
INFO:  Physical transport complete               (took 3.97 seconds total).
import_from_server
--------------------
(1 row)
```

Cette fonction nécessite que vous fournissiez les mots de passe utilisateur de la base de données. Nous vous recommandons donc de modifier les mots de passe des rôles utilisateur que vous avez utilisés une fois le transport terminé. Vous pouvez également utiliser des variables de SQL liaison pour créer des rôles d'utilisateur temporaires. Utilisez ces rôles temporaires pour le transport, puis supprimez-les une fois que vous n'en avez plus besoin.

Si votre transport n'est pas réussi, vous pouvez voir un message d'erreur similaire à ce qui suit :

```
pg_transport.num_workers=8 25% of files transported failed to download file data
```

Le message d'erreur « Impossible de télécharger les données du fichier » indique que le nombre de processus de travail n'est pas défini correctement pour la taille de la base de données. Vous devrez peut-être augmenter ou diminuer la valeur définie pour `pg_transport.num_workers`. Chaque échec indique le pourcentage d'achèvement, afin que vous puissiez voir l'impact de vos modifications. Par exemple, la modification du paramètre de 8 à 4 dans un cas a entraîné les résultats suivants :

```
pg_transport.num_workers=4 75% of files transported failed to download file data
```

Gardez à l'esprit que le paramètre `max_worker_processes` est également pris en compte pendant le processus de transport. Autrement dit, vous devrez peut-être modifier à la fois `pg_transport.num_workers` et `max_worker_processes` pour transporter correctement la base de données. L'exemple présenté a finalement fonctionné lorsque le `pg_transport.num_workers` a été réglé sur 2 :

```
pg_transport.num_workers=2 100% of files transported
```

Pour plus d'informations sur la fonction `transport.import_from_server` et ses paramètres, veuillez consulter Référence des fonctions des base de données transportables.

Référence des fonctions des base de données transportables

La `transport.import_from_server` fonction transporte une SQL base de données Postgre en l'important d'une instance de base de données source vers une instance de base de données de destination. Elle effectue cette opération en utilisant un mécanisme de transport physique de connexion de base de données.

Avant de démarrer le transport, cette fonction vérifie que les instances de base de données source et de destination sont de la même version et sont compatibles avec la migration. Elle confirme

également que l'instance de base de données de destination dispose de suffisamment d'espace pour la source.

Syntaxe

```
transport.import_from_server(
   host text,
   port int,
   username text,
   password text,
   database text,
   local_password text,
   dry_run bool
)
```

Valeur renvoyée

Aucun.

Paramètres

Le tableau ci-dessous contient les descriptions des paramètres de la fonction `transport.import_from_server`.

Paramètre	Description
`host`	Point de terminaison de l'instance de base de données source.
`port`	Entier représentant le port de l'instance de base de données source. Les SQL instances de base de données Postgre utilisent souvent le port 5432.
`username`	Utilisateur de l'instance de base de données source. Cet utilisateur doit être membre du rôle `rds_superuser` .
`password`	Mot de passe utilisateur de l'instance de base de données source.
`database`	Nom de la base de données à transporter à partir de l'instance de base de données source.
`local_password`	Mot de passe local de l'utilisateur actuel pour l'instance de base de données de destination. Cet utilisateur doit être membre du rôle `rds_superuser` .

Paramètre	Description
`dry_run`	Valeur booléenne facultative spécifiant si un essai est nécessaire. La valeur par défaut est `false`, ce qui signifie que le transport est effectué. Pour vérifier la compatibilité entre les instances de base de données source et de destination sans effectuer le transport réel, définissez `dry_run` sur `true`.

Exemple

Pour obtenir un exemple, consultez Transporter une SQL base de données Postgre vers la destination depuis la source.

Référence des paramètres des bases de données transportables

Plusieurs paramètres contrôlent le comportement de l'extension `pg_transport`. Vous trouverez ci-dessous la description de ces paramètres.

`pg_transport.num_workers`

Le nombre d'unités de travail à utiliser pour le processus de transport. La valeur par défaut est 3. Les valeurs valides vont de 1 à 32. Même les transports de base de données les plus volumineux nécessitent généralement moins de 8 unités de travail. La valeur de ce paramètre sur l'instance de base de données de destination est utilisée par la destination et la source pendant le transport.

`pg_transport.timing`

Indique s'il faut signaler les informations de synchronisation pendant le transport. La valeur par défaut est `true`, ce qui signifie que les informations de synchronisation sont signalées. Nous vous recommandons de laisser ce paramètre défini sur `true` pour que vous puissiez suivre les progrès réalisés. Pour un exemple de sortie, veuillez consulter Transporter une SQL base de données Postgre vers la destination depuis la source.

`pg_transport.work_mem`

Quantité de mémoire maximale à allouer à chaque unité de travail. La valeur par défaut est de 131072 kilo-octets (Ko) ou 262144 Ko (256 Mo), selon la version de Postgre. SQL La valeur minimale est de 64 méga-octets (65 536 Ko). Les valeurs valides sont exprimées en kilo-octets (KBs) sous forme d'unités binaires en base 2, où 1 Ko = 1024 octets.

Le transport peut utiliser moins de mémoire que spécifié dans ce paramètre. Même les transports de base de données volumineux nécessitent généralement moins de 256 Mo (262 144 Ko) de mémoire par unité de travail.

Exportation de données à partir d'une instance de base de données RDS for PostgreSQL vers Amazon S3

Vous pouvez interroger des données à partir d'une instance de base de données RDS for PostgreSQL et les exporter directement dans des fichiers stockés dans un compartiment Amazon S3. Pour ce faire, vous devez d'abord installer l'extension `aws_s3` RDS for PostgreSQL. Cette extension vous fournit les fonctions que vous utilisez pour exporter les résultats des requêtes vers Amazon S3. Vous trouverez ci-dessous comment installer l'extension et comment exporter des données vers Amazon S3.

> **Note**
>
> L'exportation intercompte vers Amazon S3 n'est pas prise en charge.

Toutes les versions actuellement disponibles de RDS for PostgreSQL prennent en charge l'exportation de données vers Amazon Simple Storage Service. Pour des informations détaillées sur les versions, consultez Amazon RDS for PostgreSQL updates (Mises à jour d'Amazon RDS for PostgreSQL) dans les notes de mise à jour d'Amazon RDS for PostgreSQL.

Si vous n'avez pas de compartiment configuré pour votre exportation, consultez les rubriques suivantes du Guide de l'utilisateur d'Amazon Simple Storage Service.

- Configuration d'Amazon S3
- Création d'un bucket

Par défaut, les données exportées de RDS pour PostgreSQL vers Amazon S3 utilisent le chiffrement côté serveur avec un. Clé gérée par AWS Si vous utilisez le chiffrement par compartiment, le compartiment Amazon S3 doit être chiffré avec une clé AWS Key Management Service (AWS KMS) (SSE-KMS). Actuellement, les compartiments chiffrés avec des clés gérées par Amazon S3 (SSE-S3) ne sont pas pris en charge.

> **Note**
>
> Vous pouvez enregistrer les données des instantanés de base de données sur Amazon S3 à l'aide de l'API AWS Management Console AWS CLI, ou Amazon RDS. Pour de plus amples

informations, veuillez consulter Exportation de données instantanées de base de données vers Amazon S3 pour Amazon RDS.

Rubriques

- Installation de l'extension aws_s3
- Présentation de l'exportation de données vers Amazon S3
- Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation
- Configuration de l'accès à un compartiment Amazon S3
- Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3
- Références de fonctions
- Résolution des problèmes d'accès à Amazon S3

Installation de l'extension aws_s3

Avant de pouvoir utiliser Amazon Simple Storage Service avec votre instance de base de données RDS for PostgreSQL, vous devez installer l'extension `aws_s3`. Cette extension fournit des fonctions pour exporter des données depuis une instance de base de données RDS for PostgreSQL vers un compartiment Amazon S3. Il fournit également des fonctions pour importer des données depuis un compartiment Amazon S3. Pour de plus amples informations, veuillez consulter Importation de données Amazon S3 dans une instance de base de données RDS for PostgreSQL d'un . L'extension `aws_s3` dépend de certaines des fonctions d'aide de l'extension `aws_commons`, qui est installée automatiquement lorsque cela est nécessaire.

Pour installer l'extension **`aws_s3`**

1. Utilisez psql (ou pgAdmin) pour vous connecter à l'instance de base de données RDS for PostgreSQL en tant qu'utilisateur disposant de privilèges `rds_superuser`. Si vous avez conservé le nom par défaut pendant le processus d'installation, vous vous connectez en tant que `postgres`.

   ```
   psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password
   ```

2. Pour installer l'extension, exécutez la commande suivante.

```
postgres=> CREATE EXTENSION aws_s3 CASCADE;
NOTICE: installing required extension "aws_commons"
CREATE EXTENSION
```

3. Pour vérifier que l'extension est installée, vous pouvez utiliser la métacommande psql `\dx`.

```
postgres=> \dx
       List of installed extensions
    Name     | Version |   Schema   |                 Description
-------------+---------+------------+---------------------------------------------
 aws_commons | 1.2     | public     | Common data types across AWS services
 aws_s3      | 1.1     | public     | AWS S3 extension for importing data from S3
 plpgsql     | 1.0     | pg_catalog | PL/pgSQL procedural language
(3 rows)
```

Les fonctions d'importation de données depuis Amazon S3 et d'exportation de données vers Amazon S3 sont désormais disponibles.

Assurez-vous que votre version de RDS for PostgreSQL prend en charge les exportations vers Amazon S3

Vous pouvez vérifier que votre version de RDS for PostgreSQL prend en charge l'exportation vers Amazon S3 en utilisant la commande `describe-db-engine-versions`. L'exemple suivant vérifie la prise en charge de la version 10.14.

```
aws rds describe-db-engine-versions --region us-east-1
--engine postgres --engine-version 10.14 | grep s3Export
```

Si la sortie inclut la chaîne `"s3Export"`, le moteur prend en charge les exportations Amazon S3. Sinon, le moteur ne les prend pas en charge.

Présentation de l'exportation de données vers Amazon S3

Pour exporter des données stockées dans un RDS for PostgreSQL vers un compartiment Amazon S3, procédez comme suit.

Pour exporter des données RDS for PostgreSQL vers S3

1. Identifiez un chemin d'accès de fichier Amazon S3 à utiliser pour exporter des données. Pour de plus amples informations sur ce processus, veuillez consulter Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation.
2. Fournissez une autorisation d'accès au compartiment Amazon S3.

   Pour exporter des données vers un fichier Amazon S3, vous devez accorder à l'instance de base de données RDS for PostgreSQL l'autorisation d'accéder au compartiment Amazon S3 que l'exportation utilisera pour le stockage. Cette opération comprend les étapes suivantes :

   1. Créez une politique IAM donnant accès à un compartiment Amazon S3 vers lequel vous souhaitez exporter.
   2. Créez un rôle IAM.
   3. Attachez la politique que vous avez créée au rôle que vous avez créé.
   4. Ajoutez ce rôle IAM à votre instance de base de données.

   Pour de plus amples informations sur ce processus, veuillez consulter Configuration de l'accès à un compartiment Amazon S3.
3. Identifiez une requête de base de données pour obtenir les données. Exportez les données de requête en appelant la fonction `aws_s3.query_export_to_s3`.

   Après avoir terminé les tâches de préparation précédentes, utilisez la fonction aws_s3.query_export_to_s3 pour exporter les résultats de requête vers Amazon S3. Pour de plus amples informations sur ce processus, veuillez consulter Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3.

Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation

Spécifiez les informations suivantes pour identifier l'emplacement dans Amazon S3 vers lequel vous souhaitez exporter des données :

- Nom du compartiment – Un compartiment est un conteneur d'objets ou de fichiers Amazon S3.

Pour plus d'informations sur le stockage de données avec Amazon S3, consultez les sections Création d'un compartiment et Utilisation d'objets dans le guide de l'utilisateur d'Amazon Simple Storage Service.

- Chemin d'accès au fichier – Le chemin d'accès au fichier identifie l'emplacement de stockage de l'exportation dans le compartiment Amazon S3. Le chemin d'accès au fichier se compose des éléments suivants :
  - Préfixe de chemin facultatif qui identifie un chemin d'accès à un dossier virtuel.
  - Préfixe de fichier qui identifie un ou plusieurs fichiers à stocker. Les exportations les plus volumineuses sont stockées dans plusieurs fichiers, chacun ayant une taille maximale d'environ 6 Go. Les noms de fichiers supplémentaires ont le même préfixe de fichier mais en ajoutant `_part`*XX*. *XX* représente 2, puis 3, et ainsi de suite.

  Par exemple, un chemin d'accès de fichier avec un dossier `exports` et un préfixe de fichier `query-1-export` sera représenté par `/exports/query-1-export`.
- AWS Région (facultatif) : AWS région dans laquelle se trouve le compartiment Amazon S3. Si vous ne spécifiez aucune valeur de AWS région, Amazon RDS enregistre vos fichiers dans Amazon S3 dans la même AWS région que l'instance de base de données du cluster exportant.

  > **Note**
  >
  > Actuellement, la AWS région doit être identique à la région de l'instance de base de données du de bases de données exportatrice.

  Pour obtenir la liste des noms de AWS régions et des valeurs associées, consultezRégions, zones de disponibilité et zones locales.

Pour conserver les informations de fichier Amazon S3 sur l'emplacement de stockage de l'exportation, vous pouvez utiliser la fonction aws_commons.create_s3_uri pour créer une structure composite `aws_commons._s3_uri_1` comme suit.

```
psql=> SELECT aws_commons.create_s3_uri(
   'amzn-s3-demo-bucket',
   'sample-filepath',
   'us-west-2'
) AS s3_uri_1 \gset
```

Vous fournissez ultérieurement cette valeur `s3_uri_1` en tant que paramètre dans l'appel à la fonction aws_s3.query_export_to_s3. Pour obtenir des exemples, consultez Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3.

Configuration de l'accès à un compartiment Amazon S3

Pour exporter des données vers Amazon S3, accordez à votre instance l'autorisation d'accéder au compartiment Amazon S3 dans lequel les fichiers doivent être stockés.

Pour cela, procédez comme suit :

Pour donner à une instance de base de données PostgreSQL l'accès à Amazon S3 via un rôle IAM

1. Créez une politique IAM.

   Cette stratégie fournit le compartiment et les autorisations d'objet permettant à votre instance de base de données PostgreSQL d'accéder à Amazon S3.

   Dans le cadre de la création de cette politique, procédez comme suit :

   a. Incluez dans la stratégie les actions obligatoires suivantes pour permettre le transfert de fichiers de votre instance de base de données PostgreSQL vers un compartiment Amazon S3 :

      - `s3:PutObject`
      - `s3:AbortMultipartUpload`

   b. Incluez l'Amazon Resource Name (ARN) qui identifie le compartiment Amazon S3 et les objets du compartiment. Le format ARN pour l'accès à Amazon S3 est le suivant : `arn:aws:s3:::amzn-s3-demo-bucket/*`

   Pour plus d'informations sur la création d'une politique IAM pour Amazon RDS for PostgreSQL, consultez Création et utilisation d'une politique IAM pour l'accès à une base de données IAM. Consultez également Didacticiel : création et attachement de votre première politique gérée par le client dans le Guide de l'utilisateur IAM.

   La AWS CLI commande suivante crée une politique IAM nommée `rds-s3-export-policy` avec ces options. Elle accorde un accès à un compartiment nommé *`amzn-s3-demo-bucket`*.

> **Warning**
>
> Nous vous recommandons de configurer votre base de données dans un VPC privé dont les politiques de point de terminaison sont configurées pour accéder à des compartiments spécifiques. Pour de plus amples informations, veuillez consulter Utilisation des stratégies de point de terminaison pour Amazon S3 dans le Amazon VPC Guide de l'utilisateur.
>
> Nous vous recommandons vivement de ne pas créer de politique avec accès à toutes les ressources. Cet accès peut constituer une menace pour la sécurité des données. Si vous créez une stratégie qui accorde à `S3:PutObject` un accès à toutes les ressources à l'aide de `"Resource":"*"`, un utilisateur disposant de privilèges d'exportation peut exporter des données vers tous les compartiments de votre compte. En outre, l'utilisateur peut exporter des données vers n'importe quel compartiment accessible publiquement en écriture dans votre région AWS .

Après avoir créé la politique, notez son ARN (Amazon Resource Name). Vous en aurez besoin par la suite pour attacher la politique à un rôle IAM.

```
aws iam create-policy  --policy-name rds-s3-export-policy  --policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Sid": "s3export",
         "Action": [
           "s3:PutObject*",
           "s3:ListBucket",
           "s3:GetObject*",
           "s3:DeleteObject*",
           "s3:GetBucketLocation",
           "s3:AbortMultipartUpload"
         ],
         "Effect": "Allow",
         "Resource": [
           "arn:aws:s3:::amzn-s3-demo-bucket/*"
         ]
       }
     ]
   }'
```

2. Créez un rôle IAM.

   L'objectif est ici de permettre à Amazon RDS d'endosser ce rôle IAM en votre nom pour accéder à vos compartiments Amazon S3. Pour plus d'informations, consultez Création d'un rôle pour déléguer des autorisations à un utilisateur IAM dans le Guide de l'utilisateur IAM.

   Nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans des politiques basées sur les ressources pour limiter les autorisations du service à une ressource spécifique. C'est le moyen le plus efficace de se protéger contre le problème du député confus.

   Si vous utilisez les deux clés de contexte de condition globale et que la valeur de `aws:SourceArn` contient l'ID de compte, la valeur de `aws:SourceAccount` et le compte indiqué dans la valeur de `aws:SourceArn` doivent utiliser le même ID de compte lorsqu'il est utilisé dans la même déclaration de politique.

   - Utilisez `aws:SourceArn` si vous souhaitez un accès interservices pour une seule ressource.
   - Utilisez `aws:SourceAccount` si vous souhaitez autoriser une ressource de ce compte à être associée à l'utilisation interservices.

   Dans la politique, veillez à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l'ARN complet de la ressource. L'exemple suivant montre comment procéder à l'aide de la AWS CLI commande pour créer un rôle nommé`rds-s3-export-role`.

   Example

   Dans Linux, macOS, ou Unix:

```
aws iam create-role  \
    --role-name rds-s3-export-role  \
    --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
```

```
            "StringEquals": {
               "aws:SourceAccount": "111122223333",
               "aws:SourceArn": "arn:aws:rds:us-east-1:111122223333:db:dbname"
               }
            }
      }
    ]
  }'
```

Dans Windows:

```
aws iam create-role  ^
    --role-name rds-s3-export-role  ^
    --assume-role-policy-document '{
     "Version": "2012-10-17",
     "Statement": [
       {
         "Effect": "Allow",
         "Principal": {
            "Service": "rds.amazonaws.com"
          },
         "Action": "sts:AssumeRole",
         "Condition": {
             "StringEquals": {
                "aws:SourceAccount": "111122223333",
                "aws:SourceArn": "arn:aws:rds:us-east-1:111122223333:db:dbname"
                }
             }
       }
     ]
  }'
```

3. Attachez la politique IAM que vous avez créée au rôle IAM que vous venez de créer.

   La AWS CLI commande suivante associe la politique créée précédemment au rôle nommé `rds-s3-export-role`. Remplacer *`your-policy-arn`* par l'ARN de stratégie que vous avez noté lors d'une étape précédente.

```
aws iam attach-role-policy  --policy-arn your-policy-arn  --role-name rds-s3-
export-role
```

4. Ajoutez le rôle IAM à l'instance de base de données. Pour ce faire, utilisez le AWS Management Console ou AWS CLI, comme décrit ci-dessous.

console

Pour ajouter un rôle IAM à l'instance de base de données PostgreSQL à l'aide de la console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Choisissez le nom de l'instance de base de données PostgreSQL pour afficher ses détails.
3. Dans l'onglet Connectivité & sécurité de la section Gérer les rôles IAM, choisissez le rôle à ajouter sous Ajouter des rôles IAM à cette instance.
4. Sous Fonctionnalité, choisissez s3Export.
5. Choisissez Ajouter un rôle.

AWS CLI

Pour ajouter un rôle IAM à une instance de base de données PostgreSQL à l'aide de la CLI

- Utilisez la commande suivante pour ajouter le rôle à l'instance de base de données PostgreSQL nommée `my-db-instance`. Remplacez *`your-role-arn`* par l'ARN de rôle que vous avez noté lors d'une étape précédente. Utilisez `s3Export` comme valeur de l'option `--feature-name`.

  Example

  Dans Linux, macOS, ou Unix:

```
aws rds add-role-to-db-instance \
   --db-instance-identifier my-db-instance \
   --feature-name s3Export \
   --role-arn your-role-arn   \
   --region your-region
```

  Dans Windows:

```
aws rds add-role-to-db-instance ^
   --db-instance-identifier my-db-instance ^
   --feature-name s3Export ^
```

```
    --role-arn your-role-arn ^
    --region your-region
```

Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3

Exportez vos données PostgreSQL vers Amazon S3 en appelant la fonction aws_s3.query_export_to_s3.

Rubriques

- Prérequis
- Appel de aws_s3.query_export_to_s3
- Exportation vers un fichier CSV qui utilise un délimiteur personnalisé
- Exportation vers un fichier binaire avec encodage

Prérequis

Avant d'utiliser la fonction `aws_s3.query_export_to_s3`, assurez-vous de remplir les conditions préalables suivantes :

- Installez les extensions PostgreSQL requises comme décrit dans Présentation de l'exportation de données vers Amazon S3.
- Déterminez vers quel emplacement Amazon S3 exporter vos données comme décrit dans Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation.
- Assurez-vous que l'instance de base de données dispose d'un accès à Amazon S3 comme décrit dans Configuration de l'accès à un compartiment Amazon S3.

Les exemples suivants utilisent une table de base de données appelée `sample_table`. Ces exemples exportent les données dans un compartiment appelé *amzn-s3-demo-bucket*. Les exemples de table et de données sont créés avec les instructions SQL suivantes dans psql.

```
psql=> CREATE TABLE sample_table (bid bigint PRIMARY KEY, name varchar(80));
psql=> INSERT INTO sample_table (bid,name) VALUES (1, 'Monday'), (2,'Tuesday'), (3,
 'Wednesday');
```

Appel de aws_s3.query_export_to_s3

Ce qui suit montre les techniques de base permettant d'appeler la fonction aws_s3.query_export_to_s3.

Ces exemples utilisent la variable `s3_uri_1` pour identifier une structure contenant les informations identifiant le fichier Amazon S3. Utilisez la fonction aws_commons.create_s3_uri pour créer la structure.

```
psql=> SELECT aws_commons.create_s3_uri(
   'amzn-s3-demo-bucket',
   'sample-filepath',
   'us-west-2'
) AS s3_uri_1 \gset
```

Bien que les paramètres varient pour les deux appels de fonction `aws_s3.query_export_to_s3` suivants, les résultats sont les mêmes pour ces exemples. Toutes les lignes de la table `sample_table` sont exportées dans un compartiment appelé *amzn-s3-demo-bucket*.

```
psql=> SELECT * FROM aws_s3.query_export_to_s3('SELECT * FROM
 sample_table', :'s3_uri_1');

psql=> SELECT * FROM aws_s3.query_export_to_s3('SELECT * FROM
 sample_table', :'s3_uri_1', options :='format text');
```

Les paramètres sont décrits comme suit :

- `'SELECT * FROM sample_table'` – Le premier paramètre est une chaîne de texte obligatoire contenant une requête SQL. Le moteur PostgreSQL exécute cette requête. Les résultats de la requête sont copiés dans le compartiment S3 identifié dans d'autres paramètres.
- `:'s3_uri_1'` – Ce paramètre est une structure qui identifie le fichier Amazon S3. Cet exemple utilise une variable pour identifier la structure créée précédemment. Vous pouvez plutôt créer la structure en incluant l'appel de fonction `aws_commons.create_s3_uri` en ligne dans l'appel de fonction `aws_s3.query_export_to_s3` comme suit.

  ```
  SELECT * from aws_s3.query_export_to_s3('select * from sample_table',
     aws_commons.create_s3_uri('amzn-s3-demo-bucket', 'sample-filepath', 'us-west-2')
  );
  ```

- `options :='format text'` – Le paramètre `options` est une chaîne de texte facultative contenant des arguments `COPY` PostgreSQL. Le processus de copie utilise les arguments et le format de la commande PostgreSQL COPY.

Si le fichier spécifié n'existe pas dans le compartiment Amazon S3, il est créé. Si le fichier existe déjà, il est remplacé. La syntaxe d'accès aux données exportées dans Amazon S3 est la suivante.

```
s3-region://bucket-name[/path-prefix]/file-prefix
```

Les exportations les plus volumineuses sont stockées dans plusieurs fichiers, chacun ayant une taille maximale d'environ 6 Go. Les noms de fichiers supplémentaires ont le même préfixe de fichier mais en ajoutant `_part`*XX*. *XX* représente 2, puis 3, et ainsi de suite. Par exemple, supposons que vous spécifiiez le chemin d'accès où vous stockez les fichiers de données comme suit.

```
s3-us-west-2://amzn-s3-demo-bucket/my-prefix
```

Si l'exportation doit créer trois fichiers de données, le compartiment Amazon S3 contient les fichiers de données suivants.

```
s3-us-west-2://amzn-s3-demo-bucket/my-prefix
s3-us-west-2://amzn-s3-demo-bucket/my-prefix_part2
s3-us-west-2://amzn-s3-demo-bucket/my-prefix_part3
```

Pour obtenir la référence complète de cette fonction et les moyens supplémentaires de l'appeler, veuillez consulter aws_s3.query_export_to_s3. Pour plus d'informations sur l'accès aux fichiers dans Amazon S3, consultez Afficher un objet dans le Guide de l'utilisateur Amazon Simple Storage Service.

Exportation vers un fichier CSV qui utilise un délimiteur personnalisé

L'exemple suivant montre comment appeler la fonction aws_s3.query_export_to_s3 pour exporter des données vers un fichier qui utilise un délimiteur personnalisé. L'exemple utilise les arguments de la commande PostgreSQL COPY pour spécifier le format CSV (valeur séparée par des virgules) et un délimiteur deux-points (:).

```
SELECT * from aws_s3.query_export_to_s3('select * from basic_test', :'s3_uri_1',
 options :='format csv, delimiter $$:$$');
```

Exportation vers un fichier binaire avec encodage

L'exemple suivant montre comment appeler la fonction aws_s3.query_export_to_s3 pour exporter des données vers un fichier binaire ayant un encodage Windows-1253.

```
SELECT * from aws_s3.query_export_to_s3('select * from basic_test', :'s3_uri_1',
 options :='format binary, encoding WIN1253');
```

Références de fonctions

Fonctions

- aws_s3.query_export_to_s3
- aws_commons.create_s3_uri

aws_s3.query_export_to_s3

Exporte un résultat de requête PostgreSQL vers un compartiment Amazon S3. L'extension `aws_s3` fournit la fonction `aws_s3.query_export_to_s3`.

Les deux paramètres requis sont `query` et `s3_info`. Ils définissent la requête à exporter et identifient le compartiment Amazon S3 vers lequel effectuer l'exportation. Un paramètre facultatif appelé `options` permet de définir différents paramètres d'exportation. Pour obtenir des exemples d'utilisation de la fonction `aws_s3.query_export_to_s3`, veuillez consulter Exportation de données de requête à l'aide de la fonction aws_s3.query_export_to_s3.

Syntaxe

```
aws_s3.query_export_to_s3(
    query text,
    s3_info aws_commons._s3_uri_1,
    options text,
    kms_key text
)
```

Paramètres d'entrée

query

Chaîne de texte obligatoire contenant une requête SQL exécutée par le moteur PostgreSQL. Les résultats de cette requête sont copiés dans un compartiment S3 identifié dans le paramètre `s3_info`.

s3_info

Type composite `aws_commons._s3_uri_1` contenant les informations suivantes sur l'objet S3 :

- `bucket` – Nom du compartiment Amazon S3 contenant le fichier.
- `file_path` – Nom du fichier Amazon S3 et chemin d'accès à celui-ci.
- `region`— La AWS région dans laquelle se trouve le compartiment. Pour obtenir la liste des noms de AWS régions et des valeurs associées, consultezRégions, zones de disponibilité et zones locales.

  Actuellement, cette valeur doit être la même AWS région que celle de l'instance de base de données du de bases de données exportatrice. La valeur par défaut est la AWS région de l'instance de base de données du de bases de données exportatrice.

Pour créer une structure composite `aws_commons._s3_uri_1`, veuillez consulter aws_commons.create_s3_uri fonction.

options

Chaîne de texte facultative contenant les arguments de la commande `COPY` de PostgreSQL. Ces arguments spécifient la façon dont les données doivent être copiées lors de l'exportation. Pour de plus amples informations, veuillez consulter la documentation sur la commande COPY de PostgreSQL.

texte kms_key

Chaîne de texte facultative contenant la clé KMS gérée par le client du compartiment S3 vers lequel exporter les données.

Autres paramètres d'entrée

Pour faciliter le test, vous pouvez utiliser un ensemble étendu de paramètres au lieu du paramètre `s3_info`. Plusieurs variations de syntaxe supplémentaires pour la fonction `aws_s3.query_export_to_s3` sont fournies ci-dessous.

Au lieu d'utiliser le paramètre `s3_info` pour identifier un fichier Amazon S3, utilisez la combinaison des paramètres `bucket`, `file_path` et `region`.

```
aws_s3.query_export_to_s3(
    query text,
    bucket text,
    file_path text,
    region text,
    options text,
    kms_key text
)
```

query

Chaîne de texte obligatoire contenant une requête SQL exécutée par le moteur PostgreSQL. Les résultats de cette requête sont copiés dans un compartiment S3 identifié dans le paramètre `s3_info`.

bucket

Chaîne de texte obligatoire comportant le nom du compartiment Amazon S3 qui contient le fichier.

file_path

Chaîne de texte obligatoire contenant le nom du fichier Amazon S3, avec le chemin d'accès à celui-ci.

region

Chaîne de texte facultative contenant la AWS région dans laquelle se trouve le compartiment. Pour obtenir la liste des noms de AWS régions et des valeurs associées, consultezRégions, zones de disponibilité et zones locales.

Actuellement, cette valeur doit être la même AWS région que celle de l'instance de base de données du de bases de données exportatrice. La valeur par défaut est la AWS région de l'instance de base de données du de bases de données exportatrice.

options

Chaîne de texte facultative contenant les arguments de la commande `COPY` de PostgreSQL. Ces arguments spécifient la façon dont les données doivent être copiées lors de l'exportation. Pour de plus amples informations, veuillez consulter la documentation sur la commande COPY de PostgreSQL.

texte kms_key

Chaîne de texte facultative contenant la clé KMS gérée par le client du compartiment S3 vers lequel exporter les données.

Paramètres de sortie

```
aws_s3.query_export_to_s3(
    OUT rows_uploaded bigint,
    OUT files_uploaded bigint,
    OUT bytes_uploaded bigint
)
```

rows_uploaded

Nombre de lignes de table qui ont été téléchargées avec succès vers Amazon S3 pour la requête donnée.

files_uploaded

Nombre de fichiers téléchargés vers Amazon S3. Les fichiers sont créés avec des tailles d'environ 6 Go. Chaque fichier supplémentaire créé voit l'élément `_part`*XX* ajouté à son nom. *XX* représente 2, puis 3, et ainsi de suite.

bytes_uploaded

Nombre total d'octets téléchargés vers Amazon S3.

Exemples

```
psql=> SELECT * from aws_s3.query_export_to_s3('select * from sample_table', 'amzn-s3-demo-bucket', 'sample-filepath');
psql=> SELECT * from aws_s3.query_export_to_s3('select * from sample_table', 'amzn-s3-demo-bucket', 'sample-filepath','us-west-2');
psql=> SELECT * from aws_s3.query_export_to_s3('select * from sample_table', 'amzn-s3-demo-bucket', 'sample-filepath','us-west-2','format text');
```

aws_commons.create_s3_uri

Crée une structure `aws_commons._s3_uri_1` pour contenir les informations relatives au fichier Amazon S3. Vous utilisez les résultats de la fonction `aws_commons.create_s3_uri` dans le

paramètre `s3_info` de la fonction aws_s3.query_export_to_s3. Pour obtenir un exemple d'utilisation de la fonction `aws_commons.create_s3_uri`, veuillez consulter Spécification du chemin d'accès au fichier Amazon S3 vers lequel effectuer l'exportation.

Syntaxe

```
aws_commons.create_s3_uri(
   bucket text,
   file_path text,
   region text
)
```

Paramètres d'entrée

bucket

Chaîne de texte obligatoire contenant le nom du compartiment Amazon S3 pour le fichier.

file_path

Chaîne de texte obligatoire contenant le nom du fichier Amazon S3, avec le chemin d'accès à celui-ci.

region

Chaîne de texte obligatoire contenant la AWS région dans laquelle se trouve le fichier. Pour obtenir la liste des noms de AWS régions et des valeurs associées, consultezRégions, zones de disponibilité et zones locales.

Résolution des problèmes d'accès à Amazon S3

Si vous rencontrez des problèmes de connexion lorsque vous essayez d'exporter des données vers Amazon S3, confirmez d'abord que les règles d'accès sortant du groupe de sécurité VPC associé à votre instance de base de données permettent la connectivité réseau. Plus précisément, le groupe de sécurité doit disposer d'une règle permettant à l'instance de base de données d'envoyer du trafic TCP vers le port 443 et vers n'importe quelle IPv4 adresse (0.0.0.0/0). Pour de plus amples informations, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

Voir également les recommandations suivantes :

- Résolution des problèmes liés à Identity and Access Amazon RDS
- Dépannage d'Amazon S3 dans le Guide de l'utilisateur Amazon Simple Storage Service.
- Dépannage d'Amazon S3 et IAM dans le Guide de l'utilisateur IAM

Invocation d'une AWS Lambda fonction depuis une instance de base de données

AWS Lambda est un service de calcul piloté par les événements qui vous permet d'exécuter du code sans provisionner ni gérer de serveurs. Il peut être utilisé avec de nombreux AWS services, notamment RDS pour PostgreSQL. Par exemple, vous pouvez utiliser des fonctions Lambda pour traiter les notifications d'événements à partir d'une base de données ou pour charger des données à partir de fichiers chaque fois qu'un nouveau fichier est chargé sur Amazon S3. Pour en savoir plus sur Lambda, consultez Qu'est-ce que c'est ? AWS Lambda dans le Guide AWS Lambda du développeur.

> **Note**
>
> L'appel d'une AWS Lambda fonction est pris en charge dans les versions RDS pour PostgreSQL suivantes :
>
> - Toutes les versions de PostgreSQL 17
> - Toutes les versions de PostgreSQL 16
> - Toutes les versions de PostgreSQL 15
> - PostgreSQL 14.1 et versions mineures ultérieures
> - PostgreSQL 13.2 et versions mineures ultérieures
> - PostgreSQL 12.6 et versions mineures ultérieures

Vous trouverez ci-après des résumés des étapes nécessaires.

Pour plus d'informations sur les fonctions Lambda, veuillez consulter Mise en route avec Lambda et Principes de base d'AWS Lambda dans le Guide du développeur AWS Lambda .

Rubriques

- Étape 1 : configurer votre instance de base de données pour les connexions sortantes vers AWS Lambda
- Étape 2 : configurer IAM pour votre instance de base de données et AWS Lambda
- Étape 3 : installer l'extension aws_lambda pour une instance de base de données RDS for PostgreSQL

- Étape 4 : utiliser les fonctions d'assistance Lambda avec votre instance de base de données RDS for PostgreSQL (Facultatif)
- Étape 5 : appeler une fonction Lambda à partir de votre instance de base de données RDS for PostgreSQL.
- Étape 6 : accorder aux autres utilisateurs l'autorisation d'appeler les fonctions Lambda
- Exemples : Invocation de fonctions Lambda depuis votre RDScluster de SQL
- Messages d'erreur de fonction Lambda
- AWS Lambdaréférence de fonction et de paramètre

Étape 1 : configurer votre instance de base de données pour les connexions sortantes vers AWS Lambda

Les fonctions Lambda s'exécutent toujours au sein d'un Amazon VPC appartenant au service. AWS Lambda Lambda applique des règles d'accès réseau et de sécurité à ce VPC, le maintient et le surveille automatiquement. Votre instance de base de données RDS for PostgreSQL envoie du trafic réseau vers le VPC du service Lambda. La façon dont vous configurez cela dépend de si votre instance de base de données est publique ou privée.

- instance de base de données publique RDS pour PostgreSQL — L'instance de base de données données est publique si elle se trouve dans un sous-réseau public de votre VPC et si la propriété « » de l'instance est. PubliclyAccessible `true` Pour trouver la valeur de cette propriété, vous pouvez utiliser la describe-db-instances AWS CLI commande. Vous pouvez également utiliser la AWS Management Console afin d'ouvrir l'onglet Connectivity & security (Connectivité et sécurité) et vérifier que l'option Publicly accessible (Accessible publiquement) est définie sur Yes (Oui). Pour vérifier que l'instance se trouve dans le sous-réseau public de votre VPC, vous pouvez utiliser la AWS Management Console ou AWS CLI.

  Pour configurer l'accès à Lambda, vous utilisez le AWS Management Console ou AWS CLI pour créer une règle sortante sur le groupe de sécurité de votre VPC. La règle de sortie indique que TCP peut utiliser le port 443 pour envoyer des paquets à n'importe quelle IPv4 adresse (0.0.0.0/0).
- Cluster de RDS privé pour instance de base de données PostgreSQL — Dans ce cas, la propriété PubliclyAccessible `false` « » de l'instance est ou se trouve dans un sous-réseau privé. Pour permettre à l'instance de fonctionner avec Lambda, vous pouvez utiliser une passerelle traduction d'adresses réseau (NAT). Pour plus d'informations, veuillez consulter Passerelles NAT. Vous pouvez également configurer votre VPC avec un point de terminaison VPC pour

Lambda. Pour de plus amples informations, consultez Points de terminaison VPC dans le Guide de l'utilisateur Amazon VPC. Le point de terminaison répond aux appels faits par votre instance de base de données RDS for PostgreSQL à vos fonctions Lambda. Le point de terminaison d'un VPC utilise sa propre résolution DNS privée. RDS for PostgreSQL ne peut pas utiliser le point de terminaison d'un VPC Lambda tant que vous n'avez pas changé la valeur du paramètre `rds.custom_dns_resolution` de sa valeur par défaut de 0 (non activée) à 1. Pour ce faire :

- Créez un groupe de paramètres DB personnalisé.
- Change la valeur du paramètre `rds.custom_dns_resolution` de sa valeur par défaut de `0` à 1.
- Modifiez votre instance de base de données pour utiliser votre groupe de paramètres de base de données personnalisé.
- Redémarrez l'instance pour que le paramètre modifié prenne effet.

Votre VPC peut désormais interagir avec le AWS Lambda VPC au niveau du réseau. Ensuite, vous configurez les autorisations à l'aide d'IAM.

Étape 2 : configurer IAM pour votre instance de base de données et AWS Lambda

L'appel de fonctions Lambda depuis votre instance de base de données RDS for PostgreSQL requiert certains privilèges. Pour configurer les privilèges requis, nous vous recommandons de créer une politique IAM qui permet d'appeler des fonctions Lambda, d'attribuer cette politique à un rôle, puis d'appliquer le rôle à votre instance de base de données. Cette approche accorde à l'instance de base de données des privilèges pour appeler la fonction Lambda spécifiée en votre nom. Les étapes suivantes expliquent comment procéder à l'aide de l' AWS CLI.

Pour configurer les autorisations IAM pour l'utilisation de votre instance Amazon RDSavec Lambda

1. Utilisez la AWS CLI commande create-policy pour créer une politique IAM qui permet à votre instance de base de données Aurora d'appeler la fonction Lambda spécifiée. (L'ID d'instruction (Sid) est une description facultative pour votre instruction de politique et n'a aucun effet sur l'utilisation.) Cette politique accorde à votre instance de base de données les autorisations minimales requises pour appeler la fonction Lambda spécifiée.

```
aws iam create-policy  --policy-name rds-lambda-policy --policy-document '{
    "Version": "2012-10-17",
    "Statement": [
```

```
        {
        "Sid": "AllowAccessToExampleFunction",
        "Effect": "Allow",
        "Action": "lambda:InvokeFunction",
        "Resource": "arn:aws:lambda:aws-region:444455556666:function:my-function"
        }
    ]
}'
```

Vous pouvez également utiliser la politique `AWSLambdaRole` prédéfinie qui vous permet d'appeler n'importe laquelle de vos fonctions Lambda. Pour de plus amples informations, veuillez consulter la rubrique Politiques IAM basées sur l'identité pour Lambda.

2. Utilisez la AWS CLI commande create-role pour créer un rôle IAM que la politique peut assumer lors de l'exécution.

```
aws iam create-role  --role-name rds-lambda-role --assume-role-policy-document '{
    "Version": "2012-10-17",
    "Statement": [
        {
        "Effect": "Allow",
        "Principal": {
            "Service": "rds.amazonaws.com"
        },
        "Action": "sts:AssumeRole"
        }
    ]
}'
```

3. Appliquez la politique au rôle à l'aide de la attach-role-policy AWS CLI commande.

```
aws iam attach-role-policy \
    --policy-arn arn:aws:iam::444455556666:policy/rds-lambda-policy \
    --role-name rds-lambda-role --region aws-region
```

4. AWS CLI Cette dernière étape permet aux utilisateurs de base de données de votre instance de base de données d'appeler des fonctions Lambda.

```
aws rds add-role-to-db-instance \
       --db-instance-identifier my-instance-name \
       --feature-name Lambda \
       --role-arn  arn:aws:iam::444455556666:role/rds-lambda-role   \
```

```
        --region aws-region
```

Une fois le VPC et les configurations IAM terminées, vous pouvez désormais installer l'extension `aws_lambda`. (Notez que vous pouvez installer l'extension à tout moment, mais tant que vous n'avez pas configuré la prise en charge du VPC et les privilèges IAM corrects, l'extension `aws_lambda` n'ajoute rien aux fonctionnalités de votre instance de base de données RDS for PostgreSQL.)

Étape 3 : installer l'extension **`aws_lambda`** pour une instance de base de données RDS for PostgreSQL

Cette extension offre à votre instance de base de données RDS for PostgreSQL la capacité d'appeler des fonctions Lambda depuis PostgreSQL.

Pour installer l'extension **`aws_lambda`** dans votre instance de base de données RDS for PostgreSQL

Utilisez la ligne de commande `psql` de PostgreSQL ou l'outil pgAdmin afin de vous connecter à votre instance de base de données RDS for PostgreSQL.

1. Connectez-vous à votre instance de base de données RDS for PostgreSQL en tant qu'utilisateur doté de privilèges `rds_superuser`. L'utilisateur `postgres` par défaut est illustré dans l'exemple.

   ```
   psql -h instance.444455556666.aws-region.rds.amazonaws.com -U postgres -p 5432
   ```

2. Installez l'extension `aws_lambda`. L'extension `aws_commons` est également requise. Elle fournit des fonctions d'assistance pour `aws_lambda` et de nombreuses autres extensions Aurora pour PostgreSQL. Si elle n'est pas déjà sur votre instance de base de données RDS for PostgreSQL, elle est installée avec `aws_lambda` comme illustré ci-dessous.

   ```
   CREATE EXTENSION IF NOT EXISTS aws_lambda CASCADE;
   NOTICE:  installing required extension "aws_commons"
   CREATE EXTENSION
   ```

L'extension `aws_lambda` est installée dans l'instance de base de données primaire de votre . Vous pouvez désormais créer des structures de commodité pour appeler vos fonctions Lambda.

Étape 4 : utiliser les fonctions d'assistance Lambda avec votre instance de base de données RDS for PostgreSQL (Facultatif)

Vous pouvez utiliser les fonctions d'assistance de l'extension `aws_commons` pour préparer les entités que vous pouvez appeler plus facilement depuis PostgreSQL. Pour ce faire, vous avez besoin des informations suivantes concernant vos fonctions Lambda :

- Function name (Nom de la fonction) — Le nom, l'Amazon Resource Name (ARN), la version ou l'alias de la fonction Lambda. La politique IAM créée dans Étape 2 : configurer IAM pour votre instance et Lambda nécessite l'ARN, nous vous recommandons donc d'utiliser l'ARN de votre fonction.
- AWS Région — (Facultatif) AWS Région dans laquelle se trouve la fonction Lambda si elle ne se trouve pas dans la même région que votre instance de base de données Aurora .

Pour conserver les informations de nom de la fonction Lambda, utilisez la fonction aws_commons.create_lambda_function_arn. Cette fonction d'assistance crée une structure composite `aws_commons._lambda_function_arn_1` avec les détails requis par la fonction d'appel. Vous trouverez ci-dessous trois autres approches pour configurer cette structure composite.

```
SELECT aws_commons.create_lambda_function_arn(
   'my-function',
   'aws-region'
) AS aws_lambda_arn_1 \gset
```

```
SELECT aws_commons.create_lambda_function_arn(
   '111122223333:function:my-function',
   'aws-region'
) AS lambda_partial_arn_1 \gset
```

```
SELECT aws_commons.create_lambda_function_arn(
   'arn:aws:lambda:aws-region:111122223333:function:my-function'
) AS lambda_arn_1 \gset
```

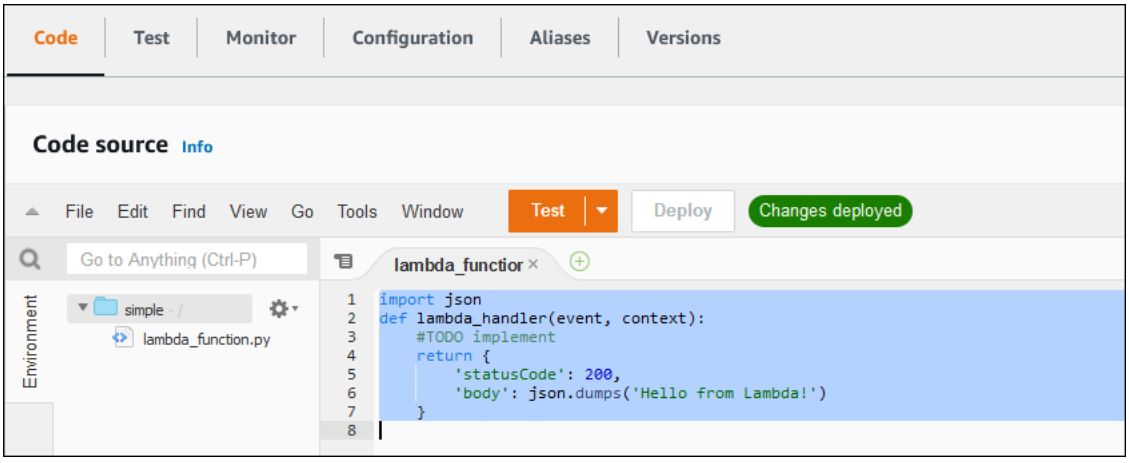
N'importe laquelle de ces valeurs peut être utilisée dans les appels à la fonction aws_lambda.invoke. Pour obtenir des exemples, consultez Étape 5 : appeler une fonction Lambda à partir de votre instance de base de données RDS for PostgreSQL..

Étape 5 : appeler une fonction Lambda à partir de votre instance de base de données RDS for PostgreSQL.

La fonction `aws_lambda.invoke` se comporte de manière synchrone ou asynchrone, en fonction du `invocation_type`. Les deux alternatives à ce paramètre sont `RequestResponse` (valeur par défaut) et `Event`, comme suit.

- **`RequestResponse`** — Ce type d'appel est synchrone. Il s'agit du comportement par défaut lorsque l'appel est effectué sans spécifier de type d'appel. La charge utile de réponse inclut les résultats de la fonction `aws_lambda.invoke`. Utilisez ce type d'appel lorsque votre flux de travail nécessite la réception des résultats de la fonction Lambda avant de continuer.
- **`Event`** — Ce type d'appel est asynchrone. La réponse n'inclut pas de charge utile contenant des résultats. Utilisez ce type d'appel lorsque votre flux de travail n'a pas besoin de résultat de la fonction Lambda pour continuer le traitement.

Pour tester simplement votre configuration, vous pouvez vous connecter à votre instance de base de données en utilisant `psql` et appeler un exemple de fonction depuis la ligne de commande. Supposons que l'une des fonctions de base soit configurée sur votre service Lambda, telle que la fonction simple Python affichée dans la capture d'écran suivante.



Pour invoquer un exemple de fonction

1. Connectez-vous à votre instance de base de données avec `psql` ou pgAdmin.

   ```
   psql -h instance.444455556666.aws-region.rds.amazonaws.com -U postgres -p 5432
   ```

2. Appelez la fonction en utilisant son ARN.

```
SELECT * from
 aws_lambda.invoke(aws_commons.create_lambda_function_arn('arn:aws:lambda:aws-
region:444455556666:function:simple', 'us-west-1'), '{"body": "Hello from
 Postgres!"}'::json );
```

La réponse se présente comme suit.

```
status_code |                        payload                         |
 executed_version | log_result
-------------+--------------------------------------------------------
+------------------+------------
         200 | {"statusCode": 200, "body": "\"Hello from Lambda!\""} | $LATEST
     |
(1 row)
```

Si votre tentative d'appel ne réussit pas, veuillez consulter la section [Messages d'erreur de fonction Lambda].

Étape 6 : accorder aux autres utilisateurs l'autorisation d'appeler les fonctions Lambda

À ce stade des procédures, vous êtes le seul, en tant que `rds_superuser`, à pouvoir appeler vos fonctions Lambda. Pour permettre à d'autres utilisateurs d'appeler les fonctions que vous avez créées, vous devez leur accorder des autorisations.

Pour accorder à d'autres personnes l'autorisation d'appeler les fonctions Lambda

1. Connectez-vous à votre instance de base de données avec `psql` ou pgAdmin.

```
psql -h instance.444455556666.aws-region.rds.amazonaws.com -U postgres -p 5432
```

2. Exécutez les commandes SQL suivantes :

```
postgres=>  GRANT USAGE ON SCHEMA aws_lambda TO db_username;
GRANT EXECUTE ON ALL FUNCTIONS IN SCHEMA aws_lambda TO db_username;
```

Exemples : Invocation de fonctions Lambda depuis votre RDScluster de SQL

Ci-dessous, vous pouvez trouver plusieurs exemples d'appel de la fonction aws_lambda.invoke. La plupart des exemples utilisent la structure composite `aws_lambda_arn_1` que vous créez Étape 4 : utiliser les fonctions d'assistance Lambda avec votre instance de base de données RDS for PostgreSQL (Facultatif) pour simplifier la transmission des détails de la fonction. Pour obtenir un exemple d'appel asynchrone, reportez-vous à la section Exemple : appel asynchrone (Event) de fonctions Lambda. Tous les autres exemples répertoriés utilisent l'appel synchrone.

Pour en savoir plus sur les types d'invocation Lambda, consultez la section Invocation de fonctions Lambda dans le AWS Lambda Guide du développeur. Pour plus d'informations sur `aws_lambda_arn_1`, consultez aws_commons.create_lambda_function_arn.

Liste d'exemples

- Exemple : appel synchrone (RequestResponse) de fonctions Lambda
- Exemple : appel asynchrone (Event) de fonctions Lambda
- Exemple : capture du journal d'exécution Lambda dans une réponse de fonction
- Exemple : inclusion du contexte client dans une fonction Lambda
- Exemple : appel d'une version spécifique d'une fonction Lambda

Exemple : appel synchrone (RequestResponse) de fonctions Lambda

Voici deux exemples d'appel synchrone de fonction Lambda. Les résultats de ces appels de fonction `aws_lambda.invoke` sont identiques.

```
SELECT * FROM aws_lambda.invoke('aws_lambda_arn_1', '{"body": "Hello from
 Postgres!"}'::json);
```

```
SELECT * FROM aws_lambda.invoke('aws_lambda_arn_1', '{"body": "Hello from
 Postgres!"}'::json, 'RequestResponse');
```

Les paramètres sont décrits comme suit :

- `:'aws_lambda_arn_1'` — Ce paramètre identifie la structure composite créée dans Étape 4 : utiliser les fonctions d'assistance Lambda avec votre instance de

base de données RDS for PostgreSQL (Facultatif), avec la fonction d'assistance `aws_commons.create_lambda_function_arn`. Vous pouvez également créer cette structure en ligne dans votre appel `aws_lambda.invoke` comme suit.

```
SELECT * FROM aws_lambda.invoke(aws_commons.create_lambda_function_arn('my-function',
 'aws-region'),
'{"body": "Hello from Postgres!"}'::json
);
```

- `'{"body": "Hello from PostgreSQL!"}'::json`— La JSON charge utile à transmettre à la fonction Lambda.
- `'RequestResponse'` – Type d'appel Lambda.

Exemple : appel asynchrone (Event) de fonctions Lambda

Voici un exemple d'appel de fonction Lambda asynchrone. Le type d'appel `Event` planifie l'appel de fonction Lambda avec la charge utile d'entrée spécifiée et renvoie une réponse immédiatement. Utiliser le type d'appel `Event` dans certains flux de travail qui ne dépendent pas des résultats de la fonction Lambda.

```
SELECT * FROM aws_lambda.invoke('aws_lambda_arn_1', '{"body": "Hello from
 Postgres!"}'::json, 'Event');
```

Exemple : capture du journal d'exécution Lambda dans une réponse de fonction

Vous pouvez inclure les 4 derniers Ko du journal d'exécution dans la réponse de la fonction à l'aide du paramètre `log_type` dans votre appel de fonction `aws_lambda.invoke`. Par défaut, ce paramètre est défini sur `None`, mais vous pouvez spécifier `Tail` afin de capturer les résultats du journal d'exécution Lambda dans la réponse, comme indiqué ci-dessous.

```
SELECT *, select convert_from(decode(log_result, 'base64'), 'utf-8') as log FROM
 aws_lambda.invoke(:'aws_lambda_arn_1', '{"body": "Hello from Postgres!"}'::json,
 'RequestResponse', 'Tail');
```

Définissez le paramètre aws_lambda.invoke de la fonction `log_type` sur `Tail` pour inclure le journal d'exécution dans la réponse. La valeur par défaut du paramètre `log_type` est `None`.

Le `log_result` qui est retourné est une chaîne `base64` encodée. Vous pouvez décoder le contenu en combinant les fonctions `decode` et `convert_from` PostgreSQL.

Pour plus d'informations sur `log_type`, consultez aws_lambda.invoke.

Exemple : inclusion du contexte client dans une fonction Lambda

La fonction `aws_lambda.invoke` possède un paramètre `context` que vous pouvez utiliser pour transférer des informations séparées de la charge utile, comme indiqué ci-dessous.

```
SELECT *, convert_from(decode(log_result, 'base64'), 'utf-8') as log FROM
 aws_lambda.invoke(:'aws_lambda_arn_1', '{"body": "Hello from Postgres!"}'::json,
 'RequestResponse', 'Tail');
```

Pour inclure le contexte du client, utilisez un JSON objet pour le `context` paramètre de la aws_lambda.invoke fonction.

Pour plus d'informations sur le paramètre `context`, veuillez consulter la référence aws_lambda.invoke.

Exemple : appel d'une version spécifique d'une fonction Lambda

Vous pouvez spécifier une version particulière d'une fonction Lambda en incluant le paramètre `qualifier` avec l'appel `aws_lambda.invoke`. Vous trouverez ci-dessous un exemple de ce procédé qui utilise '*`custom_version`*' comme alias pour la version.

```
SELECT * FROM aws_lambda.invoke('aws_lambda_arn_1', '{"body": "Hello from
 Postgres!"}'::json, 'RequestResponse', 'None', NULL, 'custom_version');
```

Vous pouvez également fournir un qualificatif de fonction Lambda avec les informations de nom de la fonction à la place, comme suit.

```
SELECT * FROM aws_lambda.invoke(aws_commons.create_lambda_function_arn('my-
function:custom_version', 'us-west-2'),
'{"body": "Hello from Postgres!"}'::json);
```

Pour de plus amples informations sur `qualifier` et d'autres paramètres, veuillez consulter la référence aws_lambda.invoke.

Messages d'erreur de fonction Lambda

Dans la liste suivante, vous trouverez des informations sur les messages d'erreur, avec les causes et les solutions possibles.

- VPCproblèmes de configuration

  VPCdes problèmes de configuration peuvent générer les messages d'erreur suivants lorsque vous essayez de vous connecter :

  ```
  ERROR:  invoke API failed
  DETAIL: AWS Lambda client returned 'Unable to connect to endpoint'.
  CONTEXT:  SQL function "invoke" statement 1
  ```

  Cette erreur est souvent due à une configuration incorrecte du groupe VPC de sécurité. Assurez-vous d'avoir une règle de sortie pour l'TCPouverture sur le port 443 de votre groupe de VPC sécurité afin de VPC pouvoir vous connecter au LambdaVPC.

  Si votre instance de base de données est privée, vérifiez la DNS configuration privée de votreVPC. Assurez-vous de définir le `rds.custom_dns_resolution` paramètre sur 1 et de configurer AWS PrivateLink comme indiqué dansÉtape 1 : configurer votre instance de base de données pour les connexions sortantes vers AWS Lambda. Pour plus d'informations, consultez la section VPC Points de terminaison de l'interface (AWS PrivateLink).

- Manque d'autorisations nécessaires pour appeler les fonctions Lambda

  Si l'un des messages d'erreur suivants s'affiche, l'utilisateur (rôle) qui appelle la fonction ne dispose pas des autorisations nécessaires.

  ```
  ERROR:  permission denied for schema aws_lambda
  ```

  ```
  ERROR:  permission denied for function invoke
  ```

  Un utilisateur (rôle) doit recevoir des autorisations spécifiques pour appeler les fonctions Lambda. Pour de plus amples informations, veuillez consulter Étape 6 : accorder aux autres utilisateurs l'autorisation d'appeler les fonctions Lambda.

- Traitement inapproprié des erreurs dans vos fonctions Lambda4

  Si une fonction Lambda génère une exception pendant le traitement de la demande, elle `aws_lambda.invoke` échoue avec une SQL erreur Postgre telle que la suivante.

  ```
  SELECT * FROM aws_lambda.invoke('aws_lambda_arn_1', '{"body": "Hello from
   Postgres!"}'::json);
  ERROR:  lambda invocation failed
  ```

```
DETAIL:  "arn:aws:lambda:us-west-2:555555555555:function:my-function" returned error
 "Unhandled", details: "<Error details string>".
```

Veillez à gérer les erreurs dans vos fonctions Lambda ou dans votre application SQL Postgre.

AWS Lambdaréférence de fonction et de paramètre

Vous trouverez ci-dessous la référence des fonctions et des paramètres à utiliser pour appeler Lambda avec .

Fonctions et paramètres

- aws_lambda.invoke
- aws_commons.create_lambda_function_arn
- paramètres aws_lambda

aws_lambda.invoke

Exécute une fonction Lambda pour une instance de base de données RDS for PostgreSQL.

Pour plus de détails sur l'appel de fonctions Lambda, consultez également la section Appel dans le Manuel du développeur AWS Lambda.

Syntaxe

JSON

```
aws_lambda.invoke(
IN function_name TEXT,
IN payload JSON,
IN region TEXT DEFAULT NULL,
IN invocation_type TEXT DEFAULT 'RequestResponse',
IN log_type TEXT DEFAULT 'None',
IN context JSON DEFAULT NULL,
IN qualifier VARCHAR(128) DEFAULT NULL,
OUT status_code INT,
OUT payload JSON,
OUT executed_version TEXT,
OUT log_result TEXT)
```

```
aws_lambda.invoke(
IN function_name aws_commons._lambda_function_arn_1,
IN payload JSON,
IN invocation_type TEXT DEFAULT 'RequestResponse',
IN log_type TEXT DEFAULT 'None',
IN context JSON DEFAULT NULL,
IN qualifier VARCHAR(128) DEFAULT NULL,
OUT status_code INT,
OUT payload JSON,
OUT executed_version TEXT,
OUT log_result TEXT)
```

JSONB

```
aws_lambda.invoke(
IN function_name TEXT,
IN payload JSONB,
IN region TEXT DEFAULT NULL,
IN invocation_type TEXT DEFAULT 'RequestResponse',
IN log_type TEXT DEFAULT 'None',
IN context JSONB DEFAULT NULL,
IN qualifier VARCHAR(128) DEFAULT NULL,
OUT status_code INT,
OUT payload JSONB,
OUT executed_version TEXT,
OUT log_result TEXT)
```

```
aws_lambda.invoke(
IN function_name aws_commons._lambda_function_arn_1,
IN payload JSONB,
IN invocation_type TEXT DEFAULT 'RequestResponse',
IN log_type TEXT DEFAULT 'None',
IN context JSONB DEFAULT NULL,
IN qualifier VARCHAR(128) DEFAULT NULL,
OUT status_code INT,
OUT payload JSONB,
OUT executed_version TEXT,
OUT log_result TEXT
)
```

Paramètres d'entrée

function_name

Nom d'identification de la fonction Lambda. La valeur peut être le nom de la fonction, un ARN ou un ARN partiel. Pour obtenir la liste des formats possibles, consultez Formats de nom de fonction Lambda dans le Manuel du développeur AWS Lambda.

payload

Entrée de la fonction Lambda. Le format peut être JSON ou JSONB. Pour de plus amples informations, veuillez consulter la documentation PostgreSQL sur les types JSON.

région

(Facultatif) Région Lambda de la fonction. Par défaut, RDS résout la Région AWS à partir de l'ARN complet dans le `function_name` ou utilise la Région de l'instance de base de données RDS for PostgreSQL. Si cette valeur de région est en conflit avec celle fournie dans l'ARN `function_name`, une erreur est déclenchée.

invocation_type

Type d'appel de la fonction Lambda. La valeur est sensible à la casse. Les valeurs possibles sont notamment les suivantes :

- `RequestResponse` – Valeur par défaut Ce type d'appel d'une fonction Lambda est synchrone et renvoie une charge utile de réponse dans le résultat. Utilisez le type d'appel `RequestResponse` lorsque votre flux de travail dépend de la réception immédiate du résultat de la fonction Lambda.
- `Event` – Ce type d'appel d'une fonction Lambda est asynchrone et retourne une réponse immédiatement sans retourner de charge utile. Utilisez le type d'appel `Event` lorsque vous n'avez pas besoin des résultats de la fonction Lambda avant que votre flux de travail ne progresse.
- `DryRun` – Ce type d'appel teste l'accès sans exécuter la fonction Lambda.

log_type

Type de journal Lambda à renvoyer dans le paramètre de sortie `log_result`. La valeur est sensible à la casse. Les valeurs possibles sont notamment les suivantes :

- Tail – Le paramètre de sortie `log_result` renvoyé inclura les 4 derniers Ko du journal d'exécution.
- None – Aucune information de journal Lambda n'est renvoyée.

context

Contexte client au format JSON ou JSONB. Les champs à utiliser incluent alors `custom` et `env`.

qualifier

Qualificateur qui identifie la version d'une fonction Lambda à appeler. Si cette valeur est en conflit avec celle fournie dans l'ARN `function_name`, une erreur est déclenchée.

Paramètres de sortie

status_code

Code de réponse d'état HTTP. Pour plus d'informations, consultez Éléments de réponse à l'appel de la fonction Lambda dans le AWS LambdaManuel du développeur .

payload

Informations renvoyées à partir de la fonction Lambda exécutée. Le format est en JSON ou JSONB.

executed_version

Version de la fonction Lambda exécutée.

log_result

Informations du journal d'exécution renvoyées si la valeur `log_type` est `Tail` lorsque la fonction Lambda a été appelée. Le résultat contient les 4 derniers Ko du journal d'exécution codé en Base64.

aws_commons.create_lambda_function_arn

Crée une structure `aws_commons._lambda_function_arn_1` pour contenir les informations de nom de fonction Lambda. Vous pouvez utiliser les résultats de la fonction `aws_commons.create_lambda_function_arn` dans le paramètre `function_name` de la fonction aws_lambda.invoke aws_lambda.invoke.

Syntaxe

```
aws_commons.create_lambda_function_arn(
    function_name TEXT,
    region TEXT DEFAULT NULL
```

```
    )
    RETURNS aws_commons._lambda_function_arn_1
```

Paramètres d'entrée

function_name

Chaîne de texte obligatoire contenant le nom de la fonction Lambda. La valeur peut être un nom de fonction, un ARN partiel ou un ARN complet.

région

Chaîne de texte facultative contenant la région AWS dans laquelle se trouve la fonction Lambda. Pour obtenir la liste des noms de régions et les valeurs associées, consultez Régions, zones de disponibilité et zones locales.

paramètres aws_lambda

Dans ce tableau, vous trouverez les paramètres associés à la `aws_lambda` fonction.

Paramètre	Description
`aws_lambda.connect_timeout_ms`	Il s'agit d'un paramètre dynamique qui définit le temps d'attente maximal lors de la connexion à AWS Lambda. Les valeurs par défaut sont`1000`. Les valeurs autorisées pour ce paramètre sont comprises entre 1 et 900 000.
`aws_lambda.request_timeout_ms`	Il s'agit d'un paramètre dynamique qui définit le temps d'attente maximal pendant l'attente d'une réponse de AWS Lambda. Les valeurs par défaut sont`3000`. Les valeurs autorisées pour ce paramètre sont comprises entre 1 et 900 000.
`aws_lambda.endpoint_override`	Spécifie le point de terminaison qui peut être utilisé pour se connecter à AWS Lambda. Une chaîne vide sélectionne le point de terminaison AWS Lambda par défaut pour la région. Vous devez redémarrer la base de données pour que cette modification de paramètre statique soit prise en compte.

Tâches courantes d'administration de bases de données pour Amazon RDS for PostgreSQL

Les administrateurs de base de données (DBAs) exécutent diverses tâches lors de l'administration d'une instance de base de données Amazon RDS for PostgreSQL. Si vous êtes administrateur de base de données déjà familier avec PostgreSQL, vous devez connaître certaines des différences importantes entre l'exécution de PostgreSQL sur votre matériel et RDS for PostgreSQL. Par exemple, comme il s'agit d'un service géré, Amazon RDS n'autorise pas l'accès shell à vos instances de base de données. Cela signifie que vous n'avez pas d'accès direct à `pg_hba.conf` et aux autres fichiers de configuration. Pour RDS for PostgreSQL, les modifications généralement apportées au fichier de configuration PostgreSQL d'une instance sur site sont apportées à un groupe de paramètres de base de données personnalisé associé à l'instance de base de données RDS for PostgreSQL. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).

Vous ne pouvez pas non plus accéder aux fichiers journaux de la même manière qu'avec une instance PostgreSQL sur site. Pour en savoir plus sur la journalisation, consultez Fichiers journaux de base de données RDS for PostgreSQL.

Autre exemple, vous n'avez pas accès au compte `superuser` PostgreSQL. Sur RDS for PostgreSQL, le rôle `rds_superuser` dispose des privilèges les plus élevés. Il est accordé à `postgres` au moment de la configuration. Que vous soyez familier avec l'utilisation de PostgreSQL sur site ou que vous n'ayez aucune expérience avec RDS for PostgreSQL, nous vous recommandons de comprendre le rôle `rds_superuser` et de vous renseigner sur l'utilisation des rôles, des utilisateurs, des groupes et des autorisations. Pour de plus amples informations, veuillez consulter Comprendre les rôles et les autorisations PostgreSQL.

Voici quelques tâches DBA courantes pour RDS for PostgreSQL.

Rubriques

- Collations prises en charge dans pour Postgre SQL
- Comprendre les rôles et les autorisations PostgreSQL
- Gestion des connexions mortes dans PostgreSQL
-
- Utilisation de mécanismes de journalisation pris en charge par RDS for PostgreSQL
- Gestion des fichiers temporaires avec PostgreSQL
- Utilisation de pgBadger pour l'analyse de journal serveur avec PostgreSQL

- Utilisation PGSnapper pour la surveillance de PostgreSQL
- Utilisation de paramètres sur votre instance de base de données RDS for PostgreSQL

Collations prises en charge dans pour Postgre SQL

Les classements sont un ensemble de règles qui déterminent la manière dont les chaînes de caractères stockées dans la base de données sont triées et comparées. Les classements jouent un rôle fondamental dans le système informatique et sont inclus dans le système d'exploitation. Les classements changent au fil du temps lorsque de nouveaux caractères sont ajoutés aux langues ou lorsque les règles de classement changent.

Les bibliothèques de classement définissent des règles et des algorithmes spécifiques pour un classement. Les bibliothèques de classement les plus populaires utilisées dans Postgre SQL sont le GNU C (glibc) et les composants d'internationalisation pour Unicode (). ICU Par défaut, utilise le classement glibc qui inclut les ordres de tri des caractères Unicode pour les séquences de caractères multi-octets.

Lorsque vous créez une nouvelle instance de base de données de RDS pour PostgreSQL, elle vérifie le classement disponible dans le système d'exploitation. Les SQL paramètres Postgre de la `CREATE DATABASE` commande `LC_COLLATE` et `LC_CTYPE` sont utilisés pour spécifier un classement, qui est le classement par défaut dans cette base de données. Vous pouvez également utiliser le paramètre `LOCALE` dans `CREATE DATABASE` pour définir ces paramètres. Cela détermine le classement par défaut pour les chaînes de caractères dans la base de données et les règles de classification des caractères sous forme de lettres, de chiffres ou de symboles. Vous pouvez également choisir un classement à utiliser sur une colonne, un index ou une requête.

pour Postgre SQL dépend de la bibliothèque glibc du système d'exploitation pour la prise en charge du classement. L'SQLinstance pour Postgre est régulièrement mise à jour avec les dernières versions du système d'exploitation. Ces mises à jour incluent parfois une version plus récente de la bibliothèque glibc. Dans de rares cas, les nouvelles versions de glibc modifient l'ordre de tri ou le classement de certains caractères, ce qui peut entraîner un tri différent des données ou la production d'entrées d'index non valides. Si vous découvrez des problèmes d'ordre de tri pour le classement lors d'une mise à jour, vous devrez peut-être reconstruire les index.

Afin de réduire les impacts possibles des mises à jour de la glibc, inclut SQL désormais une bibliothèque de classement par défaut indépendante. Cette bibliothèque de classement est disponible dans 14.6, 13.9, 12.13, 11.18, 10.23 et les versions mineures plus récentes. Elle est compatible

avec glibc 2.26-59.amzn2 et assure la stabilité de l'ordre de tri afin d'éviter des résultats de requêtes incorrects.

Comprendre les rôles et les autorisations PostgreSQL

Lorsque vous créez une instance de base de RDS pour PostgreSQL à l'aide de, un compte administrateur est créé en AWS Management Console même temps. Par défaut, son nom est `postgres`, comme illustré dans la capture d'écran ci-dessous :



Vous pouvez choisir un autre nom plutôt que d'accepter le nom par défaut (`postgres`). Le nom que vous choisissez doit commencer par une lettre et comporter entre 1 et 16 caractères alphanumériques. Par souci de simplicité, nous nous référons à ce compte utilisateur principal par sa valeur par défaut (`postgres`) tout au long de ce manuel.

Si vous utilisez le `create-db-instance` AWS CLI plutôt que le AWS Management Console, vous créez le nom en le transmettant avec le `master-username` paramètre dans la commande. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS.

Que vous utilisiez l'API AWS Management Console AWS CLI, la ou l'API Amazon RDS, que vous utilisiez le `postgres` nom par défaut ou que vous choisissiez un autre nom, ce premier compte utilisateur de base de données est membre du `rds_superuser` groupe et dispose de `rds_superuser` privilèges.

Rubriques

- Comprendre le rôle rds_superuser
- Contrôle de l'accès utilisateur à la base de données PostgreSQL
- Délégation et contrôle de la gestion des mots de passe utilisateur

- Utilisation de SCRAM pour le chiffrement de mot de passe PostgreSQL

Comprendre le rôle rds_superuser

Dans PostgreSQL, un rôle peut définir un utilisateur, un groupe ou un ensemble d'autorisations spécifiques accordées à un groupe ou à un utilisateur pour divers objets de la base de données. Les commandes PostgreSQL `CREATE USER` et `CREATE GROUP` ont été remplacées par la commande `CREATE ROLE` plus générique avec des propriétés spécifiques permettant de distinguer les utilisateurs de la base de données. Un utilisateur de base de données peut être considéré comme un rôle disposant du privilège LOGIN.

> **Note**
>
> Les commandes `CREATE USER` et `CREATE GROUP` peuvent toujours être utilisées. Pour plus d'informations, consultez Database Roles dans la documentation de PostgreSQL.

L'utilisateur `postgres` est l'utilisateur de base de données disposant des privilèges les plus élevés sur votre instance de base de données RDS for PostgreSQL. Il présente les caractéristiques définies par l'instruction `CREATE ROLE` suivante.

```
CREATE ROLE postgres WITH LOGIN NOSUPERUSER INHERIT CREATEDB CREATEROLE NOREPLICATION
 VALID UNTIL 'infinity'
```

Sauf indication contraire, les propriétés `NOSUPERUSER`, `NOREPLICATION`, `INHERIT` et `VALID UNTIL 'infinity'` sont les options par défaut de CREATE ROLE.

Par défaut, `postgres` fait en sorte que des privilèges soient octroyés au rôle `rds_superuser` ainsi que des autorisations permettant de créer des rôles et des bases de données. Le rôle `rds_superuser` permet à l'utilisateur `postgres` d'effectuer les opérations suivantes :

- Ajoutez les extensions qu'il est possible d'utiliser avec Amazon RDS. Pour de plus amples informations, veuillez consulter Utilisation des SQL fonctionnalités de Postgre prises en charge par Amazon RDS pour Postgre SQL
- Créer des rôles pour les utilisateurs et leur accorder des privilèges. Pour plus d'informations, consultez CREATE ROLE et GRANT dans la documentation de PostgreSQL.
- Créer des bases de données. Pour plus d'informations, consultez CREATE DATABASE dans la documentation de PostgreSQL.

- Accorder des privilèges `rds_superuser` aux rôles utilisateur qui ne disposent pas de ces privilèges, et révoquer les privilèges si nécessaire. Nous vous recommandons d'accorder ce rôle uniquement aux utilisateurs effectuant des tâches de super-utilisateur. En d'autres termes, vous pouvez attribuer ce rôle aux administrateurs de base de données (DBAs) ou aux administrateurs système.
- Accorder (et révoquer) le rôle `rds_replication` aux utilisateurs de base de données qui ne possèdent pas le rôle `rds_superuser`.
- Accorder (et révoquer) le rôle `rds_password` aux utilisateurs de base de données qui ne possèdent pas le rôle `rds_superuser`.
- Obtenir des informations d'état sur toutes les connexions à la base de données en utilisant la vue `pg_stat_activity`. En cas de besoin, `rds_superuser` peut arrêter toutes les connexions à l'aide de `pg_terminate_backend` ou `pg_cancel_backend`.

Dans l'instruction `CREATE ROLE postgres...`, vous pouvez voir que le rôle utilisateur `postgres` rejette spécifiquement les autorisations `superuser` PostgreSQL. RDS for PostgreSQL étant un service géré, vous ne pouvez ni accéder au système d'exploitation hôte, ni vous connecter à l'aide du compte `superuser` PostgreSQL. La plupart des tâches qui exigent un accès `superuser` sur une instance autonome de PostgreSQL sont gérées automatiquement par Amazon RDS.

Pour plus d'informations sur l'octroi de privilèges, veuillez consulter GRANT dans la documentation de PostgreSQL.

Le rôle `rds_superuser` est l'un des nombreux rôles prédéfinis d'un Instance de base de données RDS for PostgreSQL.

> **Note**
>
> Dans PostgreSQL 13 et versions antérieures, les rôles prédéfinis s'appellent rôles par défaut.

La liste suivante répertorie certains des autres rôles prédéfinis créés automatiquement pour un nouveau Instance de base de données RDS for PostgreSQL. Les rôles prédéfinis et leurs privilèges ne peuvent pas être modifiés. Vous ne pouvez pas supprimer, renommer ou modifier les privilèges de ces rôles prédéfinis. Toute tentative de ce type génère une erreur.

- rds_password : rôle pouvant modifier les mots de passe et configurer des contraintes de mot de passe pour les utilisateurs de base de données. Ce `rds_superuser` rôle est attribué par

défaut au rôle et peut être accordé aux utilisateurs de la base de données. Pour de plus amples informations, veuillez consulter Contrôle de l'accès utilisateur à la base de données PostgreSQL.

  - Pour les versions de RDS pour PostgreSQL antérieures à 14`rds_password`, le rôle peut modifier les mots de passe et définir des contraintes de mot de passe pour les utilisateurs de base de données et les utilisateurs dotés d'un rôle. `rds_superuser` À partir de RDS pour PostgreSQL version 14 et versions ultérieures`rds_password`, le rôle peut modifier les mots de passe et configurer des contraintes de mot de passe uniquement pour les utilisateurs de base de données. Seuls les utilisateurs ayant `rds_superuser` un rôle peuvent effectuer ces actions sur d'autres utilisateurs ayant `rds_superuser` un rôle.
- rdsadmin : rôle créé pour gérer la plupart des tâches de gestion que l'administrateur qui utilise les privilèges `superuser` aurait exécutées sur une base de données PostgreSQL autonome. Ce rôle est utilisé en interne par RDS for PostgreSQL pour de nombreuses tâches de gestion.
- rdstopmgr – Rôle utilisé en interne par Amazon RDS pour prendre en charge les déploiements multi-AZ.
- rds_reserved — Rôle utilisé en interne par Amazon RDS pour réserver des connexions à la base de données.

Afficher les rôles et leurs privilèges

Vous pouvez afficher les rôles prédéfinis et leurs privilèges dans votre instance de base de données RDS pour PostgreSQL à l'aide de différentes commandes en fonction de votre version de PostgreSQL. Pour voir tous les rôles prédéfinis, vous pouvez vous connecter à votre instance de base de données RDS pour PostgreSQL et exécuter les commandes suivantes à l'aide du. `psql`

Pour les versions **`psql`** 15 et antérieures

Connectez-vous à votre instance de base de données RDS pour PostgreSQL et utilisez `\du` la commande dans psql :

```
postgres=> \du
                                                    List of roles
    Role name    |                         Attributes                          |
                 Member of
-----------------+-------------------------------------------------------------
+-------------------------------------------------
 postgres        | Create role, Create DB                                      +|
 {rds_superuser}
                 | Password valid until infinity                               |
```

```
 rds_ad          | Cannot login                                                 | {}
 rds_iam         | Cannot login                                                 | {}
 rds_password    | Cannot login                                                 | {}
 rds_replication | Cannot login                                                 | {}
 rds_superuser   | Cannot login                                                 |
{pg_monitor,pg_signal_backend,rds_password,rds_replication}
 rdsadmin        | Superuser, Create role, Create DB, Replication, Bypass RLS+| {}
                 | Password valid until infinity                              |
```

Pour les versions **`psql`** 16 et ultérieures

```
postgres=> \drg+
                            List of role grants
   Role name   |          Member of          |       Options       | Grantor
---------------+-----------------------------+---------------------+----------
 postgres      | rds_superuser               | INHERIT, SET        | rdsadmin
 rds_superuser | pg_checkpoint               | ADMIN, INHERIT, SET | rdsadmin
 rds_superuser | pg_monitor                  | ADMIN, INHERIT, SET | rdsadmin
 rds_superuser | pg_signal_backend           | ADMIN, INHERIT, SET | rdsadmin
 rds_superuser | pg_use_reserved_connections | ADMIN, INHERIT, SET | rdsadmin
 rds_superuser | rds_password                | ADMIN, INHERIT, SET | rdsadmin
 rds_superuser | rds_replication             | ADMIN, INHERIT, SET | rdsadmin
```

Pour vérifier l'appartenance à un rôle sans dépendance de version, vous pouvez utiliser la requête SQL suivante :

```
SELECT m.rolname AS "Role name", r.rolname AS "Member of"
FROM pg_catalog.pg_roles m
JOIN pg_catalog.pg_auth_members pam ON (pam.member = m.oid)
LEFT JOIN pg_catalog.pg_roles r ON (pam.roleid = r.oid)
LEFT JOIN pg_catalog.pg_roles g ON (pam.grantor = g.oid)
WHERE m.rolname !~ '^pg_'
ORDER BY 1, 2;
```

Dans la sortie, vous pouvez voir que `rds_superuser` n'est pas un rôle utilisateur de base de données (il ne peut pas se connecter), mais qu'il dispose des privilèges de nombreux autres rôles. Vous pouvez également voir que l'utilisateur de base de données `postgres` est membre du rôle `rds_superuser`. Comme mentionné précédemment, `postgres` est la valeur par défaut sur la page Create database (Créer une base de données) de la console Amazon RDS. Si vous avez choisi un autre nom, ce nom apparaît dans la liste des rôles.

Contrôle de l'accès utilisateur à la base de données PostgreSQL

Les nouvelles bases de données de PostgreSQL sont toujours créées avec un ensemble de privilèges par défaut dans le schéma `public` de la base de données, qui permet à tous les utilisateurs et rôles de base de données de créer des objets. Ces privilèges permettent aux utilisateurs de base de données de se connecter à la base de données, par exemple, et de créer des tables temporaires lorsqu'ils sont connectés.

Pour mieux contrôler l'accès des utilisateurs aux instances de base de données que vous créez sur votre instance de base de données RDS for PostgreSQL, nous vous recommandons de révoquer ces privilèges `public` par défaut. Vous accordez ensuite des privilèges spécifiques aux utilisateurs de base de données de manière plus détaillée, comme indiqué dans la procédure suivante.

Pour configurer des rôles et des privilèges pour une nouvelle instance de base de données

Supposons que vous configuriez une base de données sur une instance de base de données RDS for PostgreSQL récemment créée à l'usage de plusieurs chercheurs, qui ont tous besoin d'un accès en lecture/écriture à la base de données.

1. Utilisez `psql` (ou pgAdmin) pour vous connecter à votre instance de base de données RDS for PostgreSQL :

   ```
   psql --host=your-db-instance.666666666666.aws-region.rds.amazonaws.com --port=5432
    --username=postgres --password
   ```

   Lorsque vous y êtes invité, saisissez votre mot de passe. Le client `psql` se connecte à la base de données de connexions administratives par défaut, `postgres=>`, et l'affiche sous forme d'invite.
2. Pour empêcher les utilisateurs de base de données de créer des objets dans le schéma `public`, procédez comme suit :

   ```
   postgres=> REVOKE CREATE ON SCHEMA public FROM PUBLIC;
   REVOKE
   ```
3. Vous créez ensuite une instance de base de données :

   ```
   postgres=> CREATE DATABASE lab_db;
   CREATE DATABASE
   ```
4. Révoquez tous les privilèges du schéma `PUBLIC` sur cette nouvelle base de données.

```
postgres=> REVOKE ALL ON DATABASE lab_db FROM public;
REVOKE
```

5. Créez un rôle pour les utilisateurs de base de données.

```
postgres=> CREATE ROLE lab_tech;
CREATE ROLE
```

6. Donnez aux utilisateurs de base de données disposant de ce rôle la possibilité de se connecter à la base de données.

```
postgres=> GRANT CONNECT ON DATABASE lab_db TO lab_tech;
GRANT
```

7. Accordez à tous les utilisateurs dotés du rôle `lab_tech` tous les privilèges sur cette base de données.

```
postgres=> GRANT ALL PRIVILEGES ON DATABASE lab_db TO lab_tech;
GRANT
```

8. Créez des utilisateurs de base de données, comme suit :

```
postgres=> CREATE ROLE lab_user1 LOGIN PASSWORD 'change_me';
CREATE ROLE
postgres=> CREATE ROLE lab_user2 LOGIN PASSWORD 'change_me';
CREATE ROLE
```

9. Accordez à ces deux utilisateurs les privilèges associés au rôle lab_tech :

```
postgres=> GRANT lab_tech TO lab_user1;
GRANT ROLE
postgres=> GRANT lab_tech TO lab_user2;
GRANT ROLE
```

À ce stade, `lab_user1` et `lab_user2` peuvent se connecter à la base de données `lab_db`. Cet exemple ne respecte pas les bonnes pratiques pour une utilisation en entreprise, qui peuvent inclure la création de plusieurs instances de base de données, différents schémas et l'octroi d'autorisations limitées. Pour des informations plus complètes et des scénarios supplémentaires, consultez Managing PostgreSQL Users and Roles.

Pour plus d'informations sur les privilèges dans les bases de données PostgreSQL, veuillez consulter la commande GRANT dans la documentation PostgreSQL.

Délégation et contrôle de la gestion des mots de passe utilisateur

En tant qu'administrateur de base de données, vous souhaitez peut-être déléguer la gestion des mots de passe utilisateur. Vous souhaitez peut-être également empêcher les utilisateurs de base de données de modifier leurs mots de passe ou de reconfigurer les contraintes de mot de passe, telles que la durée de vie d'un mot de passe. Pour vous assurer que seuls les utilisateurs de base de données que vous choisissez peuvent modifier les paramètres de mot de passe, vous pouvez activer la fonctionnalité de gestion restreinte des mots de passe. Lorsque vous activez cette fonctionnalité, seuls les utilisateurs de base de données qui ont obtenu le rôle `rds_password` peuvent gérer les mots de passe.

> **Note**
>
> Pour utiliser la gestion restreinte des mots de passe, votre instance de base de données RDS for PostgreSQL doit exécuter PostgreSQL 10.6 ou versions ultérieures.

Par défaut, cette fonctionnalité est désactivée (`off`), comme illustré ci-dessous :

```
postgres=> SHOW rds.restrict_password_commands;
  rds.restrict_password_commands
--------------------------------
 off
(1 row)
```

Pour l'activer, vous utilisez un groupe de paramètres personnalisé et redéfinissez le paramètre `rds.restrict_password_commands` sur 1. Assurez-vous de redémarrer votre instance de base de données RDS for PostgreSQL pour que le réglage prenne effet.

Lorsque cette fonctionnalité est activée, les privilèges `rds_password` sont requis pour les commandes SQL suivantes :

```
CREATE ROLE myrole WITH PASSWORD 'mypassword';
CREATE ROLE myrole WITH PASSWORD 'mypassword' VALID UNTIL '2023-01-01';
ALTER ROLE myrole WITH PASSWORD 'mypassword' VALID UNTIL '2023-01-01';
ALTER ROLE myrole WITH PASSWORD 'mypassword';
```

```
ALTER ROLE myrole VALID UNTIL '2023-01-01';
ALTER ROLE myrole RENAME TO myrole2;
```

Le changement de nom d'un rôle (`ALTER ROLE myrole RENAME TO newname`) est également restreint si le mot de passe utilise l'algorithme de MD5 hachage.

Lorsque cette fonctionnalité est activée, toute tentative d'exécution de l'une de ces commandes SQL sans les autorisations de rôle `rds_password` génère l'erreur suivante :

```
ERROR: must be a member of rds_password to alter passwords
```

Nous vous recommandons de n'accorder `rds_password` qu'à quelques rôles utilisés exclusivement pour la gestion des mots de passe. Si vous accordez des privilèges `rds_password` aux utilisateurs de base de données qui ne disposent pas de privilèges `rds_superuser`, vous devez également leur accorder l'attribut `CREATEROLE`.

Assurez-vous de vérifier les exigences de mot de passe telles que la date d'expiration et le niveau de complexité requis du côté du client. Si vous utilisez votre propre utilitaire côté client pour les modifications relatives aux mots de passe, l'utilitaire doit être membre de `rds_password` et disposer des privilèges `CREATE ROLE`.

Utilisation de SCRAM pour le chiffrement de mot de passe PostgreSQL

Le mécanisme d'authentification Salted Challenge Response (SCRAM) est une alternative à l'algorithme message digest (MD5) par défaut de PostgreSQL pour chiffrer les mots de passe. Le mécanisme d'authentification SCRAM est considéré comme plus sûr que. MD5 Pour en savoir plus sur ces deux approches différentes de sécurisation des mots de passe, consultez Password Authentication (Authentification par mot de passe) dans la documentation PostgreSQL.

Nous vous recommandons d'utiliser SCRAM plutôt que MD5 comme schéma de chiffrement de mot de passe pour votre cluster de bases de données . Instance de base de données RDS for PostgreSQL. Il s'agit d'un mécanisme stimulation/réponse cryptographique qui utilise l'algorithme scram-sha-256 pour l'authentification par mot de passe et le chiffrement de mot de passe.

Vous devrez peut-être mettre à jour les bibliothèques pour vos applications clientes de sorte qu'elles prennent en charge SCRAM. Par exemple, les versions JDBC antérieures à 42.2.0 ne prennent pas en charge SCRAM. Pour plus d'informations, consultez PostgreSQL JDBC Driver (Pilote JDBC PostgreSQL) dans la documentation du pilote JDBC PostgreSQL. Pour obtenir la liste des autres

pilotes PostgreSQL prenant en charge SCRAM, consultez la liste des pilotes dans la documentation PostgreSQL.

RDS for PostgreSQL version 13.1 et ultérieures prennent en charge scram-sha-256. Ces versions vous permettent également de configurer votre instance de base de données pour qu'elle requiert SCRAM, comme indiqué dans les procédures suivantes.

Configuration de votre instance de base de données RDS for PostgreSQL de sorte à requérir SCRAM

Pour , vous pouvez exiger que l'instance de base de données RDS for PostgreSQL n'accepte que les mots de passe qui utilisent l'algorithme scram-sha-256.

> **Important**
>
> Pour les proxys RDS existants avec des bases de données PostgreSQL, si vous modifiez l'authentification de base de données pour utiliser uniquement `SCRAM`, le proxy devient indisponible pendant 60 secondes au maximum. Pour éviter ce problème, effectuez l'une des actions suivantes :
>
> - Veillez à ce que la base de données permette à la fois l'authentification `SCRAM` et `MD5`.
> - Pour utiliser uniquement l'authentification `SCRAM`, créez un nouveau proxy, migrez le trafic de votre application vers ce nouveau proxy, puis supprimez le proxy précédemment associé à la base de données.

Avant d'apporter des modifications à votre système, assurez-vous de bien comprendre le processus complet, comme suit :

- Obtenez des informations sur tous les rôles et sur le chiffrement des mots de passe pour tous les utilisateurs de base de données.
- Revérifiez les paramètres de votre instance de base de données RDS for PostgreSQL qui contrôlent le chiffrement des mots de passe.
- Si votre instance de base de données RDS for PostgreSQL utilise un groupe de paramètres par défaut, vous devez créer un groupe de paramètres de base de données personnalisé et l'appliquer à votre instance de base de données RDS for PostgreSQL de sorte à pouvoir modifier les paramètres si nécessaire. Si votre instance de base de données RDS for PostgreSQL utilise un groupe de paramètres personnalisé, vous pouvez modifier ultérieurement les paramètres nécessaires dans le processus, selon vos besoins.

- Remplacez le paramètre `password_encryption` par `scram-sha-256`.
- Informez tous les utilisateurs de la base de données qu'ils doivent mettre à jour leurs mots de passe. Faites de même pour votre compte `postgres`. Les nouveaux mots de passe sont chiffrés et stockés à l'aide de l'algorithme scram-sha-256.
- Vérifiez que tous les mots de passe utilisent le même type de chiffrement.
- Si tous les mots de passe utilisent scram-sha-256, vous pouvez modifier le paramètre `rds.accepted_password_auth_method` de `md5+scram` à `scram-sha-256`.

> **Warning**
>
> Après avoir changé `rds.accepted_password_auth_method` pour scram-sha-256 uniquement, tous les utilisateurs (rôles) avec des mots de passe chiffrés par `md5` ne peuvent pas se connecter.

Se préparer à exiger SCRAM pour votre instance de base de données RDS for PostgreSQL

Avant d'apporter des modifications à votre instance de base de données RDS for PostgreSQL, vérifiez tous les comptes utilisateurs de base de données existants. Vérifiez également le type de chiffrement utilisé pour les mots de passe. Pour ce faire, utilisez l'extension `rds_tools`. Pour savoir quelles versions de PostgreSQL sont prises en `rds_tools` charge, consultez la section Versions d'extension pour Amazon RDS for PostgreSQL.

Pour obtenir la liste des utilisateurs de base de données (rôles) et des méthodes de chiffrement des mots de passe

1. Utilisez `psql` pour vous connecter à votre instance de base de données RDS for PostgreSQL, comme suit.

   ```
   psql --host=db-name.111122223333.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password
   ```

2. Installez l'extension `rds_tools`.

   ```
   postgres=> CREATE EXTENSION rds_tools;
   CREATE EXTENSION
   ```

3. Obtenez la liste des rôles et des méthodes de chiffrement.

```
postgres=> SELECT * FROM
      rds_tools.role_password_encryption_type();
```

Vous voyez des résultats similaires à ce qui suit.

```
       rolname        | encryption_type
----------------------+-----------------
 pg_monitor           |
 pg_read_all_settings |
 pg_read_all_stats    |
 pg_stat_scan_tables  |
 pg_signal_backend    |
 lab_tester           | md5
 user_465             | md5
 postgres             | md5
(8 rows)
```

Création d'un groupe de paramètres de base de données personnalisé

> **Note**
>
> Si votre instance de base de données RDS for PostgreSQL utilise déjà un groupe de paramètres personnalisé, vous n'avez pas besoin d'en créer un.

Pour obtenir un aperçu des groupes de paramètres pour Amazon RDS, consultez Utilisation de paramètres sur votre instance de base de données RDS for PostgreSQL.

Le type de chiffrement utilisé pour les mots de passe est défini dans un paramètre, `password_encryption`. Le chiffrement autorisé par l'instance de base de données RDS for PostgreSQL est défini dans un autre paramètre, `rds.accepted_password_auth_method`. Le remplacement de l'un de ces paramètres par une valeur autre que celle par défaut requiert de créer un groupe de paramètres de base de données personnalisé et de l'appliquer à votre instance.

Vous pouvez également utiliser l'API AWS Management Console ou l'API RDS pour créer un de base de données personnalisé. Pour plus d'informations, consultez

Vous pouvez maintenant employer le groupe de paramètres personnalisés avec votre instance de base de données.

Pour créer un groupe de paramètres de base de données personnalisé

1. Utilisez la commande CLI `create-db-parameter-group` pour créer le groupe de paramètres de base de données personnalisé. Cet exemple utilise `postgres13` comme source pour ce groupe de paramètres personnalisé.

   Pour LinuxmacOS, ou Unix :

   ```
   aws rds create-db-parameter-group --db-parameter-group-name 'docs-lab-scram-passwords' \
     --db-parameter-group-family postgres13  --description 'Custom parameter group for SCRAM'
   ```

   Dans Windows :

   ```
   aws rds create-db-parameter-group --db-parameter-group-name "docs-lab-scram-passwords" ^
     --db-parameter-group-family postgres13  --description "Custom DB parameter group for SCRAM"
   ```

2. Utilisez la commande CLI `modify-db-instance` pour appliquer ce groupe de paramètres personnalisé à votre cluster de bases de données RDS for PostgreSQL.

   Pour LinuxmacOS, ou Unix :

   ```
   aws rds modify-db-instance --db-instance-identifier 'your-instance-name' \
           --db-parameter-group-name "docs-lab-scram-passwords
   ```

   Dans Windows :

   ```
   aws rds modify-db-instance --db-instance-identifier "your-instance-name" ^
           --db-parameter-group-name "docs-lab-scram-passwords
   ```

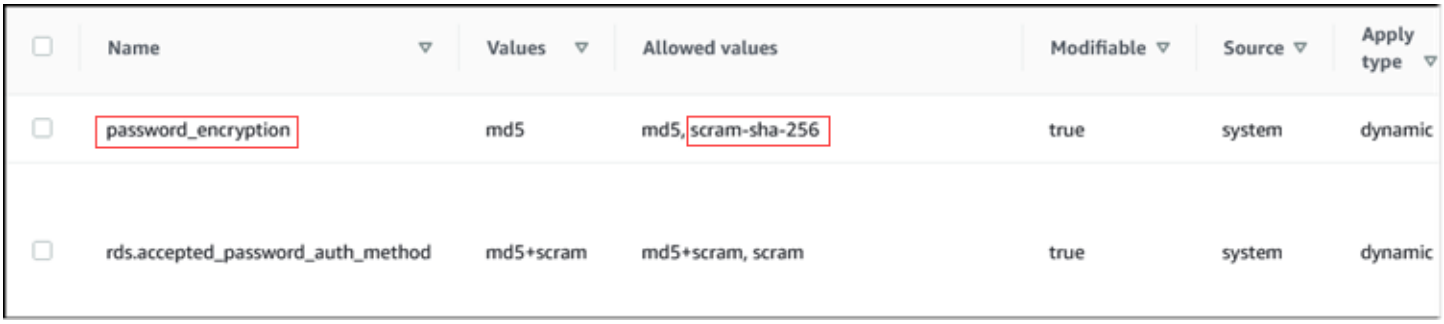
   Pour resynchroniser votre instance de base de données RDS for PostgreSQL avec votre groupe de paramètres de base de données personnalisé, vous devez redémarrer l'instance principale et toutes les autres instances du cluster. Planifiez cette opération pendant votre fenêtre de maintenance habituelle afin de minimiser l'impact sur vos utilisateurs.

Configuration du chiffrement des mots de passe pour utiliser SCRAM

Le mécanisme de chiffrement du mot de passe utilisé par une instance de base de données RDS for PostgreSQL est défini(e) dans le groupe de paramètres de base de données dans le paramètre `password_encryption`. Les valeurs autorisées incluent une valeur non définie, `md5` ou `scram-sha-256`. La valeur par défaut dépend de la version de RDS for PostgreSQL, comme suit :

- RDS for PostgreSQL versions 14 et ultérieures : la valeur par défaut est `scram-sha-256`
- RDS for PostgreSQL 13 : la valeur par défaut est `md5`

En attachant un groupe de paramètres de base de données personnalisé à votre instance de base de données RDS for PostgreSQL, vous pouvez modifier les valeurs du paramètre de chiffrement des mots de passe.



Name	Values	Allowed values	Modifiable	Source	Apply type
password_encryption	md5	md5, scram-sha-256	true	system	dynamic
rds.accepted_password_auth_method	md5+scram	md5+scram, scram	true	system	dynamic

Pour remplacer le paramètre de chiffrement des mots de passe par scram-sha-256

- Remplacez la valeur du chiffrement des mots de passe par scram-sha-256, comme indiqué ci-après. Cette modification peut être appliquée immédiatement, car le paramètre est dynamique. Aucun redémarrage n'est donc nécessaire pour que la modification soit appliquée.

  Pour LinuxmacOS, ou Unix :

  ```
  aws rds modify-db-parameter-group --db-parameter-group-name \
    'docs-lab-scram-passwords' --parameters
   'ParameterName=password_encryption,ParameterValue=scram-
  sha-256,ApplyMethod=immediate'
  ```

  Dans Windows :

  ```
  aws rds modify-db-parameter-group --db-parameter-group-name ^
  ```

```
  "docs-lab-scram-passwords" --parameters
 "ParameterName=password_encryption,ParameterValue=scram-
sha-256,ApplyMethod=immediate"
```

Migration des mots de passe des rôles utilisateur vers SCRAM

Vous pouvez migrer les mots de passe pour les rôles d'utilisateur vers SCRAM comme décrit ci-dessous.

Pour migrer les mots de passe des utilisateurs (rôles) de base de données MD5 vers SCRAM

1. Connectez-vous en tant qu'utilisateur administrateur (nom d'utilisateur par défaut, `postgres`) comme suit.

```
psql --host=db-name.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password
```

2. Vérifiez la valeur du paramètre `password_encryption` sur votre instance de base de données RDS for PostgreSQL à l'aide de la commande suivante.

```
postgres=> SHOW password_encryption;
 password_encryption
---------------------
 md5
 (1 row)
```

3. Remplacez la valeur de ce paramètre par scram-sha-256. Pour de plus amples informations, veuillez consulter Configuration du chiffrement des mots de passe pour utiliser SCRAM.
4. Vérifiez à nouveau la valeur pour vous assurer qu'elle est maintenant réglée sur `scram-sha-256`, comme suit.

```
postgres=> SHOW password_encryption;
 password_encryption
---------------------
 scram-sha-256
 (1 row)
```

5. Demandez à tous les utilisateurs de base de données de modifier leurs mots de passe. Veillez également à modifier votre propre mot de passe pour le compte `postgres` (utilisateur de base de données avec privilèges `rds_superuser`).

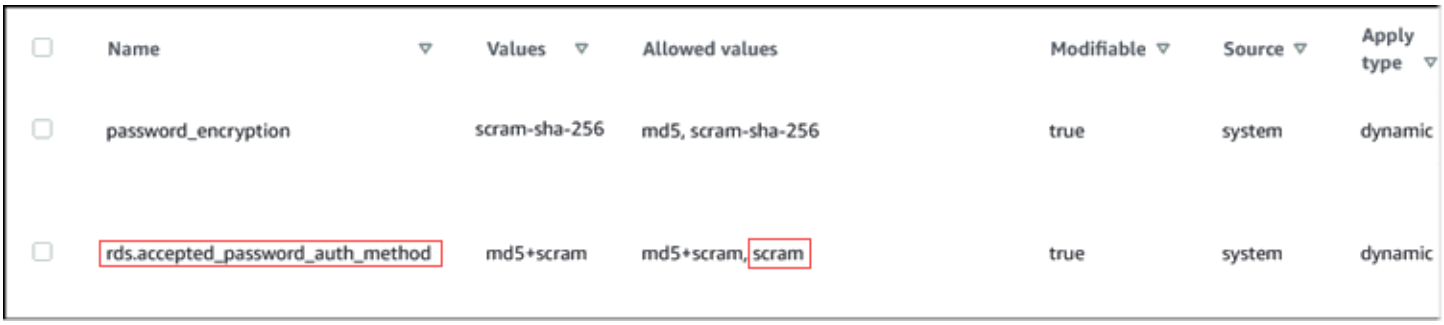
```
labdb=> ALTER ROLE postgres WITH LOGIN PASSWORD 'change_me';
ALTER ROLE
```

6. Répétez l'opération pour toutes les bases de données de votre Instance de base de données RDS for PostgreSQL.

Modification du paramètre de sorte à utiliser SCRAM

Il s'agit de la dernière étape du processus. Une fois que vous avez effectué la modification de la procédure suivante, tous les comptes utilisateurs (rôles) qui utilisent toujours le chiffrement `md5` pour les mots de passe ne pourront pas se connecter au Instance de base de données RDS for PostgreSQL.

Le paramètre `rds.accepted_password_auth_method` spécifie la méthode de chiffrement acceptée par l'instance de base de données RDS for PostgreSQL pour un mot de passe utilisateur pendant le processus de connexion. La valeur par défaut est `md5+scram`, ce qui signifie que l'une des méthodes est acceptée. L'image suivante indique la valeur par défaut de ce paramètre.



Name	Values	Allowed values	Modifiable	Source	Apply type
password_encryption	scram-sha-256	md5, scram-sha-256	true	system	dynamic
rds.accepted_password_auth_method	md5+scram	md5+scram, scram	true	system	dynamic

Les valeurs autorisées pour ce paramètre sont `md5+scram` ou `scram`. Si la valeur de ce paramètre est remplacée par `scram`, le paramètre devient obligatoire.

Pour modifier la valeur du paramètre afin d'exiger l'authentification SCRAM pour les mots de passe

1. Vérifiez que tous les mots de passe utilisateur de toutes les bases de données de votre instance de base de données RDS for PostgreSQL utilisent `scram-sha-256` pour le chiffrement des mots de passe. Pour ce faire, interrogez `rds_tools` pour obtenir le rôle (utilisateur) et le type de chiffrement, comme suit.

```
postgres=> SELECT * FROM rds_tools.role_password_encryption_type();
  rolname        | encryption_type
  ---------------------+-----------------
```

```
  pg_monitor           |
  pg_read_all_settings |
  pg_read_all_stats    |
  pg_stat_scan_tables  |
  pg_signal_backend    |
  lab_tester           | scram-sha-256
  user_465             | scram-sha-256
  postgres             | scram-sha-256
  ( rows)
```

2. Répétez la requête sur toutes les instances de base de données de votre Instance de base de données RDS for PostgreSQL.

   Si tous les mots de passe utilisent scram-sha-256, vous pouvez continuer.

3. Remplacez la valeur de l'authentification par mot de passe acceptée par scram-sha-256, comme suit.

   Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-parameter-group --db-parameter-group-name 'docs-lab-scram-
passwords' \
  --parameters
 'ParameterName=rds.accepted_password_auth_method,ParameterValue=scram,ApplyMethod=immediat
```

   Dans Windows :

```
aws rds modify-db-parameter-group --db-parameter-group-name "docs-lab-scram-
passwords" ^
  --parameters
 "ParameterName=rds.accepted_password_auth_method,ParameterValue=scram,ApplyMethod=immediat
```

Gestion des connexions mortes dans PostgreSQL

Les connexions sont interrompues lorsqu'une session de base de données reste active sur le serveur alors que l'application cliente a été abandonnée ou interrompue de manière anormale. Cette situation se produit généralement lorsque les processus clients se bloquent ou se terminent de manière inattendue sans fermer correctement leurs connexions à la base de données ou annuler les demandes en cours.

PostgreSQL identifie et nettoie efficacement les connexions inactives lorsque les processus du serveur sont inactifs ou tentent d'envoyer des données aux clients. Cependant, la détection est difficile pour les sessions inactives, en attente de saisie par le client ou en cours d'exécution active de requêtes. Pour gérer ces scénarios, PostgreSQL `tcp_keepalives_*` fournit des paramètres`tcp_user_timeout`, et. `client_connection_check_interval`

Rubriques

- Comprendre TCP Keepalive
-
- Cas d'utilisation des paramètres TCP Keepalive
- Bonnes pratiques

Comprendre TCP Keepalive

TCP Keepalive est un mécanisme au niveau du protocole qui permet de maintenir et de vérifier l'intégrité de la connexion. Chaque connexion TCP gère les paramètres au niveau du noyau qui régissent le comportement de keepalive. Lorsque le temporisateur Keepalive expire, le système effectue les opérations suivantes :

- Envoie un paquet de sonde sans aucune donnée et avec l'indicateur ACK activé.
- Attend une réponse du point de terminaison distant conformément aux TCP/IP spécifications.
- Gère l'état de la connexion en fonction de la réponse ou de l'absence de réponse.

Paramètre	Description	Valeurs par défaut
`tcp_keepalives_idle`	Spécifie le nombre de secondes d'inactivité avant l'envoi du message keepalive.	300
`tcp_keepalives_interval`	Spécifie le nombre de secondes entre les retransmissions de messages keepalive non reconnus.	30

Paramètre	Description	Valeurs par défaut
`tcp_keepalives_count`	Nombre maximum de messages KeepAlive perdus avant de déclarer la connexion interrompue	2
`tcp_user_timeout`	Spécifie la durée (en millisecondes) pendant laquelle les données non reconnues peuvent rester avant de fermer de force la connexion.	0
`client_connection_check_interval`	Définit l'intervalle (en millisecondes) pour vérifier l'état de la connexion client lors de requêtes de longue durée. Cela garantit une détection plus rapide des connexions fermées.	0

Cas d'utilisation des paramètres TCP Keepalive

Maintien en vie des sessions inactives

Pour empêcher les connexions inactives d'être interrompues par des pare-feux ou des routeurs pour cause d'inactivité :

- Configurez `tcp_keepalives_idle` pour envoyer des paquets keepalive à intervalles réguliers.

Détecter les connexions mortes

Pour détecter rapidement les connexions inactives :

- Ajustez `tcp_keepalives_idletcp_keepalives_interval`, et`tcp_keepalives_count`. Par exemple, avec les paramètres par défaut d'Aurora PostgreSQL, il faut environ une minute (2 sondes × 30 secondes) pour détecter une connexion interrompue. La réduction de ces valeurs peut accélérer la détection.

- `tcp_user_timeout`À utiliser pour spécifier le temps d'attente maximal pour un accusé de réception.

Les paramètres TCP keepalive aident le noyau à détecter les connexions mortes, mais PostgreSQL risque de ne pas agir tant que le socket n'est pas utilisé. Si une session exécute une longue requête, il est possible que les connexions inactives ne soient détectées qu'une fois la requête terminée. Dans PostgreSQL 14 et versions supérieures`client_connection_check_interval`, il est possible d'accélérer la détection des connexions mortes en interrogeant régulièrement le socket lors de l'exécution de la requête.

Bonnes pratiques

- Définissez des intervalles de conservation raisonnables : réglez `tcp_user_timeout` `tcp_keepalives_count` et `tcp_keepalives_interval` pour équilibrer la vitesse de détection et l'utilisation des ressources. `tcp_keepalives_idle`
- Optimisez pour votre environnement : alignez les paramètres sur le comportement du réseau, les politiques de pare-feu et les besoins des sessions.
- Tirez parti des fonctionnalités de PostgreSQL : utilisez-les `client_connection_check_interval` dans PostgreSQL 14 et versions supérieures pour des contrôles de connexion efficaces.

Nous vous conseillons vivement d'utiliser la fonction autovacuum afin de maintenir l'intégrité de votre instance de base de données PostgreSQL. La fonction autovacuum automatise le lancement des commandes VACUUM et ANALYZE. Elle vérifie les tables ayant eu un grand nombre de tuples insérés, mis à jour ou supprimés. Après cette vérification, elle récupère le stockage en supprimant les données ou les tuples obsolètes de la base de données PostgreSQL.

Par défaut, l'autovacuum est activé pour les instances de base de données RDS pour PostgreSQL PostgreSQL que vous créez à l'aide de l'un des groupes de paramètres de base de données PostgreSQL par défaut. Les autres paramètres de configuration associés à la fonction autovacuum sont également définis par défaut. Comme ces valeurs par défaut sont relativement génériques, vous pouvez bénéficier du réglage de certains paramètres associés à la fonction d'autovacuum pour votre charge de travail spécifique.

Vous trouverez ci-dessous plus d'informations sur l'autovacuum et sur la façon de régler certains de ses paramètres sur votre instance de base de données RDS pour PostgreSQL PostgreSQL.

Pour des informations de haut niveau, voirBonnes pratiques pour utiliser les moteurs de stockage PostgreSQL.

Rubriques

- Allocation de mémoire pour la fonction autovacuum
- Réduction de la probabilité de renvoi à la ligne de l'ID de transaction
- Déterminer si les tables de votre base de données ont besoin d'une opération VACUUM
- Déterminer les tables actuellement éligibles pour autovacuum
- Déterminer si autovacuum est en cours d'exécution et pour combien de temps
- Réalisation d'un gel manuel du processus vacuum
- Réindexation d'une table pendant l'exécution du processus autovacuum
- Gestion de la fonction autovacuum avec de grands index
- Autres paramètres qui affectent la fonction d'autovacuum
- Définition des paramètres d'autovacuum au niveau de la table
- Enregistrement des activités d'autovacuum et de vacuum
- Comprendre le comportement de l'autovacuum avec des bases de données non valides

Allocation de mémoire pour la fonction autovacuum

L'un des paramètres les plus importants qui influencent les performances d'autovacuum est `autovacuum_work_mem`. Dans RDS pour PostgreSQL PostgreSQL versions 14 et antérieuresautovacuum_work_mem, le paramètre est défini sur -1, ce qui indique que le paramètre de est utilisé à la place. `maintenance_work_mem` Pour toutes les autres versions, `autovacuum_work_mem` est déterminé par GREATEST ({DBInstanceClassMemory/32768}, 65536).

Les opérations d'aspiration manuelles utilisent toujours ce `maintenance_work_mem` réglage, le réglage par défaut étant GREATEST ({DBInstanceClassMemory/63963136 *1024}, 65536), et il peut également être ajusté au niveau de la session à l'aide de la commande pour des opérations manuelles plus ciblées. `SET VACUUM`

`autovacuum_work_mem`Détermine la mémoire permettant à Autovacuum de contenir les identifiants des tuples morts (`pg_stat_all_tables.n_dead_tup`) pour les index d'aspiration.

Lorsque vous effectuez des calculs pour déterminer la valeur du `autovacuum_work_mem` paramètre, tenez compte des points suivants :

- Si vous définissez le paramètre trop bas, le processus d'aspiration devra peut-être scanner la table plusieurs fois pour terminer son travail. Ces nombreuses analyses peuvent avoir un impact négatif sur les performances. Pour les instances plus importantes, le réglage `maintenance_work_mem` `autovacuum_work_mem` d'au moins 1 Go peut améliorer les performances des tables d'aspiration contenant un grand nombre de tuples morts. Cependant, dans les versions 16 et antérieures de PostgreSQL, l'utilisation de la mémoire de Vacuum est limitée à 1 Go, ce qui est suffisant pour traiter environ 179 millions de tuples morts en un seul passage. Si une table contient plus de tuples morts que cela, vacuum devra effectuer plusieurs passages dans les index de la table, ce qui augmentera considérablement le temps requis. À partir de la version 17 de PostgreSQL, il n'y a pas de limite de 1 Go et Autovacuum peut traiter plus de 179 millions de tuples en utilisant des arbres de base.

  Un identifiant de tuple a une taille de 6 octets. Pour estimer la mémoire nécessaire pour vider un index d'une table, recherchez le nombre de tuples morts, puis multipliez ce nombre par 6 pour déterminer la mémoire requise pour vider l'index en un seul passage. `pg_stat_all_tables.n_dead_tup` Vous pouvez utiliser la requête suivante :

  ```
  SELECT
      relname AS table_name,
      n_dead_tup,
      pg_size_pretty(n_dead_tup * 6) AS estimated_memory
  FROM
      pg_stat_all_tables
  WHERE
      relname = 'name_of_the_table';
  ```

- Le paramètre `autovacuum_work_mem` fonctionne en conjonction avec le paramètre `autovacuum_max_workers`. Chaque travailleur `autovacuum_max_workers` peut utiliser la mémoire que vous allouez. Si vous avez beaucoup de petites tables, allouez plus de `autovacuum_max_workers` et moins de `autovacuum_work_mem`. Si vos tables sont volumineuses (plus de 100 Go), allouez plus de mémoire et réduisez le nombre de processus de travail. Vous devez avoir alloué suffisamment de mémoire pour pouvoir prendre en charge votre plus grande table. Assurez-vous donc que la combinaison des processus de travail et de la mémoire est égale à la mémoire totale que vous souhaitez allouer.

Réduction de la probabilité de renvoi à la ligne de l'ID de transaction

Dans certains cas, les valeurs du groupe de paramètres associées à la fonction autovacuum peuvent ne pas être suffisamment agressives pour empêcher le renvoi à la ligne de l'ID de transaction. Pour résoudre ce problème, RDS pour PostgreSQL PostgreSQL fournit un mécanisme qui adapte automatiquement les valeurs des paramètres d'autovacuum. Une explication détaillée du renvoi à la ligne de l'ID de transaction figure dans la documentation PostgreSQL.

L'autovacuum adaptatif est activé par défaut pour les instances RDS pour PostgreSQL PostgreSQL avec le paramètre dynamique défini sur ON. `rds.adaptive_autovacuum` Nous vous recommandons vivement de garder cette option activée. Toutefois, pour désactiver le réglage adaptatif des paramètres d'autovacuum, définissez le paramètre `rds.adaptive_autovacuum` sur 0 ou OFF.

L'encapsulation des identifiants de transaction est toujours possible même lorsqu'Amazon RDS Aurora (RDS) ajuste les paramètres d'autovacuum. Nous vous encourageons à implémenter une CloudWatch alarme Amazon pour l'encapsulation des identifiants de transaction. Pour plus d'informations, consultez l'article Implémenter un système d'alerte précoce pour l'encapsulation des identifiants de transaction dans RDS pour PostgreSQL sur le blog de base de données. AWS

Lorsque le réglage adaptatif des paramètres d'aspiration automatique est activé, Amazon RDS commence à ajuster les paramètres d'aspiration automatique lorsque la CloudWatch métrique `MaximumUsedTransactionIDs` atteint la valeur du `autovacuum_freeze_max_age` paramètre ou 500 000 000, la valeur la plus élevée étant retenue.

Amazon RDS continue à ajuster les paramètres pour la fonction autovacuum si une table continue à s'orienter vers le renvoi à la ligne de l'ID de transaction. Chacun de ces ajustements dédie plus de ressources à la fonction d'autovacuum pour éviter le renvoi à la ligne. Amazon RDS met à jour les paramètres suivants associés à la fonction d'autovacuum :

- autovacuum_vacuum_cost_delay
- autovacuum_vacuum_cost_limit
- `autovacuum_work_mem`
- autovacuum_naptime

RDS modifie ces paramètres seulement si la nouvelle valeur rend la fonction d'autovacuum plus agressive. Ces paramètres sont modifiés dans la mémoire sur l'instance de base de données. Les

valeurs figurant dans le groupe de paramètres ne sont pas modifiées. Pour afficher les paramètres en mémoire actuels, utilisez la commande SQL SHOW de PostgreSQL.

Chaque fois que Amazon RDS modifie l'un de ces paramètres d'autovacuum, il génère un événement pour l'instance de base de données concernée. Cet événement est visible sur AWS Management Console et via l'API Amazon RDS. Une fois que la `MaximumUsedTransactionIDs` CloudWatch métrique est revenue en dessous du seuil, Amazon RDS réinitialise les paramètres relatifs à l'autovacuum en mémoire aux valeurs spécifiées dans le groupe de paramètres. Il génère ensuite un autre événement correspondant à cette modification.

Déterminer si les tables de votre base de données ont besoin d'une opération VACUUM

Vous pouvez utiliser la requête suivante pour afficher le nombre de transactions dégelées dans une base de données. La `datfrozenxid` colonne de la `pg_database` ligne d'une base de données est la limite inférieure de la transaction normale qui IDs apparaît dans cette base de données. Cette colonne représente le minimum des valeurs `relfrozenxid` par table au sein de la base de données.

```
SELECT datname, age(datfrozenxid) FROM pg_database ORDER BY age(datfrozenxid) desc
 limit 20;
```

Par exemple, les résultats de l'exécution de la requête précédente pourraient être les suivants.

```
datname    | age
mydb       | 1771757888
template0  | 1721757888
template1  | 1721757888
rdsadmin   | 1694008527
postgres   | 1693881061
(5 rows)
```

Lorsque l'âge d'une base de données atteint 2 milliards de transactions IDs, l'ID de transaction (XID) est contourné et la base de données passe en lecture seule. Vous pouvez utiliser cette requête pour produire une métrique et l'exécuter plusieurs fois par jour. Par défaut, autovacuum est défini pour conserver un âge de transactions inférieur à 200,000,000 (`autovacuum_freeze_max_age`).

Un exemple de politique de surveillance peut ressembler à ceci :

- Définissez la valeur `autovacuum_freeze_max_age` sur 200 millions de transactions.

- Si une table atteint 500 millions de transactions débloquées, cela déclenche une alarme de faible gravité. Ce n'est pas une valeur déraisonnable, mais elle peut indiquer que la fonction d'autovacuum ne suit pas.
- Si l'âge d'une table atteint 1 milliard, cela doit être considéré comme une alarme exigeant une action. En général, il est conseillé de conserver des âges plus proches de `autovacuum_freeze_max_age` pour des raisons de performances. Nous vous recommandons d'enquêter en appliquant les recommandations suivantes.
- Si une table atteint les 1,5 million de transactions non vidées, elle déclenche une alarme de haute gravité. Selon la rapidité avec laquelle votre base de données utilise les transactions IDs, cette alarme peut indiquer que le système n'a plus le temps d'exécuter Autovacuum. Dans ce cas, nous vous recommandons une résolution immédiate.

Si une table enfreint constamment ces seuils, vous devez continuer à modifier vos paramètres d'autovacuum. Par défaut, l'utilisation manuelle de VACUUM (pour lequel les retards basés sur les coûts sont désactivés) est plus agressive que le processus d'autovacuum par défaut, mais elle est également plus intrusive pour le système dans son ensemble.

Nous vous recommandons la procédure suivante :

- Gardez tout cela à l'esprit et activez un mécanisme de surveillance afin de connaître l'âge de vos transactions les plus anciennes.

  Pour plus d'informations sur la création d'un processus qui vous avertit de l'encapsulation des identifiants de transaction, consultez le billet de blog consacré à la AWS base de données Implémentation d'un système d'alerte précoce pour l'encapsulation des identifiants de transaction dans Amazon RDS for PostgreSQL.
- Pour les tables plus occupées, procédez régulièrement au gel manuel du processus de vacuum pendant une fenêtre de maintenance, en plus de compter sur la fonction d'autovacuum. Pour plus d'informations sur le gel manuel du processus vacuum, veuillez consulter Réalisation d'un gel manuel du processus vacuum.

Déterminer les tables actuellement éligibles pour autovacuum

Souvent, une ou deux tables ont besoin d'une opération VACUUM. Les tables dont la valeur `relfrozenxid` est supérieure au nombre de transactions dans `autovacuum_freeze_max_age` sont toujours ciblées par la fonction d'autovacuum. Sinon, si le nombre de tuples devenus obsolètes depuis le dernier VACUUM dépasse le seuil de vide, la table est mise sous vide.

Le seuil d'autovacuum est défini comme suit :

```
Vacuum-threshold = vacuum-base-threshold + vacuum-scale-factor * number-of-tuples
```

où le `vacuum base threshold` estautovacuum_vacuum_threshold, l'`vacuum scale factor`est `autovacuum_vacuum_scale_factor` et l'`number of tuples`estpg_class.reltuples.

Pendant que vous êtes connecté à votre base de données, exécutez la requête suivante pour afficher la liste des tables qu'autovacuum considère comme éligibles pour une action vacuum.

```
WITH vbt AS (SELECT setting AS autovacuum_vacuum_threshold FROM
pg_settings WHERE name = 'autovacuum_vacuum_threshold'),
vsf AS (SELECT setting AS autovacuum_vacuum_scale_factor FROM
pg_settings WHERE name = 'autovacuum_vacuum_scale_factor'),
fma AS (SELECT setting AS autovacuum_freeze_max_age FROM pg_settings WHERE name =
 'autovacuum_freeze_max_age'),
sto AS (select opt_oid, split_part(setting, '=', 1) as param,
split_part(setting, '=', 2) as value from (select oid opt_oid, unnest(reloptions)
 setting from pg_class) opt)
SELECT '"'||ns.nspname||'"."'||c.relname||'"' as relation,
pg_size_pretty(pg_table_size(c.oid)) as table_size,
age(relfrozenxid) as xid_age,
coalesce(cfma.value::float, autovacuum_freeze_max_age::float)
 autovacuum_freeze_max_age,
(coalesce(cvbt.value::float, autovacuum_vacuum_threshold::float) +
coalesce(cvsf.value::float,autovacuum_vacuum_scale_factor::float) * c.reltuples)
AS autovacuum_vacuum_tuples, n_dead_tup as dead_tuples FROM
pg_class c join pg_namespace ns on ns.oid = c.relnamespace
join pg_stat_all_tables stat on stat.relid = c.oid join vbt on (1=1) join vsf on (1=1)
 join fma on (1=1)
left join sto cvbt on cvbt.param = 'autovacuum_vacuum_threshold' and c.oid =
 cvbt.opt_oid
left join sto cvsf on cvsf.param = 'autovacuum_vacuum_scale_factor' and c.oid =
 cvsf.opt_oid
left join sto cfma on cfma.param = 'autovacuum_freeze_max_age' and c.oid = cfma.opt_oid
WHERE c.relkind = 'r' and nspname <> 'pg_catalog'
AND (age(relfrozenxid) >= coalesce(cfma.value::float, autovacuum_freeze_max_age::float)
OR coalesce(cvbt.value::float, autovacuum_vacuum_threshold::float) +
coalesce(cvsf.value::float,autovacuum_vacuum_scale_factor::float) *
c.reltuples <= n_dead_tup)
ORDER BY age(relfrozenxid) DESC LIMIT 50;
```

Déterminer si autovacuum est en cours d'exécution et pour combien de temps

Si vous avez besoin de vider manuellement une table, vous devez déterminer si autovacuum est en cours d'exécution. Si c'est le cas, vous devrez peut-être ajuster les paramètres pour le rendre plus efficace, ou désactiver temporairement l'aspirateur automatique afin de pouvoir l'exécuter VACUUM manuellement.

Utilisez la requête suivante pour déterminer si autovacuum est en cours d'exécution, pendant combien de temps il a été en cours d'exécution et s'il est en attente sur une autre session.

```
SELECT datname, usename, pid, state, wait_event, current_timestamp - xact_start AS
 xact_runtime, query
FROM pg_stat_activity
WHERE upper(query) LIKE '%VACUUM%'
ORDER BY xact_start;
```

Après l'exécution de la requête, vous devez obtenir un résultat similaire à ce qui suit.

```
 datname | usename  |  pid  | state  | wait_event |      xact_runtime       | query
 --------+----------+-------+--------+------------+-------------------------
+-----------------------------------------------------------------------------------------------
 mydb    | rdsadmin | 16473 | active |            | 33 days 16:32:11.600656 |
 autovacuum: VACUUM ANALYZE public.mytable1 (to prevent wraparound)
 mydb    | rdsadmin | 22553 | active |            | 14 days 09:15:34.073141 |
 autovacuum: VACUUM ANALYZE public.mytable2 (to prevent wraparound)
 mydb    | rdsadmin | 41909 | active |            | 3 days 02:43:54.203349  |
 autovacuum: VACUUM ANALYZE public.mytable3
 mydb    | rdsadmin |   618 | active |            | 00:00:00                |
 SELECT datname, usename, pid, state, wait_event, current_timestamp - xact_start AS
 xact_runtime, query+
         |          |       |        |            |                         | FROM
 pg_stat_activity
           +
         |          |       |        |            |                         | WHERE
 query like '%VACUUM%'
          +
         |          |       |        |            |                         | ORDER BY
 xact_start;
       +
```

Plusieurs problèmes peuvent occasionner des longueurs d'exécution (plusieurs jours) d'une session autovacuum. Le problème le plus courant est que la valeur de votre paramètre `maintenance_work_mem` est trop basse pour la taille de la table ou pour la fréquence des mises à jour.

Nous vous recommandons d'utiliser la formule suivante pour définir la valeur du paramètre `maintenance_work_mem`.

```
GREATEST({DBInstanceClassMemory/63963136*1024},65536)
```

De courtes sessions autovacuum peuvent également indiquer des problèmes :

- Cela peut indiquer un nombre `autovacuum_max_workers` insuffisant pour votre charge de travail. Dans ce cas, vous devez indiquer le nombre d'exécutants.
- Cela peut indiquer une corruption d'index (la fonction d'autovacuum se bloque et redémarre sur la même relation, mais ne progresse pas). Dans ce cas, exécutez un `vacuum freeze verbose` *`table`* manuel pour voir la cause exacte.

Réalisation d'un gel manuel du processus vacuum

Vous pouvez effectuer un gel manuel sur une table pour laquelle un processus vacuum est déjà en cours. C'est utile si vous avez identifié une table avec un âge proche de 2 milliards de transactions (ou supérieur à tous les seuils que vous surveillez).

Les étapes suivantes sont fournies à titre informatif et il existe plusieurs variantes de ce processus. Par exemple, pendant le test, supposons que vous trouviez que la valeur du paramètre `maintenance_work_mem` a été définie trop bas et que vous devez agir immédiatement sur une table. Toutefois, vous ne voulez pas renvoyer l'instance à l'expéditeur pour le moment. À l'aide des requêtes des sections précédentes, vous déterminez quelle table pose problème et remarquez une session autovacuum en cours d'exécution depuis longtemps. Vous savez que vous devez modifier le paramètre `maintenance_work_mem`, mais vous devez également agir immédiatement et effectuer un processus vacuum sur la table concernée. La procédure suivante montre ce que vous devez faire dans cette situation.

Pour procéder manuellement au gel du processus vacuum

1. Ouvrez les deux sessions de la base de données contenant la table sur laquelle vous voulez effectuer le processus vacuum. Pour la seconde session, utilisez « écran » ou un autre utilitaire qui gère la session si votre connexion est abandonnée.
2. Dans la première session, obtenez le PID de la session autovacuum en cours d'exécution sur la table.

   Exécutez la requête suivante pour obtenir le PID de la session autovacuum.

   ```
   SELECT datname, usename, pid, current_timestamp - xact_start
   AS xact_runtime, query
   FROM pg_stat_activity WHERE upper(query) LIKE '%VACUUM%' ORDER BY
   xact_start;
   ```
3. Dans la deuxième session, calculez la quantité de mémoire dont vous avez besoin pour cette opération. Dans cet exemple, nous déterminons que nous pouvons nous permettre d'utiliser jusqu'à 2 Go de mémoire pour cette opération. Nous affectons donc 2 Go à `maintenance_work_mem` pour la session en cours.

   ```
   SET maintenance_work_mem='2 GB';
   SET
   ```
4. Dans la deuxième session, émettez une commande `vacuum freeze verbose` pour la table. Le paramètre de mode détaillé est utile, car il vous permet de voir l'activité bien qu'il n'existe actuellement aucun rapport d'avancement de cette opération dans PostgreSQL.

   ```
   \timing on
   Timing is on.
   vacuum freeze verbose pgbench_branches;
   ```

   ```
   INFO:  vacuuming "public.pgbench_branches"
   INFO:  index "pgbench_branches_pkey" now contains 50 row versions in 2 pages
   DETAIL:  0 index row versions were removed.
   0 index pages have been deleted, 0 are currently reusable.
   CPU 0.00s/0.00u sec elapsed 0.00 sec.
   INFO:  index "pgbench_branches_test_index" now contains 50 row versions in 2 pages
   DETAIL:  0 index row versions were removed.
   0 index pages have been deleted, 0 are currently reusable.
   CPU 0.00s/0.00u sec elapsed 0.00 sec.
   ```

```
INFO:  "pgbench_branches": found 0 removable, 50 nonremovable row versions
     in 43 out of 43 pages
DETAIL:  0 dead row versions cannot be removed yet.
There were 9347 unused item pointers.
0 pages are entirely empty.
CPU 0.00s/0.00u sec elapsed 0.00 sec.
VACUUM
Time: 2.765 ms
```

5. Au cours de la première session, si l'autovacuum bloquait la session d'aspiration, cela `pg_stat_activity` indique que votre session T d'aspiration est en attente. Dans ce cas, mettez fin au processus d'aspiration automatique comme suit.

```
SELECT pg_terminate_backend('the_pid');
```

> **Note**
>
> Certaines versions inférieures d'Amazon RDS Amazon ne peuvent pas mettre fin à un processus d'aspiration automatique à l'aide de la commande précédente et échouent avec l'erreur suivante :. `ERROR: 42501: must be a superuser to terminate superuser process LOCATION: pg_terminate_backend, signalfuncs.c:227`

À ce stade, votre session commence. Autovacuum redémarre immédiatement car cette table est probablement la plus haute de sa liste de travail.

6. Lancez votre commande `vacuum freeze verbose` dans la session 2, puis terminez le processus autovacuum de la session 1.

Réindexation d'une table pendant l'exécution du processus autovacuum

Si un index a été corrompu, la fonction d'autovacuum continue à traiter la table et échoue. Si vous essayez d'effectuer un processus vacuum manuel dans cette situation, vous recevez un message d'erreur similaire à ce qui suit.

```
postgres=>  vacuum freeze pgbench_branches;
ERROR: index "pgbench_branches_test_index" contains unexpected
   zero page at block 30521
```

```
HINT: Please REINDEX it.
```

Lorsque l'index est corrompu et que la fonction d'autovacuum tente de s'exécuter sur la table, vous vous heurtez à une session autovacuum déjà en cours d'exécution. Lorsque vous émettez une commande REINDEX, vous retirez un verrou exclusif sur la table. Les opérations d'écriture sont bloquées, ainsi que les opérations de lecture qui utilisent cet index spécifique.

Pour réindexer une table lorsque la fonction d'autovacuum est en cours d'exécution sur la table

1. Ouvrez les deux sessions de la base de données contenant la table sur laquelle vous voulez effectuer le processus vacuum. Pour la seconde session, utilisez « écran » ou un autre utilitaire qui gère la session si votre connexion est abandonnée.
2. Dans la première session, obtenez le PID de la session autovacuum en cours d'exécution sur la table.

   Exécutez la requête suivante pour obtenir le PID de la session autovacuum.

   ```
   SELECT datname, usename, pid, current_timestamp - xact_start
   AS xact_runtime, query
   FROM pg_stat_activity WHERE upper(query) like '%VACUUM%' ORDER BY
   xact_start;
   ```

3. Dans la deuxième session, émettez la commande reindex.

   ```
   \timing on
   Timing is on.
   reindex index pgbench_branches_test_index;
   REINDEX
     Time: 9.966 ms
   ```

4. Dans la première session, si autovacuum provoquait un blocage du processus, vous voyez dans `pg_stat_activity` que l'attente a la valeur « T » pour votre session vacuum. Dans ce cas, vous mettrez fin au processus autovacuum.

   ```
   SELECT pg_terminate_backend('the_pid');
   ```

   À ce stade, votre session commence. Il est important de noter que la fonction d'autovacuum redémarre immédiatement parce que cette table figure probablement tout en haut de sa liste de tâches.

5. Lancez votre commande dans la session 2, puis terminez le processus autovacuum de la session 1.

Gestion de la fonction autovacuum avec de grands index

Dans le cadre de son fonctionnement, la fonction autovacuum effectue plusieurs phases de mise à vide lorsqu'elle s'exécute sur une table. Avant que la table ne soit nettoyée, tous ses index sont d'abord vidés. Lorsque vous supprimez plusieurs grands index, cette phase consomme beaucoup de temps et de ressources. Par conséquent, il est recommandé de contrôler le nombre d'index d'une table et d'éliminer les index inutilisés.

Pour ce processus, vérifiez d'abord la taille globale de l'index. Déterminez ensuite s'il existe des index potentiellement inutilisés qui peuvent être supprimés comme le montrent les exemples suivants.

Pour vérifier la taille de la table et de ses index

```
postgres=> select pg_size_pretty(pg_relation_size('pgbench_accounts'));
pg_size_pretty
6404 MB
(1 row)
```

```
postgres=> select pg_size_pretty(pg_indexes_size('pgbench_accounts'));
pg_size_pretty
11 GB
(1 row)
```

Dans cet exemple, la taille des index est supérieure à celle de la table. Cette différence peut entraîner des problèmes de performances car les index sont surchargés ou inutilisés, ce qui a une incidence sur la fonction autovacuum ainsi que sur les opérations d'insertion.

Pour vérifier la présence d'index inutilisés

À l'aide de la vue `pg_stat_user_indexes`, vous pouvez vérifier la fréquence d'utilisation d'un index avec la colonne `idx_scan`. Dans l'exemple suivant, les index non utilisés ont la valeur `idx_scan` définie sur `0`.

```
postgres=> select * from pg_stat_user_indexes where relname = 'pgbench_accounts' order
 by idx_scan desc;
```

```
relid  | indexrelid | schemaname | relname          | indexrelname          | idx_scan
 | idx_tup_read | idx_tup_fetch
-------+------------+------------+------------------+-----------------------+----------
+--------------+--------------
16433  | 16454      | public     | pgbench_accounts | index_f               | 6
 | 6            | 0
16433  | 16450      | public     | pgbench_accounts | index_b               | 3
 | 199999       | 0
16433  | 16447      | public     | pgbench_accounts | pgbench_accounts_pkey | 0
 | 0            | 0
16433  | 16452      | public     | pgbench_accounts | index_d               | 0
 | 0            | 0
16433  | 16453      | public     | pgbench_accounts | index_e               | 0
 | 0            | 0
16433  | 16451      | public     | pgbench_accounts | index_c               | 0
 | 0            | 0
16433  | 16449      | public     | pgbench_accounts | index_a               | 0
 | 0            | 0
(7 rows)
```

```
postgres=> select schemaname, relname, indexrelname, idx_scan from pg_stat_user_indexes
 where relname = 'pgbench_accounts' order by idx_scan desc;

schemaname  | relname          | indexrelname          | idx_scan
------------+------------------+-----------------------+----------
public      | pgbench_accounts | index_f               | 6
public      | pgbench_accounts | index_b               | 3
public      | pgbench_accounts | pgbench_accounts_pkey | 0
public      | pgbench_accounts | index_d               | 0
public      | pgbench_accounts | index_e               | 0
public      | pgbench_accounts | index_c               | 0
public      | pgbench_accounts | index_a               | 0
(7 rows)
```

Note

Ces statistiques sont incrémentielles à partir du moment où elles sont réinitialisées. Supposons que vous disposiez d'un index qui n'est utilisé qu'à la fin d'un trimestre ou uniquement pour un rapport spécifique. Il est possible que cet index n'ait pas été utilisé depuis la réinitialisation des statistiques. Pour plus d'informations, consultez Statistics

Functions (Fonctions statistiques). Les index utilisés pour renforcer l'unicité ne seront pas analysés et ne devraient pas être identifiés comme des index inutilisés. Pour identifier les index inutilisés, vous devez avoir une connaissance approfondie de l'application et de ses requêtes.

Pour vérifier quand les statistiques ont été réinitialisées pour la dernière fois pour une base de données, utilisez pg_stat_database

```
postgres=> select datname, stats_reset from pg_stat_database where datname =
 'postgres';

datname   | stats_reset
----------+-------------------------------
postgres  | 2022-11-17 08:58:11.427224+00
(1 row)
```

Mise à vide d'une table le plus rapidement possible

RDS for PostgreSQL versions 12 et ultérieures

Si vous avez trop d'index dans une grande table, il se peut que votre instance de base de données soit proche du renvoi à la ligne de l'ID de transaction (XID), c'est-à-dire lorsque le compteur XID revient à zéro. Si elle n'est pas vérifiée, cette situation peut entraîner une perte de données. Toutefois, vous pouvez rapidement vider la table sans nettoyer les index. Dans RDS for PostgreSQL versions 12 et ultérieures, vous pouvez utiliser VACUUM avec la clause INDEX_CLEANUP.

```
postgres=> VACUUM (INDEX_CLEANUP FALSE, VERBOSE TRUE) pgbench_accounts;

INFO: vacuuming "public.pgbench_accounts"
INFO: table "pgbench_accounts": found 0 removable, 8 nonremovable row versions in 1 out
 of 819673 pages
DETAIL: 0 dead row versions cannot be removed yet, oldest xmin: 7517
Skipped 0 pages due to buffer pins, 0 frozen pages.
CPU: user: 0.01 s, system: 0.00 s, elapsed: 0.01 s.
```

Si une session de mise à vide automatique est déjà en cours, vous devez y mettre fin pour démarrer le processus VACUUM manuel. Pour plus d'informations sur le gel manuel du processus vacuum, consultez Réalisation d'un gel manuel du processus vacuum.

> **Note**
>
> Le fait d'ignorer régulièrement le nettoyage de l'index entraîne un gonflement de l'index, ce qui dégrade les performances d'analyse. L'index conserve les lignes mortes et le tableau conserve les pointeurs de ligne morte. Par conséquent, il `pg_stat_all_tables.n_dead_tup` augmente jusqu'à ce qu'un aspirateur automatique ou un ASPIRATEUR manuel avec nettoyage d'index s'exécute. Il est recommandé d'utiliser cette procédure uniquement pour empêcher l'encapsulation des identifiants de transaction.

RDS for PostgreSQL versions 11 et ultérieures

Toutefois, dans RDS for PostgreSQL versions 11 et ultérieures, la seule façon de permettre au processus vacuum de se terminer plus rapidement est de réduire le nombre d'index sur une table. La suppression d'un index peut affecter les plans de requête. Nous vous recommandons de supprimer d'abord les index inutilisés, puis de les supprimer lorsque le renvoi à la ligne de l'ID de transaction est très proche. Une fois le processus vacuum terminé, vous pouvez recréer ces index.

Autres paramètres qui affectent la fonction d'autovacuum

Cette requête affiche les valeurs de certains des paramètres qui ont un impact direct sur la fonction d'autovacuum et son comportement. Les paramètres d'autovacuum sont décrits en détails dans la documentation PostgreSQL.

```
SELECT name, setting, unit, short_desc
FROM pg_settings
WHERE name IN (
'autovacuum_max_workers',
'autovacuum_analyze_scale_factor',
'autovacuum_naptime',
'autovacuum_analyze_threshold',
'autovacuum_analyze_scale_factor',
'autovacuum_vacuum_threshold',
'autovacuum_vacuum_scale_factor',
'autovacuum_vacuum_threshold',
'autovacuum_vacuum_cost_delay',
'autovacuum_vacuum_cost_limit',
'vacuum_cost_limit',
'autovacuum_freeze_max_age',
'maintenance_work_mem',
```

```
'vacuum_freeze_min_age');
```

Tous ces paramètres affectent la fonction d'autovacuum, mais les plus importants sont :

- maintenance_work_mem
- autovacuum_freeze_max_age
- autovacuum_max_workers
- autovacuum_vacuum_cost_delay
- autovacuum_vacuum_cost_limit

Définition des paramètres d'autovacuum au niveau de la table

Les paramètres de stockage liés à la fonction d'autovacuum peuvent être définis au niveau de la table, ce qui peut être plus judicieux que de modifier le comportement de toute la base de données. Pour les grandes tables, vous devrez peut-être définir des paramètres agressifs et il est déconseillé de faire en sorte que la fonction d'autovacuum se comporte de cette manière pour toutes les tables.

La requête suivante affiche les tables qui ont actuellement des options en place au niveau de la table.

```
SELECT relname, reloptions
FROM pg_class
WHERE reloptions IS NOT null;
```

Par exemple, cela peut être utile sur les tables qui sont beaucoup plus grandes que le reste de vos tables. Supposez que vous avez une table de 300 Go et 30 autres tables de moins de 1 Go. Dans ce cas, vous pouvez définir des paramètres spécifiques pour votre grande table afin de ne pas modifier le comportement de l'intégralité de votre système.

```
ALTER TABLE mytable set (autovacuum_vacuum_cost_delay=0);
```

Cela permet de désactiver le retard d'autovacuum basé sur les coûts pour cette table au détriment d'une plus grande utilisation des ressources sur votre système. Normalement, l'autovacuum s'arrête pour `autovacuum_vacuum_cost_delay` à chaque fois que `autovacuum_cost_limit` est atteinte. Pour plus de détails, consultez la SQL documentation Postgre sur l'aspiration basée sur les coûts.

Enregistrement des activités d'autovacuum et de vacuum

Les informations sur les activités d'autovacuum sont envoyées au `postgresql.log` en fonction du niveau spécifié dans le paramètre `rds.force_autovacuum_logging_level`. Voici les valeurs autorisées pour ce paramètre et les SQL versions de Postgre pour lesquelles cette valeur est le paramètre par défaut :

- `disabled`(Postger SQL 10, Postger 9.6) SQL
- `debug5`, `debug4`, `debug3`, `debug2`, `debug1`
- `info`(Poster SQL 12, Poster 11) SQL
- `notice`
- `warning`(Postgre SQL 13 et versions ultérieures)
- `error`, journal, `fatal`, `panic`

`rds.force_autovacuum_logging_level` fonctionne avec le paramètre `log_autovacuum_min_duration`. La valeur du paramètre `log_autovacuum_min_duration` est le seuil (en millisecondes) au-dessus duquel les actions autovacuum sont enregistrées. Une valeur de `-1` n'enregistre rien, tandis qu'une valeur de 0 enregistre toutes les actions. Comme avec `rds.force_autovacuum_logging_level`, valeurs par défaut pour `log_autovacuum_min_duration` dépendent de la version, comme suit :

- `10000 ms`— Postgre SQL 14, Postgre SQL 13, Postgre SQL 12 et Postgre 11 SQL
- `(empty)`— Aucune valeur par défaut pour Postgre SQL 10 et SQL Postgre 9.6

Nous vous recommandons de définir le `rds.force_autovacuum_logging_level` à la valeur `WARNING`. Nous vous recommandons également de définir `log_autovacuum_min_duration` à une valeur comprise entre 1000 et 5000. Un paramètre de 5000 journaux d'activité qui prend plus de 5000 millisecondes. Tout paramètre autre que -1 enregistre également les messages si l'action autovacuum est ignorée en raison d'un verrouillage en conflit ou d'une perte simultanée de relations. Pour plus d'informations, consultez la section Aspiration automatique dans la documentation de SQL Postgre.

Pour résoudre les problèmes, vous pouvez modifier le paramètre `rds.force_autovacuum_logging_level` à l'un des niveaux de débogage, de `debug1` jusqu'à `debug5` pour obtenir les informations les plus détaillées. Nous vous recommandons d'utiliser les

paramètres de débogage pendant de courtes périodes et à des fins de dépannage uniquement. Pour en savoir plus, consultez Quand se connecter dans la SQL documentation Postgre.

> **Note**
>
> Postgre SQL permet au `rds_superuser` compte de visualiser les sessions Autovacuum dans. `pg_stat_activity` Par exemple, vous pouvez identifier et mettre fin à la session qui bloque l'exécution d'une commande ou empêche la commande de s'exécuter plus lentement qu'une commande vacuum exécutée manuellement.

Comprendre le comportement de l'autovacuum avec des bases de données non valides

Une nouvelle valeur `-2` est introduite dans la `datconnlimit` colonne du `pg_database` catalogue pour indiquer que les bases de données qui ont été interrompues au milieu de l'opération DROP DATABASE ne sont pas valides.

Cette nouvelle valeur est disponible dans les versions suivantes d' RDS pour PostgreSQL :

- 15.4 et toutes les versions supérieures
- Version 14.9 et versions ultérieures
- Versions 13.12 et supérieures
- Versions 12.16 et supérieures
- Versions 11.21 et supérieures

Les bases de données non valides n'affectent pas la capacité d'Autovacuum à bloquer les fonctionnalités des bases de données valides. Autovacuum ignore les bases de données non valides. Par conséquent, les opérations de vide régulières continueront à fonctionner correctement et efficacement pour toutes les bases de données valides de votre environnement PostgreSQL.

Rubriques

- Surveillance de l'identifiant de transaction
- Ajustement de la requête de surveillance
- Résolution du problème de base de données non valide

Surveillance de l'identifiant de transaction

Cette `age(datfrozenxid)` fonction est couramment utilisée pour surveiller l'âge des identifiants de transaction (XID) des bases de données afin d'empêcher l'encapsulation des identifiants de transaction.

Les bases de données non valides étant exclues d'Autovacuum, leur compteur d'ID de transaction (XID) peut atteindre la valeur maximale de`2 billion`, contourner et poursuivre ce cycle indéfiniment. `- 2 billion` Une requête typique pour surveiller l'encapsulation des identifiants de transaction peut ressembler à ceci :

```
SELECT max(age(datfrozenxid)) FROM pg_database;
```

Cependant, avec l'introduction de la valeur -2 pour`datconnlimit`, les bases de données non valides peuvent fausser les résultats de cette requête. Étant donné que ces bases de données ne sont pas valides et ne doivent pas faire l'objet de contrôles de maintenance réguliers, elles peuvent générer des faux positifs, ce qui vous laisse penser que le taux `age(datfrozenxid)` est supérieur à ce qu'il est réellement.

Ajustement de la requête de surveillance

Pour garantir une surveillance précise, vous devez ajuster votre requête de surveillance afin d'exclure les bases de données non valides. Suivez cette requête recommandée :

```
SELECT
    max(age(datfrozenxid))
FROM
    pg_database
WHERE
    datconnlimit <> -2;
```

Cette requête garantit que seules les bases de données valides sont prises en compte dans le `age(datfrozenxid)` calcul, ce qui reflète fidèlement l'âge des identifiants de transaction dans votre environnement PostgreSQL.

Résolution du problème de base de données non valide

Lorsque vous tentez de vous connecter à une base de données non valide, un message d'erreur semblable au suivant peut s'afficher :

```
postgres=> \c db1
connection to server at "mydb.xxxxxxxxxx.us-west-2.rds.amazonaws.com" (xx.xx.xx.xxx),
 port xxxx failed: FATAL:  cannot connect to invalid database "db1"
HINT:  Use DROP DATABASE to drop invalid databases.
Previous connection kept
```

En outre, si le `log_min_messages` paramètre est défini sur `DEBUG2` ou supérieur, vous remarquerez peut-être que les entrées de journal suivantes indiquent que le processus d'aspiration automatique ignore la base de données non valide :

```
2024-07-30 05:59:00 UTC::@:[32000]:DEBUG:  autovacuum: skipping invalid database "db6"
2024-07-30 05:59:00 UTC::@:[32000]:DEBUG:  autovacuum: skipping invalid database "db1"
```

Pour résoudre le problème, suivez les instructions `HINT` fournies lors de la tentative de connexion. Connectez-vous à n'importe quelle base de données valide à l'aide de votre compte principal RDS ou d'un compte de base de données doté du `rds_superuser` rôle, et supprimez les bases de données non valides.

```
SELECT
    'DROP DATABASE ' || quote_ident(datname) || ';'
FROM
    pg_database
WHERE
    datconnlimit = -2 \gexec
```

Dans PostgreSQL, la mise sous vide est essentielle pour garantir l'intégrité de la base de données, car elle permet de récupérer de l'espace de stockage et d'éviter les problèmes d'encapsulation des identifiants de transaction. Cependant, il peut arriver que l'aspirateur ne fonctionne pas comme vous le souhaitez, ce qui peut entraîner une dégradation des performances, une surcharge de stockage et même avoir un impact sur la disponibilité de votre instance de base de données par encapsulation des identifiants de transaction. Il est donc essentiel d'identifier et de résoudre ces

problèmes pour optimiser les performances et la disponibilité des bases de données. Consultez Comprendre l'autovacuum dans les environnements Amazon RDS for PostgreSQL pour en savoir plus sur l'autovacuum.

Cette `postgres_get_av_diag()` fonction permet d'identifier les problèmes qui empêchent ou retardent la progression de l'aspirateur agressif. Des suggestions sont fournies, qui peuvent inclure des commandes pour résoudre le problème lorsqu'il est identifiable ou des conseils pour des diagnostics supplémentaires lorsque le problème n'est pas identifiable. Les bloqueurs d'aspiration agressifs sont signalés lorsque leur âge dépasse le seuil d'autoaspiration adaptatif de 500 millions de transactions fixé par RDS. IDs

Quel est l'âge de l'identifiant de transaction ?

La `age()` fonction de transaction IDs calcule le nombre de transactions survenues depuis le plus ancien identifiant de transaction dégelé pour une base de données (`pg_database.datfrozenxid`) ou une table (`pg_class.relfrozenxid`). Cette valeur indique l'activité de la base de données depuis la dernière opération de vide agressif et met en évidence la charge de travail probable pour les prochains processus VACUUM.

Qu'est-ce qu'un aspirateur agressif ?

Une opération VACUUM agressive effectue une numérisation complète de toutes les pages d'un tableau, y compris celles qui sont généralement ignorées en mode normal VACUUMs. Cette analyse approfondie vise à « geler » les transactions IDs qui approchent de leur âge maximum, évitant ainsi une situation connue sous le nom d'encapsulation des identifiants de transaction.

`postgres_get_av_diag()`Pour signaler un bloqueur, celui-ci doit dater d'au moins 500 millions de transactions.

Rubriques

-
-
-
-
-
-

La `postgres_get_av_diag()` fonction est actuellement disponible dans les versions RDS pour PostgreSQL PostgreSQL suivantes :

- 17.2 et versions supérieures 17
- 16.7 et versions ultérieures 16
- 15.11 et versions ultérieures 15
- Versions 14.16 et supérieures 14
- 13.19 et versions ultérieures 13

Pour pouvoir l'utiliser`postgres_get_av_diag()`, créez l'`rds_tools`extension.

```
postgres=> CREATE EXTENSION rds_tools ;
CREATE EXTENSION
```

Vérifiez que l'extension est installée.

```
postgres=> \dx rds_tools
             List of installed extensions
   Name    | Version |  Schema   |                    Description
 ----------+---------+-----------
+------------------------------------------------------------
 rds_tools |   1.8   | rds_tools | miscellaneous administrative functions for RDS
 PostgreSQL
 1 row
```

Vérifiez que la fonction est créée.

```
postgres=> SELECT
    proname function_name,
    pronamespace::regnamespace function_schema,
    proowner::regrole function_owner
FROM
    pg_proc
WHERE
    proname = 'postgres_get_av_diag';
    function_name     | function_schema | function_owner
----------------------+-----------------+----------------
 postgres_get_av_diag | rds_tools       | rds_superuser
(1 row)
```

La requête doit être exécutée dans la base de données avec l'identifiant de transaction le plus ancien pour obtenir des résultats précis. Pour plus d'informations sur l'utilisation de la base de données avec l'identifiant de transaction le plus ancien, voir Non connecté à la base de données ayant l'identifiant de transaction le plus ancien

```
SELECT
    blocker,
    DATABASE,
    blocker_identifier,
    wait_event,
    TO_CHAR(autovacuum_lagging_by, 'FM9,999,999,999') AS autovacuum_lagging_by,
    suggestion,
    suggested_action
FROM (
    SELECT
        *
    FROM
        rds_tools.postgres_get_av_diag ()
    ORDER BY
        autovacuum_lagging_by DESC) q;
```

La `postgres_get_av_diag()` fonction renvoie un tableau contenant les informations suivantes :

bloqueur

Spécifie la catégorie d'activité de base de données qui bloque le vide.

- Déclaration active
- État Idle in transaction (Transaction inactive)
- Transaction préparée
- Emplacement de réplication logique
- Réplique de lecture avec emplacement de réplication physique
- Réplique de lecture avec réplication en continu
- Tables temporaires

database

Spécifie le nom de la base de données, le cas échéant et si elle est prise en charge. Il s'agit de la base de données dans laquelle l'activité est en cours et bloque ou bloquera l'autovacuum. Il s'agit de la base de données à laquelle vous devez vous connecter et agir.

identifiant_bloqueur

Spécifie l'identifiant de l'activité qui bloque ou bloquera l'autovacuum. L'identifiant peut être un identifiant de processus accompagné d'une instruction SQL, d'une transaction préparée, de l'adresse IP d'une réplique lue et du nom du slot de réplication, logique ou physique.

événement d'attente

Spécifie l'événement d' de la session de blocage et s'applique aux bloqueurs suivants :

- Déclaration active
- État Idle in transaction (Transaction inactive)

autovacum_lagging_by

Spécifie le nombre de transactions pour lesquelles Autovacuum est en retard dans son carnet de travail par catégorie.

suggestion

Spécifie des suggestions pour résoudre le bloqueur. Ces instructions incluent le nom de la base de données dans laquelle l'activité existe, le cas échéant, le numéro de processus (PID) de la session, le cas échéant, et les mesures à prendre.

action_suggérée

Suggère l'action à entreprendre pour résoudre le bloqueur.

Autovacuum effectue des aspirations agressives et abaisse l'âge de la transaction IDs en dessous du seuil spécifié par le `autovacuum_freeze_max_age` paramètre de votre instance RDS. Vous pouvez suivre cet âge à l'aide de la CloudWatch métrique Amazon`MaximumUsedTransactionIDs`.

Pour trouver le paramètre de `autovacuum_freeze_max_age` (dont la valeur par défaut est de 200 millions de transactions IDs) pour votre instance Amazon RDS, vous pouvez utiliser la requête suivante :

```
SELECT
    TO_CHAR(setting::bigint, 'FM9,999,999,999') autovacuum_freeze_max_age
FROM
    pg_settings
WHERE
```

```
    name = 'autovacuum_freeze_max_age';
```

Notez que les bloqueurs d'aspiration agressifs `postgres_get_av_diag()` ne sont vérifiés que lorsque leur âge dépasse le seuil de 500 millions de transactions d'aspiration automatique adaptatif d'Amazon RDS. IDs `postgres_get_av_diag()`Pour détecter les bloqueurs, ceux-ci doivent dater d'au moins 500 millions de transactions.

La `postgres_get_av_diag()` fonction identifie les types de bloqueurs suivants :

Rubriques

- Déclaration active
- État Idle in transaction (Transaction inactive)
- Transaction préparée
- Emplacement de réplication logique
- Réplicas en lecture
- Tables temporaires

Déclaration active

Dans PostgreSQL, une instruction active est une instruction SQL en cours d'exécution par la base de données. Cela inclut les requêtes, les transactions ou toutes les opérations en cours. Lors de la surveillance via`pg_stat_activity`, la colonne d'état indique que le processus avec le PID correspondant est actif.

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle identifie une instruction active.

```
blocker               | Active statement
database              | my_database
blocker_identifier    | SELECT pg_sleep(20000);
wait_event            | Timeout:PgSleep
autovacuum_lagging_by | 568,600,871
suggestion            | Connect to database "my_database", review carefully and you
 may consider terminating the process using suggested_action. For more information, see
 Working with PostgreSQL autovacuum in the Amazon RDS User Guide.
suggested_action      | {"SELECT pg_terminate_backend (29621);"}
```

Action suggérée

En suivant les instructions de la `suggestion` colonne, l'utilisateur peut se connecter à la base de données dans laquelle l'instruction active est présente et, comme indiqué dans la `suggested_action` colonne, il est conseillé de revoir attentivement l'option permettant de mettre fin à la session. Si la résiliation est sûre, vous pouvez utiliser la `pg_terminate_backend()` fonction pour mettre fin à la session. Cette action peut être effectuée par un administrateur (tel que le compte principal RDS) ou par un utilisateur disposant des `pg_terminate_backend()` privilèges requis.

> **Warning**
>
> Une session interrompue annulera (`ROLLBACK`) les modifications apportées. En fonction de vos besoins, vous souhaiterez peut-être exécuter à nouveau le relevé. Cependant, il est recommandé de ne le faire qu'une fois que le processus d'aspiration automatique a terminé son opération de vide agressif.

État Idle in transaction (Transaction inactive)

Une instruction idle in transaction fait référence à toute session qui a ouvert une transaction explicite (par exemple en émettant une `BEGIN` instruction), effectué un certain travail et attendant maintenant que le client passe plus de travail ou signale la fin de la transaction en émettant un `COMMITROLLBACK`, ou `END` (ce qui se traduirait par un implicite`COMMIT`).

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle identifie une `idle in transaction` instruction en tant que bloqueur.

```
blocker               | idle in transaction
database              | my_database
blocker_identifier    | INSERT INTO tt SELECT * FROM tt;
wait_event            | Client:ClientRead
autovacuum_lagging_by | 1,237,201,759
suggestion            | Connect to database "my_database", review carefully and you
 may consider terminating the process using suggested_action. For more information, see
 Working with PostgreSQL autovacuum in the Amazon RDS User Guide.
suggested_action      | {"SELECT pg_terminate_backend (28438);"}
```

Action suggérée

Comme indiqué dans la `suggestion` colonne, vous pouvez vous connecter à la base de données dans laquelle la session de transaction inactive est présente et terminer la session à l'aide de cette

`pg_terminate_backend()` fonction. L'utilisateur peut être votre administrateur (compte principal RDS) ou un utilisateur disposant du `pg_terminate_backend()` privilège.

> **Warning**
>
> Une session interrompue annulera (`ROLLBACK`) les modifications apportées. En fonction de vos besoins, vous souhaiterez peut-être exécuter à nouveau le relevé. Cependant, il est recommandé de ne le faire qu'une fois que le processus d'aspiration automatique a terminé son opération de vide agressif.

Transaction préparée

PostgreSQL autorise les transactions qui font partie d'une stratégie de validation en deux phases appelée transactions préparées. Elles sont activées en réglant le `max_prepared_transactions` paramètre sur une valeur différente de zéro. Les transactions préparées sont conçues pour garantir qu'une transaction est durable et reste disponible même après une panne de base de données, un redémarrage ou une déconnexion du client. Comme les transactions ordinaires, un identifiant de transaction leur est attribué et peuvent affecter l'autovacuum. S'il est laissé dans un état préparé, l'autovacuum ne peut pas effectuer de congélation et peut entraîner l'encapsulation de l'identifiant de transaction.

Lorsque les transactions sont laissées préparées indéfiniment sans être résolues par un gestionnaire de transactions, elles deviennent des transactions préparées orphelines. La seule façon de résoudre ce problème est de valider ou d'annuler la transaction à l'aide des `ROLLBACK PREPARED` commandes `COMMIT PREPARED` or, respectivement.

> **Note**
>
> Sachez qu'une sauvegarde effectuée lors d'une transaction préparée contiendra toujours cette transaction après restauration. Reportez-vous aux informations suivantes pour savoir comment localiser et clôturer de telles transactions.

La `postgres_get_av_diag()` fonction affiche le résultat suivant lorsqu'elle identifie un bloqueur qui est une transaction préparée.

```
blocker             | Prepared transaction
database            | my_database
```

```
blocker_identifier    | myptx
wait_event            | Not applicable
autovacuum_lagging_by | 1,805,802,632
suggestion            | Connect to database "my_database" and consider either COMMIT
 or ROLLBACK the prepared transaction using suggested_action. For more information, see
 Working with PostgreSQL autovacuum in the Amazon RDS User Guide.
suggested_action      | {"COMMIT PREPARED 'myptx';",[OR],"ROLLBACK PREPARED 'myptx';"}
```

Action suggérée

Comme indiqué dans la colonne de suggestions, connectez-vous à la base de données où se trouve la transaction préparée. Sur la base de la `suggested_action` colonne, examinez attentivement s'il faut exécuter l'`COMMIT`une ou l'autre`ROLLBACK`, et quelle est l'action appropriée.

Pour surveiller les transactions préparées en général, PostgreSQL propose une vue de catalogue appelée. `pg_prepared_xacts` Vous pouvez utiliser la requête suivante pour trouver les transactions préparées.

```
SELECT
    gid,
    prepared,
    owner,
    database,
    transaction AS oldest_xmin
FROM
    pg_prepared_xacts
ORDER BY
    age(transaction) DESC;
```

Emplacement de réplication logique

L'objectif d'un slot de réplication est de conserver les modifications non consommées jusqu'à ce qu'elles soient répliquées sur un serveur cible. Pour plus d'informations, consultez la section Réplication logique de PostgreSQL.

Il existe deux types de slots de réplication logiques.

Emplacements de réplication logiques inactifs

Lorsque la réplication est terminée, les journaux de transactions non consommés ne peuvent pas être supprimés et le slot de réplication devient inactif. Bien qu'aucun emplacement de réplication logique inactif ne soit actuellement utilisé par un abonné, il reste sur le serveur, ce qui entraîne la

rétention des fichiers WAL et empêche la suppression des anciens journaux de transactions. Cela peut augmenter l'utilisation du disque et empêcher spécifiquement Autovacuum de nettoyer les tables de catalogue internes, car le système doit empêcher le remplacement des informations LSN. Si rien n'est fait, cela peut entraîner une saturation du catalogue, une dégradation des performances et un risque accru de vide global, ce qui peut entraîner une interruption des transactions.

Emplacements de réplication logiques actifs mais lents

Parfois, la suppression des tuples morts du catalogue est retardée en raison de la dégradation des performances de la réplication logique. Ce retard de réplication ralentit la mise à jour du `catalog_xmin` et peut entraîner un gonflement du catalogue et un vide général.

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle trouve un emplacement de réplication logique en tant que bloqueur.

```
blocker               | Logical replication slot
database              | my_database
blocker_identifier    | slot1
wait_event            | Not applicable
autovacuum_lagging_by | 1,940,103,068
suggestion            | Ensure replication is active and resolve any lag for the slot
 if active. If inactive, consider dropping it using the command in suggested_action.
 For more information, see Working with PostgreSQL autovacuum in the Amazon RDS User
 Guide.
suggested_action      | {"SELECT pg_drop_replication_slot('slot1') FROM
 pg_replication_slots WHERE active = 'f';"}
```

Action suggérée

Pour résoudre ce problème, vérifiez la configuration de réplication afin de détecter tout problème lié au schéma cible ou aux données susceptibles de mettre fin au processus d'application. Les raisons les plus courantes sont les suivantes :

- Colonnes manquantes
- Type de données incompatible
- Incompatibilité des données
- Table manquante

Si le problème est lié à des problèmes d'infrastructure :

- Problèmes de réseau - Comment résoudre les problèmes liés à une base de données Amazon RDS dans un état réseau incompatible ? .
- La base de données ou l'instance de base de données n'est pas disponible pour les raisons suivantes :
  - L'instance de réplication n'a plus d'espace de stockage : consultez les instances de base de données Amazon RDS à court de stockage pour obtenir des informations sur l'ajout de stockage.
  - Paramètres incompatibles - Passez en revue Comment puis-je réparer une instance de base de données Amazon RDS bloquée avec le statut de paramètres incompatibles ? pour plus d'informations sur la manière de résoudre le problème.

Si votre instance est en dehors du AWS réseau ou est activée AWS EC2, consultez votre administrateur pour savoir comment résoudre les problèmes liés à la disponibilité ou à l'infrastructure.

Supprimer le slot inactif

> **Warning**
>
> Attention : Avant de supprimer un slot de réplication, assurez-vous qu'il n'est pas en cours de réplication, qu'il est inactif et qu'il est dans un état irrécupérable. La suppression prématurée d'un emplacement peut perturber la réplication ou entraîner une perte de données.

Après avoir confirmé que le slot de réplication n'est plus nécessaire, supprimez-le pour permettre à Autovacuum de continuer. Cette condition `active = 'f'` garantit que seul un emplacement inactif est supprimé.

```
SELECT pg_drop_replication_slot('slot1') WHERE active ='f'
```

Réplicas en lecture

Lorsque le `hot_standby_feedback` paramètre est activé pour les répliques de lecture Amazon RDS, il empêche l'aspiration automatique sur la base de données principale de supprimer les lignes mortes qui pourraient encore être nécessaires aux requêtes exécutées sur la réplique en lecture. Cela concerne tous les types de répliques physiques en lecture, y compris celles gérées avec ou sans emplacements de réplication. Ce comportement est nécessaire car les requêtes exécutées sur le réplica de secours nécessitent que ces lignes restent disponibles sur le réplica principal pour éviter les conflits de requêtes et les annulations.

Réplique de lecture avec emplacement de réplication physique

Ces emplacements garantissent que la base de données principale conserve les fichiers journaux d'écriture anticipée essentiels jusqu'à ce que la réplique les traite, préservant ainsi la cohérence des données même en cas de perturbation du réseau.

À partir de RDS pour PostgreSQL Aurora PostgreSQL , toutes les répliques utilisent des emplacements de réplication. Dans les versions antérieures, seules les répliques entre régions utilisaient des emplacements de réplication.

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle trouve une réplique en lecture avec un slot de réplication physique comme bloqueur.

```
blocker               | Read replica with physical replication slot
database              |
blocker_identifier    | rds_us_west_2_db_xxxxxxxxxxxxxxxxxxxx
wait_event            | Not applicable
autovacuum_lagging_by | 554,080,689
suggestion            | Run the following query on the replica
 "rds_us_west_2_db_xxxxxxxxxxxxxxxxxxx" to find the long running query:

                      | SELECT * FROM pg_catalog.pg_stat_activity WHERE
 backend_xmin::text::bigint = 757989377;

                      | Review carefully and you may consdier terminating the query on
 read replica using suggested_action. For more information, see Working with PostgreSQL
 autovacuum in the Amazon RDS User Guide.                                    +
          |
suggested_action      | {"SELECT pg_terminate_backend(pid) FROM
 pg_catalog.pg_stat_activity WHERE backend_xmin::text::bigint = 757989377;","
                                                                            +
                      | [OR]

                                                +
                      | ","Disable hot_standby_feedback","

                                                +
                      | [OR]

                                                +
                      | ","Delete the read replica if not needed"}
```

Réplique de lecture avec réplication en continu

Amazon RDS permet de configurer des répliques de lecture sans emplacement de réplication physique dans les anciennes versions, jusqu'à la version 13. Cette approche réduit les frais généraux en permettant au système principal de recycler les fichiers WAL de manière plus agressive, ce qui est avantageux dans les environnements où l'espace disque est limité et peut être toléré de manière occasionnelle ReplicaLag. Cependant, sans emplacement, le mode veille doit rester synchronisé pour éviter de manquer des fichiers WAL. Amazon RDS utilise des fichiers WAL archivés pour aider la réplique à rattraper son retard, mais ce processus nécessite une surveillance attentive et peut être lent.

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle trouve une réplique de lecture en continu comme bloqueur.

```
blocker               | Read replica with streaming replication slot
database              | Not applicable
blocker_identifier    | xx.x.x.xxx/xx
wait_event            | Not applicable
autovacuum_lagging_by | 610,146,760
suggestion            | Run the following query on the replica "xx.x.x.xxx" to find the
 long running query:
                                                                                +
                      | SELECT * FROM pg_catalog.pg_stat_activity WHERE
 backend_xmin::text::bigint = 348319343;

               +
                      | Review carefully and you may consdier terminating the query on
 read replica using suggested_action. For more information, see Working with PostgreSQL
 autovacuum in the Amazon RDS User Guide.                                       +
                      |
suggested_action      | {"SELECT pg_terminate_backend(pid) FROM
 pg_catalog.pg_stat_activity WHERE backend_xmin::text::bigint = 348319343;","

                       +
                      | [OR]

                                                                                +
                      | ","Disable hot_standby_feedback","

                                                                                +
                      | [OR]

                                                                                +
                      | ","Delete the read replica if not needed"}
```

Action suggérée

Comme recommandé dans la `suggested_action` colonne, examinez attentivement ces options pour débloquer l'autovacuum.

- Terminer la requête — En suivant les instructions de la colonne de suggestions, vous pouvez vous connecter à la réplique de lecture, comme indiqué dans la colonne suggested _action. Il est conseillé de revoir attentivement l'option permettant de mettre fin à la session. Si la résiliation est considérée comme sûre, vous pouvez utiliser la `pg_terminate_backend( )` fonction pour mettre fin à la session. Cette action peut être effectuée par un administrateur (tel que le compte principal RDS) ou par un utilisateur disposant du privilège pg_terminate_backend () requis.

  Vous pouvez exécuter la commande SQL suivante sur la réplique en lecture pour mettre fin à la requête qui empêche le vide sur le serveur principal de nettoyer les anciennes lignes. La valeur de `backend_xmin` est indiquée dans la sortie de la fonction :

  ```
  SELECT
      pg_terminate_backend(pid)
  FROM
      pg_catalog.pg_stat_activity
  WHERE
      backend_xmin::text::bigint = backend_xmin;
  ```

- Désactiver le retour en mode veille à chaud : pensez à désactiver le `hot_standby_feedback` paramètre s'il entraîne des retards de mise sous vide importants.

  Le `hot_standby_feedback` paramètre permet à une réplique en lecture d'informer le principal de son activité de requête, empêchant ainsi le principal de vider les tables ou les lignes utilisées en mode veille. Bien que cela garantisse la stabilité des requêtes en mode veille, cela peut retarder considérablement le passage à l'aspirateur sur le serveur principal. La désactivation de cette fonction permet au principal de passer l'aspirateur sans attendre que le serveur d'attente rattrape son retard. Toutefois, cela peut entraîner des annulations de requêtes ou des échecs en mode veille s'il tente d'accéder à des lignes qui ont été supprimées par le serveur principal.
- Supprimer le réplica lu si ce n'est pas nécessaire — Si le réplica lu n'est plus nécessaire, vous pouvez le supprimer. Cela supprimera la surcharge de réplication associée et permettra au serveur principal de recycler les journaux de transactions sans être freiné par le réplica.

Tables temporaires

Les tables temporaires, créées à l'aide du TEMPORARY mot-clé, résident dans le schéma temporaire, par exemple pg_temp_xxx, et ne sont accessibles qu'à la session qui les a créées. Les tables temporaires sont supprimées à la fin de la session. Cependant, ces tables sont invisibles pour le processus d'aspiration automatique de PostgreSQL et doivent être nettoyées manuellement par la session qui les a créées. Essayer d'aspirer la table des températures d'une autre session n'a aucun effet.

Dans des circonstances exceptionnelles, une table temporaire existe sans qu'elle soit détenue par une session active. Si la session propriétaire se termine de façon inattendue en raison d'un crash fatal, d'un problème réseau ou d'un événement similaire, il est possible que la table temporaire ne soit pas nettoyée et qu'elle soit considérée comme une table « orpheline ». Lorsque le processus Autovacuum de PostgreSQL détecte une table temporaire orpheline, il enregistre le message suivant :

```
LOG: autovacuum: found orphan temp table \"%s\".\"%s\" in database \"%s\"
```

La `postgres_get_av_diag()` fonction affiche une sortie similaire à la suivante lorsqu'elle identifie une table temporaire en tant que bloqueur. Pour que la fonction affiche correctement la sortie relative aux tables temporaires, elle doit être exécutée dans la même base de données où ces tables existent.

```
blocker               | Temporary table
database              | my_database
blocker_identifier    | pg_temp_14.ttemp
wait_event            | Not applicable
autovacuum_lagging_by | 1,805,802,632
suggestion            | Connect to database "my_database". Review carefully, you
 may consider dropping temporary table using command in suggested_action. For more
 information, see Working with PostgreSQL autovacuum in the Amazon RDS User Guide.
suggested_action      | {"DROP TABLE ttemp;"}
```

Action suggérée

Suivez les instructions fournies dans la `suggestion` colonne de sortie pour identifier et supprimer la table temporaire qui empêche l'exécution d'Autovacuum. Utilisez la commande suivante pour supprimer la table temporaire signalée par`postgres_get_av_diag()`. Remplacez le nom de la table en fonction de la sortie fournie par la `postgres_get_av_diag()` fonction.

```
DROP TABLE my_temp_schema.my_temp_table;
```

La requête suivante peut être utilisée pour identifier les tables temporaires :

```
SELECT
    oid,
    relname,
    relnamespace::regnamespace,
    age(relfrozenxid)
FROM
    pg_class
WHERE
relpersistence = 't'
ORDER BY
    age(relfrozenxid) DESC;
```

Cette section explore d'autres raisons qui peuvent empêcher l'utilisation de l'aspirateur de progresser. Ces problèmes ne sont actuellement pas directement identifiables par la `postgres_get_av_diag()` fonction.

Rubriques

- Pages non valides
- Incohérence de l'index
- Taux de transaction exceptionnellement élevé

Pages non valides

Une erreur de page non valide se produit lorsque PostgreSQL détecte une incompatibilité dans le checksum d'une page lors de l'accès à cette page. Le contenu est illisible, ce qui empêche l'autovacuum de geler les tuples. Cela arrête efficacement le processus de nettoyage. L'erreur suivante est inscrite dans le journal de PostgreSQL :

```
WARNING:  page verification failed, calculated checksum YYYYY but expected XXXX
ERROR:  invalid page in block ZZZZZ of relation base/XXXXX/XXXXX
CONTEXT:  automatic vacuum of table myschema.mytable
```

Déterminer le type d'objet

```
ERROR: invalid page in block 4305910 of relation base/16403/186752608
WARNING: page verification failed, calculated checksum 50065 but expected 60033
```

À partir du message d'erreur, le chemin `base/16403/186752608` fournit les informations suivantes :

- « base » est le nom du répertoire situé sous le répertoire de données PostgreSQL.
- « 16403 » est l'OID de la base de données, que vous pouvez consulter dans le catalogue du `pg_database` système.
- « 186752608 » est le`relfilenode`, que vous pouvez utiliser pour rechercher le schéma et le nom de l'objet dans le catalogue du `pg_class` système.

En vérifiant le résultat de la requête suivante dans la base de données concernée, vous pouvez déterminer le type d'objet. La requête suivante récupère les informations d'objet pour oid : 186752608. Remplacez l'OID par celui correspondant à l'erreur que vous avez rencontrée.

```
SELECT
    relname AS object_name,
    relkind AS object_type,
    nspname AS schema_name
FROM
    pg_class c
    JOIN pg_namespace n ON c.relnamespace = n.oid
WHERE
    c.oid = 186752608;
```

Pour plus d'informations, consultez la `pg_class`documentation de PostgreSQL pour connaître tous les types d'objets pris en charge, indiqués par `relkind` la colonne dans. `pg_class`

Orientations

La solution la plus efficace à ce problème dépend de la configuration de votre instance Amazon RDS spécifique et du type de données concerné par la page incohérente.

Si le type d'objet est un index :

Il est recommandé de reconstruire l'index.

- Utilisation de l'**`CONCURRENTLY`**option — Avant la version 12 de PostgreSQL, la reconstruction d'un index nécessitait un verrou de table exclusif, limitant ainsi l'accès à la table. Avec PostgreSQL

version 12 et versions ultérieures, cette option permet le verrouillage au niveau CONCURRENTLY des lignes, ce qui améliore considérablement la disponibilité de la table. Voici la commande :

```
REINDEX INDEX ix_name CONCURRENTLY;
```

Bien qu'elle CONCURRENTLY soit moins perturbatrice, elle peut être plus lente sur des tables très occupées. Si possible, envisagez de créer l'indice pendant les périodes de faible trafic.

Pour plus d'informations, consultez la documentation PostgreSQL REINDEX.

- À l'aide de l'**INDEX_CLEANUP FALSE**option — Si les index sont volumineux et qu'on estime qu'il faut beaucoup de temps pour les terminer, vous pouvez débloquer Autovacuum en exécutant un manuel VACUUM FREEZE tout en excluant les index. Cette fonctionnalité est disponible dans PostgreSQL version 12 et versions ultérieures.

  Le contournement des index vous permettra d'éviter le processus d'élimination de l'index incohérent et d'atténuer le problème d'encapsulation. Toutefois, cela ne résoudra pas le problème sous-jacent de page non valide. Pour résoudre complètement le problème de page non valide, vous devez tout de même reconstruire l'index.

Si le type d'objet est une vue matérialisée :

Si une erreur de page non valide se produit sur une vue matérialisée, connectez-vous à la base de données concernée et actualisez-la pour corriger la page non valide :

Actualisez la vue matérialisée :

```
REFRESH MATERIALIZED VIEW schema_name.materialized_view_name;
```

Si l'actualisation échoue, essayez de recréer :

```
DROP MATERIALIZED VIEW schema_name.materialized_view_name;
CREATE MATERIALIZED VIEW schema_name.materialized_view_name AS query;
```

L'actualisation ou la recréation de la vue matérialisée permet de la restaurer sans affecter les données de la table sous-jacente.

Pour tous les autres types d'objets :

Pour tous les autres types d'objets, contactez l' AWS assistance.

Incohérence de l'index

Un indice logiquement incohérent peut empêcher l'autovacuum de progresser. Les erreurs suivantes ou des erreurs similaires sont enregistrées pendant la phase de vide de l'index ou lorsque des instructions SQL accèdent à l'index.

```
ERROR: right sibling's left-link doesn't match:block 5 links to 10 instead of expected
 2 in index ix_name
```

```
ERROR: failed to re-find parent key in index "XXXXXXXXXX" for deletion target page XXX
CONTEXT:  while vacuuming index index_name of relation schema.table
```

Orientations

Reconstruisez l'index ou ignorez les index en utilisant `INDEX_CLEANUP` le manuel`VACUUM FREEZE`. Pour plus d'informations sur la façon de reconstruire l'index, voir Si le type d'objet est un index.

- Utilisation de l'option CONCURRENTLY : avant la version 12 de PostgreSQL, la reconstruction d'un index nécessitait un verrouillage de table exclusif, limitant ainsi l'accès à la table. Avec la version 12 de PostgreSQL et les versions ultérieures, l'option CONCURRENTLY permet le verrouillage au niveau des lignes, ce qui améliore considérablement la disponibilité de la table. Voici la commande :

  ```
  REINDEX INDEX ix_name CONCURRENTLY;
  ```

  CONCURRENTLY est moins perturbateur, mais il peut être plus lent lorsque les tables sont occupées. Si possible, envisagez de créer l'indice pendant les périodes de faible trafic. Pour plus d'informations, consultez REINDEX dans la documentation de PostgreSQL.
- À l'aide de l'option INDEX_CLEANUP FALSE : si les index sont volumineux et qu'on estime qu'ils nécessitent beaucoup de temps pour être terminés, vous pouvez débloquer l'autovacuum en exécutant un VACUUM FREEZE manuel tout en excluant les index. Cette fonctionnalité est disponible dans PostgreSQL version 12 et versions ultérieures.

  Le contournement des index vous permettra d'éviter le processus d'élimination de l'index incohérent et d'atténuer le problème d'encapsulation. Toutefois, cela ne résoudra pas le problème sous-jacent de page non valide. Pour résoudre complètement le problème de page non valide, vous devez tout de même reconstruire l'index.

Taux de transaction exceptionnellement élevé

Dans PostgreSQL, les taux de transactions élevés peuvent avoir un impact significatif sur les performances d'Autovacuum, ce qui ralentit le nettoyage des tuples morts et augmente le risque d'encapsulation des identifiants de transaction. Vous pouvez surveiller le taux de transaction en mesurant la différence `max(age(datfrozenxid))` entre deux périodes, généralement par seconde. En outre, vous pouvez utiliser les indicateurs de compteur suivants de RDS Performance Insights pour mesurer le taux de transaction (la somme de xact_commit et xact_rollback), qui est le nombre total de transactions.

Compteur	Type	Unité	Métrique
xact_commit	Transactions	Validations par seconde	db.Transactions.xact_commit
xact_rollback	Transactions	Restaurations par seconde	db.Transactions.xact_rollback

Une augmentation rapide indique une charge de transactions élevée, qui peut surcharger Autovacuum, provoquant un gonflement, des problèmes de verrouillage et des problèmes de performances potentiels. Cela peut avoir un impact négatif sur le processus d'aspiration automatique de plusieurs manières :

- Activité liée à la table : La table en question peut faire l'objet d'un volume élevé de transactions, ce qui peut entraîner des retards.
- Ressources du système L'ensemble du système est peut-être surchargé, ce qui rend difficile pour Autovacuum d'accéder aux ressources nécessaires pour fonctionner efficacement.

Envisagez les stratégies suivantes pour permettre à l'autovacuum de fonctionner plus efficacement et de suivre ses tâches :

1. Réduisez le taux de transaction si possible. Envisagez de regrouper ou de regrouper des transactions similaires dans la mesure du possible.
2. Ciblez des tables fréquemment mises à jour avec un `VACUUM FREEZE` fonctionnement manuel tous les soirs, toutes les semaines ou toutes les deux semaines en dehors des heures de pointe.

3. Envisagez d'étendre votre classe d'instance pour allouer davantage de ressources système afin de gérer le volume élevé de transactions et d'autovacuum.

Cette section décrit les facteurs qui contribuent souvent au ralentissement des performances de l'aspirateur et explique comment résoudre ces problèmes.

Rubriques

- Aspirez de grands index
- Trop de tables ou de bases de données pour être vidées
- Un aspirateur agressif (pour éviter tout enroulement) fonctionne

Aspirez de grands index

VACUUM fonctionne selon des phases séquentielles : initialisation, analyse des tas, aspiration des index et des tas, nettoyage des index, troncature des tas et nettoyage final. Pendant le scan en tas, le processus élague les pages, les défragmente et les fige. Une fois l'analyse du tas terminée, VACUUM nettoie les index, renvoie les pages vides au système d'exploitation et effectue les dernières tâches de nettoyage, telles que le nettoyage de la carte de l'espace libre et la mise à jour des statistiques.

L'aspiration de l'index peut nécessiter plusieurs passages lorsque `maintenance_work_mem` (ou`autovacuum_work_mem`) est insuffisante pour traiter l'index. Dans PostgreSQL 16 et versions antérieures, une limite de mémoire de 1 Go pour le stockage des IDs tuples morts imposait souvent des transmissions multiples sur des index volumineux. PostgreSQL 17 `TidStore` introduit, qui alloue de la mémoire de manière dynamique au lieu d'utiliser un tableau à allocation unique. Cela permet de supprimer la contrainte de 1 Go, d'utiliser la mémoire de manière plus efficace et de réduire le besoin de plusieurs analyses d'index pour chaque index.

Les index volumineux peuvent toujours nécessiter plusieurs passes dans PostgreSQL 17 si la mémoire disponible ne permet pas de traiter l'intégralité des index en une seule fois. Généralement, les index plus grands contiennent davantage de tuples morts qui nécessitent plusieurs passes.

Détection des opérations de vide lentes

La `postgres_get_av_diag()` fonction peut détecter le ralentissement des opérations d'aspiration en raison d'une mémoire insuffisante. Pour plus d'informations sur cette fonction, consultez.

La `postgres_get_av_diag()` fonction émet les notifications suivantes lorsque la mémoire disponible n'est pas suffisante pour terminer l'aspiration de l'index en un seul passage.

`rds_tools`1,8

```
NOTICE: Your database is currently running aggressive vacuum to prevent wraparound and
 it might be slow.
```

```
NOTICE: The current setting of autovacuum_work_mem is "XXX" and might not be
 sufficient. Consider increasing the setting, and if necessary, scaling up the Amazon
 RDS instance class for more memory.
       Additionally, review the possibility of manual vacuum with exclusion of indexes
 using (VACUUM (INDEX_CLEANUP FALSE, VERBOSE TRUE) table_name;).
```

`rds_tools`1,9

```
NOTICE: Your database is currently running aggressive vacuum to prevent wraparound and
 it might be slow.
```

```
NOTICE: The current setting of autovacuum_work_mem is XX might not be sufficient.
 Consider increasing the setting to XXX, and if necessary, scaling up the RDS instance
 class for more
       memory. The suggested value is an estimate based on the current number of dead
 tuples for the table being vacuumed, which might not fully reflect the latest state.
 Additionally, review the possibility of manual
       vacuum with exclusion of indexes using (VACUUM (INDEX_CLEANUP FALSE, VERBOSE
 TRUE) table_name;). For more information, see
       Working with PostgreSQL autovacuum in the Amazon Amazon RDS User Guide
       .
```

> **Note**
>
> La `postgres_get_av_diag()` fonction repose sur l'`pg_stat_all_tables.n_dead_tup`estimation de la quantité de mémoire requise pour l'aspiration à index.

Lorsque la `postgres_get_av_diag()` fonction identifie une opération d'aspiration lente qui nécessite plusieurs analyses d'index en raison d'une insuffisance`autovacuum_work_mem`, elle génère le message suivant :

```
NOTICE: Your vacuum is performing multiple index scans due to insufficient
 autovacuum_work_mem:XXX for index vacuuming.
        For more information, see Working with PostgreSQL autovacuum in the Amazon
 Amazon RDS User Guide.
```

Orientations

Vous pouvez appliquer les solutions de contournement suivantes à l'aide du manuel `VACUUM FREEZE` pour accélérer le gel du tableau.

Augmentez la mémoire pour passer l'aspirateur

Comme suggéré par la `postgres_get_av_diag()` fonction, il est conseillé d'augmenter le `autovacuum_work_mem` paramètre pour faire face aux contraintes de mémoire potentielles au niveau de l'instance. Bien qu'il `autovacuum_work_mem` s'agisse d'un paramètre dynamique, il est important de noter que pour que le nouveau paramètre de mémoire prenne effet, le démon autovacuum doit redémarrer ses serveurs de travail. Pour ce faire, procédez comme suit :

1. Vérifiez que le nouveau paramètre est en place.
2. Arrêtez les processus en cours d'exécution d'autovacuum.

Cette approche garantit que l'allocation de mémoire ajustée est appliquée aux nouvelles opérations d'autovacuum.

Pour des résultats plus immédiats, pensez à effectuer manuellement une `VACUUM FREEZE` opération avec un `maintenance_work_mem` réglage accru au cours de votre session :

```
SET maintenance_work_mem TO '1GB';
VACUUM FREEZE VERBOSE table_name;
```

Si vous utilisez Amazon RDS et que vous constatez que vous avez besoin de mémoire supplémentaire pour prendre en charge des valeurs plus élevées pour `maintenance_work_mem` ou`autovacuum_work_mem`, envisagez de passer à une classe d'instance avec plus de mémoire. Cela peut fournir les ressources nécessaires pour améliorer les opérations d'aspiration manuelles et automatiques, ce qui se traduit par une amélioration des performances globales de l'aspirateur et des bases de données.

Désactiver INDEX_CLEANUP

Le mode manuel VACUUM de PostgreSQL version 12 et versions ultérieures permet d'ignorer la phase de nettoyage de l'index, tandis que l'aspirateur automatique d'urgence dans PostgreSQL version 14 et versions ultérieures le fait automatiquement en fonction du paramètre. vacuum_failsafe_age

> **Warning**
>
> Le fait de ne pas nettoyer l'index peut entraîner un gonflement de l'index et avoir un impact négatif sur les performances des requêtes. Pour pallier ce problème, pensez à réindexer ou à aspirer les index concernés pendant une période de maintenance.

Pour obtenir des conseils supplémentaires sur la gestion des index volumineux, reportez-vous à la documentation surGestion de la fonction autovacuum avec de grands index .

Aspirateur à indice parallèle

À partir de PostgreSQL 13, les index peuvent être nettoyés et nettoyés en parallèle par défaut VACUUM manuellement, avec un processus d'aspiration assigné à chaque index. Cependant, pour que PostgreSQL puisse déterminer si une opération sous vide peut être exécutée en parallèle, des critères spécifiques doivent être remplis :

- Il doit y avoir au moins deux index.
- Le max_parallel_maintenance_workers paramètre doit être défini sur au moins 2.
- La taille de l'index doit dépasser la min_parallel_index_scan_size limite, qui est par défaut de 512 Ko.

Vous pouvez ajuster le max_parallel_maintenance_workers paramètre en fonction du nombre de v CPUs disponibles sur votre instance Amazon RDS et du nombre d'index sur la table afin d'optimiser le délai d'aspiration.

Pour plus d'informations, consultez Parallel vacuuming in Amazon RDS for PostgreSQL et Amazon Aurora PostgreSQL.

Trop de tables ou de bases de données pour être vidées

Comme indiqué dans la documentation The Autovacuum Daemon de PostgreSQL, le démon autovacuum fonctionne par le biais de plusieurs processus. Cela inclut un lanceur d'aspiration automatique permanent chargé de démarrer les processus de travail d'aspiration automatique pour

chaque base de données du système. Le lanceur programme ces travailleurs pour qu'ils initient environ toutes les `autovacuum_naptime` secondes par base de données.

Avec les bases de données « N », un nouveau travailleur commence environ toutes les [`autovacuum_naptime`/N secondes]. Cependant, le nombre total de travailleurs simultanés est limité par le `autovacuum_max_workers` paramètre. Si le nombre de bases de données ou de tables à nettoyer dépasse cette limite, la base de données ou table suivante sera traitée dès qu'un travailleur sera disponible.

Lorsque de nombreuses tables ou bases de données volumineuses doivent être nettoyées simultanément, tous les opérateurs d'autovacuum disponibles peuvent être occupés pendant une période prolongée, ce qui retarde la maintenance des autres tables et bases de données. Dans les environnements où les taux de transactions sont élevés, ce goulot d'étranglement peut rapidement s'aggraver et potentiellement entraîner des problèmes de vide généraux au sein de votre instance Amazon RDS.

Lorsqu'il `postgres_get_av_diag()` détecte un nombre élevé de tables ou de bases de données, il fournit la recommandation suivante :

```
NOTICE: Your database is currently running aggressive vacuum to prevent wraparound and
 it might be slow.
```

```
NOTICE: The current setting of autovacuum_max_workers:3 might not be sufficient.
 Consider increasing the setting and, if necessary, consider scaling up the Amazon RDS
 instance class for more workers.
```

Orientations

Augmenter autovacuum_max_workers

Pour accélérer l'aspiration, nous vous recommandons de régler le `autovacuum_max_workers` paramètre afin de permettre à un plus grand nombre d'aspirateurs automatiques de travailler simultanément. Si les problèmes de performances persistent, envisagez de faire évoluer votre instance Amazon RDS vers une classe contenant plus de vCPUs, ce qui peut améliorer encore les capacités de traitement parallèle.

Un aspirateur agressif (pour éviter tout enroulement) fonctionne

L'âge de la base de données (MaximumUsedTransactionIDs) dans PostgreSQL ne diminue que lorsqu'un vide agressif (pour empêcher toute encapsulation) est terminé avec succès. Jusqu'à ce que ce vide soit terminé, l'âge continuera d'augmenter en fonction du taux de transaction.

La `postgres_get_av_diag()` fonction génère ce qui suit `NOTICE` lorsqu'elle détecte un vide agressif. Cependant, il ne déclenche cette sortie qu'après que le vide a été actif pendant au moins deux minutes.

```
NOTICE: Your database is currently running aggressive vacuum to prevent wraparound,
 monitor autovacuum performance.
```

Pour plus d'informations sur l'aspirateur agressif, voir Lorsqu'un aspirateur agressif fonctionne déjà.

Vous pouvez vérifier si un aspirateur agressif est en cours à l'aide de la requête suivante :

```
SELECT
    a.xact_start AS start_time,
    v.datname "database",
    a.query,
    a.wait_event,
    v.pid,
    v.phase,
    v.relid::regclass,
    pg_size_pretty(pg_relation_size(v.relid)) AS heap_size,
    (
        SELECT
            string_agg(pg_size_pretty(pg_relation_size(i.indexrelid)) || ':' ||
 i.indexrelid::regclass || chr(10), ', ')
        FROM
            pg_index i
        WHERE
            i.indrelid = v.relid
    ) AS index_sizes,
    trunc(v.heap_blks_scanned * 100 / NULLIF(v.heap_blks_total, 0)) AS step1_scan_pct,
    v.index_vacuum_count || '/' || (
        SELECT
            count(*)
        FROM
            pg_index i
        WHERE
```

```
            i.indrelid = v.relid
    ) AS step2_vacuum_indexes,
    trunc(v.heap_blks_vacuumed * 100 / NULLIF(v.heap_blks_total, 0)) AS
 step3_vacuum_pct,
    age(CURRENT_TIMESTAMP, a.xact_start) AS total_time_spent_sofar
FROM
    pg_stat_activity a
    INNER JOIN pg_stat_progress_vacuum v ON v.pid = a.pid;
```

Vous pouvez déterminer s'il s'agit d'un vide agressif (pour éviter toute distorsion) en vérifiant la colonne de requête dans la sortie. L'expression « pour empêcher l'enveloppement » indique qu'il s'agit d'un aspirateur agressif.

```
query                 | autovacuum: VACUUM public.t3 (to prevent wraparound)
```

Supposons, par exemple, que vous disposiez d'un bloqueur à l'âge d'un milliard de transactions et d'une table nécessitant un aspirateur agressif pour éviter toute encapsulation au même âge de transaction. De plus, il existe un autre bloqueur à l'âge de 750 millions de transactions. Après avoir éliminé le bloqueur à l'âge d'un milliard de transactions, l'âge des transactions ne tombera pas immédiatement à 750 millions. Il restera élevé jusqu'à ce que la table nécessitant un vide agressif ou toute transaction portant sur un âge de plus de 750 millions de dollars soit terminée. Pendant cette période, l'âge des transactions de votre cluster PostgreSQL continuera d'augmenter. Une fois le processus d'aspiration terminé, l'âge des transactions tombera à 750 millions, mais recommencera à augmenter jusqu'à ce que l'aspiration soit terminée. Ce cycle se poursuivra tant que ces conditions persisteront, jusqu'à ce que l'âge des transactions atteigne finalement le niveau configuré pour votre instance Amazon RDS, spécifié par`autovacuum_freeze_max_age`.

La `postgres_get_av_diag()` fonction fournit les messages NOTICE suivants :

Lorsque l'âge n'a pas encore atteint le seuil de surveillance

Le seuil de surveillance `postgres_get_av_diag()` pour identifier les bloqueurs est de 500 millions de transactions par défaut. Si `postgres_get_av_diag()` génère le NOTICE suivant, cela indique que l'âge de la transaction n'a pas encore atteint ce seuil.

```
NOTICE: postgres_get_av_diag() checks for blockers that prevent aggressive vacuums
 only, it does so only after exceeding dvb_threshold which is 500,000,000 and age of
 this PostgreSQL cluster is currently at 2.
```

Non connecté à la base de données dont l'identifiant de transaction est le plus ancien

La `postgres_get_av_diag()` fonction fournit le résultat le plus précis lorsqu'elle est connectée à la base de données dont l'identifiant de transaction est le plus ancien. La base de données dont l'âge de l'identifiant de transaction est le plus ancien indiqué par `postgres_get_av_diag()` sera différente de « my_database » dans votre cas. Si vous n'êtes pas connecté à la bonne base de données, le NOTICE suivant est généré :

```
NOTICE: You are not connected to the database with the age of oldest transaction
 ID. Connect to my_database database and run postgres_get_av_diag() for accurate
 reporting.
```

La connexion à la base de données dont l'âge de transaction est le plus élevé est importante pour les raisons suivantes :

- Identification des bloqueurs de tables temporaires : les métadonnées des tables temporaires étant spécifiques à chaque base de données, elles se trouvent généralement dans la base de données dans laquelle elles ont été créées. Toutefois, si une table temporaire se trouve être le principal bloqueur et qu'elle se trouve dans la base de données contenant la transaction la plus ancienne, cela peut être trompeur. La connexion à la base de données appropriée garantit l'identification précise du bloqueur de table temporaire.
- Diagnostic des aspirateurs lents : Les métadonnées de l'index et les informations relatives au nombre de tables sont spécifiques à la base de données et sont nécessaires pour diagnostiquer les problèmes d'aspiration lente.

La base de données avec la transaction la plus ancienne par âge se trouve sur une base de données rdsadmin ou template0

Dans certains cas, les `rdsadmin` `template0` bases de données peuvent être identifiées comme étant celles dont l'identifiant de transaction est le plus ancien. Dans ce cas, `postgres_get_av_diag()` émettra l'AVIS suivant :

```
NOTICE: The database with the age of oldest transaction ID is rdsadmin or template0,
 reach out to support if the reported blocker is in rdsadmin or template0.
```

Vérifiez que le bloqueur répertorié ne provient d'aucune de ces deux bases de données. Si la présence du bloqueur est signalée dans l'un `rdsadmin` ou l'autre`template0`, contactez le support car ces bases de données ne sont pas accessibles aux utilisateurs et nécessitent une intervention.

Il est très peu probable que la base de `template0` données `rdsadmin` ou la base de données contienne un bloqueur supérieur.

Lorsqu'un aspirateur agressif est déjà en marche

La `postgres_get_av_diag()` fonction est conçue pour signaler lorsqu'un processus de vide agressif est en cours d'exécution, mais elle ne déclenche cette sortie que lorsque le vide est actif pendant au moins 1 minute. Ce délai intentionnel contribue à réduire les risques de faux positifs. En attendant, la fonction garantit que seuls les aspirateurs efficaces et significatifs sont signalés, ce qui permet une surveillance plus précise et plus fiable de l'activité du vide.

La `postgres_get_av_diag()` fonction génère l'AVIS suivant lorsqu'elle détecte un ou plusieurs aspirateurs agressifs en cours de réalisation.

```
NOTICE: Your database is currently running aggressive vacuum to prevent wraparound,
 monitor autovacuum performance.
```

Comme indiqué dans l'AVIS, continuez à surveiller les performances de l'aspirateur. Pour plus d'informations sur l'aspirateur agressif, voir Un aspirateur agressif (pour éviter tout enroulement) fonctionne

Lorsque l'aspirateur automatique est désactivé

La `postgres_get_av_diag()` fonction génère le NOTICE suivant si l'autovacuum est désactivé sur votre instance de base de données :

```
NOTICE: Autovacuum is OFF, we strongly recommend to enable it, no restart is
 necessary.
```

Autovacuum est une fonctionnalité essentielle de votre instance de base de données PostgreSQL RDS pour PostgreSQL qui garantit le bon fonctionnement de la base de données. Il supprime automatiquement les anciennes versions de lignes, récupère de l'espace de stockage et évite le surpeuplement des tables, ce qui contribue à garantir l'efficacité des tables et des index pour des performances optimales. En outre, il protège contre l'encapsulation des identifiants de transaction, qui peut interrompre les transactions sur votre instance Amazon RDS. La désactivation de l'autovacuum peut entraîner une baisse à long terme des performances et de la stabilité de la base de données. Nous vous conseillons de le garder allumé en permanence.

> **Note**
>
> La désactivation de l'aspirateur automatique n'arrête pas les aspirateurs agressifs. Elles se produiront toujours une fois que vos tables auront atteint le `autovacuum_freeze_max_age` seuil.

Le nombre de transactions restantes est extrêmement bas

La `postgres_get_av_diag()` fonction génère l'AVIS suivant lorsqu'un aspirateur enveloppant est imminent. Cet AVIS est émis lorsque votre instance Amazon RDS est à 100 millions de transactions avant que de nouvelles transactions ne soient potentiellement rejetées.

```
WARNING: Number of transactions remaining is critically low, resolve issues with
 autovacuum or perform manual VACUUM FREEZE before your instance stops accepting
 transactions.
```

Votre action immédiate est requise pour éviter les interruptions de service de la base de données. Vous devez surveiller de près vos opérations d'aspiration et envisager de lancer manuellement une opération `VACUUM FREEZE` sur la base de données concernée afin d'éviter les échecs de transaction.

Utilisation de mécanismes de journalisation pris en charge par RDS for PostgreSQL

Il existe plusieurs paramètres, extensions et autres éléments configurables que vous pouvez définir pour journaliser les activités qui se produisent sur votre instance de base de données PostgreSQL. Tel est le cas des éléments suivants :

- Le paramètre`log_statement` peut être utilisé pour consigner l'activité utilisateur dans votre base de données PostgreSQL. Pour en savoir plus sur la journalisation RDS for PostgreSQL et sur la façon de surveiller les journaux, veuillez consulter Fichiers journaux de base de données RDS for PostgreSQL.
- Le paramètre `rds.force_admin_logging_level` journalise les actions effectuées par l'utilisateur interne Amazon RDS (rdsadmin) dans les bases de données sur l'instance de base de données. Il écrit la sortie dans le journal d'erreurs PostgreSQL. Les valeurs autorisées sont

`disabled`, `debug5`, `debug4`, `debug3`, `debug2`, `debug1`, `info`, `notice`, `warning`, `error`, journal,`fatal` et `panic`. La valeur par défaut est `disabled`.

- Le paramètre `rds.force_autovacuum_logging_level` peut être configuré pour capturer diverses opérations d'autovacuum dans le journal des erreurs PostgreSQL. Pour de plus amples informations, veuillez consulter Enregistrement des activités d'autovacuum et de vacuum.
- L'extension PostgreSQL Audit (pgAudit) peut être installée et configurée pour capturer des activités au niveau de la session ou au niveau de l'objet. Pour de plus amples informations, veuillez consulter Utilisation pgAudit pour enregistrer l'activité de la base de données.
- L'extension `log_fdw` vous permet d'accéder au journal du moteur de base de données à l'aide de SQL. Pour de plus amples informations, veuillez consulter Utilisation de l'extension log_fdw pour accéder au journal de base de données en utilisant SQL.
- La bibliothèque `pg_stat_statements` est spécifiée par défaut pour le paramètre `shared_preload_libraries` dans RDS for PostgreSQL 10 et versions ultérieures. C'est cette bibliothèque que vous pouvez utiliser pour analyser les requêtes en cours d'exécution. Assurez-vous que `pg_stat_statements` est défini dans votre groupe de paramètres de base de données. Pour plus d'informations sur la surveillance de votre instance de base de données RDS for PostgreSQL à l'aide des informations fournies par cette bibliothèque, veuillez consulter Statistiques SQL pour RDS PostgreSQL.
- Le paramètre `log_hostname` capture dans le journal le nom d'hôte de chaque connexion client. Pour RDS for PostgreSQL versions 12 et ultérieures, ce paramètre est défini sur `off` par défaut. Si vous l'activez, veillez à surveiller les temps de connexion des sessions. Lorsqu'il est activé, le service utilise la demande de recherche inversée DNS pour obtenir le nom d'hôte du client qui établit la connexion et pour l'ajouter au journal PostgreSQL. Cela a un impact notable au cours de la connexion à la session. Nous vous recommandons d'activer ce paramètre à des fins de dépannage uniquement.

D'une manière générale, le but de la journalisation est de permettre au DBA de surveiller, d'ajuster les performances et de résoudre les problèmes. La plupart des journaux sont chargés automatiquement sur Amazon CloudWatch ou Performance Insights. Ici, ils sont triés et regroupés pour fournir des métriques complètes pour votre instance de base de données. Pour en savoir plus sur la surveillance et les métriques d'Amazon RDS, veuillez consulter Surveillance des métriques dans une RDS instance Amazon.

Gestion des fichiers temporaires avec PostgreSQL

Dans PostgreSQL, une requête complexe peut effectuer plusieurs opérations de tri ou de hachage en même temps, chaque opération utilisant la mémoire d'instance pour stocker les résultats jusqu'à la valeur spécifiée dans le paramètre. `work_mem` Lorsque la mémoire de l'instance n'est pas suffisante, des fichiers temporaires sont créés pour stocker les résultats. Ils sont écrits sur le disque pour terminer l'exécution de la requête. Par la suite, ces fichiers sont automatiquement supprimés une fois la requête terminée. Dans RDS pour PostgreSQL, ces fichiers sont stockés dans Amazon EBS sur le volume de données. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS. Vous pouvez surveiller la `FreeStorageSpace` métrique publiée dans CloudWatch pour vous assurer que votre instance de base de données dispose de suffisamment d'espace de stockage disponible. Pour plus d'informations, consultez `FreeStorageSpace` .

Nous recommandons d'utiliser des d'instances Amazon RDS Optimized Read pour les charges de travail impliquant plusieurs requêtes simultanées qui augmentent l'utilisation de fichiers temporaires. Ces d'instances utilisent un stockage local au niveau des blocs basé sur un disque SSD (Non-Volatile Memory Express) pour placer les fichiers temporaires. NVMe Pour de plus amples informations, veuillez consulter Amélioration des performances des requêtes pour RDS for PostgreSQL avec Lectures optimisées pour Amazon RDS.

Vous pouvez utiliser les paramètres et fonctions suivants pour gérer les fichiers temporaires dans votre instance.

- **`temp_file_limit`** : ce paramètre annule toute requête dépassant la taille des fichiers temp_files en Ko. Cette limite empêche toute requête de s'exécuter indéfiniment et de consommer de l'espace disque avec des fichiers temporaires. Vous pouvez estimer la valeur à l'aide des résultats du paramètre `log_temp_files`. Nous vous recommandons d'examiner le comportement de la charge de travail et de définir la limite en fonction de l'estimation. L'exemple suivant présente la manière dont une requête est annulée lorsqu'elle dépasse la limite.

```
postgres=>select * from pgbench_accounts, pg_class, big_table;
```

```
ERROR: temporary file size exceeds temp_file_limit (64kB)
```

- **`log_temp_files`** : ce paramètre envoie des messages au fichier postgresql.log lorsque les fichiers temporaires d'une session sont supprimés. Ce paramètre produit des journaux lorsqu'une

requête est terminée avec succès. Par conséquent, cela peut ne pas aider à résoudre les requêtes actives et de longue durée.

L'exemple suivant montre que lorsque la requête aboutit, les entrées sont journalisées dans le fichier postgresql.log pendant que les fichiers temporaires sont nettoyés.

```
2023-02-06 23:48:35 UTC:205.251.233.182(12456):adminuser@postgres:[31236]:LOG:
 temporary file: path "base/pgsql_tmp/pgsql_tmp31236.5", size 140353536
2023-02-06 23:48:35 UTC:205.251.233.182(12456):adminuser@postgres:[31236]:STATEMENT:
  select a.aid from pgbench_accounts a, pgbench_accounts b where a.bid=b.bid order by
 a.bid limit 10;
2023-02-06 23:48:35 UTC:205.251.233.182(12456):adminuser@postgres:[31236]:LOG:
 temporary file: path "base/pgsql_tmp/pgsql_tmp31236.4", size 180428800
2023-02-06 23:48:35 UTC:205.251.233.182(12456):adminuser@postgres:[31236]:STATEMENT:
  select a.aid from pgbench_accounts a, pgbench_accounts b where a.bid=b.bid order by
 a.bid limit 10;
```

- **pg_ls_tmpdir** : cette fonction disponible auprès de RDS pour PostgreSQL versions 13 et ultérieures offre une visibilité sur l'utilisation actuelle des fichiers temporaires. La requête terminée n'apparaît pas dans les résultats de la fonction. Dans l'exemple suivant, vous pouvez visualiser les résultats de cette fonction.

```
postgres=>select * from pg_ls_tmpdir();
```

```
      name       |    size    |      modification
-----------------+------------+------------------------
 pgsql_tmp8355.1 | 1072250880 | 2023-02-06 22:54:56+00
 pgsql_tmp8351.0 | 1072250880 | 2023-02-06 22:54:43+00
 pgsql_tmp8327.0 | 1072250880 | 2023-02-06 22:54:56+00
 pgsql_tmp8351.1 |  703168512 | 2023-02-06 22:54:56+00
 pgsql_tmp8355.0 | 1072250880 | 2023-02-06 22:54:00+00
 pgsql_tmp8328.1 |  835031040 | 2023-02-06 22:54:56+00
 pgsql_tmp8328.0 | 1072250880 | 2023-02-06 22:54:40+00
(7 rows)
```

```
postgres=>select query from pg_stat_activity where pid = 8355;
```

```
query
------------------------------------------------------------------------------------
select a.aid from pgbench_accounts a, pgbench_accounts b where a.bid=b.bid order by
 a.bid
(1 row)
```

Le nom du fichier inclut l'ID de traitement (PID) de la session qui a généré le fichier temporaire. Une requête plus avancée, comme dans l'exemple suivant, effectue la somme des fichiers temporaires pour chaque PID.

```
postgres=>select replace(left(name, strpos(name, '.')-1),'pgsql_tmp','') as pid,
 count(*), sum(size) from pg_ls_tmpdir() group by pid;
```

```
 pid  | count |   sum
------+-------+------------------
 8355 |     2 | 2144501760
 8351 |     2 | 2090770432
 8327 |     1 | 1072250880
 8328 |     2 | 2144501760
(4 rows)
```

- **pg_stat_statements** : si vous activez le paramètre pg_stat_statements, vous pouvez consulter l'utilisation moyenne des fichiers temporaires par appel. Vous pouvez identifier le query_id de la requête et l'utiliser pour examiner l'utilisation des fichiers temporaires, comme indiqué dans l'exemple suivant.

```
postgres=>select queryid from pg_stat_statements where query like 'select a.aid from
 pgbench%';
```

```
       queryid
----------------------
 -7170349228837045701
(1 row)
```

```
postgres=>select queryid, substr(query,1,25), calls, temp_blks_read/calls
 temp_blks_read_per_call, temp_blks_written/calls temp_blks_written_per_call from
 pg_stat_statements where queryid = -7170349228837045701;
```

```
       queryid        |          substr           | calls | temp_blks_read_per_call |
 temp_blks_written_per_call
----------------------+---------------------------+-------+-------------------------
+---------------------------
 -7170349228837045701 | select a.aid from pgbench |    50 |                  239226 |
                     388678
(1 row)
```

- **Performance Insights** : dans le tableau de bord Performance Insights, vous pouvez consulter l'utilisation des fichiers temporaires en activant les métriques temp_bytes et temp_files. Vous pouvez ensuite voir la moyenne de ces deux métriques et voir comment elles correspondent à la charge de travail des requêtes. La vue de Performance Insights n'affiche pas spécifiquement les requêtes qui génèrent les fichiers temporaires. Toutefois, lorsque vous associez Performance Insights à la requête indiquée pour `pg_ls_tmpdir`, vous pouvez dépanner, analyser et déterminer les modifications apportées à la charge de travail de vos requêtes.

  Pour plus d'informations sur la façon d'analyser les métriques et les requêtes avec Performance Insights, consultezAnalyse des métriques à l'aide du tableau de bord de Performance Insights.

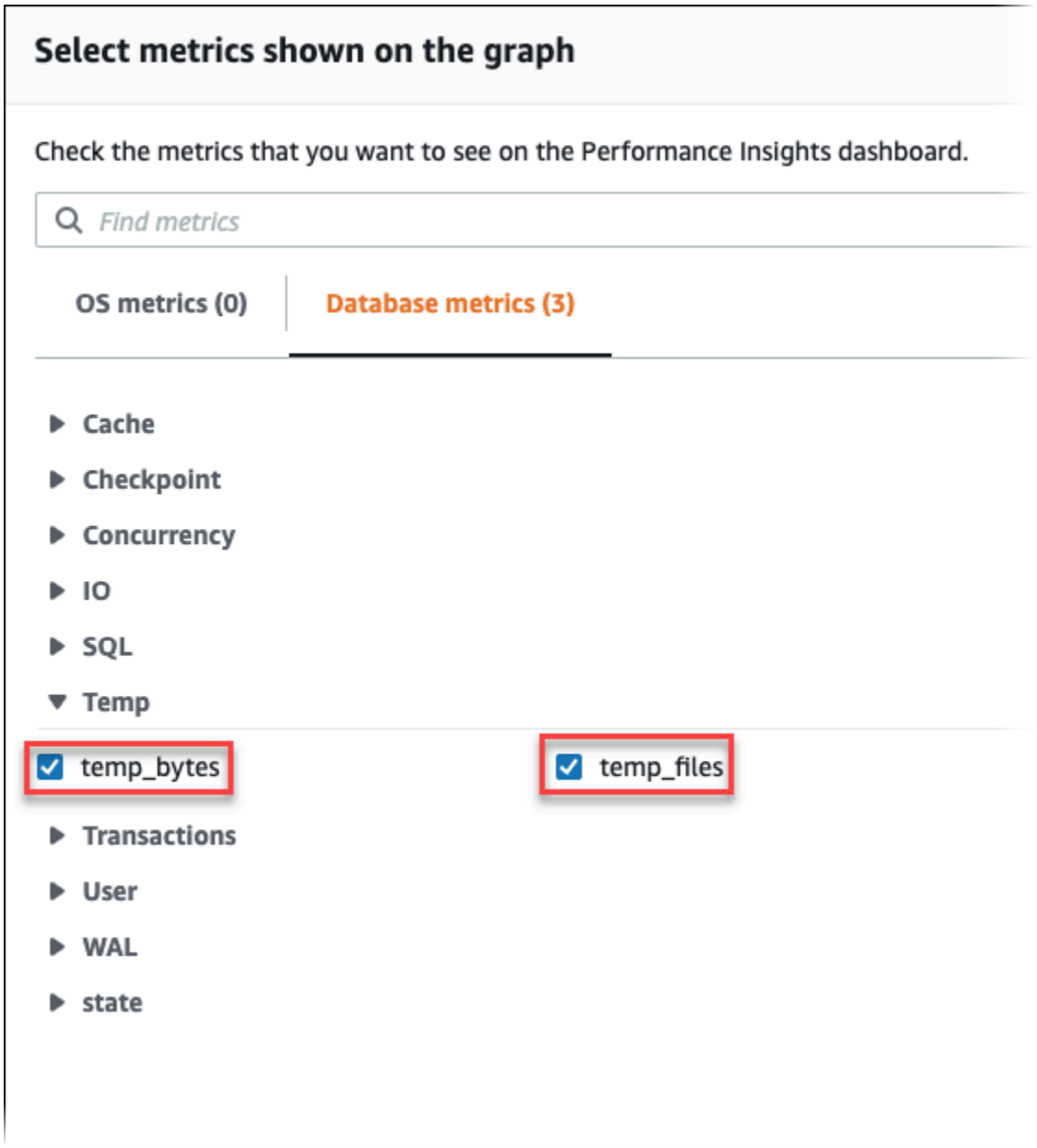
  Pour un exemple d'affichage de l'utilisation de fichiers temporaires avec Performance Insights, voir Visualisation de l'utilisation des fichiers temporaires avec Performance Insights

Visualisation de l'utilisation des fichiers temporaires avec Performance Insights

Vous pouvez utiliser Performance Insights pour visualiser l'utilisation des fichiers temporaires en activant les métriques temp_bytes et temp_files. La vue dans Performance Insights ne montre pas les requêtes spécifiques qui génèrent des fichiers temporaires. Toutefois, lorsque vous combinez Performance Insights avec la requête indiquée pour`pg_ls_tmpdir`, vous pouvez résoudre les problèmes, analyser et déterminer les modifications de la charge de travail de vos requêtes.

1. Dans le tableau de bord de Performance Insights, choisissez Gérer les métriques.

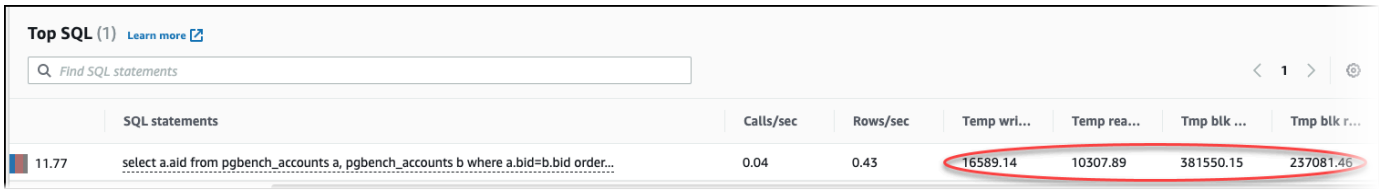
2. Choisissez Métriques de base de données et sélectionnez les métriques temp_bytes et temp_files comme indiqué dans l'image suivante.



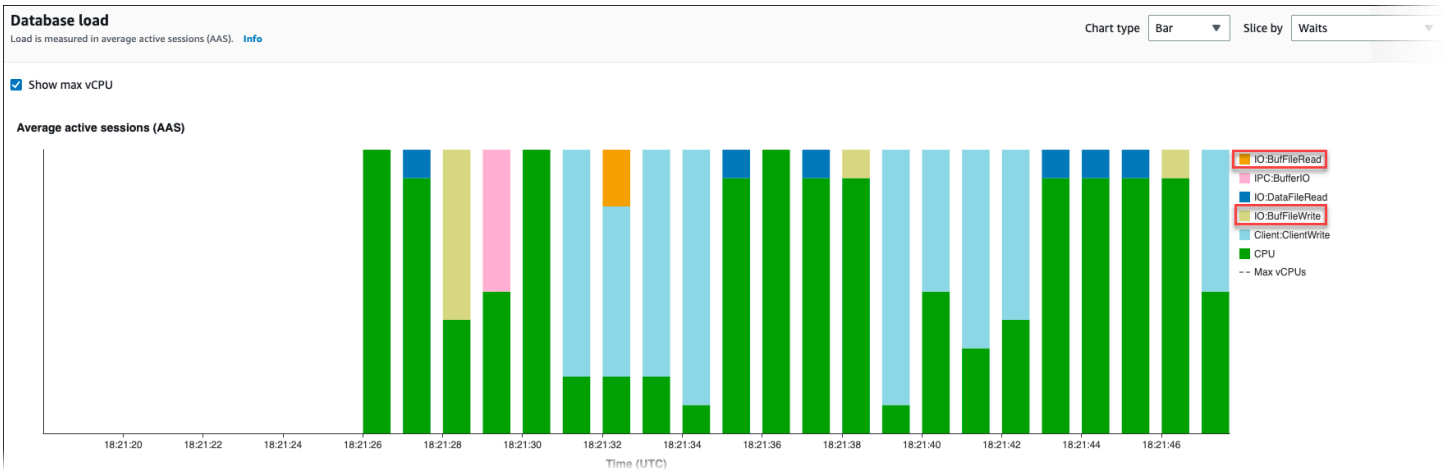
3. Dans l'SQLonglet supérieur, cliquez sur l'icône Préférences.
4. Dans la fenêtre des préférences, activez les statistiques suivantes pour qu'elles apparaissent dans l'SQLonglet supérieur et choisissez Continuer.

   - Nombre d'écritures temporaires/seconde
   - Nombre de lectures temporaires/seconde
   - Écritures/appels en bloc temporaires
   - Lectures/appels en bloc temporaires

5. Le fichier temporaire est décomposé lorsqu'il est associé à la requête affichée pour `pg_ls_tmpdir`, comme le montre l'exemple suivant.



Les événements `IO:BufFileRead` et `IO:BufFileWrite` se produisent lorsque les requêtes les plus importantes de votre charge de travail créent souvent des fichiers temporaires. Vous pouvez utiliser Performance Insights pour identifier les principales requêtes en attente `IO:BufFileRead` et `IO:BufFileWrite` en consultant le nombre moyen de sessions actives (AAS) dans les SQL sections Chargement de base de données et Top.



Pour plus d'informations sur la façon d'analyser les requêtes les plus importantes et la charge par événement d'attente à l'aide de l'analyse des performances, consultez Vue d'ensemble de l'SQLonglet Top. Vous devez identifier et ajuster les requêtes qui entraînent une augmentation de l'utilisation des fichiers temporaires et des événements d'attente associés. Pour plus d'informations sur ces événements d'attente et les mesures correctives, voir IO : BufFileRead et IO : BufFileWrite .

> **Note**
>
> Le paramètre `work_mem` contrôle le moment où la mémoire de l'opération de tri est insuffisante et les résultats sont écrits dans des fichiers temporaires. Nous vous recommandons de ne pas modifier la valeur de ce paramètre au-delà de la valeur par défaut, car cela permettrait à chaque session de base de données de consommer davantage de mémoire. En outre, une session unique qui effectue des jointures et des tris complexes peut

effectuer des opérations parallèles au cours desquelles chaque opération consomme de la mémoire.
Il est recommandé de définir ce paramètre au niveau de la session à l'aide de la commande `SET work_mem` lorsque vous disposez d'un rapport volumineux comportant plusieurs jointures et tris. La modification n'est alors appliquée qu'à la session en cours et ne modifie pas la valeur de manière globale.

Utilisation de pgBadger pour l'analyse de journal serveur avec PostgreSQL

Vous pouvez utiliser un analyseur de journaux tel que pgBadger pour analyser les journaux PostgreSQL. La documentation pgBadger indique que le motif %l (ligne de journal pour la session ou le processus) doit faire partie du préfixe. Cependant, si vous fournissez le `log_line_prefix` RDS actuel en tant que paramètre à pgBadger, il devrait quand même produire un rapport.

Par exemple, la commande suivante formate correctement un fichier journal Amazon RDS for PostgreSQL daté du 04/02/2014 à l'aide de pgBadger.

```
./pgbadger -f stderr -p '%t:%r:%u@%d:[%p]:' postgresql.log.2014-02-04-00
```

Utilisation PGSnapper pour la surveillance de PostgreSQL

Vous pouvez l'utiliser PGSnapper pour faciliter la collecte périodique de statistiques et de mesures relatives aux performances d'Amazon RDS for PostgreSQL. Pour plus d'informations, consultez Surveiller les performances d'Amazon RDS for PostgreSQL à l'aide de. PGSnapper

Utilisation de paramètres sur votre instance de base de données RDS for PostgreSQL

Dans certains cas, vous pouvez créer une instance de base de données RDS for PostgreSQL sans spécifier de groupe de paramètres personnalisé. Si tel est le cas, votre instance de base de données est créée à l'aide du groupe de paramètres par défaut de la version de PostgreSQL que vous choisissez. Par exemple, supposons que vous créez une instance de base de données RDS for PostgreSQL à l'aide de PostgreSQL 13.3. Dans ce cas, l'instance de base de données est créée à l'aide des valeurs du groupe de paramètres pour les versions PostgreSQL 13, `default.postgres13`.

Vous pouvez créer votre propre groupe de paramètres de base de données avec des paramètres personnalisés. Vous devez le faire si vous souhaitez modifier les paramètres de l'instance de base de données RDS for PostgreSQL à partir de leurs valeurs par défaut. Pour savoir comment procéder, veuillez consulter la section Groupes de paramètres pour (Amazon RDS).

Vous pouvez suivre les paramètres de votre instance de base de données RDS for PostgreSQL de plusieurs manières différentes. Vous pouvez utiliser l'API AWS Management Console, la AWS CLI, ou l'API Amazon RDS. Vous pouvez également interroger les valeurs à partir de la table `pg_settings` PostgreSQL de votre instance, comme illustré ci-dessous.

```
SELECT name, setting, boot_val, reset_val, unit
 FROM pg_settings
 ORDER BY name;
```

Pour plus d'informations sur les valeurs renvoyées par cette requête, veuillez consulter `pg_settings` dans la documentation PostgreSQL.

Soyez particulièrement prudent lorsque vous modifiez les paramètres de `max_connections` et `shared_buffers` sur votre instance de base de données RDS for PostgreSQL. Par exemple, supposons que vous modifiiez les paramètres de `max_connections` ou `shared_buffers` et vous utilisez des valeurs trop élevées pour votre charge de travail réelle. Dans ce cas, votre instance de base de données RDS for PostgreSQL ne démarrera pas. Si cela se produit, une erreur telle que la suivante s'affiche dans le `postgres.log`.

```
2018-09-18 21:13:15 UTC::@:[8097]:FATAL:  could not map anonymous shared memory: Cannot
 allocate memory
2018-09-18 21:13:15 UTC::@:[8097]:HINT:  This error usually means that PostgreSQL's
 request for a shared memory segment
exceeded available memory or swap space. To reduce the request size (currently
 3514134274048 bytes), reduce
PostgreSQL's shared memory usage, perhaps by reducing shared_buffers or
 max_connections.
```

Toutefois, notez que vous ne pouvez modifier aucune valeur des paramètres contenus dans les groupes de paramètres RDS for PostgreSQL DB par défaut. Pour modifier n'importe quel paramètre, créez d'abord un groupe de paramètres de base de données personnalisé. Modifiez ensuite les paramètres de ce groupe personnalisé, puis appliquez le groupe de paramètres personnalisé à votre instance de base de données RDS for PostgreSQL. Pour en savoir plus, veuillez consulter la section Groupes de paramètres pour (Amazon RDS).

Il existe deux types de paramètres dans RDS pour PostgreSQL.

- Paramètres statiques : les paramètres statiques exigent que l'instance de base de données RDS for PostgreSQL soit réinitialisée après une modification afin que la nouvelle valeur puisse prendre effet.
- Paramètres dynamiques : les paramètres dynamiques ne nécessitent pas de réinitialisation après avoir modifié leurs paramètres.

> **Note**
>
> Si votre instance de base de données RDS for PostgreSQL utilise votre propre groupe de paramètres de base de données personnalisé, vous pouvez modifier les valeurs des paramètres dynamiques sur l'instance en cours d'exécution. Pour ce faire AWS Management Console, vous pouvez utiliser l'API AWS CLI, ou l'API Amazon RDS.

Vous pouvez également modifier des valeurs de paramètres, si vous disposez des privilèges nécessaires, en utilisant les commandes `ALTER DATABASE`, `ALTER ROLE` et `SET`.

Liste de paramètres d'instance de base de données RDS for PostgreSQL

Le tableau suivant répertorie certains (mais pas tous) des paramètres disponibles (mais pas tous) dans une instance de base de données RDS for PostgreSQL. Pour afficher tous les paramètres disponibles, utilisez la describe-db-parameters AWS CLI commande. Par exemple, pour obtenir la liste de tous les paramètres disponibles dans le groupe de paramètres par défaut pour RDS for PostgreSQL version 13, procédez comme suit.

```
aws rds describe-db-parameters --db-parameter-group-name default.postgres13
```

Vous pouvez également utiliser la console. Choisissez Parameter groups (Groupes de paramètres) dans le menu Amazon RDS, puis choisissez le groupe de paramètres parmi ceux disponibles dans votre Région AWS.

Nom du paramètre	Type d'application	Description
application_name	Répartition dynamique	Définit le nom de l'application à indiquer dans les statistiques et les journaux.
archive_command	Répartition dynamique	Définit la commande shell qui sera appelée pour archiver un fichier WAL.
array_nulls	Répartition dynamique	Autorise l'entrée d'éléments NULL dans les tableaux.
authentication_timeout	Répartition dynamique	Définit le délai maximum autorisé pour procéder à l'authentification du client.
autovacuum	Répartition dynamique	Démarre le sous-processus autovacuum.

Nom du paramètre	Type d'application	Description
autovacuum_analyze_scale_factor	Répartition dynamique	Nombre de tuples insérés, mis à jour ou supprimés avant analyse en tant que partie de reltuples.
autovacuum_analyze_threshold	Répartition dynamique	Nombre minimum de tuples insérés, mis à jour ou supprimés avant analyse.
autovacuum_freeze_max_age	Statique	Âge auquel lancer le processus autovacuum sur une table pour empêcher le renvoi à la ligne de l'ID de transaction.
autovacuum_naptime	Répartition dynamique	Temps de repos entre les exécutions autovacuum.
autovacuum_max_workers	Statique	Définit le nombre maximum de processus autovacuum qui peuvent être exécutés simultanément.
autovacuum_vacuum_cost_delay	Répartition dynamique	Valeur du coût de retard du processus vacuum en millisecondes, pour le processus autovacuum.
autovacuum_vacuum_cost_limit	Répartition dynamique	Coût cumulé qui provoque l'endormissement du processus vacuum, pour le processus autovacuum.
autovacuum_vacuum_scale_factor	Répartition dynamique	Nombre de tuples mis à jour ou supprimés avant le processus vacuum en tant que partie de reltuples.
autovacuum_vacuum_threshold	Répartition dynamique	Nombre de tuples mis à jour ou supprimés avant le processus vacuum.

Nom du paramètre	Type d'application	Description
backslash_quote	Répartition dynamique	Définit si une barre oblique inverse (\) peut être utilisée dans les littéraux de chaîne.
bgwriter_delay	Répartition dynamique	Délai d'inactivité entre les tours d'activité du processus d'écriture en arrière-plan.
bgwriter_lru_maxpages	Répartition dynamique	Nombre maximum de pages récemment utilisées qui peuvent être vidées à chaque tour par le processus d'écriture en tâche de fond.
bgwriter_lru_multiplier	Répartition dynamique	Multiple de l'utilisation moyenne de tampons à libérer par tour.
bytea_output	Répartition dynamique	Configure le format de sortie pour les valeurs de type octets.
check_function_bodies	Répartition dynamique	Vérifie les corps des fonctions pendant la fonction CREATE FUNCTION.
checkpoint_completion_target	Répartition dynamique	Temps nécessaire pour vider les tampons sales au moment de la vérification, sous la forme d'une fraction de temps entre deux points de vérification.
checkpoint_segments	Répartition dynamique	Définit la distance maximale dans les segments de journaux entre deux points de vérification automatique de journal WAL.
checkpoint_timeout	Répartition dynamique	Définit le temps maximum entre deux points de vérification automatique des WAL.

Nom du paramètre	Type d'application	Description
checkpoint_warning	Répartition dynamique	Active les avertissements si des segments de points de vérification sont remplis à une fréquence supérieure à ce paramètre.
client_connection_check_interval	Répartition dynamique	Définit l'intervalle de temps entre les vérifications de déconnexion lors de l'exécution des requêtes.
client_encoding	Répartition dynamique	Définit l'encodage du jeu de caractères du client.
client_min_messages	Répartition dynamique	Définit les niveaux des messages envoyés au client.
commit_delay	Répartition dynamique	Définit la durée, en micro-secondes, entre le déclenchement de la sonde transaction-commit et le vidage de WAL vers le disque.
commit_siblings	Répartition dynamique	Définit le nombre minimum de transactions ouvertes simultanément avant d'atteindre le délai commit_delay.
constraint_exclusion	Répartition dynamique	Autorise le planificateur à utiliser des contraintes pour optimiser les requêtes.
cpu_index_tuple_cost	Répartition dynamique	Définit l'estimation faite par le planificateur du coût de traitement de chaque entrée d'index pendant la vérification d'un index.
cpu_operator_cost	Répartition dynamique	Définit l'estimation faite par le planificateur du coût de traitement de chaque opérateur ou appel de fonction.

Nom du paramètre	Type d'application	Description
cpu_tuple_cost	Répartition dynamique	Définit l'estimation faite par le planificateur du coût de traitement de chaque ligne.
cursor_tuple_fraction	Répartition dynamique	Définit l'estimation faite par le planificateur de la fraction des lignes d'un curseur qui sera récupérée.
datestyle	Répartition dynamique	Définit le format d'affichage des valeurs de type date et heure.
deadlock_timeout	Répartition dynamique	Définit le délai d'attente au niveau d'un verrou avant blocage.
debug_pretty_print	Répartition dynamique	Indente les affichages des arborescence d'analyse et de planification.
debug_print_parse	Répartition dynamique	Enregistre l'arborescence d'analyse de chaque requête.
debug_print_plan	Répartition dynamique	Enregistre le plan d'exécution de chaque requête.
debug_print_rewritten	Répartition dynamique	Enregistre l'arbre d'interprétation réécrit de chaque requête.
default_statistics_target	Répartition dynamique	Définit la cible des statistiques par défaut.

Nom du paramètre	Type d'application	Description
default_tablespace	Répartition dynamique	Définit l'espace de table par défaut dans lequel créer des tables et des index.
default_transaction_deferrable	Répartition dynamique	Définit le statut reportable des nouvelles transactions.
default_transaction_isolation	Répartition dynamique	Définit le niveau d'isolation de transaction de chaque nouvelle transaction.
default_transaction_read_only	Répartition dynamique	Définit le statut en lecture seule des nouvelles transactions.
default_with_oids	Répartition dynamique	Crée de nouvelles tables avec object IDs (OIDs) par défaut.
effective_cache_size	Répartition dynamique	Définit l'estimation faite par le planificateur de la taille du cache du disque.
effective_io_concurrency	Répartition dynamique	Nombre de demandes simultanées pouvant être traitées de manière efficace par le sous-système du disque.
enable_bitmapscan	Répartition dynamique	Autorise l'utilisation de plans de parcours de bitmap par le planificateur.
enable_hashagg	Répartition dynamique	Autorise l'utilisation de plans d'agrégation hachée par le planificateur.

Nom du paramètre	Type d'application	Description
enable_hashjoin	Répartition dynamique	Autorise l'utilisation de plans de jointures de hachage par le planificateur.
enable_indexscan	Répartition dynamique	Autorise l'utilisation de plans de parcours d'index par le planificateur.
enable_material	Répartition dynamique	Autorise l'utilisation de la matérialisation par le planificateur.
enable_mergejoin	Répartition dynamique	Autorise l'utilisation de plans de jointures de fusion par le planificateur.
enable_nestloop	Répartition dynamique	Autorise l'utilisation de plans de jointures de boucles imbriquées par le planificateur.
enable_seqscan	Répartition dynamique	Autorise l'utilisation de plans de parcours séquentiels par le planificateur.
enable_sort	Répartition dynamique	Autorise l'utilisation des étapes de tri explicite par le planificateur.
enable_tidscan	Répartition dynamique	Autorise l'utilisation de plans de parcours de TID par le planificateur.
escape_string_warning	Répartition dynamique	Avertit sur l'utilisation des barres obliques inverses (\) dans des littéraux de chaîne ordinaires.

Nom du paramètre	Type d'application	Description
extra_float_digits	Répartition dynamique	Définit le nombre de chiffres affichés pour les valeurs à virgule flottante.
from_collapse_limit	Répartition dynamique	Définit la taille FROM-list au-delà de laquelle les sous-requêtes ne sont pas regroupées.
fsync	Répartition dynamique	Force la synchronisation des mises à jour sur le disque.
full_page_writes	Répartition dynamique	Ecrit les pages complètes dans les WAL lors de la première modification après un point de vérification.
geqo	Répartition dynamique	Active l'optimisation génétique des requêtes.
geqo_effort	Répartition dynamique	geqo_effort est utilisé pour définir la valeur par défaut pour les autres paramètres GEQO.
geqo_generations	Répartition dynamique	GEQO : nombre d'itérations de l'algorithme.
geqo_pool_size	Répartition dynamique	GEQO : nombre d'individus au sein d'une population.
geqo_seed	Répartition dynamique	GEQO : valeur initiale pour la sélection des chemins au hasard.

Nom du paramètre	Type d'application	Description
geqo_selection_bias	Répartition dynamique	GEQO : pression de sélectivité au sein de la population.
geqo_threshold	Répartition dynamique	Définit le seuil d'éléments FROM au-delà duquel GEQO est utilisé.
gin_fuzzy_search_limit	Répartition dynamique	Définit le résultat maximum autorisé pour la recherche exacte par GIN.
hot_standby_feedback	Répartition dynamique	Détermine si une instance de secours envoie des messages de commentaire aux instances principales ou de veille en amont.
intervalstyle	Répartition dynamique	Définit le format d'affichage des valeurs de type intervalle.
join_collapse_limit	Répartition dynamique	Définit la taille FROM-list au-delà de laquelle les constructions JOIN ne sont pas mises à plat.
lc_messages	Répartition dynamique	Définit la langue d'affichage des messages.
lc_monetary	Répartition dynamique	Définit la locale à utiliser pour le formatage des montants monétaires.
lc_numeric	Répartition dynamique	Définit la locale à utiliser pour le formatage des nombres.

Nom du paramètre	Type d'application	Description
lc_time	Répartition dynamique	Définit la locale à utiliser pour le formatage des valeurs de date et d'heure.
log_autovacuum_min_duration	Répartition dynamique	Définit la durée minimum d'exécution au-delà de laquelle les actions autovacuum seront enregistrées.
log_checkpoints	Répartition dynamique	Enregistre les points de vérification.
log_connections	Répartition dynamique	Enregistre toutes les connexions réussies.
log_disconnections	Répartition dynamique	Enregistre la fin d'une session, y compris sa durée.
log_duration	Répartition dynamique	Enregistre la durée de chaque instruction SQL terminée.
log_error_verbosity	Répartition dynamique	Définit la quantité de détails dans les messages enregistrés.
log_executor_stats	Répartition dynamique	Ecrit les statistiques de performance de l'exécuteur dans le journal du serveur.
log_filename	Répartition dynamique	Définit le modèle de nom de fichier pour les fichiers journaux.

Nom du paramètre	Type d'application	Description
log_file_mode	Répartition dynamique	Définit les autorisations de fichier pour les fichiers journaux. La valeur par défaut est 0644.
log_hostname	Répartition dynamique	Enregistre le nom de l'hôte dans les journaux de connexion. À partir de PostgreSQL 12 et des versions ultérieures, ce paramètre est « désactivé » par défaut. Lorsqu'il est activé, la connexion utilise la recherche inversée DNS pour obtenir le nom d'hôte qui est capturé dans les journaux de connexion. Si vous activez ce paramètre, vous devez surveiller son impact sur le temps nécessaire à l'établissement des connexions.
log_line_prefix	Répartition dynamique	Contrôle les informations préfixées à chaque ligne de journal.
log_lock_waits	Répartition dynamique	Enregistre les longs temps d'attente pour l'acquisition d'un verrou.
log_min_duration_statement	Répartition dynamique	Définit la durée minimum d'exécution au-delà de laquelle les instructions seront enregistrées.
log_min_error_statement	Répartition dynamique	Déclenche l'enregistrement de toutes les instructions générant une erreur à ce niveau ou à un niveau supérieur.
log_min_messages	Répartition dynamique	Définit les niveaux des messages qui sont enregistrés.

Nom du paramètre	Type d'application	Description
log_parser_stats	Répartition dynamique	Ecrit les statistiques de performance de l'analyseur dans le journal du serveur.
log_planner_stats	Répartition dynamique	Ecrit les statistiques de performance du planificateur dans le journal du serveur.
log_rotation_age	Répartition dynamique	Déclenchement de la rotation de fichier journal automatique au-delà d'un délai de N minutes.
log_rotation_size	Répartition dynamique	Déclenchement de la rotation de fichier journal automatique au-delà de N kilo-octets.
log_statement	Répartition dynamique	Définit le type d'instructions enregistrées.
log_statement_stats	Répartition dynamique	Ecrit les statistiques de performance cumulées dans le journal du serveur.
log_temp_files	Répartition dynamique	Enregistre l'utilisation des fichiers temporaires dont la taille est supérieure à cette taille en kilo-octets.
log_timezone	Répartition dynamique	Définit le fuseau horaire à utiliser dans les messages de journaux.
log_truncate_on_rotation	Répartition dynamique	Tronquez les fichiers journaux existants du même nom pendant la rotation des journaux.

Nom du paramètre	Type d'application	Description
logging_collector	Statique	Démarrez un sous-processus pour capturer la sortie stderr et/ou csvlogs dans des fichiers journaux.
maintenance_work_mem	Répartition dynamique	Définit la quantité maximum de mémoire que peuvent utiliser les opérations de maintenance.
max_connections	Statique	Définit le nombre maximum de connexions simultanées.
max_files_per_process	Statique	Définit le nombre maximum de fichiers ouverts simultanément pour chaque processus serveur.
max_locks_per_transaction	Statique	Définit le nombre maximum de verrous par transaction.
max_pred_locks_per_transaction	Statique	Définit le nombre maximum de verrous de prédicat par transaction.
max_prepared_transactions	Statique	Définit le nombre maximum de transactions préparées simultanément.
max_stack_depth	Répartition dynamique	Définit la profondeur maximum de la pile, en kilo-octets.
max_standby_archive_delay	Répartition dynamique	Définit le délai maximum avant l'annulation des requêtes lorsqu'un serveur hot standby traite des données WAL archivées.
max_standby_streaming_delay	Répartition dynamique	Définit le délai maximum avant l'annulation des requêtes lorsqu'un serveur hot standby traite des données WAL diffusées.

Nom du paramètre	Type d'application	Description

`max_wal_size`	Répartition dynamique	Définit la taille (en Mo) de journal WAL qui déclenche un point de vérification.
		• Pour RDS pour PostgreSQL 15 et versions antérieures, il est de 2 Go.
		• Pour RDS pour PostgreSQL 16 et versions ultérieures, RDS pour PostgreSQL se configure automatiquement en fonction de la taille de stockage allouée à votre instance :
		  • 6 Go pour les instances avec un espace de stockage alloué de 100 Go ou plus.
		  • 2 Go pour les instances dont le stockage alloué est inférieur à 100 Go.
		
		Utilisez la commande suivante sur votre instance de base de données Amazon RDS for PostgreSQL pour voir sa valeur actuelle :
		`SHOW max_wal_size;`

`min_wal_size`	Répartition dynamique	Définit la taille minimale à laquelle réduire le journal WAL. Pour PostgreSQL version 9.6 ou antérieure, la taille `min_wal_size` est exprimée en unités de 16 Mo. Pour PostgreSQL version 10 ou supérieure, la taille `min_wal_size` est exprimée en unités de 1 Mo.
`quote_all_identifiers`	Répartition dynamique	Ajoute des guillemets (") à tous les identificateurs lors de la génération de fragments SQL.

Nom du paramètre	Type d'application	Description
`random_page_cost`	Répartition dynamique	Définit l'estimation faite par le planificateur du coût d'une page de disque extraite de façon non séquentielle. Ce paramètre n'a aucune valeur sauf si la gestion du plan de requête (QPM) est activée. Lorsque QPM est activée, la valeur par défaut de ce paramètre est 4.
`rds.adaptive_autovacuum`	Répartition dynamique	Règle automatiquement les paramètres d'autovacuum chaque fois que les seuils d'ID de transaction sont dépassés.
`rds.force_ssl`	Répartition dynamique	Nécessite l'utilisation de connexions SSL. La valeur par défaut est définie sur 1 (activé) pour RDS for PostgreSQL version 15. Toutes les autres versions majeures 14 et antérieures de RDS for PostgreSQL ont la valeur par défaut définie sur 0 (désactivé).
`rds.local_volume_spill_enabled`	Statique	Permet d'écrire des fichiers de déversement logiques sur le volume local.
`rds.log_retention_period`	Répartition dynamique	Définit la rétention des journaux de telle manière qu'Amazon RDS supprime les journaux PostgreSQL antérieurs à n minutes.
`rds.rds_superuser_reserved_connections`	Statique	Définit le nombre d'emplacements de connexion réservés pour rds_superusers. Ce paramètre n'est disponible que dans les versions 15 et antérieures. Pour plus d'informations, consultez la documentation de PostgreSQL reserved_connections.

Nom du paramètre	Type d'application	Description
`rds.replica_identity_full`	Répartition dynamique	Lorsque vous définissez ce paramètre suron, il remplace le paramètre d'identité de réplique sur `FULL` pour toutes les tables de base de données. Cela signifie que toutes les valeurs des colonnes sont écrites dans le journal d'écriture anticipée (WAL), quels que soient vos `REPLICA IDENTITY FULL` paramètres. **Note** L'activation de ce paramètre peut augmenter les IOPS de votre instance de base de données en raison de la journalisation WAL supplémentaire.
`rds.restrict_password_commands`	Statique	Limite la gestion des mots de passe aux utilisateurs auxquels le rôle `rds_password` a été affecté. Pour la restriction par mot de passe, définissez ce paramètre sur 1. La valeur par défaut est 0.
`search_path`	Répartition dynamique	Définit l'ordre de recherche des schémas pour les noms pour lesquels le schéma n'est pas précisé.
`seq_page_cost`	Répartition dynamique	Définit l'estimation faite par le planificateur du coût d'une page de disque extraite de façon séquentielle.
`session_replication_role`	Répartition dynamique	Définit le comportement des sessions concernant les déclencheurs et les règles de réécriture.

Nom du paramètre	Type d'application	Description
shared_buffers	Statique	Définit le nombre de tampons de mémoire partagée utilisés par le serveur.
shared_preload_libraries	Statique	Répertorie les bibliothèques partagées à précharger dans l'instance de base de données RDS for PostgreSQL. Les valeurs prises en charge incluent auto_explain, orafce, pgaudit, pglogical, pg_bigm, pg_cron, pg_hint_plan, pg_prewarm, pg_similarity, pg_stat_statements, pg_tle, pg_transport, plprofiler et plrust.
ssl	Répartition dynamique	Active les connexions SSL.
sql_inheritance	Répartition dynamique	Entraîne l'ajout par défaut de sous-tables dans plusieurs commandes.
ssl_renegotiation_limit	Répartition dynamique	Définit la quantité de trafic à envoyer et recevoir avant de renégocier les clés de chiffrement.
standard_conforming_strings	Répartition dynamique	Entraîne les chaînes ... à traiter littéralement les barres obliques inverses.
statement_timeout	Répartition dynamique	Définit la durée maximum de toute instruction.
synchronize_seqscans	Répartition dynamique	Active les analyses séquentielles synchronisées.

Nom du paramètre	Type d'application	Description
synchronous_commit	Répartition dynamique	Définit le niveau de synchronisation des transactions actuelles.
tcp_keepalives_count	Répartition dynamique	Nombre maximum de paquets TCP keepalive.
tcp_keepalives_idle	Répartition dynamique	Délai entre les émissions de paquets TCP keepalive.
tcp_keepalives_interval	Répartition dynamique	Délai entre les envois de paquets TCP keepalive.
temp_buffers	Répartition dynamique	Définit le nombre maximum de tampons temporaires utilisés par chaque session.
temp_file_limit	Répartition dynamique	Définit la taille maximale en Ko que peuvent atteindre les fichiers temporaires.
temp_tablespaces	Répartition dynamique	Définit l'espace de table à utiliser pour les tables et fichiers de tri temporaires.

Nom du paramètre	Type d'application	Description
`timezone`	Répartition dynamique	Définit le fuseau horaire pour l'affichage et l'interprétation de la date et de l'heure. L'Internet Assigned Numbers Authority (IANA) publie de nouveaux fuseaux horaires sur https://www.iana.org/time-zones plusieurs fois par an. Chaque fois que RDS publie une nouvelle version de maintenance mineure de PostgreSQL, elle est livrée avec les dernières données de fuseau horaire au moment de la publication. Lorsque vous utilisez les dernières versions de RDS for PostgreSQL, vous disposez de données de fuseau horaire récentes provenant de RDS. Pour vous assurer que votre instance de base de données dispose de données de fuseau horaire récentes, nous vous recommandons de passer à une version supérieure du moteur de base de données. Vous ne pouvez pas modifier les tables de fuseau horaire des instances de base de données PostgreSQL manuellement. RDS ne modifie ni ne réinitialise les données de fuseau horaire des instances de base de données en cours d'exécution. Les nouvelles données de fuseau horaire ne sont installées que lorsque vous effectuez une mise à niveau de la version du moteur de base de données.
`track_activities`	Répartition dynamique	Collecte des informations sur les commandes en cours d'exécution.
`track_activity_query_size`	Statique	Définit la taille réservée pour pg_stat_activity.current_query, en octets.

Nom du paramètre	Type d'application	Description
track_counts	Répartition dynamique	Active la collecte de statistiques sur l'activité de la base de données.
track_functions	Répartition dynamique	Active la collecte de statistiques au niveau de la fonction sur l'activité de la base de données.
track_io_timing	Répartition dynamique	Active la collecte de statistiques de durée sur l'activité I/O de la base de données.
transaction_deferrable	Répartition dynamique	Indique si une transaction sérialisable en lecture seule doit être différée jusqu'à ce qu'elle puisse être démarrée sans échec de sérialisation possible.
transaction_isolation	Répartition dynamique	Définit le niveau d'isolation des transactions actuelles.
transaction_read_only	Répartition dynamique	Définit le statut en lecture seule des transactions actuelles.
transform_null_equals	Répartition dynamique	Traite l'expression =NULL en tant que IS NULL.
update_process_title	Répartition dynamique	Met à jour le titre du processus pour indiquer la commande SQL active.

Nom du paramètre	Type d'application	Description
vacuum_cost_delay	Répartition dynamique	Valeur du coût de délai du processus vacuum en millisecondes.
vacuum_cost_limit	Répartition dynamique	Coût cumulé qui provoque l'endormissement du processus vacuum.
vacuum_cost_page_dirty	Répartition dynamique	Coût du processus vacuum pour une page salie par le processus vacuum.
vacuum_cost_page_hit	Répartition dynamique	Coût du processus vacuum pour une page trouvée dans le cache des tampons.
vacuum_cost_page_miss	Répartition dynamique	Coût du processus vacuum pour une page non trouvée dans le cache des tampons.
vacuum_defer_cleanup_age	Répartition dynamique	Nombre de transactions pendant lesquelles le processus vacuum et le nettoyage hot seront reportés à plus tard, le cas échéant.
vacuum_freeze_min_age	Répartition dynamique	Âge limite auquel le processus vacuum doit figer une ligne de tableau.
vacuum_freeze_table_age	Répartition dynamique	Âge auquel le processus vacuum effectue une analyse complète de la table pour figer des lignes.
wal_buffers	Statique	Définit le nombre de tampons de page de disque dans la mémoire partagée pour les WAL.

Nom du paramètre	Type d'application	Description
wal_writer_delay	Répartition dynamique	Délai d'inactivité de l'enregistreur des WAL entre les actions de vidage WAL.
work_mem	Répartition dynamique	Définit la quantité maximum de mémoire que peuvent utiliser les espaces de travail des requêtes.
xmlbinary	Répartition dynamique	Définit la façon dont les valeurs binaires doivent être codées en XML.
xmloption	Répartition dynamique	Définit si des données XML dans des opérations d'analyse ou de sérialisation implicites doivent être considérées comme des documents ou des fragments de contenu.

Amazon RDS utilise les unités PostgreSQL par défaut pour tous les paramètres. Le tableau suivant présente l'unité par défaut PostgreSQL pour chaque paramètre.

Nom du paramètre	Unit
archive_timeout	s
authentication_timeout	s
autovacuum_naptime	s
autovacuum_vacuum_cost_delay	ms
bgwriter_delay	ms
checkpoint_timeout	s

Nom du paramètre	Unit
checkpoint_warning	s
deadlock_timeout	ms
effective_cache_size	8 Ko
lock_timeout	ms
log_autovacuum_min_duration	ms
log_min_duration_statement	ms
log_rotation_age	minutes
log_rotation_size	Ko
log_temp_files	Ko
maintenance_work_mem	Ko
max_stack_depth	Ko
max_standby_archive_delay	ms
max_standby_streaming_delay	ms
post_auth_delay	s
pre_auth_delay	s
segment_size	8 Ko
shared_buffers	8 Ko
statement_timeout	ms
ssl_renegotiation_limit	Ko
tcp_keepalives_idle	s

Nom du paramètre	Unit
tcp_keepalives_interval	s
temp_file_limit	Ko
work_mem	Ko
temp_buffers	8 Ko
vacuum_cost_delay	ms
wal_buffers	8 Ko
wal_receiver_timeout	ms
wal_segment_size	B
wal_sender_timeout	ms
wal_writer_delay	ms
wal_receiver_status_interval	s

Réglage avec les événements d'attente pour RDS for PostgreSQL

Les événements d'attente constituent un outil de réglage important pour RDS for PostgreSQL. Lorsque vous parvenez à déterminer pourquoi les sessions sont en attente de ressources et ce qu'elles font, vous êtes mieux à même de réduire les goulets d'étranglement. Vous pouvez utiliser les informations de cette section pour déterminer les causes possibles et les actions correctives à mettre en œuvre. Cette section décrit également les concepts de base du réglage de PostgreSQL.

Les événements d'attente présentés dans cette section sont spécifiques à RDS for PostgreSQL.

Rubriques

- Concepts essentiels à connaître pour le réglage de RDS for PostgreSQL
- Événements d'attente RDS for PostgreSQL
- Cliente : ClientRead
- Cliente : ClientWrite
- CPU
- IO:BufFileRead et IO:BufFileWrite
- IO : DataFileRead
- IO : WALWrite
- IPC : événements d'attente parallèles
- Lock:advisory
- Lock:extend
- Lock:Relation
- Lock:transactionid
- Lock:tuple
- LWLock:BufferMapping (LWLock:buffer_mapping)
- LWLock:BufferIO (IPC:BufferIO)
- LWLock:buffer_content (BufferContent)
- LWLock:lock_manager (:lockmanager) LWLock
- LWLock:pg_stat_statements
- Timeout:PgSleep
- Timeout:VacuumDelay

Concepts essentiels à connaître pour le réglage de RDS for PostgreSQL

Avant de procéder au réglage de votre base de données RDS for PostgreSQL, découvrez ce que sont les événements d'attente et pourquoi ils se produisent. Examinez également l'architecture de base de RDS for PostgreSQL en termes de mémoire et de disque. Un diagramme d'architecture très utile est disponible dans le wikibook PostgreSQL.

Rubriques

- Événements d'attente RDS for PostgreSQL
- Mémoire RDS for PostgreSQL
- Processus RDS for PostgreSQL

Événements d'attente RDS for PostgreSQL

Un événement d'attente indique que la session est en attente d'une ressource. Par exemple, l'événement d'attente `Client:ClientRead` se produit quand RDS for PostgreSQL attend de recevoir des données du client. Les sessions attendent généralement des ressources telles que les suivantes.

- Accès monothread à une mémoire tampon, par exemple lorsqu'une session tente de modifier une mémoire tampon
- Ligne verrouillée par une autre session
- Lecture d'un fichier de données
- Écriture de fichier journal

Par exemple, pour répondre à une requête, la session peut effectuer une analyse complète de la table. Si les données ne sont pas déjà en mémoire, la session attend que le disque soit I/O terminé. Lorsque les mémoires tampons sont lues en mémoire, la session peut être contrainte d'attendre parce que d'autres sessions accèdent aux mêmes mémoires tampons. La base de données enregistre les attentes à l'aide d'un événement d'attente prédéfini. Ces événements sont regroupés en catégories.

En soi, un événement d'attente individuel n'indique pas un problème de performances. Par exemple, si les données demandées ne sont pas en mémoire, il est nécessaire de les lire sur le disque. Si une session verrouille une ligne pour une mise à jour, une autre session attend que la ligne soit déverrouillée pour pouvoir la mettre à jour. Une validation nécessite d'attendre la fin de l'écriture

dans un fichier journal. Les attentes font partie intégrante du fonctionnement normal d'une base de données.

D'un autre côté, un grand nombre d'événements d'attente indiquent généralement un problème de performances. Dans ce cas, vous pouvez utiliser les données des événements d'attente pour déterminer où les sessions passent du temps. Par exemple, si plusieurs heures sont désormais nécessaires à l'exécution d'un rapport qui ne prend habituellement que quelques minutes, vous pouvez identifier les événements d'attente qui contribuent le plus au temps d'attente total. La détermination des causes des principaux événements d'attente peut vous permettre d'apporter des modifications qui auront pour effet d'améliorer les performances. Par exemple, si votre session est en attente sur une ligne qui a été verrouillée par une autre session, vous pouvez mettre fin à la session à l'origine du verrouillage.

Mémoire RDS for PostgreSQL

La mémoire RDS for PostgreSQL se décompose en deux parties : la mémoire partagée et la mémoire locale.

Rubriques

- Mémoire partagée dans RDS for PostgreSQL
- Mémoire locale dans RDS for PostgreSQL

Mémoire partagée dans RDS for PostgreSQL

RDS for PostgreSQL alloue de la mémoire partagée au démarrage de l'instance. La mémoire partagée se décompose en sous-zones. Les sections suivantes décrivent les plus importantes.

Rubriques

- Mémoires tampons partagées
- Mémoires tampons WAL (Write-Ahead Log)

Mémoires tampons partagées

Le groupe de mémoires tampons partagées est une zone de mémoire RDS for PostgreSQL qui contient toutes les pages actuellement ou précédemment utilisées par les connexions d'applications. Une page correspond à la version mémoire d'un bloc de disque. Le groupe de mémoires tampons partagées met en cache les blocs de données lus sur le disque. Le groupe réduit la nécessité de relire les données à partir du disque, ce qui améliore l'efficacité de la base de données.

Chaque table et chaque index est stocké sous la forme d'un tableau de pages de taille fixe. Chaque bloc contient plusieurs tuples, qui correspondent à des lignes. Un tuple peut être stocké sur n'importe quelle page.

Le groupe de mémoires tampons partagées dispose d'une mémoire limitée. Si une nouvelle demande requiert une page qui n'est pas en mémoire, et qu'il n'y a plus de mémoire disponible, RDS for PostgreSQL expulse une page moins fréquemment utilisée pour répondre à la demande. La politique d'expulsion est implémentée par un algorithme de balayage horaire.

Le paramètre `shared_buffers` détermine la quantité de mémoire que le serveur consacre à la mise en cache des données. La valeur par défaut est définie sur `{DBInstanceClassMemory/32768}` octets, en fonction de la mémoire disponible pour l'instance de base de données.

Mémoires tampons WAL (Write-Ahead Log)

Une mémoire tampon WAL (Write-Ahead Log) contient des données de transaction que RDS for PostgreSQL écrit ultérieurement sur un stockage permanent. Le mécanisme WAL permet à RDS for PostgreSQL d'effectuer les opérations suivantes :

- Récupérer des données après une défaillance
- Réduisez le disque I/O en évitant les écritures fréquentes sur le disque

Quand un client modifie des données, RDS for PostgreSQL écrit les modifications dans la mémoire tampon WAL. Lorsque le client émet une commande `COMMIT`, le processus d'écriture WAL écrit les données de transaction dans le fichier WAL.

Le `wal_level` paramètre détermine la quantité d'informations écrites dans le WAL, avec des valeurs possibles telles que `minimal`, `replica` et `logical`.

Mémoire locale dans RDS for PostgreSQL

Chaque processus backend alloue de la mémoire locale pour le traitement des requêtes.

Rubriques

- Zone de mémoire de travail
- Zone de mémoire des travaux de maintenance
- Zone de mémoire tampon temporaire

Zone de mémoire de travail

La zone de mémoire de travail contient des données temporaires pour les requêtes qui effectuent des opérations de tri et de hachage. Par exemple, une requête contenant une clause `ORDER BY` effectue un tri. Les requêtes utilisent des tables de hachage dans les jointures de hachage et les agrégations.

`work_mem`Paramètre indiquant la quantité de mémoire à utiliser par les opérations de tri internes et les tables de hachage avant d'écrire sur des fichiers de disque temporaires, mesurée en mégaoctets. La valeur par défaut est de 4 Mo. Plusieurs sessions peuvent s'exécuter simultanément et chacune peut exécuter des opérations de maintenance en parallèle. La mémoire de travail totale utilisée peut donc être un multiple du paramètre `work_mem`.

Zone de mémoire des travaux de maintenance

La zone de mémoire des travaux de maintenance met les données en cache pour les opérations de maintenance. Ces opérations incluent l'opération VACUUM, la création d'un index et l'ajout de clés étrangères.

Le `maintenance_work_mem` paramètre indique la quantité maximale de mémoire à utiliser par les opérations de maintenance, mesurée en mégaoctets. La valeur par défaut est de 64 Mo. Une session de base de données ne peut exécuter qu'une seule opération de maintenance à la fois.

Zone de mémoire tampon temporaire

La zone de mémoire tampon temporaire met en cache les tables temporaires pour chaque session de base de données.

Chaque session alloue des mémoires tampons temporaires en fonction des besoins jusqu'à la limite que vous spécifiez. Lorsque la session se termine, le serveur efface le contenu des mémoires tampons.

Le `temp_buffers` paramètre définit le nombre maximum de tampons temporaires utilisés par chaque session, mesuré en mégaoctets. La valeur par défaut est de 8 Mo. Avant la première utilisation de tables temporaires au sein d'une session, vous pouvez modifier la valeur `temp_buffers`.

Processus RDS for PostgreSQL

RDS for PostgreSQL utilise plusieurs processus.

Rubriques

- Processus postmaster

- Processus backend
- Processus d'arrière-plan

Processus postmaster

Le processus postmaster est le premier qui est lancé lorsque vous démarrez RDS for PostgreSQL. Les principales responsabilités du processus postmaster sont les suivantes :

- Créer et surveiller les processus d'arrière-plan
- Recevoir les requêtes d'authentification des processus clients, et les authentifier avant d'autoriser la base de données à traiter les requêtes

Processus backend

Si le processus postmaster authentifie une requête client, il crée un nouveau processus backend, également appelé processus postgres. Un processus client se connecte à un seul processus backend. Le processus client et le processus backend communiquent directement sans intervention du processus postmaster.

Processus d'arrière-plan

Le processus postmaster crée plusieurs processus qui effectuent différentes tâches backend. Les plus importants sont les suivants :

- Dispositif d'écriture WAL

  RDS for PostgreSQL écrit les données contenues dans la mémoire tampon WAL (Write-Ahead Log) dans les fichiers journaux. Le principe de l'approche WAL est que la base de données ne peut pas écrire les modifications dans les fichiers de données tant que la base de données n'a pas écrit les enregistrements de journal décrivant ces modifications sur le disque. Le mécanisme WAL réduit les E/S disque et permet à RDS for PostgreSQL d'utiliser les journaux pour restaurer la base de données après une défaillance.
- Dispositif d'écriture d'arrière-plan

  Ce processus écrit périodiquement les pages modifiées des mémoires tampons vers les fichiers de données. Une page est considérée comme modifiée lorsqu'un processus backend la modifie en mémoire.
- Démon autovacuum

Le démon est composé des éléments suivants :

- Le lanceur autovacuum
- Les processus employés autovacuum

Lorsque la fonction autovacuum est activée, elle recherche les tables dans lesquelles un grand nombre de tuples ont été insérés, mis à jour ou supprimés. Les responsabilités du démon sont les suivantes :

- Récupérer ou réutiliser l'espace disque occupé par les lignes mises à jour ou supprimées
- Mettre à jour les statistiques utilisées par le planificateur
- Protéger contre la perte d'anciennes données en raison du renvoi à la ligne de l'ID de transaction

La fonction autovacuum automatise l'exécution des commandes `VACUUM` et `ANALYZE`. `VACUUM`présente les variantes suivantes : standard et complet. La variante standard s'exécute parallèlement à d'autres opérations de base de données. `VACUUM FULL` requiert un verrou exclusif sur la table sur laquelle il travaille. Il ne peut donc pas être exécuté en parallèle avec des opérations qui accèdent à la même table. `VACUUM`crée un I/O trafic important, ce qui peut nuire aux performances des autres sessions actives.

Événements d'attente RDS for PostgreSQL

Le tableau suivant répertorie les événements d'attente liés à RDS for PostgreSQL, qui révèlent souvent des problèmes de performances, et résume leurs causes et les actions correctives les plus courantes.

Événement d'attente	Définition
Cliente : ClientRead	Cet événement se produit quand RDS for PostgreSQL attend de recevoir des données du client.
Cliente : ClientWrite	Cet événement se produit quand RDS for PostgreSQL attend d'écrire des données sur le client.
CPU	Cet événement se produit lorsqu'un thread est actif dans l'UC ou qu'il est en attente d'UC.

Événement d'attente	Définition
IO:BufFileRead et IO:BufFileWrite	Ces événements se produisent quand RDS for PostgreSQL crée des fichiers temporaires.
IO : DataFileRead	Cet événement se produit lorsqu'une connexion attend qu'un processus backend lise une page requise à partir du stockage parce que la page n'est pas disponible dans la mémoire partagée.
IO : WALWrite	Cet événement indique quand RDS for PostgreSQL est en attente de l'écriture des tampons de journal d'écriture anticipée (WAL) dans un fichier WAL.
Lock:advisory	Cet événement se produit lorsqu'une application PostgreSQL utilise un verrou pour coordonner l'activité sur plusieurs sessions.
Lock:extend	Cet événement se produit lorsqu'un processus backend attend de verrouiller une relation pour l'étendre alors qu'un autre processus présente un verrou sur cette relation dans le même but.
Lock:Relation	Cet événement se produit lorsqu'une requête attend d'acquérir un verrou sur une table ou une vue actuellement verrouillée par une autre transaction.
Lock:transactionid	Cet événement se produit lorsqu'une transaction attend un verrou de niveau ligne.
Lock:tuple	Cet événement se produit lorsqu'un processus backend attend d'acquérir un verrou sur un tuple.
LWLock:BufferMapping (LWLock:buffer_mapping)	Cet événement se produit lorsqu'une session attend d'associer un bloc de données à une mémoire tampon dans le groupe de mémoires tampons partagées.

Événement d'attente	Définition
LWLock:BufferIO (IPC:BufferIO)	Cet événement se produit lorsque RDS pour PostgreSQL attend que d'autres processus input/output terminent leurs opérations (E/S) lorsqu'ils tentent simultanément d'accéder à une page.
LWLock:buffer_content (BufferContent)	Cet événement se produit lorsqu'une session attend de lire ou d'écrire une page de données en mémoire alors que celle-ci est verrouillée en écriture dans une autre session.
LWLock:lock_manager (:lockmanager) LWLock	Cet événement se produit lorsque le moteur RDS for PostgreSQL conserve la zone de mémoire du verrou partagé pour allouer, vérifier et libérer un verrou quand il est impossible d'utiliser un verrou à chemin d'accès rapide.
Timeout:PgSleep	Cet événement se produit lorsqu'un processus serveur a appelé la fonction `pg_sleep` et attend l'expiration du délai de mise en veille.
Timeout:VacuumDelay	Cet événement indique que le processus vacuum est en veille car la limite de coût estimée a été atteinte.

Cliente : ClientRead

L'événement `Client:ClientRead` se produit quand RDS for PostgreSQL attend de recevoir des données du client.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour RDS for PostgreSQL versions 10 et ultérieures.

Contexte

Une instance de base de données RDS for PostgreSQL attend de recevoir des données du client. L'instance de base de données RDS for PostgreSQL doit recevoir les données du client avant de pouvoir envoyer plus de données au client. La période pendant laquelle l'instance attend avant de recevoir les données du client est un événement `Client:ClientRead`.

Causes probables de l'augmentation du nombre d'événements d'attente

Les principales causes de l'événement d'attente `Client:ClientRead` sont les suivantes :

Latence réseau accrue

La latence réseau peut être accrue entre l'instance de base de données RDS for PostgreSQL et le client. Une latence réseau plus élevée augmente le temps de réception des données du client par l'instance de base de données.

Charge accrue sur le client

Le client peut être soumis à une pression exercée sur l'UC ou à une saturation du réseau. Une augmentation de la charge sur le client peut retarder la transmission des données du client vers l'instance de base de données RDS for PostgreSQL.

Nombre excessif d'allers-retours réseau

Un grand nombre d'allers-retours réseau entre l'instance de base de données RDS for PostgreSQL et le client peut retarder la transmission des données du client vers l'instance de base de données RDS for PostgreSQL.

Opération de copie importante

Lors d'une opération de copie, les données sont transférées du système de fichiers du client vers l'instance de base de données RDS for PostgreSQL. L'envoi d'une grande quantité de données à l'instance de base de données peut retarder la transmission des données du client vers l'instance de base de données.

Connexion client inactive

Quand un client se connecte à l'instance de base de données RDS for PostgreSQL alors que son état est `idle in transaction`, l'instance de base de données peut attendre que le client envoie plus de données ou émette une commande. Une connexion dans ces conditions peut entraîner une augmentation des événements `Client:ClientRead`.

PgBouncer utilisé pour le regroupement de connexions

PgBouncer possède un paramètre de configuration réseau de bas niveau appelé`pkt_buf`, qui est défini sur 4 096 par défaut. Si la charge de travail envoie des paquets de requêtes de plus de 4 096 octets PgBouncer, nous vous recommandons d'augmenter le `pkt_buf` paramètre à 8 192. Si le nouveau paramètre ne permet pas de réduire le nombre d'événements `Client:ClientRead`, nous vous recommandons d'augmenter le paramètre `pkt_buf` en le définissant sur des valeurs plus élevées, telles que 16 384 ou 32 768. Si le texte de la requête est volumineux, un paramètre plus élevé peut être particulièrement utile.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Placement des clients dans la même zone de disponibilité et le même sous-réseau VPC que l'instance
- Procédez à la mise à l'échelle du client
- Utilisez des instances de la génération actuelle
- Augmentez la bande passante réseau
- Surveillez les valeurs maximales des métriques de performances réseau
- Surveillez les transactions dont l'état est « idle in transaction » (transaction inactive)

Placement des clients dans la même zone de disponibilité et le même sous-réseau VPC que l'instance

Pour réduire la latence réseau et augmenter le débit, placez les clients dans la même zone de disponibilité et le même sous-réseau de cloud privé virtuel (VPC) que l'instance de base de données RDS for PostgreSQL. Veillez à ce que les clients soient géographiquement aussi proches que possible de l'instance de base de données.

Procédez à la mise à l'échelle du client

À l'aide d'Amazon CloudWatch ou d'autres indicateurs de l'hôte, déterminez si votre client est actuellement limité par le processeur ou la bande passante du réseau, ou les deux. Si le client est soumis à des contraintes, mettez-le à l'échelle en conséquence.

Utilisez des instances de la génération actuelle

La classe d'instance de base de données que vous utilisez ne prend peut-être pas en charge les trames jumbo. Si vous exécutez votre application sur Amazon EC2, pensez à utiliser une instance de génération actuelle pour le client. Configurez également l'unité de transmission maximale (MTU) sur le système d'exploitation client. Cette technique peut réduire le nombre d'allers-retours réseau et augmenter le débit du réseau. Pour plus d'informations, consultez la section Cadres Jumbo (9001 MTU) dans le guide de l'utilisateur Amazon EC2.

Pour plus d'informations sur les classes d'instances de base de données, consultez Classes d'instances de base de données . Pour déterminer quelle classe d'instance de base de données est l'équivalent d'un type d'instance Amazon EC2, placez `db.` avant le nom du type d'instance Amazon EC2. Par exemple, l'instance Amazon EC2 `r5.8xlarge` est l'équivalent de la classe d'instance de base de données `db.r5.8xlarge`.

Augmentez la bande passante réseau

Utilisez `NetworkReceiveThroughput` les CloudWatch métriques `NetworkTransmitThroughput` Amazon pour surveiller le trafic réseau entrant et sortant sur l'instance de base de données. Ces métriques peuvent vous aider à déterminer si la bande passante réseau est suffisante pour votre charge de travail.

Si la bande passante réseau est insuffisante, augmentez-la. Si le AWS client ou votre instance de base de données atteint les limites de bande passante du réseau, le seul moyen d'augmenter la bande passante est d'augmenter la taille de votre instance de base de données. Pour plus d'informations, consultez Types de classes d'instance de base de données.

Pour plus d'informations sur CloudWatch les métriques, consultez CloudWatch Métriques Amazon pour Amazon RDS.

Surveillez les valeurs maximales des métriques de performances réseau

Si vous utilisez des clients Amazon EC2, Amazon EC2 fournit des valeurs maximales pour les métriques de performances réseau, y compris pour la bande passante réseau entrante et sortante

agrégée. Il assure également le suivi des connexions pour garantir un retour optimal des paquets et un accès aux services locaux de liaison pour des services tels que le système de noms de domaine (DNS). Pour surveiller ces valeurs maximales, utilisez un pilote réseau amélioré à jour et surveillez les performances réseau de votre client.

Pour plus d'informations, consultez Surveiller les performances réseau de votre instance Amazon EC2 dans le guide de l'utilisateur Amazon EC2 et Surveiller les performances réseau de votre instance Amazon EC2 dans le guide de l'utilisateur Amazon EC2.

Surveillez les transactions dont l'état est « idle in transaction » (transaction inactive)

Déterminez si le nombre de connexions `idle in transaction` est croissant. Pour ce faire, surveillez la colonne `state` de la table `pg_stat_activity`. Vous pouvez identifier la source des connexions en exécutant une requête semblable à la suivante.

```
select client_addr, state, count(1) from pg_stat_activity
where state like 'idle in transaction%'
group by 1,2
order by 3 desc
```

Cliente : ClientWrite

L'événement `Client:ClientWrite` se produit quand RDS for PostgreSQL attend d'écrire des données sur le client.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour RDS for PostgreSQL versions 10 et ultérieures.

Contexte

Un processus client doit lire toutes les données reçues d'un cluster de bases de données RDS for PostgreSQL avant que ce cluster puisse envoyer plus de données. La période pendant laquelle le cluster attend avant d'envoyer plus de données au client est un événement `Client:ClientWrite`.

Une diminution du débit réseau entre l'instance de base de données RDS for PostgreSQL et le client peut provoquer cet événement. Cet événement peut également se produire lorsque le client est soumis à une pression exercée sur l'UC ou à une saturation du réseau.. On parle de pression exercée sur l'UC lorsque l'UC est entièrement utilisée et que des tâches attendent du temps UC. On parle de saturation du réseau lorsque le réseau situé entre la base de données et le client transporte plus de données qu'il ne peut en gérer.

Causes probables de l'augmentation du nombre d'événements d'attente

Les principales causes de l'événement d'attente `Client:ClientWrite` sont les suivantes :

Latence réseau accrue

La latence réseau peut être accrue entre l'instance de base de données RDS for PostgreSQL et le client. Une latence réseau plus élevée augmente le temps nécessaire au client pour recevoir les données.

Charge accrue sur le client

Le client peut être soumis à une pression exercée sur l'UC ou à une saturation du réseau. Une augmentation de la charge sur le client retarde la réception des données de l'instance de base de données RDS for PostgreSQL.

Gros volume de données envoyé au client

L'instance de base de données RDS for PostgreSQL peut envoyer une grande quantité de données au client. Un client peut ne pas être en mesure de recevoir les données aussi rapidement que le cluster les envoie. Certaines activités comme la copie d'une table volumineuse peuvent entraîner une augmentation des événements `Client:ClientWrite`.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Placez les clients dans la même zone de disponibilité et le même sous-réseau VPC que le cluster
- Utilisez des instances de la génération actuelle
- Réduisez la quantité de données envoyée au client
- Procédez à la mise à l'échelle du client

Placez les clients dans la même zone de disponibilité et le même sous-réseau VPC que le cluster

Pour réduire la latence réseau et augmenter le débit, placez les clients dans la même zone de disponibilité et le même sous-réseau de cloud privé virtuel (VPC) que l'instance de base de données RDS for PostgreSQL.

Utilisez des instances de la génération actuelle

La classe d'instance de base de données que vous utilisez ne prend peut-être pas en charge les trames jumbo. Si vous exécutez votre application sur Amazon EC2, pensez à utiliser une instance de génération actuelle pour le client. Configurez également l'unité de transmission maximale (MTU) sur le système d'exploitation client. Cette technique peut réduire le nombre d'allers-retours réseau et augmenter le débit du réseau. Pour plus d'informations, consultez la section Cadres Jumbo (9001 MTU) dans le guide de l'utilisateur Amazon EC2.

Pour plus d'informations sur les classes d'instances de base de données, consultez Classes d'instances de base de données . Pour déterminer quelle classe d'instance de base de données est l'équivalent d'un type d'instance Amazon EC2, placez `db.` avant le nom du type d'instance Amazon EC2. Par exemple, l'instance Amazon EC2 `r5.8xlarge` est l'équivalent de la classe d'instance de base de données `db.r5.8xlarge`.

Réduisez la quantité de données envoyée au client

Lorsque cela est possible, ajustez votre application pour réduire la quantité de données que l'instance de base de données RDS for PostgreSQL envoie au client. Ces ajustements permettent d'éviter les conflits liés à l'UC et au réseau sur le client.

Procédez à la mise à l'échelle du client

À l'aide d'Amazon CloudWatch ou d'autres indicateurs de l'hôte, déterminez si votre client est actuellement limité par le processeur ou la bande passante du réseau, ou les deux. Si le client est soumis à des contraintes, mettez-le à l'échelle en conséquence.

CPU

Cet événement se produit lorsqu'un thread est actif dans l'UC ou qu'il est en attente d'UC.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente s'appliquent à toutes les versions de RDS for PostgreSQL.

Contexte

L'unité centrale (UC) est le composant d'un ordinateur qui exécute les instructions. Par exemple, les instructions de l'UC effectuent des opérations arithmétiques et échangent des données en mémoire. Si une requête augmente le nombre d'instructions qu'elle exécute par le biais du moteur de base de données, le temps passé à exécuter la requête augmente. La planification du temps UC consiste à accorder du temps UC à un processus. La planification est orchestrée par le noyau du système d'exploitation.

Rubriques

- Comment savoir quand cette attente se produit
- Métrique DBLoadCPU
- Métriques os.cpuUtilization
- Cause probable de la planification du temps UC

Comment savoir quand cette attente se produit

Cet événement d'attente `CPU` indique qu'un processus backend est actif dans l'UC ou qu'il est en attente d'UC. Cela se produit lorsqu'une requête contient les informations suivantes :

- La colonne `pg_stat_activity.state` indique la valeur `active`.

- Les colonnes `wait_event_type` et `wait_event` de `pg_stat_activity` indiquent toutes les deux `null`.

Pour voir les processus backend qui utilisent l'UC ou sont en attente de celle-ci, exécutez la requête suivante.

```
SELECT *
FROM   pg_stat_activity
WHERE  state = 'active'
AND    wait_event_type IS NULL
AND    wait_event IS NULL;
```

Métrique DBLoadCPU

La métrique Performance Insights de l'UC est `DBLoadCPU`. La valeur de `DBLoadCPU` peut être différente de la valeur de la métrique Amazon CloudWatch `CPUUtilization`. Cette dernière est collectée à partir de l'hyperviseur pour une instance de base de données.

Métriques os.cpuUtilization

Les métriques du système d'exploitation Performance Insights fournissent des informations détaillées sur l'utilisation de l'UC. Par exemple, vous pouvez afficher les métriques suivantes :

- `os.cpuUtilization.nice.avg`
- `os.cpuUtilization.total.avg`
- `os.cpuUtilization.wait.avg`
- `os.cpuUtilization.idle.avg`

Performance Insights présente l'utilisation du processeur par le moteur de base de données sous la forme `os.cpuUtilization.nice.avg`.

Cause probable de la planification du temps UC

Le noyau du système d'exploitation (SE) gère la planification pour le processeur. Quand le processeur est actif, il se peut qu'un processus doive attendre avant d'être planifié. Le processeur est actif pendant qu'il effectue des calculs. Il est également actif quand il dispose d'un thread inactif qui ne s'exécute pas, c'est-à-dire d'un thread inactif qui attend des E/S mémoire. Ce type d'E/S domine la charge de travail de base de données standard.

Les processus sont susceptibles d'attendre d'être planifiés sur une UC lorsque les conditions suivantes sont réunies :

- La métrique CloudWatch `CPUUtilization` est proche de 100 %.
- La charge moyenne est supérieure au nombre de vCPU, ce qui indique une charge importante. Vous trouverez la métrique `loadAverageMinute` dans la section relative aux métriques du système d'exploitation de Performance Insights.

Causes probables de l'augmentation du nombre d'événements d'attente

Un événement d'attente UC trop fréquent peut révéler un problème de performances dont les principales causes sont les suivantes.

Rubriques

- Causes probables des pics soudains
- Causes probables d'une fréquence élevée sur le long terme
- Cas particuliers

Causes probables des pics soudains

Les causes les plus probables des pics soudains sont les suivantes :

- Votre application a ouvert un trop grand nombre de connexions simultanées à la base de données. Ce scénario est connu sous le nom de « connection storm » (tempête de connexions).
- La charge de travail de votre application a connu l'un des changements suivants :
  - Nouvelles requêtes
  - Augmentation de la taille du jeu de données
  - Maintenance ou création d'index
  - Nouvelles fonctions
  - Nouveaux opérateurs
  - Augmentation des exécutions de requêtes parallèles
- Vos plans d'exécution des requêtes ont changé. Dans certains cas, un changement peut entraîner une augmentation des mémoires tampons. Par exemple, la requête utilise désormais une analyse séquentielle alors qu'elle utilisait auparavant un index. Dans ce cas, les requêtes ont besoin de plus d'UC pour atteindre le même objectif.

Causes probables d'une fréquence élevée sur le long terme

Causes les plus probables de la répétition des événements sur une longue période :

- Un trop grand nombre de processus backend s'exécutent simultanément sur l'UC. Ces processus peuvent être des employés parallèles.
- Les performances des requêtes ne sont pas optimales car elles nécessitent un grand nombre de mémoires tampons.

Cas particuliers

Si aucune des causes probables ne s'avère être la bonne, les situations suivantes peuvent se produire :

- L'UC échange des processus en entrée et en sortie.
- Le processeur est peut-être en train de gérer les entrées des tables de pages si la fonctionnalité huge pages (grandes pages) a été désactivée. Cette fonctionnalité de gestion de la mémoire est activée par défaut pour toutes les classes d'instances de base de données autres que les classes micro, petites et moyennes. Pour de plus amples informations, veuillez consulter De grandes pages RDS pour Postgre SQL .

Actions

Si l'événement d'attente CPU domine l'activité de la base de données, cela n'indique pas nécessairement un problème de performance. Ne réagissez à cet événement qu'en cas de dégradation des performances.

Rubriques

- Vérifiez que la base de données n'est pas à l'origine de l'augmentation du nombre d'événements d'attente UC
- Déterminez si le nombre de connexions a augmenté
- Réagissez aux changements de charge de travail

Vérifiez que la base de données n'est pas à l'origine de l'augmentation du nombre d'événements d'attente UC

Examinez la métrique `os.cpuUtilization.nice.avg` dans Performance Insights. Si cette valeur est bien inférieure à l'utilisation de l'UC, cela signifie que des processus non liés à la base de données contribuent majoritairement aux événements d'attente UC.

Déterminez si le nombre de connexions a augmenté

Examinez la métrique `DatabaseConnections` dans Amazon CloudWatch. L'action à entreprendre varie selon que ce nombre augmente ou diminue pendant la période d'augmentation des événements d'attente UC.

Les connexions ont augmenté

Si le nombre de connexions a augmenté, comparez le nombre de processus backend utilisant l'UC au nombre de vCPU. Les scénarios possibles sont les suivants :

- Le nombre de processus backend utilisant l'UC est inférieur au nombre de vCPU.

  Dans ce cas, le nombre de connexions n'est pas un problème. Cependant, vous pouvez toujours essayer de réduire l'utilisation de l'UC.
- Le nombre de processus backend utilisant l'UC est supérieur au nombre de vCPU.

  Dans ce cas, procédez comme suit :

  - Réduisez le nombre de processus backend connectés à votre base de données. Par exemple, implémentez une solution de regroupement des connexions telle que RDS Proxy. Pour en savoir plus, veuillez consulter la section Proxy Amazon RDS .
  - Augmentez la taille de votre instance pour bénéficier d'un plus grand nombre de vCPU.
  - Redirigez certaines charges de travail en lecture seule vers des nœuds de lecture, le cas échéant.

Les connexions n'ont pas augmenté

Examinez les métriques `blks_hit` dans Performance Insights. Recherchez une corrélation entre une augmentation de `blks_hit` et l'utilisation de l'UC. Les scénarios possibles sont les suivants :

- L'utilisation de l'UC et `blks_hit` sont corrélés.

Dans ce cas, recherchez les principales instructions SQL liées à l'utilisation de l'UC ainsi que les modifications apportées au plan. Vous pouvez utiliser l'une des techniques suivantes :

- Décrivez les plans manuellement et comparez-les au plan d'exécution attendu.
- Recherchez une augmentation des accès en bloc par seconde et des accès en bloc locaux par seconde. Dans la section Top SQL (Principaux éléments SQL) du tableau de bord Performance Insights, choisissez Preferences (Préférences).

- L'utilisation de l'UC et `blks_hit` ne sont pas corrélés.

  Dans ce cas, déterminez si l'une des situations suivantes se produit :

  - L'application se connecte et se déconnecte rapidement de la base de données.

    Diagnostiquez ce comportement en activant `log_connections` et `log_disconnections`, puis en analysant les journaux PostgreSQL. Pensez à utiliser l'analyseur de journaux `pgbadger`. Pour de plus amples informations, veuillez consulter https://github.com/darold/pgbadger.

  - Le système d'exploitation est surchargé.

    Dans ce cas, Performance Insights montre que les processus backend utilisent l'UC plus longtemps que d'habitude. Recherchez des preuves dans les métriques Performance Insights `os.cpuUtilization` ou dans la métrique CloudWatch `CPUUtilization`. Si le système d'exploitation est surchargé, consultez les métriques de surveillance améliorée pour approfondir le diagnostic. Plus précisément, examinez la liste des processus et le pourcentage d'UC utilisé par chaque processus.

  - Les principales instructions SQL utilisent trop d'UC.

    Examinez les instructions liées à l'utilisation de l'UC pour voir si elles peuvent en utiliser moins. Exécutez une commande `EXPLAIN` et concentrez-vous sur les nœuds du plan qui ont le plus d'impact. Utilisez un visualiseur de plan d'exécution PostgreSQL. Pour essayer cet outil, consultez http://explain.dalibo.com/.

Réagissez aux changements de charge de travail

Si votre charge de travail a changé, recherchez les types de changements suivants :

Nouvelles requêtes

Déterminez si les nouvelles requêtes sont attendues. Si oui, assurez-vous que leur plan d'exécution et le nombre d'exécutions par seconde sont attendus.

Augmentation de la taille du jeu de données

Déterminez si un partitionnement, s'il n'est pas déjà implémenté, peut être utile. Cette stratégie peut réduire le nombre de pages qu'une requête doit récupérer.

Maintenance ou création d'index

Vérifiez si le calendrier de maintenance est attendu. Une bonne pratique consiste à planifier des activités de maintenance en dehors des périodes de pointe.

Nouvelles fonctions

Déterminez si ces fonctions s'exécutent comme prévu lors des tests. Plus précisément, déterminez si le nombre d'exécutions par seconde est attendu.

Nouveaux opérateurs

Déterminez s'ils fonctionnent comme prévu lors des tests.

Augmentation du nombre de requêtes exécutées en parallèle

Déterminez si l'une des situations suivantes s'est produite :

- Les relations ou index impliqués ont soudainement augmenté en termes de taille, de sorte qu'ils sont considérablement différents de `min_parallel_table_scan_size` ou `min_parallel_index_scan_size`.
- Des modifications récentes ont été apportées à `parallel_setup_cost` ou `parallel_tuple_cost`.
- Des modifications récentes ont été apportées à `max_parallel_workers` ou `max_parallel_workers_per_gather`.

IO:BufFileRead et IO:BufFileWrite

Les événements `IO:BufFileRead` et `IO:BufFileWrite` se produisent quand RDS for PostgreSQL crée des fichiers temporaires. Lorsque des opérations requièrent plus de mémoire que n'en confèrent les paramètres de mémoire de travail définis, elles écrivent des données temporaires sur un stockage permanent. Cette opération est parfois appelée « déversement sur disque ».

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente

- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

`IO:BufFileRead` et `IO:BufFileWrite` se rapportent à la zone de mémoire de travail et à la zone de mémoire des travaux de maintenance. Pour plus d'informations sur ces zones de mémoire locale, consultez Consommation des ressources dans la documentation PostgreSQL.

La valeur par défaut du paramètre `work_mem` est 4 Mo. Si une session effectue des opérations en parallèle, chaque employé gérant le parallélisme utilise 4 Mo de mémoire. Par conséquent, définissez `work_mem` prudemment. Si vous augmentez trop la valeur, une base de données exécutant plusieurs sessions peut utiliser trop de mémoire. Si vous définissez une valeur trop faible, RDS for PostgreSQL crée des fichiers temporaires dans le stockage local. Les I/O disque de ces fichiers temporaires peuvent réduire les performances.

Si vous observez la séquence d'événements suivante, votre base de données génère peut-être des fichiers temporaires :

1. Diminution soudaine et brutale de la disponibilité
2. Récupération rapide de l'espace libre

Vous pouvez également observer un schéma en dents de scie. Ce schéma peut indiquer que votre base de données crée constamment de petits fichiers.

Causes probables de l'augmentation du nombre d'événements d'attente

En général, ces événements d'attente sont provoqués par des opérations qui utilisent plus de mémoire que n'en allouent les paramètres `work_mem` ou `maintenance_work_mem`. Pour compenser, les opérations écrivent dans des fichiers temporaires. Les principales causes des événements `IO:BufFileRead` et `IO:BufFileWrite` sont les suivantes :

Requêtes nécessitant plus de mémoire qu'il n'en existe dans la zone de mémoire de travail

Les requêtes présentant les caractéristiques suivantes utilisent la zone de mémoire de travail :

- Jointures par hachage
- `ORDER BY` Clause
- `GROUP BY` Clause
- `DISTINCT`
- Fonctions de fenêtrage
- `CREATE TABLE AS SELECT`
- Actualisation de la vue matérialisée

Instructions nécessitant plus de mémoire qu'il n'en existe dans la zone de mémoire des travaux de maintenance

Les instructions suivantes utilisent la zone de mémoire des travaux de maintenance :

- `CREATE INDEX`
- `CLUSTER`

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Identifiez le problème
- Examinez vos requêtes de jointure
- Examinez vos requêtes ORDER BY et GROUP BY
- Évitez d'utiliser l'opération DISTINCT
- Envisagez d'utiliser des fonctions de fenêtrage à la place des fonctions GROUP BY
- Examinez les vues matérialisées et les instructions CTAS
- Reconstruction des index à l'aide de pg_repack
- Augmentez maintenance_work_mem lorsque vous mettez des tables en cluster
- Réglez la mémoire de manière à éviter IO:BufFileRead et IO:BufFileWrite

Identifiez le problème

Imaginons une situation dans laquelle Performance Insights n'est pas activé et dans laquelle vous soupçonnez que les événements `IO:BufFileRead` et `IO:BufFileWrite` se produisent plus

souvent qu'à l'accoutumée. Pour identifier la source du problème, vous pouvez définir le paramètre `log_temp_files` de manière à consigner toutes les requêtes qui génèrent un nombre de Ko de fichiers temporaires supérieur au seuil spécifié. Par défaut, `log_temp_files` est défini sur `-1`, ce qui désactive cette fonctionnalité de journalisation. Si vous définissez ce paramètre sur `0`, RDS for PostgreSQL consigne tous les fichiers temporaires. Si vous définissez la valeur `1024`, RDS for PostgreSQL consigne toutes les requêtes qui produisent des fichiers temporaires de plus de 1 Mo. Pour en savoir plus sur `log_temp_files`, consultez Error Reporting and Logging dans la documentation PostgreSQL.

Examinez vos requêtes de jointure

Il est probable que votre requête utilise des jointures. Par exemple, la requête suivante joint quatre tables.

```
SELECT *
       FROM "order"
 INNER JOIN order_item
       ON (order.id = order_item.order_id)
 INNER JOIN customer
       ON (customer.id = order.customer_id)
 INNER JOIN customer_address
       ON (customer_address.customer_id = customer.id AND
           order.customer_address_id = customer_address.id)
 WHERE customer.id = 1234567890;
```

Les pics d'utilisation des fichiers temporaires peuvent être dus à un problème dans la requête proprement dite. Par exemple, une clause rompue peut ne pas filtrer correctement les jointures. Prenons la deuxième jointure interne de l'exemple suivant.

```
SELECT *
       FROM "order"
 INNER JOIN order_item
       ON (order.id = order_item.order_id)
 INNER JOIN customer
       ON (customer.id = customer.id)
 INNER JOIN customer_address
       ON (customer_address.customer_id = customer.id AND
           order.customer_address_id = customer_address.id)
 WHERE customer.id = 1234567890;
```

La requête précédente joint par erreur `customer.id` à `customer.id`, générant un produit cartésien entre chaque client et chaque commande. Ce type de jointure accidentelle génère des fichiers temporaires volumineux. Selon la taille des tables, une requête cartésienne peut même saturer le stockage. Votre application peut présenter des jointures cartésiennes lorsque les conditions suivantes sont réunies :

- Vous observez des baisses importantes et brutales de la disponibilité du stockage, suivies d'une récupération rapide.
- Aucun index n'est créé.
- Aucune instruction `CREATE TABLE FROM SELECT` n'est émise.
- Aucune vue matérialisée n'est actualisée.

Pour savoir si les tables sont jointes à l'aide des clés appropriées, examinez votre requête et les directives de mappage objet-relationnel. Gardez à l'esprit que certaines requêtes de votre application ne sont pas appelées en permanence, et que certaines requêtes sont générées dynamiquement.

Examinez vos requêtes ORDER BY et GROUP BY

Dans certains cas, une clause `ORDER BY` peut entraîner un nombre excessif de fichiers temporaires. Considérez les directives suivantes :

- N'incluez des colonnes dans une clause `ORDER BY` que lorsqu'elles doivent être classées. Cette directive est particulièrement importante pour les requêtes qui renvoient des milliers de lignes et spécifient de nombreuses colonnes dans la clause `ORDER BY`.
- N'hésitez pas à créer des index pour accélérer les clauses `ORDER BY` lorsqu'elles correspondent à des colonnes qui présentent le même ordre croissant ou décroissant. Les index partiels sont préférables car ils sont plus petits. Les index de petite taille sont lus et parcourus plus rapidement.
- Si vous créez des index pour des colonnes qui peuvent accepter des valeurs nulles, déterminez si vous souhaitez que les valeurs nulles soient stockées à la fin ou au début des index.

  Si possible, réduisez le nombre de lignes à classer en filtrant l'ensemble de résultats. Si vous utilisez des instructions ou des sous-requêtes liées à la clause `WITH`, n'oubliez pas qu'une requête interne génère un ensemble de résultats et le transmet à la requête externe. Plus le nombre de lignes qu'une requête peut filtrer est élevé, moins elle a de classement à effectuer.
- Si vous n'avez pas besoin de l'ensemble de résultats complet, utilisez la clause `LIMIT`. Par exemple, si vous avez uniquement besoin des cinq premières lignes, une requête utilisant la clause

LIMIT ne continue pas à générer des résultats. La requête a ainsi besoin de moins de mémoire et de moins de fichiers temporaires.

Une requête qui utilise une clause GROUP BY peut également avoir besoin de fichiers temporaires. Les requêtes GROUP BY résument les valeurs à l'aide de fonctions telles que les suivantes :

- COUNT
- AVG
- MIN
- MAX
- SUM
- STDDEV

Pour régler les requêtes GROUP BY, suivez les recommandations relatives aux requêtes ORDER BY.

Évitez d'utiliser l'opération DISTINCT

Dans la mesure du possible, évitez d'utiliser l'opération DISTINCT pour supprimer les lignes en double. Plus votre requête renvoie de lignes inutiles et en double, plus l'opération DISTINCT devient coûteuse. Si possible, ajoutez des filtres dans la clause WHERE, même si vous utilisez les mêmes filtres pour différentes tables. Un filtrage de la requête et une jointure correctes vous permettent d'améliorer les performances et de réduire l'utilisation des ressources. Ils vous permettent également d'éviter les rapports et les résultats incorrects.

Si vous devez utiliser DISTINCT pour plusieurs lignes d'une même table, n'hésitez pas à créer un index composite. Le regroupement de plusieurs colonnes dans un index peut améliorer le temps nécessaire à l'évaluation des lignes distinctes. De plus, si vous utilisez RDS for PostgreSQL version 10 ou ultérieure, vous pouvez corréler les statistiques entre plusieurs colonnes à l'aide de la commande CREATE STATISTICS.

Envisagez d'utiliser des fonctions de fenêtrage à la place des fonctions GROUP BY

Avec GROUP BY, vous modifiez l'ensemble de résultats, puis récupérez le résultat agrégé. Avec les fonctions de fenêtrage, vous pouvez agréger les données sans modifier l'ensemble de résultats. Une fonction de fenêtrage utilise la clause OVER pour effectuer des calculs sur les ensembles définis par la requête, en corrélant une ligne avec une autre. Les fonctions de fenêtrage vous permettent d'utiliser toutes les fonctions GROUP BY ainsi que les fonctions suivantes :

- RANK
- ARRAY_AGG
- ROW_NUMBER
- LAG
- LEAD

Pour réduire le nombre de fichiers temporaires générés par une fonction de fenêtrage, supprimez les doublons d'un même ensemble de résultats lorsque vous avez besoin de deux agrégations distinctes. Considérons la requête suivante :

```
SELECT sum(salary) OVER (PARTITION BY dept ORDER BY salary DESC) as sum_salary
     , avg(salary) OVER (PARTITION BY dept ORDER BY salary ASC) as avg_salary
  FROM empsalary;
```

Vous pouvez réécrire la requête en utilisant la clause `WINDOW` comme suit.

```
SELECT sum(salary) OVER w as sum_salary
         , avg(salary) OVER w as_avg_salary
    FROM empsalary
  WINDOW w AS (PARTITION BY dept ORDER BY salary DESC);
```

Par défaut, le planificateur d'exécution RDS for PostgreSQL regroupe les nœuds similaires afin de ne pas dupliquer les opérations. Toutefois, en utilisant une déclaration explicite pour le bloc de fenêtres, vous pouvez gérer la requête plus facilement. Vous pouvez également améliorer les performances en empêchant la duplication.

Examinez les vues matérialisées et les instructions CTAS

Lorsqu'une vue matérialisée est actualisée, elle exécute une requête. Cette requête peut contenir une opération telle que `GROUP BY`, `ORDER BY` ou `DISTINCT`. Lors d'une actualisation, vous pouvez observer un grand nombre de fichiers temporaires et les événements d'attente `IO:BufFileWrite` et `IO:BufFileRead`. De même, lorsque vous créez une table basée sur une instruction `SELECT`, l'instruction `CREATE TABLE` exécute une requête. Pour réduire le nombre de fichiers temporaires nécessaires, optimisez la requête.

Reconstruction des index à l'aide de pg_repack

Lorsque vous créez un index, le moteur classe l'ensemble de résultats. À mesure que la taille des tables augmente et que les valeurs de la colonne indexée se diversifient, les fichiers temporaires ont besoin de plus d'espace. Dans la plupart des cas, vous ne pouvez pas empêcher la création de fichiers temporaires pour les tables volumineuses sans modifier la zone de mémoire des travaux de maintenance. Pour plus d'informations sur `maintenance_work_mem`, consultez `maintenance_work_mem` dans la documentation PostgreSQL.

Une solution de contournement possible lors de la recréation d'un index volumineux consiste à utiliser l'extension pg_repack. Pour en savoir plus, consultez Reorganize tables in PostgreSQL databases with minimal locks dans la documentation pg_repack. Pour obtenir des informations sur la configuration de l'extension dans votre instance de base de données RDS for PostgreSQL, consultez Réduction du ballonnement des tables et des index avec l'extension pg_repack.

Augmentez maintenance_work_mem lorsque vous mettez des tables en cluster

La commande `CLUSTER` met en cluster la table spécifiée par table_name à partir d'un index existant spécifié par index_name. RDS for PostgreSQL recrée physiquement la table en suivant l'ordre d'un index donné.

Lorsque le stockage magnétique était prédominant, la mise en cluster était courante car le débit de stockage était limité. Maintenant que le stockage SSD est plus répandu, la mise en cluster est moins fréquente. Toutefois, en mettant des tables en cluster, vous pouvez encore bénéficier d'une légère amélioration des performances en fonction de la taille de la table, de l'index, de la requête, etc.

Si vous exécutez la commande `CLUSTER` et observez les événements d'attente `IO:BufFileWrite` et `IO:BufFileRead`, réglez `maintenance_work_mem`. Augmentez la taille de la mémoire en la définissant sur une valeur relativement élevée. Une valeur élevée permettra au moteur d'utiliser davantage de mémoire pour l'opération de mise en cluster.

Réglez la mémoire de manière à éviter IO:BufFileRead et IO:BufFileWrite

Dans certaines situations, vous devez régler la mémoire. Votre objectif est d'équilibrer la mémoire entre les zones de consommation suivantes à l'aide des paramètres appropriés, comme suit.

- La valeur `work_mem`
- La mémoire restant après déduction de la valeur `shared_buffers`
- Nombre maximal de connexions ouvertes et en cours d'utilisation, qui est limité par `max_connections`

Pour plus d'informations sur le réglage de la mémoire, consultez Consommation des ressources dans la documentation PostgreSQL.

Augmentez la taille de la zone de mémoire de travail

Dans certains cas, votre seule option consiste à augmenter la mémoire utilisée par votre session. Si vos requêtes sont correctement écrites et utilisent les bonnes clés pour les jointures, augmentez la valeur `work_mem`.

Pour savoir combien de fichiers temporaires une requête génère, définissez `log_temp_files` sur `0`. Si vous définissez la valeur `work_mem` sur la valeur maximale identifiée dans les journaux, vous empêchez la requête de générer des fichiers temporaires. Toutefois, `work_mem` définit le maximum par nœud du plan pour chaque connexion ou employé parallèle. Si la base de données compte 5 000 connexions, et si chacune d'entre elles utilise 256 Mio de mémoire, le moteur a besoin de 1,2 Tio de RAM. Votre instance risque donc de manquer de mémoire.

Réservez suffisamment de mémoire pour le groupe de mémoires tampons partagées

Votre base de données utilise des zones de mémoire telles que le groupe de mémoires tampons partagées, et pas seulement la zone de mémoire de travail. Prenez en compte les besoins de ces zones de mémoire supplémentaires avant d'augmenter `work_mem`.

Par exemple, supposons que votre classe d'instances RDS for PostgreSQL est db.r5.2xlarge. Cette classe dispose de 64 Gio de mémoire. Par défaut, 25 % de la mémoire est réservée pour le groupe de mémoires tampons partagées. Après avoir soustrait la quantité allouée à la zone de mémoire partagée, il reste 16 384 Mo. Évitez d'allouer la mémoire restante exclusivement à la zone de mémoire de travail, car le système d'exploitation et le moteur ont également besoin de mémoire.

La mémoire que vous pouvez allouer à `work_mem` dépend de la classe d'instance. Si vous utilisez une classe d'instance plus importante, vous disposerez de plus de mémoire. Toutefois, dans l'exemple précédent, vous ne pouvez pas utiliser plus de 16 Gio. Sinon votre instance devient indisponible lorsqu'elle est à court de mémoire. Pour récupérer l'instance en cas d'indisponibilité, les services d'automatisation de RDS for PostgreSQL redémarrent automatiquement.

Gérez le nombre de connexions

Supposons que votre instance de base de données compte 5 000 connexions simultanées. Chaque connexion utilise au moins 4 Mio de `work_mem`. La forte consommation de mémoire des connexions est susceptible de dégrader les performances. En réponse, vous disposez des options suivantes :

- Passez à une classe d'instance supérieure.
- Réduisez le nombre de connexions simultanées à la base de données à l'aide d'un proxy ou d'un regroupement de connexions.

Pour les proxies, utilisez Amazon RDS Proxy, pgBouncer ou un regroupement de connexions basé sur votre application. Cette solution réduit la charge de l'UC. Elle réduit également le risque lorsque toutes les connexions ont besoin de la zone de mémoire de travail. Lorsque les connexions à la base de données sont moins nombreuses, vous pouvez augmenter la valeur de `work_mem`. De cette façon, vous réduisez l'occurrence des événements d'attente `IO:BufFileRead` et `IO:BufFileWrite`. De plus, les requêtes en attente d'accès à la zone de mémoire de travail s'accélèrent de manière significative.

IO : DataFileRead

L'événement `IO:DataFileRead` se produit lorsqu'une connexion attend qu'un processus backend lise une page requise à partir du stockage parce que la page n'est pas disponible dans la mémoire partagée.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'allongement des temps d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

Toutes les requêtes et opérations en langage de manipulation de données (DML) accèdent aux pages du groupe de mémoires tampons. Les instructions qui peuvent induire des lectures sont : `SELECT`, `UPDATE` et `DELETE`. Par exemple, une instruction `UPDATE` peut lire des pages à partir de tables ou d'index. Si la page demandée ou mise à jour ne se trouve pas dans le groupe de mémoires tampons partagées, cette lecture peut provoquer l'événement `IO:DataFileRead`.

Comme le groupe de mémoires tampons partagées est limité, il peut être saturé. Dans ce cas, les requêtes de pages qui ne sont pas en mémoire forcent la base de données à lire des blocs sur le disque. Si l'événement `IO:DataFileRead` se produit fréquemment, votre groupe de mémoires tampons partagées est peut-être trop petit pour prendre en charge votre charge de travail. Ce problème se pose avec acuité pour les requêtes `SELECT` qui lisent un grand nombre de lignes qui ne rentrent pas dans le groupe de mémoires tampons. Pour plus d'informations sur le groupe de mémoires tampons, consultez Consommation des ressources dans la documentation PostgreSQL.

Causes probables de l'allongement des temps d'attente

Les principales causes de l'événement `IO:DataFileRead` sont les suivantes :

Pics de connexion

Il se peut que plusieurs connexions génèrent le même nombre d'événements IO : DataFileRead wait. Dans ce cas, un pic (augmentation soudaine et importante) d'événements `IO:DataFileRead` peut se produire.

Instructions SELECT et DML effectuant des analyses séquentielles

Votre application est peut-être en train d'effectuer une nouvelle opération. Ou une opération existante peut changer suite à un nouveau plan d'exécution. Dans ce cas, recherchez les tables (en particulier les tables volumineuses) qui présentent une valeur `seq_scan` plus élevée. Pour les trouver, interrogez `pg_stat_user_tables`. Pour suivre les requêtes qui génèrent plus d'opérations de lecture, utilisez l'extension `pg_stat_statements`.

CTAS et CREATE INDEX pour les jeux de données volumineux

Un CTAS est une instruction `CREATE TABLE AS SELECT`. Si vous exécutez un CTAS en utilisant un jeu de données volumineux comme source, ou si vous créez un index sur une table volumineuse, l'événement `IO:DataFileRead` peut se produire. Lorsque vous créez un index, la base de données peut avoir besoin de lire l'objet entier à l'aide d'une analyse séquentielle. Un CTAS génère des lectures `IO:DataFile` lorsque les pages ne sont pas en mémoire.

Exécution simultanée de plusieurs processus employés vacuum

Les processus employés vacuum peuvent être déclenchés manuellement ou automatiquement. Nous vous recommandons d'adopter une stratégie vacuum agressive. Toutefois, lorsqu'une table comporte de nombreuses lignes mises à jour ou supprimées, le nombre d'attentes `IO:DataFileRead` augmente. Une fois l'espace récupéré, le temps vacuum passé sur `IO:DataFileRead` diminue.

Ingestion de grandes quantités de données

Lorsque votre application ingère de grandes quantités de données, les opérations `ANALYZE` peuvent être plus fréquentes. Le processus `ANALYZE` peut être déclenché par un lanceur autovacuum, ou être appelé manuellement.

L'opération `ANALYZE` lit un sous-ensemble de la table. Le nombre de pages à analyser est calculé en multipliant 30 par la valeur `default_statistics_target`. Pour de plus amples informations, veuillez consulter la documentation sur PostgreSQL. Le paramètre `default_statistics_target` accepte des valeurs comprises entre 1 et 10 000, la valeur par défaut étant 100.

Pénurie de ressources

Si l'UC ou la bande passante réseau de l'instance sont entièrement utilisés, l'événement `IO:DataFileRead` peut se produire plus fréquemment.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Vérifiez les filtres de prédicat pour détecter les requêtes qui génèrent des attentes
- Minimisez l'effet des opérations de maintenance
- Réagissez à un nombre élevé de connexions

Vérifiez les filtres de prédicat pour détecter les requêtes qui génèrent des attentes

Supposons que vous identifiez des requêtes spécifiques qui génèrent des événements d'attente `IO:DataFileRead`. Vous pouvez les identifier à l'aide des techniques suivantes :

- Performance Insights
- Vues catalogue telles que celles fournies par l'extension `pg_stat_statements`
- Vue catalogue `pg_stat_all_tables`, si elle affiche périodiquement un nombre accru de lectures physiques
- Vue `pg_statio_all_tables`, si elle montre que les compteurs `_read` sont en augmentation

Nous vous recommandons de déterminer quels filtres sont utilisés dans le prédicat (clause `WHERE`) de ces requêtes. Suivez ces instructions :

- Exécutez la commande `EXPLAIN`. Dans la sortie, identifiez les types d'analyses utilisés. Une analyse séquentielle n'indique pas nécessairement un problème. Les requêtes qui utilisent des analyses séquentielles produisent naturellement plus d'événements `IO:DataFileRead` par rapport aux requêtes qui utilisent des filtres.

  Vérifiez que la colonne répertoriée dans la clause `WHERE` est indexée. Si ce n'est pas le cas, envisagez de créer un index pour cette colonne. Cette approche évite les analyses séquentielles et réduit le nombre d'événements `IO:DataFileRead`. Si une requête comporte des filtres restrictifs et continue à produire des analyses séquentielles, assurez-vous que les index appropriés sont utilisés.
- Déterminez si la requête accède à une table très volumineuse. Dans certains cas, le partitionnement d'une table peut améliorer les performances, en permettant à la requête de ne lire que les partitions nécessaires.
- Examinez la cardinalité (nombre total de lignes) de vos opérations de jointure. Notez le caractère restrictif des valeurs que vous transmettez dans les filtres de la clause `WHERE`. Si possible, ajustez votre requête pour réduire le nombre de lignes transmises à chaque étape du plan.

Minimisez l'effet des opérations de maintenance

Les opérations de maintenance telles que `VACUUM` et `ANALYZE` sont importantes. Nous vous recommandons de ne pas les désactiver parce que vous trouvez des événements d'attente `IO:DataFileRead` liés à ces opérations de maintenance. Les approches suivantes peuvent minimiser l'effet de ces opérations :

- Exécutez les opérations de maintenance manuellement pendant les heures creuses. Cette technique empêche la base de données d'atteindre le seuil des opérations automatiques.
- Pour les tables très volumineuses, partitionnez la table. Cette technique permet de réduire les frais liés aux opérations de maintenance. La base de données accède uniquement aux partitions qui nécessitent une maintenance.
- Lorsque vous intégrez de grandes quantités de données, pensez à désactiver la fonction d'analyse automatique.

La fonction autovacuum est automatiquement déclenchée pour une table lorsque la formule suivante est vraie.

```
pg_stat_user_tables.n_dead_tup > (pg_class.reltuples x autovacuum_vacuum_scale_factor)
 + autovacuum_vacuum_threshold
```

La vue `pg_stat_user_tables` et le catalogue `pg_class` comportent plusieurs lignes. Une ligne peut correspondre à une ligne de votre table. Cette formule suppose que les `reltuples` sont destinés à une table spécifique. Les paramètres `autovacuum_vacuum_scale_factor` (0,20 par défaut) et `autovacuum_vacuum_threshold` (50 tuples par défaut) sont généralement définis globalement pour l'ensemble de l'instance. Vous pouvez toutefois définir des valeurs différentes pour une table spécifique.

Rubriques

- Recherche des tables qui consomment de l'espace inutilement
- Recherche des index qui consomment de l'espace inutilement
- Recherchez les tables éligibles au processus autovacuum

Recherche des tables qui consomment de l'espace inutilement

Pour rechercher les tables qui consomment inutilement de l'espace, vous pouvez utiliser les fonctions issues de l'extension PostgreSQL `pgstattuple`. Cette extension (module) est disponible par défaut sur toutes les instances de base de données RDS for PostgreSQL et peut être instanciée sur l'instance à l'aide de la commande suivante.

```
CREATE EXTENSION pgstattuple;
```

Pour plus d'informations sur cette extension, consultez pgstattuple dans la documentation PostgreSQL.

Vous pouvez vérifier qu'il n'existe pas de gonflement de tables et d'index dans votre application. Pour en savoir plus, consultez Diagnostic du gonflement de la table et de l'index.

Recherche des index qui consomment de l'espace inutilement

Pour rechercher les index gonflés et estimer la quantité d'espace inutilement consommée sur les tables pour lesquelles vous disposez de privilèges de lecture, vous pouvez exécuter la requête suivante.

```
-- WARNING: rows with is_na = 't' are known to have bad statistics ("name" type is not
 supported).
-- This query is compatible with PostgreSQL 8.2 and later.

SELECT current_database(), nspname AS schemaname, tblname, idxname,
 bs*(relpages)::bigint AS real_size,
  bs*(relpages-est_pages)::bigint AS extra_size,
  100 * (relpages-est_pages)::float / relpages AS extra_ratio,
  fillfactor, bs*(relpages-est_pages_ff) AS bloat_size,
  100 * (relpages-est_pages_ff)::float / relpages AS bloat_ratio,
  is_na
  -- , 100-(sub.pst).avg_leaf_density, est_pages, index_tuple_hdr_bm,
  -- maxalign, pagehdr, nulldatawidth, nulldatahdrwidth, sub.reltuples, sub.relpages
  -- (DEBUG INFO)
FROM (
  SELECT coalesce(1 +
       ceil(reltuples/floor((bs-pageopqdata-pagehdr)/(4+nulldatahdrwidth)::float)), 0
       -- ItemIdData size + computed avg size of a tuple (nulldatahdrwidth)
    ) AS est_pages,
    coalesce(1 +
       ceil(reltuples/floor((bs-pageopqdata-pagehdr)*fillfactor/
(100*(4+nulldatahdrwidth)::float))), 0
    ) AS est_pages_ff,
    bs, nspname, table_oid, tblname, idxname, relpages, fillfactor, is_na
    -- , stattuple.pgstatindex(quote_ident(nspname)||'.'||quote_ident(idxname)) AS
 pst,
    -- index_tuple_hdr_bm, maxalign, pagehdr, nulldatawidth, nulldatahdrwidth,
 reltuples
    -- (DEBUG INFO)
  FROM (
    SELECT maxalign, bs, nspname, tblname, idxname, reltuples, relpages, relam,
 table_oid, fillfactor,
      ( index_tuple_hdr_bm +
          maxalign - CASE -- Add padding to the index tuple header to align on MAXALIGN
            WHEN index_tuple_hdr_bm%maxalign = 0 THEN maxalign
            ELSE index_tuple_hdr_bm%maxalign
          END
        + nulldatawidth + maxalign - CASE -- Add padding to the data to align on
 MAXALIGN
            WHEN nulldatawidth = 0 THEN 0
            WHEN nulldatawidth::integer%maxalign = 0 THEN maxalign
            ELSE nulldatawidth::integer%maxalign
          END
```

```
      )::numeric AS nulldatahdrwidth, pagehdr, pageopqdata, is_na
      -- , index_tuple_hdr_bm, nulldatawidth -- (DEBUG INFO)
    FROM (
      SELECT
        i.nspname, i.tblname, i.idxname, i.reltuples, i.relpages, i.relam, a.attrelid
AS table_oid,
        current_setting('block_size')::numeric AS bs, fillfactor,
        CASE -- MAXALIGN: 4 on 32bits, 8 on 64bits (and mingw32 ?)
          WHEN version() ~ 'mingw32' OR version() ~ '64-bit|x86_64|ppc64|ia64|amd64'
THEN 8
          ELSE 4
        END AS maxalign,
        /* per page header, fixed size: 20 for 7.X, 24 for others */
        24 AS pagehdr,
        /* per page btree opaque data */
        16 AS pageopqdata,
        /* per tuple header: add IndexAttributeBitMapData if some cols are null-able */
        CASE WHEN max(coalesce(s.null_frac,0)) = 0
          THEN 2 -- IndexTupleData size
          ELSE 2 + (( 32 + 8 - 1 ) / 8)
          -- IndexTupleData size + IndexAttributeBitMapData size ( max num filed per
index + 8 - 1 /8)
        END AS index_tuple_hdr_bm,
        /* data len: we remove null values save space using it fractionnal part from
stats */
        sum( (1-coalesce(s.null_frac, 0)) * coalesce(s.avg_width, 1024)) AS
nulldatawidth,
        max( CASE WHEN a.atttypid = 'pg_catalog.name'::regtype THEN 1 ELSE 0 END ) > 0
AS is_na
      FROM pg_attribute AS a
        JOIN (
          SELECT nspname, tbl.relname AS tblname, idx.relname AS idxname,
            idx.reltuples, idx.relpages, idx.relam,
            indrelid, indexrelid, indkey::smallint[] AS attnum,
            coalesce(substring(
              array_to_string(idx.reloptions, ' ')
               from 'fillfactor=([0-9]+)')::smallint, 90) AS fillfactor
          FROM pg_index
            JOIN pg_class idx ON idx.oid=pg_index.indexrelid
            JOIN pg_class tbl ON tbl.oid=pg_index.indrelid
            JOIN pg_namespace ON pg_namespace.oid = idx.relnamespace
          WHERE pg_index.indisvalid AND tbl.relkind = 'r' AND idx.relpages > 0
        ) AS i ON a.attrelid = i.indexrelid
        JOIN pg_stats AS s ON s.schemaname = i.nspname
```

```
          AND ((s.tablename = i.tblname AND s.attname =
 pg_catalog.pg_get_indexdef(a.attrelid, a.attnum, TRUE))
          -- stats from tbl
          OR  (s.tablename = i.idxname AND s.attname = a.attname))
          -- stats from functional cols
        JOIN pg_type AS t ON a.atttypid = t.oid
      WHERE a.attnum > 0
      GROUP BY 1, 2, 3, 4, 5, 6, 7, 8, 9
    ) AS s1
  ) AS s2
    JOIN pg_am am ON s2.relam = am.oid WHERE am.amname = 'btree'
) AS sub
-- WHERE NOT is_na
ORDER BY 2,3,4;
```

Recherchez les tables éligibles au processus autovacuum

Pour rechercher les tables éligibles au processus autovacuum, exécutez la requête suivante.

```
--This query shows tables that need vacuuming and are eligible candidates.
--The following query lists all tables that are due to be processed by autovacuum.
-- During normal operation, this query should return very little.
WITH  vbt AS (SELECT setting AS autovacuum_vacuum_threshold
              FROM pg_settings WHERE name = 'autovacuum_vacuum_threshold')
    , vsf AS (SELECT setting AS autovacuum_vacuum_scale_factor
              FROM pg_settings WHERE name = 'autovacuum_vacuum_scale_factor')
    , fma AS (SELECT setting AS autovacuum_freeze_max_age
              FROM pg_settings WHERE name = 'autovacuum_freeze_max_age')
    , sto AS (SELECT opt_oid, split_part(setting, '=', 1) as param,
                split_part(setting, '=', 2) as value
              FROM (SELECT oid opt_oid, unnest(reloptions) setting FROM pg_class) opt)
SELECT
    '"'||ns.nspname||'"."'||c.relname||'"' as relation
    , pg_size_pretty(pg_table_size(c.oid)) as table_size
    , age(relfrozenxid) as xid_age
    , coalesce(cfma.value::float, autovacuum_freeze_max_age::float)
 autovacuum_freeze_max_age
    , (coalesce(cvbt.value::float, autovacuum_vacuum_threshold::float) +
        coalesce(cvsf.value::float,autovacuum_vacuum_scale_factor::float) *
 c.reltuples)
        as autovacuum_vacuum_tuples
    , n_dead_tup as dead_tuples
FROM pg_class c
JOIN pg_namespace ns ON ns.oid = c.relnamespace
```

```
JOIN pg_stat_all_tables stat ON stat.relid = c.oid
JOIN vbt on (1=1)
JOIN vsf ON (1=1)
JOIN fma on (1=1)
LEFT JOIN sto cvbt ON cvbt.param = 'autovacuum_vacuum_threshold' AND c.oid =
 cvbt.opt_oid
LEFT JOIN sto cvsf ON cvsf.param = 'autovacuum_vacuum_scale_factor' AND c.oid =
 cvsf.opt_oid
LEFT JOIN sto cfma ON cfma.param = 'autovacuum_freeze_max_age' AND c.oid = cfma.opt_oid
WHERE c.relkind = 'r'
AND nspname <> 'pg_catalog'
AND (
    age(relfrozenxid) >= coalesce(cfma.value::float, autovacuum_freeze_max_age::float)
    or
    coalesce(cvbt.value::float, autovacuum_vacuum_threshold::float) +
      coalesce(cvsf.value::float,autovacuum_vacuum_scale_factor::float) * c.reltuples
 <= n_dead_tup
    -- or 1 = 1
)
ORDER BY age(relfrozenxid) DESC;
```

Réagissez à un nombre élevé de connexions

Lorsque vous surveillez Amazon CloudWatch, vous constaterez peut-être que les `DatabaseConnections` statistiques augmentent. Cette augmentation indique un nombre accru de connexions à votre base de données. Nous vous recommandons l'approche suivante :

- Limitez le nombre de connexions que l'application peut ouvrir avec chaque instance. Si votre application dispose d'une fonction intégrée de regroupement des connexions, définissez-la sur un nombre raisonnable de connexions. Basez ce nombre sur ce que les vCPU de votre instance sont en mesure de paralléliser.

  Si votre application n'utilise pas de fonction de regroupement des connexions, envisagez d'utiliser un proxy Amazon RDS ou une autre solution. Cette approche permet à votre application d'ouvrir plusieurs connexions à l'aide de l'équilibreur de charge. L'équilibreur peut alors ouvrir un nombre restreint de connexions avec la base de données. Comme le nombre de connexions exécutées en parallèle est moindre, votre instance de base de données effectue moins de changements de contexte dans le noyau. Les requêtes progressent plus rapidement, ce qui entraîne une diminution des événements d'attente. Pour de plus amples informations, veuillez consulter Proxy Amazon RDS .

- Dans la mesure du possible, tirez parti des réplicas en lecture pour RDS for PostgreSQL. Lorsque votre application exécute une opération en lecture seule, envoyez ces demandes aux réplicas en lecture. Cette technique réduit la pression des E/S sur le nœud (d'écriture) principal.
- Envisagez une augmentation d'échelle de votre instance de base de données. Une classe d'instances de plus grande capacité offre davantage de mémoire, ce qui permet à RDS for PostgreSQL de disposer d'un groupe de mémoires tampons partagées plus important pour stocker les pages. Cette plus grande taille confère également à l'instance de base de données davantage de vCPU pour gérer les connexions. Ce plus grand nombre de vCPU est particulièrement utile lorsque les opérations qui génèrent des événements d'attente `IO:DataFileRead` sont des écritures.

IO : WALWrite

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'allongement des temps d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour RDS for PostgreSQL versions 10 et ultérieures.

Contexte

L'activité de la base de données qui génère les données WAL remplit d'abord les tampons WAL, puis écrit sur le disque de manière asynchrone. L'événement d'attente `IO:WALWrite` est généré quand la session SQL attend la fin de l'écriture des données WAL sur le disque afin de pouvoir libérer l'appel COMMIT de la transaction.

Causes probables de l'allongement des temps d'attente

Si cet événement d'attente se produit souvent, vous devez examiner votre charge de travail, le type des mises à jour effectuées par votre charge de travail et leur fréquence. En particulier, recherchez les types d'activités suivants.

Activité DML intense

La modification des données dans les tables de base de données ne se produit pas instantanément. Une insertion dans une table peut nécessiter l'attente d'une insertion ou d'une mise à jour dans la même table par un autre client. Les instructions du langage de manipulation de données (DML) permettant de modifier les valeurs des données (INSERT, UPDATE, DELETE, COMMIT, ROLLBACK TRANSACTION) peuvent provoquer des conflits qui obligent le fichier WAL à attendre que les tampons soient vidés. Cette situation est capturée dans les métriques Analyse des performances d'Amazon RDS suivantes, qui indiquent une activité DML intense.

- `tup_inserted`
- `tup_updated`
- `tup_deleted`
- `xact_rollback`
- `xact_commit`

Pour plus d'informations sur ces métriques, consultez Compteurs Performance Insights pour Amazon RDS for PostgreSQL.

Activité des points de vérification fréquents

Les points de contrôle fréquents contribuent à augmenter le nombre de fichiers WAL. Dans RDS for PostgreSQL, les écritures de pages complètes sont toujours « activées ». Les écritures de pages complètes favorisent la protection contre les pertes de données. Toutefois, lorsque les points de vérification sont trop fréquents, le système peut rencontrer des problèmes de performances globales. Cela est particulièrement vrai sur les systèmes à activité DML intense. Dans certains cas, vous pouvez trouver des messages d'erreur dans votre fichier `postgresql.log` indiquant que « les points de vérification se produisent trop fréquemment ».

Lors du réglage des points de vérification, nous vous recommandons de bien équilibrer les performances par rapport au temps nécessaire prévu pour récupérer en cas d'arrêt anormal.

Actions

Nous recommandons les actions suivantes pour réduire le nombre de cet événement d'attente.

Rubriques

- Réduisez le nombre de validations
- Surveillance de vos points de vérification
- Augmenter les E/S
- Volume de journal dédié (DLV)

Réduisez le nombre de validations

Pour réduire le nombre de validations, vous pouvez combiner les instructions en blocs de transactions. Utilisez Analyse des performances d'Amazon RDS pour examiner le type des requêtes en cours d'exécution. Vous pouvez également déplacer les opérations de maintenance importantes à des heures creuses. Par exemple, créez des index ou utilisez les opérations `pg_repack` en dehors des heures de production.

Surveillance de vos points de vérification

Vous pouvez surveiller deux paramètres pour voir à quelle fréquence votre instance de base de données RDS for PostgreSQL écrit dans le fichier WAL pour les points de vérification.

- `log_checkpoints` – Ce paramètre est « activé » par défaut. Il provoque l'envoi d'un message au journal PostgreSQL pour chaque point de vérification. Ces messages de journal incluent le nombre de tampons écrits, le temps passé à les écrire et le nombre de fichiers WAL ajoutés, supprimés ou recyclés pour le point de vérification donné.

  Pour plus d'informations sur ce paramètre, consultez Error Reporting and Logging (Signalisation et journalisation des erreurs) dans la documentation PostgreSQL.
- `checkpoint_warning` – Ce paramètre définit une valeur seuil (en secondes) pour la fréquence des points de vérification, au-dessus de laquelle un avertissement est généré. Par défaut, ce paramètre n'est pas défini dans RDS for PostgreSQL. Vous pouvez définir la valeur de ce paramètre pour recevoir un avertissement lorsque les modifications de base de données dans votre instance de base de données RDS for PostgreSQL sont écrites à une vitesse que les fichiers WAL ne sont pas dimensionnés pour gérer. Par exemple, supposons que vous définissiez ce paramètre sur 30. Si votre instance RDS for PostgreSQL doit écrire des modifications plus souvent que toutes les 30 secondes, l'avertissement indiquant que « les points de vérification se produisent trop fréquemment » est envoyé dans le journal PostgreSQL. Cela peut indiquer que votre valeur `max_wal_size` doit être augmentée.

  Pour plus d'informations, consultez Write Ahead Log dans la documentation PostgreSQL.

Augmenter les E/S

Ce type d'input/output (IO) wait event can remediated by scaling the input/outputopérations par seconde (IOPs) permet d'accélérer les E/S. La mise à l'échelle des E/S est préférable à la mise à l'échelle du processeur, car la mise à l'échelle du processeur peut entraîner encore plus de conflits d'E/S en gérant plus de travail et aggraver ainsi le goulot d'étranglement d'E/S. En règle générale, nous recommandons d'envisager le réglage de votre charge de travail avant d'exécuter des opérations de mise à l'échelle.

Volume de journal dédié (DLV)

Vous pouvez utiliser un volume dédié aux journaux (DLV) pour une instance de base de données qui utilise le stockage IOPS provisionnés (PIOPS) en utilisant la console Amazon RDS, l' AWS CLI ou l'API Amazon RDS. Un DLV déplace les journaux de transactions de base de données PostgreSQL vers un volume de stockage distinct du volume contenant les tables de base de données. Pour de plus amples informations, veuillez consulter Volume de journal dédié (DLV).

IPC : événements d'attente parallèles

Les informations suivantes `IPC:parallel wait events` indiquent qu'une session attend une communication inter-processus liée aux opérations d'exécution de requêtes en parallèle.

- `IPC:BgWorkerStartup`- Un processus attend qu'un processus de travail parallèle termine sa séquence de démarrage. Cela se produit lors de l'initialisation des travailleurs pour l'exécution de requêtes en parallèle.
- `IPC:BgWorkerShutdown`- Un processus attend qu'un processus de travail parallèle termine sa séquence d'arrêt. Cela se produit pendant la phase de nettoyage de l'exécution parallèle des requêtes.
- `IPC:ExecuteGather`- Un processus attend de recevoir des données provenant de processus de travail parallèles pendant l'exécution de la requête. Cela se produit lorsque le processus leader doit recueillir des résultats auprès de ses employés.
- `IPC:ParallelFinish`- Un processus attend que les travailleurs parallèles terminent leur exécution et publient leurs résultats finaux. Cela se produit pendant la phase d'achèvement de l'exécution parallèle des requêtes.

Rubriques

- Versions de moteur prises en charge

- Contexte
- Causes probables de l'allongement des temps d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente s'appliquent à toutes les versions d'Aurora PostgreSQL.

Contexte

L'exécution parallèle de requêtes dans PostgreSQL implique la collaboration de plusieurs processus pour traiter une seule requête. Lorsqu'une requête est jugée appropriée pour la parallélisation, un processus leader est coordonné avec un ou plusieurs processus de travail parallèle sur la base du réglage des `max_parallel_workers_per_gather` paramètres. Le processus du leader répartit le travail entre les travailleurs, chaque travailleur traite sa partie des données de manière indépendante et les résultats sont transmis au processus du leader.

> **Note**
>
> Chaque travailleur parallèle fonctionne comme un processus distinct dont les besoins en ressources sont similaires à ceux d'une session utilisateur complète. Cela signifie qu'une requête parallèle avec 4 travailleurs peut consommer jusqu'à 5 fois plus de ressources (processeur, mémoire, I/O bande passante) par rapport à une requête non parallèle, car le processus principal et chaque processus de travail conservent leurs propres allocations de ressources. Par exemple, des paramètres tels que `work_mem` sont appliqués individuellement à chaque travailleur, multipliant potentiellement l'utilisation totale de la mémoire entre tous les processus.

L'architecture de requêtes parallèles comprend trois composants principaux :

- Processus leader : processus principal qui initie l'opération parallèle, divise la charge de travail et assure la coordination avec les processus de travail.
- Processus de travail : processus d'arrière-plan qui exécutent des parties de la requête en parallèle.
- Fusion Gather/Gather : opérations qui combinent les résultats de plusieurs processus de travail jusqu'au leader

Lors d'une exécution parallèle, les processus doivent communiquer entre eux par le biais de mécanismes de communication inter-processus (IPC). Ces événements d'attente IPC se produisent au cours de différentes phases :

- Démarrage du travailleur : lorsque des travailleurs parallèles sont initialisés
- Échange de données : lorsque les travailleurs traitent les données et envoient les résultats au responsable
- Arrêt du programme de travail : lorsque l'exécution parallèle est terminée et que les travailleurs sont interrompus
- Points de synchronisation : lorsque les processus doivent se coordonner ou attendre que d'autres processus terminent leurs tâches

Il est essentiel de comprendre ces événements d'attente pour diagnostiquer les problèmes de performances liés à l'exécution parallèle de requêtes, en particulier dans les environnements à forte concurrence dans lesquels plusieurs requêtes parallèles peuvent s'exécuter simultanément.

Causes probables de l'allongement des temps d'attente

Plusieurs facteurs peuvent contribuer à augmenter le nombre d'événements d'attente IPC liés au parallèle :

Haute simultanéité des requêtes parallèles

Lorsque de nombreuses requêtes parallèles sont exécutées simultanément, cela peut entraîner une contention des ressources et une augmentation des temps d'attente pour les opérations IPC. Cela est particulièrement courant dans les systèmes comportant des volumes de transactions ou des charges de travail analytiques élevés.

Plans de requêtes parallèles sous-optimaux

Si le planificateur de requêtes choisit des plans parallèles inefficaces, cela peut entraîner une parallélisation inutile ou une mauvaise répartition du travail entre les employés. Cela peut entraîner une augmentation des temps d'attente pour l'IPC, en particulier pour les événements `IPC:ExecuteGather` et `IPC:ParallelFinish` les événements. Ces problèmes de planification sont souvent dus à des statistiques périmées et table/index à des excès.

Démarrage et arrêt fréquents des travailleurs parallèles

Les requêtes de courte durée qui déclenchent et interrompent fréquemment des tâches parallèles peuvent entraîner une augmentation du nombre `IPC:BgWorkerStartup`

d'`IPC:BgWorkerShutdown`événements. Cela se produit souvent dans les charges de travail OLTP comportant de nombreuses petites requêtes parallélisables.

Contraintes liées aux ressources

Un processeur, une mémoire ou une I/O capacité limités peuvent entraîner des blocages lors de l'exécution en parallèle, ce qui augmente les temps d'attente pour tous les événements IPC. Par exemple, si le processeur est saturé, les processus de travail peuvent prendre plus de temps à démarrer ou à traiter leur partie du travail.

Structures de requêtes complexes

Les requêtes comportant plusieurs niveaux de parallélisme (par exemple, des jointures parallèles suivies d'agrégations parallèles) peuvent entraîner des modèles IPC plus complexes et des temps d'attente potentiellement accrus, en particulier pour les événements. `IPC:ExecuteGather`

Ensembles de résultats volumineux

Les requêtes qui produisent de grands ensembles de résultats peuvent entraîner une augmentation des temps d'`IPC:ExecuteGather`attente, car le processus principal passe plus de temps à collecter et à traiter les résultats des processus de travail.

La compréhension de ces facteurs peut aider à diagnostiquer et à résoudre les problèmes de performances liés à l'exécution parallèle de requêtes dans Aurora PostgreSQL.

Actions

Lorsque vous voyez des temps d'attente liés à une requête parallèle, cela signifie généralement qu'un processus principal coordonne ou attend des processus de travail parallèles. Ces temps d'attente sont courants lors de l'exécution de plans parallèles. Vous pouvez étudier et atténuer l'impact de ces temps d'attente en surveillant l'utilisation des travailleurs parallèles, en revoyant les paramètres et en ajustant l'exécution des requêtes et l'allocation des ressources.

Rubriques

- Analyser les plans de requêtes pour détecter un parallélisme inefficace
- Surveiller l'utilisation des requêtes parallèles
- Vérifiez et ajustez les paramètres des requêtes parallèles
- Optimisation de l'allocation des ressources
- Étudier la gestion des connexions

- Passez en revue et optimisez les opérations de maintenance

Analyser les plans de requêtes pour détecter un parallélisme inefficace

L'exécution parallèle de requêtes peut souvent entraîner une instabilité du système, des pics de processeur et des variations imprévisibles des performances des requêtes. Il est essentiel d'analyser de manière approfondie si le parallélisme améliore réellement votre charge de travail spécifique. Utilisez EXPLAIN ANALYZE pour passer en revue les plans d'exécution de requêtes parallèles.

Désactivez temporairement le parallélisme au niveau de la session pour comparer l'efficacité du plan :

```
SET max_parallel_workers_per_gather = 0;
EXPLAIN ANALYZE <your_query>;
```

Réactivez le parallélisme et comparez :

```
RESET max_parallel_workers_per_gather;
EXPLAIN ANALYZE <your_query>;
```

Si la désactivation du parallélisme donne des résultats meilleurs ou plus cohérents, envisagez de le désactiver pour des requêtes spécifiques au niveau de la session à l'aide des commandes SET. Pour un impact plus large, vous pouvez désactiver le parallélisme au niveau de l'instance en ajustant les paramètres pertinents dans votre groupe de paramètres de base de données. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Surveiller l'utilisation des requêtes parallèles

Utilisez les requêtes suivantes pour avoir une meilleure visibilité sur l'activité et la capacité des requêtes parallèles :

Vérifiez les processus de travail parallèle actifs :

```
SELECT
    COUNT(*)
FROM
    pg_stat_activity
WHERE
    backend_type = 'parallel worker';
```

Cette requête indique le nombre de processus de travail parallèle actifs. Une valeur élevée peut indiquer que votre `max_parallel_workers` est configuré avec une valeur élevée et vous pourriez envisager de la réduire.

Vérifiez les requêtes parallèles simultanées :

```
SELECT
    COUNT(DISTINCT leader_pid)
FROM
    pg_stat_activity
WHERE
    leader_pid IS NOT NULL;
```

Cette requête renvoie le nombre de processus leaders distincts qui ont lancé des requêtes parallèles. Un nombre élevé indique que plusieurs sessions exécutent des requêtes parallèles simultanément, ce qui peut augmenter la demande en termes de processeur et de mémoire.

Vérifiez et ajustez les paramètres des requêtes parallèles

Passez en revue les paramètres suivants pour vous assurer qu'ils correspondent à votre charge de travail :

- `max_parallel_workers`: nombre total de travailleurs parallèles au cours de toutes les sessions.
- `max_parallel_workers_per_gather`: Nombre maximum de travailleurs par requête.

Pour les charges de travail OLAP, l'augmentation de ces valeurs peut améliorer les performances. Pour les charges de travail OLTP, les valeurs les plus faibles sont généralement préférées.

```
SHOW max_parallel_workers;
```

```
SHOW max_parallel_workers_per_gather;
```

Optimisation de l'allocation des ressources

Surveillez l'utilisation du processeur et envisagez d'ajuster le nombre de v CPUs s'il est constamment élevé et si votre application bénéficie de requêtes parallèles. Assurez-vous que la mémoire disponible est suffisante pour les opérations parallèles.

- Utilisez les métriques Performance Insights pour déterminer si le système est lié au processeur.
- Chaque parallel worker utilise le sien`work_mem`. Assurez-vous que l'utilisation totale de la mémoire respecte les limites de l'instance.

Les requêtes parallèles peuvent consommer beaucoup plus de ressources que les requêtes non parallèles, car chaque processus de travail est un processus complètement distinct qui a à peu près le même impact sur le système qu'une session utilisateur supplémentaire. Cela doit être pris en compte lors du choix d'une valeur pour ce paramètre, ainsi que lors de la configuration d'autres paramètres contrôlant l'utilisation des ressources, tels que`work_mem`. Pour de plus amples informations, veuillez consulter la documentation sur PostgreSQL. Les limites de ressources telles que celles qui `work_mem` sont appliquées individuellement à chaque travailleur, ce qui signifie que l'utilisation totale peut être beaucoup plus élevée dans tous les processus qu'elle ne le serait normalement pour un seul processus.

Envisagez d'augmenter v CPUs ou de régler les paramètres de mémoire si votre charge de travail est fortement parallélisée.

Étudier la gestion des connexions

En cas d'épuisement des connexions, passez en revue les stratégies de regroupement des connexions des applications. Envisagez d'implémenter le regroupement de connexions au niveau de l'application s'il n'est pas déjà utilisé.

Passez en revue et optimisez les opérations de maintenance

Coordonnez la création d'index et les autres tâches de maintenance pour éviter les conflits de ressources. Envisagez de planifier ces opérations en dehors des heures de pointe. Évitez de planifier de lourdes opérations de maintenance (par exemple, des constructions d'index parallèles) pendant les périodes de forte charge de requêtes utilisateur. Ces opérations peuvent consommer des travailleurs parallèles et avoir un impact sur les performances des requêtes régulières.

Lock:advisory

L'événement `Lock:advisory` se produit lorsqu'une application PostgreSQL utilise un verrou pour coordonner l'activité sur plusieurs sessions.

Rubriques

- Versions de moteur pertinentes
- Contexte
- Causes
- Actions

Versions de moteur pertinentes

Ces informations sur les événements d'attente s'appliquent à RDS for PostgreSQL 9.6 et versions ultérieures.

Contexte

Les verrous consultatifs PostgreSQL sont des verrous coopératifs de niveau application explicitement verrouillés et déverrouillés par le code d'application de l'utilisateur. Une application peut utiliser des verrous consultatifs PostgreSQL pour coordonner l'activité sur plusieurs sessions. Contrairement aux verrous standard, de niveau objet ou ligne, l'application dispose d'un contrôle total sur la durée de vie du verrou. Pour en savoir plus, consultez Advisory Locks dans la documentation PostgreSQL.

Les verrous consultatifs peuvent être libérés avant la fin d'une transaction ou être maintenus par une session sur plusieurs transactions. Cela ne s'applique pas aux verrous implicites appliqués par le système, comme un verrou exclusif d'accès à une table acquis par une instruction `CREATE INDEX`.

Pour accéder à la description des fonctions utilisées pour acquérir (verrouiller) et libérer (déverrouiller) les verrous consultatifs, consultez Advisory Lock Functions dans la documentation PostgreSQL.

Les verrous consultatifs sont implémentés au-dessus du système de verrouillage PostgreSQL standard et sont visibles dans la vue système `pg_locks`.

Causes

Ce type de verrou est exclusivement contrôlé par une application qui l'utilise explicitement. Les verrous consultatifs qui sont acquis pour chaque ligne dans le cadre d'une requête peuvent provoquer un pic de verrous ou une accumulation à long terme.

Ces effets se produisent lorsque la requête est exécutée d'une manière qui acquiert des verrous sur plus de lignes que celles renvoyées par la requête. L'application doit finir par libérer chaque verrou, mais si des verrous sont acquis sur des lignes qui ne sont pas renvoyées, l'application ne peut pas tous les trouver.

L'exemple suivant est extrait de la section Advisory Locks de la documentation PostgreSQL.

```
SELECT pg_advisory_lock(id) FROM foo WHERE id > 12345 LIMIT 100;
```

Dans cet exemple, la clause `LIMIT` ne peut arrêter la sortie de la requête que lorsque les lignes ont déjà été sélectionnées en interne et que leurs valeurs d'ID ont été verrouillées. Cela peut se produire soudainement lorsqu'un volume de données croissant amène le planificateur à choisir un plan d'exécution différent qui n'a pas été testé lors de la phase de développement. Dans ce cas, l'accumulation se produit parce que l'application appelle explicitement `pg_advisory_unlock` pour chaque valeur d'ID verrouillée. Mais elle ne trouve pas l'ensemble de verrous acquis sur les lignes qui n'ont pas été renvoyées. Comme les verrous sont acquis au niveau de la session, ils ne sont pas libérés automatiquement à la fin de la transaction.

Les pics de tentatives de verrouillage bloquées peuvent également être liés à des conflits involontaires. Lors de ces conflits, des parties non liées de l'application partagent par erreur le même espace d'ID de verrou.

Actions

Examinez la façon dont les verrous consultatifs sont utilisés par l'application et indiquez où et quand, dans le flux d'application, chaque type de verrou consultatif est acquis et libéré.

Déterminez si une session acquiert trop de verrous ou si une session longue ne libère pas les verrous suffisamment tôt, ce qui entraîne une accumulation lente des verrous. Vous pouvez corriger une accumulation lente de verrous au niveau de la session en mettant fin à la session à l'aide de `pg_terminate_backend(pid)`.

Un client en attente d'un verrou consultatif apparaît dans `pg_stat_activity` avec `wait_event_type=Lock` et `wait_event=advisory`. Vous pouvez obtenir des valeurs de

verrouillage spécifiques en interrogeant la vue système `pg_locks` sur le même `pid`, à la recherche de `locktype=advisory` et `granted=f`.

Vous pouvez ensuite identifier la session de blocage en interrogeant `pg_locks` sur le même verrou consultatif doté de `granted=t`, comme illustré dans l'exemple suivant.

```
SELECT blocked_locks.pid AS blocked_pid,
         blocking_locks.pid AS blocking_pid,
         blocked_activity.usename AS blocked_user,
         blocking_activity.usename AS blocking_user,
         now() - blocked_activity.xact_start AS blocked_transaction_duration,
         now() - blocking_activity.xact_start AS blocking_transaction_duration,
         concat(blocked_activity.wait_event_type,':',blocked_activity.wait_event) AS
 blocked_wait_event,
         concat(blocking_activity.wait_event_type,':',blocking_activity.wait_event) AS
 blocking_wait_event,
         blocked_activity.state AS blocked_state,
         blocking_activity.state AS blocking_state,
         blocked_locks.locktype AS blocked_locktype,
         blocking_locks.locktype AS blocking_locktype,
         blocked_activity.query AS blocked_statement,
         blocking_activity.query AS blocking_statement
    FROM pg_catalog.pg_locks blocked_locks
    JOIN pg_catalog.pg_stat_activity blocked_activity ON blocked_activity.pid =
 blocked_locks.pid
    JOIN pg_catalog.pg_locks blocking_locks
        ON blocking_locks.locktype = blocked_locks.locktype
        AND blocking_locks.DATABASE IS NOT DISTINCT FROM blocked_locks.DATABASE
        AND blocking_locks.relation IS NOT DISTINCT FROM blocked_locks.relation
        AND blocking_locks.page IS NOT DISTINCT FROM blocked_locks.page
        AND blocking_locks.tuple IS NOT DISTINCT FROM blocked_locks.tuple
        AND blocking_locks.virtualxid IS NOT DISTINCT FROM blocked_locks.virtualxid
        AND blocking_locks.transactionid IS NOT DISTINCT FROM
 blocked_locks.transactionid
        AND blocking_locks.classid IS NOT DISTINCT FROM blocked_locks.classid
        AND blocking_locks.objid IS NOT DISTINCT FROM blocked_locks.objid
        AND blocking_locks.objsubid IS NOT DISTINCT FROM blocked_locks.objsubid
        AND blocking_locks.pid != blocked_locks.pid
    JOIN pg_catalog.pg_stat_activity blocking_activity ON blocking_activity.pid =
 blocking_locks.pid
    WHERE NOT blocked_locks.GRANTED;
```

Toutes les fonctions d'API des verrous consultatifs comportent deux ensembles d'arguments, soit un argument `bigint`, soit deux arguments `integer` :

- Pour les fonctions d'API comportant un argument `bigint`, les 32 bits supérieurs figurent dans `pg_locks.classid` et les 32 bits inférieurs se trouvent dans `pg_locks.objid`.
- Pour les fonctions d'API comportant deux arguments `integer`, le premier argument est `pg_locks.classid` et le deuxième est `pg_locks.objid`.

La valeur `pg_locks.objsubid` indique quelle forme d'API a été utilisée : 1 pour un argument `bigint` et 2 pour deux arguments `integer`.

Lock:extend

L'événement `Lock:extend` se produit lorsqu'un processus backend attend de verrouiller une relation pour l'étendre alors qu'un autre processus présente un verrou sur cette relation dans le même but.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

L'événement `Lock:extend` indique qu'un processus backend attend d'étendre une relation sur laquelle un autre processus backend détient un verrou pendant qu'il étend cette relation. Comme un seul processus à la fois peut étendre une relation, le système génère un événement d'attente `Lock:extend`. Les opérations `INSERT`, `COPY` et `UPDATE` peuvent générer cet événement.

Causes probables de l'augmentation du nombre d'événements d'attente

Un événement `Lock:extend` trop fréquent peut révéler un problème de performances dont les causes sont généralement les suivantes :

Augmentation du nombre d'insertions ou de mises à jour simultanées dans la même table

Le nombre de sessions simultanées associées à des requêtes d'insertion ou de mise à jour peut augmenter.

Bande passante réseau insuffisante

La bande passante réseau de l'instance de base de données peut être insuffisante pour répondre aux besoins de communication de stockage de la charge de travail actuelle. Cela peut contribuer à une latence de stockage qui entraîne une augmentation des événements `Lock:extend`.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Réduisez les insertions et les mises à jour simultanées dans la même relation
- Augmentez la bande passante réseau

Réduisez les insertions et les mises à jour simultanées dans la même relation

Tout d'abord, déterminez s'il y a une augmentation des métriques `tup_inserted` et `tup_updated`, et une augmentation concomitante de cet événement d'attente. Si tel est le cas, déterminez quelles relations sont en conflit pour les opérations d'insertion et de mise à jour. Pour ce faire, interrogez la vue `pg_stat_all_tables` afin de connaître les valeurs des champs `n_tup_ins` et `n_tup_upd`. Pour en savoir plus sur la vue `pg_stat_all_tables`, consultez pg_stat_all_tables dans la documentation PostgreSQL.

Pour en savoir plus sur les requêtes de blocage et les requêtes bloquées, interrogez `pg_stat_activity` comme dans l'exemple suivant :

```
SELECT
    blocked.pid,
```

```
    blocked.usename,
    blocked.query,
    blocking.pid AS blocking_id,
    blocking.query AS blocking_query,
    blocking.wait_event AS blocking_wait_event,
    blocking.wait_event_type AS blocking_wait_event_type
FROM pg_stat_activity AS blocked
JOIN pg_stat_activity AS blocking ON blocking.pid = ANY(pg_blocking_pids(blocked.pid))
where
blocked.wait_event = 'extend'
and blocked.wait_event_type = 'Lock';

   pid  | usename  |            query             | blocking_id |
            blocking_query                          | blocking_wait_event |
 blocking_wait_event_type
  ------+----------+------------------------------+-------------
+-------------------------------------------------------------------
+--------------------+--------------------------
   7143 |  myuser  | insert into tab1 values (1); |        4600 | INSERT INTO tab1 (a)
 SELECT s FROM generate_series(1,1000000) s; | DataFileExtend      | IO
```

Après avoir identifié les relations qui contribuent à l'augmentation des événements `Lock:extend`, utilisez les techniques suivantes pour réduire les conflits :

- Déterminez si vous pouvez utiliser le partitionnement pour réduire les conflits relatifs à la même table. La séparation des tuples insérés ou mis à jour dans différentes partitions peut réduire les conflits. Pour plus d'informations sur le partitionnement, voir Gestion des partitions PostgreSQL avec l'extension pg_partman.
- Si l'événement d'attente est principalement dû à une activité de mise à jour, vous pouvez réduire la valeur du facteur de remplissage de la relation. Cela peut réduire les requêtes de nouveaux blocs lors de la mise à jour. Le facteur de remplissage est un paramètre de stockage relatif à une table qui détermine la quantité maximale d'espace requise pour remplir une page de la table. Il est exprimé en pourcentage de l'espace total d'une page. Pour en savoir plus sur le facteur de remplissage, consultez CREATE TABLE dans la documentation PostgreSQL.

> **Important**
>
> Si vous modifiez le facteur de remplissage, nous vous recommandons vivement de tester votre système, car en fonction de votre charge de travail, la modification de cette valeur peut nuire aux performances.

Augmentez la bande passante réseau

Pour déterminer si la latence d'écriture a augmenté, consultez la métrique `WriteLatency` dans CloudWatch. Le cas échéant, utilisez les métriques Amazon CloudWatch `WriteThroughput` et `ReadThroughput` pour surveiller le trafic lié au stockage sur l'instance de base de données. Ces métriques peuvent vous aider à déterminer si la bande passante réseau est suffisante pour l'activité de stockage de votre charge de travail.

Si la bande passante réseau est insuffisante, augmentez-la. Si votre instance de base de données atteint les limites de sa bande passante réseau, le seul moyen d'augmenter la bande passante est d'augmenter la taille de l'instance de base de données.

Pour de plus amples informations sur les métriques CloudWatch, veuillez consulter Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS. Pour en savoir plus sur les performances réseau de chaque classe d'instance de base de données, consultez Mesures au CloudWatch niveau de l'instance Amazon pour Amazon RDS.

Lock:Relation

L'événement `Lock:Relation` se produit lorsqu'une requête attend d'acquérir un verrou sur une table ou une vue (relation) actuellement verrouillée par une autre transaction.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

La plupart des commandes PostgreSQL utilisent implicitement des verrous pour contrôler les accès simultanés aux données des tables. Vous pouvez également utiliser ces verrous explicitement dans

le code de votre application grâce à la commande LOCK. De nombreux modes de verrouillage ne sont pas compatibles entre eux et peuvent bloquer des transactions lorsqu'ils tentent d'accéder au même objet. Dans ce cas, RDS for PostgreSQL génère un événement Lock:Relation. Voici quelques exemples courants :

- Des verrous exclusifs tels que ACCESS EXCLUSIVE peuvent bloquer tous les accès simultanés. Les opérations en langage de définition de données (DDL) telles que DROP TABLE, TRUNCATE, VACUUM FULL et CLUSTER acquièrent implicitement des verrous ACCESS EXCLUSIVE. ACCESS EXCLUSIVE est également le mode de verrouillage par défaut pour les instructions LOCK TABLE qui ne spécifient pas explicitement de mode.
- L'utilisation de CREATE INDEX (without CONCURRENT) sur une table entre en conflit avec les instructions du langage de manipulation de données (DML) UPDATE, DELETE et INSERT, qui acquièrent des verrous ROW EXCLUSIVE.

Pour en savoir plus sur les verrous de niveau table et les modes de verrouillage conflictuels, consultez Explicit Locking dans la documentation PostgreSQL.

Le déblocage lié aux requêtes et transactions de blocage s'effectue généralement de l'une des manières suivantes :

- Requête de blocage – L'application peut annuler la requête ou l'utilisateur peut mettre fin au processus. Le moteur peut également forcer la requête à se terminer en raison de l'expiration du délai d'attente d'une instruction de la session ou d'un mécanisme de détection de blocage.
- Transaction de blocage – Une transaction met fin à son blocage lorsqu'elle exécute une instruction ROLLBACK ou COMMIT. Les restaurations sont également automatiques lorsque les sessions sont déconnectées par un client ou suite à des problèmes de réseau, ou lorsqu'elles sont interrompues. Les sessions peuvent être interrompues lorsque le moteur de base de données est arrêté, lorsque le système est à court de mémoire, etc.

Causes probables de l'augmentation du nombre d'événements d'attente

Lorsque l'événement Lock:Relation se produit plus fréquemment que la normale, il peut indiquer un problème de performances. Les causes sont généralement les suivantes :

Augmentation du nombre de sessions simultanées avec des verrous de table conflictuels

Le nombre de sessions simultanées peut augmenter lorsque des requêtes verrouillent la même table avec des modes de verrouillage conflictuels.

Opérations de maintenance

Les opérations de maintenance liées à l'état comme `VACUUM` et `ANALYZE` peuvent considérablement augmenter le nombre de verrous en conflit. `VACUUM FULL` acquiert un verrou `ACCESS EXCLUSIVE`, et `ANALYSE` acquiert un verrou `SHARE UPDATE EXCLUSIVE`. Ces deux types de verrous peuvent provoquer un événement d'attente `Lock:Relation`. Les opérations de maintenance des données d'application telles que l'actualisation d'une vue matérialisée peuvent également augmenter le nombre de requêtes et de transactions bloquées.

Verrous sur les instances de lecture

Il peut y avoir un conflit entre les verrous relationnels des volumes en écriture et des volumes en lecture. Actuellement, seuls les verrous relationnels `ACCESS EXCLUSIVE` sont répliqués vers les instances de lecture. Cependant, le verrou relationnel `ACCESS EXCLUSIVE` entrera en conflit avec tout verrou relationnel `ACCESS SHARE` détenu par le volume en lecture. Cela peut entraîner une augmentation des événements d'attente de relation de verrouillage sur le volume en lecture.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Réduisez l'impact des instructions SQL de blocage
- Minimisez l'effet des opérations de maintenance

Réduisez l'impact des instructions SQL de blocage

Pour réduire l'impact des instructions SQL de blocage, si possible, modifiez le code de votre application. Voici deux techniques courantes pour réduire les blocages :

- Utilisez l'option `NOWAIT` – Certaines commandes SQL, telles que les instructions `SELECT` et `LOCK` prennent en charge cette option. La directive `NOWAIT` annule la requête liée à la demande de verrou si le verrou ne peut pas être acquis immédiatement. Cette technique permet d'éviter qu'une session de blocage ne provoque un empilement de sessions bloquées derrière elle.

  Par exemple, supposons que la transaction A attende un verrou détenu par la transaction B. Si B demande un verrou sur une table qui est verrouillée par la transaction C, la transaction A peut être bloquée jusqu'à ce que la transaction C se termine. Mais si la transaction B utilise

`NOWAIT` lorsqu'elle demande le verrou sur C, elle peut échouer rapidement et faire en sorte que la transaction A n'ait pas à attendre indéfiniment.

- Utilisez `SET lock_timeout` – Définissez une valeur `lock_timeout` afin de limiter le délai d'attente d'une instruction SQL pour acquérir un verrou sur une relation. Si le verrou n'est pas acquis dans le délai spécifié, la transaction qui a demandé celui-ci est annulée. Définissez cette valeur au niveau de la session.

Minimisez l'effet des opérations de maintenance

Les opérations de maintenance telles que `VACUUM` et `ANALYZE` sont importantes. Nous vous recommandons de ne pas les désactiver parce que vous trouvez des événements d'attente `Lock:Relation` liés à ces opérations de maintenance. Les approches suivantes peuvent minimiser l'effet de ces opérations :

- Exécutez les opérations de maintenance manuellement pendant les heures creuses.
- Pour réduire les attentes `Lock:Relation` causées par les tâches autovacuum, procédez aux réglages nécessaires de la fonction autovacuum. Pour en savoir plus sur le réglage de la fonction autovacuum, consultez Utilisation de la fonction autovacuum de PostgreSQL sur Amazon RDS dans le Guide de l'utilisateur Amazon RDS.

Lock:transactionid

L'événement `Lock:transactionid` se produit lorsqu'une transaction attend un verrou au niveau ligne.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

L'événement `Lock:transactionid` se produit lorsqu'une transaction tente d'acquérir un verrou de niveau ligne qui a déjà été accordé à une transaction exécutée en même temps. La session qui présente l'événement d'attente `Lock:transactionid` est bloquée à cause de ce verrou. Une fois la transaction de blocage terminée dans une instruction `COMMIT` ou `ROLLBACK`, la transaction bloquée peut se poursuivre.

La sémantique de contrôle de simultanéité multiversion de RDS for PostgreSQL garantit l'absence de blocage des processus de lecture par les processus d'écriture, et vice versa. Pour que des conflits se produisent au niveau ligne, les transactions de blocage et les transactions bloquées doivent émettre des instructions conflictuelles des types suivants :

- `UPDATE`
- `SELECT … FOR UPDATE`
- `SELECT … FOR KEY SHARE`

L'instruction `SELECT … FOR KEY SHARE` est un cas particulier. La base de données utilise la clause `FOR KEY SHARE` pour optimiser les performances de l'intégrité référentielle. La présence d'un verrou de niveau ligne sur une ligne peut bloquer les commandes `INSERT`, `UPDATE` et `DELETE` sur d'autres tables qui font référence à la ligne.

Causes probables de l'augmentation du nombre d'événements d'attente

Un événement trop fréquent est généralement dû à des instructions `UPDATE`, `SELECT … FOR UPDATE` ou `SELECT … FOR KEY SHARE` combinées aux conditions suivantes.

Rubriques

- Forte simultanéité
- État Idle in transaction (Transaction inactive)
- Transactions de longue durée

Forte simultanéité

RDS for PostgreSQL peut utiliser une sémantique de verrouillage détaillée de niveau ligne. La probabilité de conflits au niveau ligne augmente lorsque les conditions suivantes sont réunies :

- Une charge de travail à forte simultanéité se dispute les mêmes lignes.

- La simultanéité augmente.

État Idle in transaction (Transaction inactive)

Parfois, la colonne `pg_stat_activity.state` affiche la valeur `idle in transaction`. Cette valeur apparaît pour les sessions qui ont entamé une transaction, mais qui n'ont pas encore émis de commande `COMMIT` ou `ROLLBACK`. Si la valeur `pg_stat_activity.state` n'est pas `active`, la requête affichée dans `pg_stat_activity` est la plus récente à avoir été exécutée. La session de blocage ne traite pas activement une requête, car une transaction ouverte comporte un verrou.

Si une transaction inactive a acquis un verrou au niveau ligne, cela peut empêcher d'autres sessions de l'acquérir. Cette condition entraîne l'apparition fréquente de l'événement d'attente `Lock:transactionid`. Pour diagnostiquer le problème, examinez la sortie de `pg_stat_activity` et `pg_locks`.

Transactions de longue durée

Les transactions qui s'exécutent depuis longtemps comportent des verrous pendant longtemps. Ces verrous de longue durée peuvent bloquer l'exécution d'autres transactions.

Actions

Le verrouillage de ligne correspond à un conflit entre les instructions `UPDATE`, `SELECT … FOR UPDATE` ou `SELECT … FOR KEY SHARE`. Avant de rechercher une solution, déterminez quand ces instructions sont exécutées sur la même ligne. Utilisez ces informations pour choisir une des stratégies décrites dans les sections suivantes.

Rubriques

- Réagissez à une forte simultanéité
- Réagissez aux transactions inactives
- Réagissez aux transactions de longue durée

Réagissez à une forte simultanéité

En cas de problème lié à la simultanéité, essayez l'une des techniques suivantes :

- Réduisez la simultanéité dans l'application. Par exemple, réduisez le nombre de sessions actives.
- Implémentez un groupe de connexions. Pour savoir comment regrouper des connexions à l'aide de RDS Proxy, consultez Proxy Amazon RDS .

- Concevez l'application ou le modèle de données de manière à éviter les instructions `UPDATE` et `SELECT … FOR UPDATE` conflictuelles. Vous pouvez également réduire le nombre de clés étrangères accessibles par les instructions `SELECT … FOR KEY SHARE`.

Réagissez aux transactions inactives

Si `pg_stat_activity.state` indique `idle in transaction`, utilisez les stratégies suivantes :

- Si possible, activez la validation automatique. Cette approche empêche les transactions de bloquer d'autres transactions en attendant une instruction `COMMIT` ou `ROLLBACK`.
- Recherchez les chemins de code qui ne contiennent pas d'instruction `COMMIT`, `ROLLBACK` ou `END`.
- Assurez-vous que la logique de gestion des exceptions de votre application comporte toujours un chemin vers une `end of transaction` valide.
- Assurez-vous que votre application traite les résultats des requêtes après avoir mis fin à la transaction avec `COMMIT` ou `ROLLBACK`.

Réagissez aux transactions de longue durée

Si des transactions de longue durée sont à l'origine de l'apparition fréquente de `Lock:transactionid`, essayez les stratégies suivantes :

- N'utilisez pas de verrous de ligne dans les transactions de longue durée.
- Limitez la longueur des requêtes en implémentant la validation automatique chaque fois que possible.

Lock:tuple

L'événement `Lock:tuple` se produit lorsqu'un processus backend attend d'acquérir un verrou sur un tuple.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

L'événement `Lock:tuple` indique qu'un backend attend d'acquérir un verrou sur un tuple alors qu'un autre moteur détient un verrou conflictuel sur le même tuple. Le tableau suivant illustre un scénario dans lequel les sessions génèrent l'événement `Lock:tuple`.

Heure	Session 1	Session 2	Session 3
t1	Démarre une transacti on.		
t2	Met à jour la ligne 1.		
t3		Met à jour la ligne 1. La session acquiert un verrou exclusif sur le tuple, puis attend que la session 1 libère le verrou par le biais d'une validation ou d'une restauration.	
t4			Met à jour la ligne 1. La session attend que la session 2 libère le verrou exclusif sur le tuple.

Vous pouvez également simuler cet événement d'attente à l'aide de l'outil de définition de points de référence `pgbench`. Configurez un nombre élevé de sessions simultanées pour mettre à jour la même ligne au sein d'une table avec un fichier SQL personnalisé.

Pour en savoir plus sur les modes de verrouillage conflictuels, consultez Explicit Locking dans la documentation PostgreSQL. Pour en savoir plus sur `pgbench`, consultez pgbench dans la documentation PostgreSQL.

Causes probables de l'augmentation du nombre d'événements d'attente

Un événement de ce type trop fréquent peut révéler un problème de performances dont les causes sont généralement les suivantes :

- Un grand nombre de sessions simultanées tentent d'acquérir un verrou conflictuel pour le même tuple en exécutant des instructions `UPDATE` ou `DELETE`.
- Les sessions à forte simultanéité exécutent une instruction `SELECT` en utilisant les modes de verrouillage `FOR UPDATE` ou `FOR NO KEY UPDATE`.
- Divers facteurs poussent les groupes d'applications ou de connexions à ouvrir davantage de sessions pour exécuter les mêmes opérations. Comme de nouvelles sessions tentent de modifier les mêmes lignes, la charge de base de données peut augmenter et un événement `Lock:tuple` peut apparaître.

Pour en savoir plus, consultez Row-Level Locks dans la documentation PostgreSQL.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente.

Rubriques

- Examinez la logique de votre application
- Recherchez la session de blocage
- Réduisez la simultanéité lorsqu'elle est forte
- Résolvez les problèmes liés aux goulots d'étranglement

Examinez la logique de votre application

Déterminez si une session de blocage est restée longtemps dans l'état `idle in transaction`. Si tel est le cas, la solution à court terme peut consister à mettre fin à la session de blocage. Vous pouvez utiliser la fonction `pg_terminate_backend`. Pour en savoir plus sur cette fonction, consultez Server Signaling Functions dans la documentation PostgreSQL.

Pour une solution à long terme, procédez comme suit :

- Modifiez la logique de l'application.

- Utilisez le paramètre `idle_in_transaction_session_timeout`. Ce paramètre met fin à toute session associée à une transaction ouverte qui est restée inactive plus longtemps que la durée spécifiée. Pour en savoir plus, consultez Client Connection Defaults dans la documentation PostgreSQL.
- Chaque fois que possible, utilisez la validation automatique. Pour en savoir plus, consultez SET AUTOCOMMIT dans la documentation PostgreSQL.

Recherchez la session de blocage

Pendant l'événement d'attente `Lock:tuple`, identifiez le blocage et la session bloquée en déterminant quels verrous dépendent les uns des autres. Pour en savoir plus, consultez Lock dependency information dans le wiki PostgreSQL.

L'exemple suivant interroge toutes les sessions, en y appliquant le filtre `tuple` et en les classant par `wait_time`.

```
SELECT blocked_locks.pid AS blocked_pid,
         blocking_locks.pid AS blocking_pid,
         blocked_activity.usename AS blocked_user,
         blocking_activity.usename AS blocking_user,
         now() - blocked_activity.xact_start AS blocked_transaction_duration,
         now() - blocking_activity.xact_start AS blocking_transaction_duration,
         concat(blocked_activity.wait_event_type,':',blocked_activity.wait_event) AS
 blocked_wait_event,
         concat(blocking_activity.wait_event_type,':',blocking_activity.wait_event) AS
 blocking_wait_event,
         blocked_activity.state AS blocked_state,
         blocking_activity.state AS blocking_state,
         blocked_locks.locktype AS blocked_locktype,
         blocking_locks.locktype AS blocking_locktype,
         blocked_activity.query AS blocked_statement,
         blocking_activity.query AS blocking_statement
    FROM pg_catalog.pg_locks blocked_locks
    JOIN pg_catalog.pg_stat_activity blocked_activity ON blocked_activity.pid =
 blocked_locks.pid
    JOIN pg_catalog.pg_locks blocking_locks
        ON blocking_locks.locktype = blocked_locks.locktype
        AND blocking_locks.DATABASE IS NOT DISTINCT FROM blocked_locks.DATABASE
        AND blocking_locks.relation IS NOT DISTINCT FROM blocked_locks.relation
        AND blocking_locks.page IS NOT DISTINCT FROM blocked_locks.page
        AND blocking_locks.tuple IS NOT DISTINCT FROM blocked_locks.tuple
```

```
        AND blocking_locks.virtualxid IS NOT DISTINCT FROM blocked_locks.virtualxid
        AND blocking_locks.transactionid IS NOT DISTINCT FROM
blocked_locks.transactionid
        AND blocking_locks.classid IS NOT DISTINCT FROM blocked_locks.classid
        AND blocking_locks.objid IS NOT DISTINCT FROM blocked_locks.objid
        AND blocking_locks.objsubid IS NOT DISTINCT FROM blocked_locks.objsubid
        AND blocking_locks.pid != blocked_locks.pid
    JOIN pg_catalog.pg_stat_activity blocking_activity ON blocking_activity.pid =
blocking_locks.pid
    WHERE NOT blocked_locks.GRANTED;
```

Réduisez la simultanéité lorsqu'elle est forte

L'événement `Lock:tuple` peut se produire constamment, en particulier lorsque la charge de travail est élevée. Dans ce cas, réduisez la simultanéité pour les lignes très occupées. Souvent, quelques lignes seulement contrôlent une file d'attente ou la logique booléenne, ce qui explique pourquoi ces lignes sont très occupées.

Vous pouvez réduire la simultanéité en utilisant différentes approches basées sur les besoins métier, la logique de l'application et le type de charge de travail. Par exemple, vous pouvez effectuer les opérations suivantes :

- Redéfinissez la logique de votre table et de vos données pour réduire la simultanéité.
- Modifiez la logique de l'application pour réduire la simultanéité au niveau ligne.
- Exploitez et redéfinissez les requêtes avec des verrous de niveau ligne.
- Utilisez la clause `NOWAIT` lors des nouvelles tentatives.
- Envisagez d'utiliser un contrôle de simultanéité logique optimiste et à verrouillage hybride.
- Envisagez de modifier le niveau d'isolement de la base de données.

Résolvez les problèmes liés aux goulots d'étranglement

L'événement `Lock:tuple` peut se produire avec des goulots d'étranglement tels qu'une pénurie d'UC ou une saturation de la bande passante d'Amazon EBS. Pour réduire les goulots d'étranglement, adoptez les approches suivantes :

- Procédez à une augmentation d'échelle de votre type de classe d'instance.
- Optimisez les requêtes gourmandes en ressources.
- Modifiez la logique de l'application.

- Archivez les données rarement consultées.

LWLock:BufferMapping (LWLock:buffer_mapping)

Cet événement se produit lorsqu'une session attend d'associer un bloc de données à une mémoire tampon dans le groupe de mémoires tampons partagées.

> **Note**
>
> Cet événement est nommé `LWLock:BufferMapping` pour RDS for PostgreSQL 13 et versions ultérieures. Pour RDS for PostgreSQL version 12 et versions antérieures, cet événement est nommé `LWLock:buffer_mapping`.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente s'appliquent à RDS for PostgreSQL 9.6 et versions ultérieures.

Contexte

Le groupe de mémoires tampons partagées est une zone de mémoire PostgreSQL qui contient toutes les pages actuellement ou précédemment utilisées par les processus. Lorsqu'un processus a besoin d'une page, il la lit dans le groupe de mémoires tampons partagées. Le paramètre `shared_buffers` définit la taille de la mémoire tampon partagée et réserve une zone de mémoire pour stocker la table et les pages d'index. Si vous modifiez ce paramètre, veillez à redémarrer la base de données.

L'événement d'attente `LWLock:buffer_mapping` se produit dans les scénarios suivants :

- Un processus recherche une page dans la table des mémoires tampons et acquiert un verrou de mappage de mémoire tampon partagée.

- Un processus charge une page dans le groupe de mémoires tampons et acquiert un verrou exclusif de mappage de mémoire tampon.
- Un processus supprime une page du groupe et acquiert un verrou exclusif de mappage de mémoire tampon.

Causes

Lorsque cet événement se produit plus souvent qu'à l'accoutumée, indiquant un possible problème de performances, la base de données effectue une pagination dans et hors du groupe de mémoires tampons partagées. Les causes sont généralement les suivantes :

- Requêtes volumineuses
- Index et tables gonflés
- Analyses de tables complètes
- Taille de groupe partagé inférieure à celle de l'ensemble de travail

Actions

Nous vous recommandons différentes actions en fonction de l'origine de votre événement d'attente.

Rubriques

- Surveillez les métriques liées à la mémoire tampon
- Évaluez votre stratégie d'indexation
- Réduisez le nombre de mémoires tampons qui doivent être allouées rapidement

Surveillez les métriques liées à la mémoire tampon

Lorsque les événements d'attente `LWLock:buffer_mapping` atteignent un pic, examinez le taux d'accès à la mémoire tampon. Vous pouvez utiliser ces métriques pour mieux comprendre ce qui se passe dans le cache de mémoire tampon. Examinez les métriques suivantes :

`blks_hit`

Cette métrique de compteur Performance Insights indique le nombre de blocs qui ont été récupérés à partir du groupe de mémoires tampons partagées. Après l'apparition de l'événement d'attente `LWLock:buffer_mapping`, vous pouvez observer un pic dans `blks_hit`.

`blks_read`

Cette mesure de compteur Performance Insights indique le nombre de blocs pour lesquels une lecture des I/O a été nécessaire dans le groupe de mémoires tampons partagées. Vous observerez peut-être un pic de `blks_read` dans la période précédant l'événement d'attente `LWLock:buffer_mapping`.

Évaluez votre stratégie d'indexation

Pour vous assurer que votre stratégie d'indexation ne nuit pas aux performances, vérifiez les éléments suivants :

Gonflement des index

Assurez-vous que le gonflement des index et des tables n'entraîne pas la lecture de pages inutiles dans la mémoire tampon partagée. Si vos tables contiennent des lignes inutilisées, archivez les données et supprimez les lignes des tables. Vous pouvez ensuite reconstruire les index des tables redimensionnées.

Index pour les requêtes fréquemment utilisées

Pour déterminer si vos index sont optimaux, surveillez les métriques du moteur de base de données dans Performance Insights. La métrique `tup_returned` indique le nombre de lignes lues. La métrique `tup_fetched` indique le nombre de lignes renvoyées au client. Si la métrique `tup_returned` est nettement supérieure à la métrique `tup_fetched`, les données risquent de ne pas être correctement indexées. De plus, les statistiques de votre table ne sont peut-être pas à jour.

Réduisez le nombre de mémoires tampons qui doivent être allouées rapidement

Pour réduire le nombre d'événements d'attente `LWLock:buffer_mapping`, essayez de réduire le nombre de mémoires tampons qui doivent être allouées rapidement. Une stratégie consiste à effectuer des opérations par lots de plus petite taille. Vous pouvez obtenir des lots plus petits en partitionnant vos tables.

LWLock:BufferIO (IPC:BufferIO)

L'`LWLock:BufferIO`événement se produit lorsque RDS pour PostgreSQL attend que d'autres processus input/output terminent leurs opérations (E/S) lorsqu'ils tentent simultanément d'accéder à une page. Le but est que la même page soit lue dans la mémoire tampon partagée.

Rubriques

- Versions de moteur pertinentes
- Contexte
- Causes
- Actions

Versions de moteur pertinentes

Ces informations sur les événements d'attente s'appliquent à toutes les versions de RDS for PostgreSQL. Pour RDS for PostgreSQL 12 et versions antérieures, cet événement d'attente est nommé lwlock:buffer_io, tandis qu'il est nommé lwlock:bufferio dans la version RDS for PostgreSQL 13. Depuis la version RDS for PostgreSQL 14, l'événement d'attente BufferIO est passé du type d'événement d'attente (IPC:Bufferio) `LWLock` à `IPC`.

Contexte

Chaque mémoire tampon partagée possède un I/O verrou associé à l'événement d'`LWLock:BufferIO`attente, chaque fois qu'un bloc (ou une page) doit être récupéré en dehors du pool de mémoire tampon partagée.

Ce verrou est utilisé pour gérer plusieurs sessions qui ont toutes besoin d'accéder au même bloc. Ce bloc doit être lu en dehors du groupe de mémoires tampons partagées, défini par le paramètre `shared_buffers`.

Dès que la page est lue dans le groupe de mémoires tampons partagées, le verrou `LWLock:BufferIO` est libéré.

> **Note**
>
> L'événement d'attente `LWLock:BufferIO` précède l'événement d'attente IO : DataFileRead. L'événement d'attente `IO:DataFileRead` se produit lorsque les données sont lues à partir du stockage.

Pour en savoir plus sur les verrous légers, consultez Présentation du verrouillage.

Causes

Les principales causes de l'événement d'attente `LWLock:BufferIO` sont les suivantes :

- Plusieurs backends ou connexions essayant d'accéder à la même page qui est également en attente d'une opération I/O
- Rapport entre la taille du groupe de mémoires tampons partagées (défini par le paramètre `shared_buffers`) et le nombre de mémoires tampons nécessaires à la charge de travail actuelle
- La taille du groupe de mémoires tampons partagées n'est pas bien équilibrée par rapport au nombre de pages expulsées par d'autres opérations
- Index volumineux ou gonflés qui obligent le moteur à lire plus de pages que nécessaire dans le groupe de mémoires tampons partagées
- Absence d'index qui oblige le moteur de base de données à lire plus de pages que nécessaire dans les tables
- Points de contrôle trop fréquents ou nécessité de vider un trop grand nombre de pages modifiées
- Pics soudains de connexions à la base de données tentant d'effectuer des opérations sur la même page

Actions

Nous vous recommandons différentes actions en fonction de l'origine de votre événement d'attente :

- Réglez `max_wal_size` et `checkpoint_timeout` en fonction de l'heure de pointe de votre charge de travail si vous constatez que `LWLock:BufferIO` coïncide avec des baisses de la métrique `BufferCacheHitRatio`. Identifiez ensuite la requête qui pourrait en être la cause.
- Recherchez les index inutilisés et supprimez-les.
- Utilisez des tables partitionnées (qui comportent également des index partitionnés). Cela permet de limiter la réorganisation des index et de réduire son impact.
- Évitez d'indexer inutilement des colonnes.
- Empêchez les pics soudains de connexions à la base de données en utilisant un groupe de connexions.
- En guise de bonne pratique, limitez le nombre maximal de connexions à la base de données.

LWLock:buffer_content (BufferContent)

L'événement `LWLock:buffer_content` se produit lorsqu'une session attend de lire ou d'écrire une page de données en mémoire alors que celle-ci est verrouillée en écriture dans une autre

session. Dans RDS for PostgreSQL 13 et versions ultérieures, cet événement d'attente est appelé `BufferContent`.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

Pour lire ou manipuler des données, PostgreSQL y accède via des mémoires tampons partagées. Pour lire à partir de la mémoire tampon, un processus obtient un verrou léger (LWLock) sur le contenu de la mémoire tampon en mode partagé. Pour écrire dans la mémoire tampon, il obtient ce verrou en mode exclusif. Les verrous partagés permettent à d'autres processus d'acquérir simultanément des verrous partagés sur ce contenu. Les verrous exclusifs empêchent les autres processus d'obtenir tout type de verrou sur ce contenu.

L'événement `LWLock:buffer_content` (`BufferContent`) indique que plusieurs processus tentent d'obtenir un verrou sur le contenu d'une mémoire tampon spécifique.

Causes probables de l'augmentation du nombre d'événements d'attente

Un événement `LWLock:buffer_content` (`BufferContent`) trop fréquent peut révéler un problème de performances dont les causes sont généralement les suivantes :

Augmentation des mises à jour simultanées des mêmes données

> Le nombre de sessions simultanées associées à des requêtes de mise à jour du même contenu de mémoire tampon peut augmenter. Ce conflit peut être plus marqué sur les tables contenant beaucoup d'index.

Les données de la charge de travail ne sont pas en mémoire

Lorsque les données traitées par la charge de travail active ne sont pas en mémoire, la fréquence de ces événements d'attente peut augmenter. Cet effet est dû au fait que les processus détenant des verrous peuvent les conserver plus longtemps pendant qu'ils effectuent des opérations d'I/O disque.

Utilisation excessive de contraintes de clé étrangère

Les contraintes de clé étrangère peuvent augmenter la durée pendant laquelle un processus conserve un verrou de contenu de mémoire tampon. Cet effet est dû au fait que les opérations de lecture ont besoin d'un verrou de contenu de mémoire tampon partagée sur la clé référencée pendant la mise à jour de cette clé.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente. Vous pouvez identifier les événements `LWLock:buffer_content` (`BufferContent`) en utilisant Amazon RDS Performance Insights ou en interrogeant la vue `pg_stat_activity`.

Rubriques

- Améliorez l'efficacité en mémoire
- Réduisez l'utilisation des contraintes de clé étrangère
- Supprimez les index inutilisés
- Augmentation de la taille du cache lors de l'utilisation de séquences

Améliorez l'efficacité en mémoire

Pour que les données de la charge de travail active aient plus de chances d'être mises en mémoire, partitionnez les tables ou procédez à une augmentation d'échelle de votre classe d'instance. Pour plus d'informations sur les classes d'instances de base de données, consultez Classes d'instances de base de données .

Réduisez l'utilisation des contraintes de clé étrangère

Examinez les charges de travail qui présentent un nombre élevé d'événements d'attente `LWLock:buffer_content` (`BufferContent`) pour déterminer si des contraintes de clé étrangère sont utilisées. Supprimez les contraintes de clé étrangère inutiles.

Supprimez les index inutilisés

Pour les charges de travail qui présentent un nombre élevé d'événements d'attente `LWLock:buffer_content` (`BufferContent`), identifiez les index inutilisés et supprimez-les.

Augmentation de la taille du cache lors de l'utilisation de séquences

Si vos tables utilisent des séquences, augmentez la taille du cache pour éliminer les conflits sur les pages des séquences et les pages d'index. Chaque séquence correspond à une page unique en mémoire partagée. Le cache prédéfini s'applique à chaque connexion. Cela peut ne pas être suffisant pour gérer la charge de travail lorsque de nombreuses sessions simultanées obtiennent une valeur de séquence.

LWLock:lock_manager (:lockmanager) LWLock

Cet événement se produit lorsque le moteur RDS for PostgreSQL conserve la zone de mémoire du verrou partagé pour allouer, vérifier et libérer un verrou quand il est impossible d'utiliser un verrou à chemin d'accès rapide.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente s'appliquent à RDS for PostgreSQL 9.6 et versions ultérieures. Pour les versions de RDS for PostgreSQL antérieures à la version 13, le nom de cet événement d'attente est `LWLock:lock_manager`. Pour RDS for PostgreSQL 13 et versions ultérieures, le nom de cet événement d'attente est `LWLock:lockmanager`.

Contexte

Lorsque vous émettez une instruction SQL, RDS for PostgreSQL enregistre des verrous pour protéger la structure, les données et l'intégrité de votre base de données pendant les opérations simultanées. Le moteur peut atteindre cet objectif en utilisant un verrou à chemin d'accès rapide ou

non rapide. Un verrou à chemin d'accès non rapide est plus coûteux et génère plus de frais qu'un verrou à chemin d'accès rapide.

Verrouillage à chemin d'accès rapide

Pour réduire les frais liés aux verrous qui sont fréquemment acquis et libérés, mais qui entrent rarement en conflit, les processus backend peuvent utiliser le verrouillage à chemin d'accès rapide. La base de données utilise ce mécanisme pour les verrous qui répondent aux critères suivants :

- Ils utilisent la méthode de verrouillage DEFAULT.
- Ils représentent un verrou sur une relation de base de données plutôt qu'une relation partagée.
- Il s'agit de verrous faibles qui sont peu susceptibles d'entrer en conflit.
- Le moteur peut rapidement vérifier qu'aucun verrou conflictuel ne peut exister.

Le moteur ne peut pas utiliser de verrouillage à chemin d'accès rapide lorsque l'une des conditions suivantes est vraie :

- Le verrou ne répond pas aux critères précédents.
- Il n'y a plus d'emplacements disponibles pour le processus backend.

Pour adapter vos requêtes au verrouillage rapide des chemins, vous pouvez utiliser la requête suivante.

```
SELECT count(*), pid, mode, fastpath
  FROM pg_locks
 WHERE fastpath IS NOT NULL
 GROUP BY 4,3,2
 ORDER BY pid, mode;
 count | pid  |      mode       | fastpath
-------+------+-----------------+----------
16 | 9185 | AccessShareLock | t
336 | 9185 | AccessShareLock | f
1 | 9185 | ExclusiveLock   | t
```

La requête suivante affiche uniquement le total sur la base de données.

```
SELECT count(*), mode, fastpath
  FROM pg_locks
```

```
 WHERE fastpath IS NOT NULL
 GROUP BY 3,2
 ORDER BY mode,1;
count |      mode       | fastpath
-------+-----------------+----------
16 | AccessShareLock | t
337 | AccessShareLock | f
1 | ExclusiveLock   | t
(3 rows)
```

Pour en savoir plus sur le verrouillage à chemin d'accès rapide, consultez fast path dans le fichier README du gestionnaire de verrous PostgreSQL et pg-locks dans la documentation PostgreSQL.

Exemple de problème de mise à l'échelle pour le gestionnaire de verrous

Dans cet exemple, une table nommée `purchases` stocke cinq ans de données, partitionnées par jour. Chaque partition possède deux index. La séquence d'événements suivante se produit :

1. Vous interrogez des données réparties sur différents jours, ce qui oblige la base de données à lire de nombreuses partitions.
2. La base de données crée une entrée de verrou pour chaque partition. Si les index de partition font partie du chemin d'accès de l'optimiseur, la base de données crée également une entrée de verrou pour eux.
3. Lorsque le nombre d'entrées de verrou demandées pour le même processus backend est supérieur à 16, ce qui correspond à la valeur de `FP_LOCK_SLOTS_PER_BACKEND`, le gestionnaire de verrous utilise la méthode de verrouillage à chemin d'accès non rapide.

Les applications modernes peuvent comporter des centaines de sessions. Si des sessions simultanées interrogent le parent sans élaguer correctement les partitions, la base de données peut créer des centaines, voire des milliers, de verrous à chemin d'accès non rapide. Généralement, lorsque cette simultanéité est supérieure au nombre de vCPUs, l'événement d'`LWLock:lock_manager`attente apparaît.

> **Note**
>
> L'événement d'attente `LWLock:lock_manager` n'est pas lié au nombre de partitions ou d'index contenus dans un schéma de base de données. Il est plutôt lié au nombre de verrous à chemin d'accès non rapide que la base de données doit contrôler.

Causes probables de l'augmentation du nombre d'événements d'attente

Lorsque l'événement d'attente `LWLock:lock_manager` se produit plus souvent qu'à l'accoutumée, indiquant un possible problème de performances, les causes les plus probables des pics soudains sont les suivantes :

- Les sessions actives simultanées exécutent des requêtes qui n'utilisent pas de verrous à chemin d'accès rapide. Ces sessions dépassent également le nombre maximum de vCPU.
- Un grand nombre de sessions actives simultanées accèdent à une table fortement partitionnée. Chaque partition possède plusieurs index.
- La base de données subit une tempête de connexions. Par défaut, certaines applications et certains logiciels de regroupement de connexions créent davantage de connexions lorsque la base de données est lente. Cette pratique aggrave le problème. Réglez le logiciel de regroupement de connexions de manière à éviter les tempêtes de connexions.
- Un grand nombre de sessions interrogent une table parente sans élaguer les partitions.
- Un langage de définition de données (DDL), un langage de manipulation de données (DML) ou une commande de maintenance verrouille exclusivement une relation occupée ou des tuples fréquemment consultés ou modifiés.

Actions

L'événement d'attente `CPU` n'est pas nécessairement lié à un problème de performances. Ne réagissez à cet événement que lorsque les performances se dégradent et que cet événement d'attente domine la charge de la base de données.

Rubriques

- Élaguez les partitions
- Supprimez les index inutiles
- Réglez vos requêtes pour qu'elles utilisent le verrouillage à chemin d'accès rapide
- Procédez au réglage d'autres événements d'attente
- Réduisez les goulots d'étranglement matériels
- Utilisez une fonction de regroupement de connexions
- Mise à niveau de votre version de RDS for PostgreSQL

Élaguez les partitions

L'élagage des partitions est une stratégie d'optimisation des requêtes pour tables partitionnées de manière déclarative, qui exclut les partitions inutiles des analyses de tables, améliorant ainsi les performances. L'élagage des partitions est activé par défaut. S'il est désactivé, activez-le comme suit.

```
SET enable_partition_pruning = on;
```

Les requêtes peuvent tirer parti de l'élagage des partitions lorsque leur clause `WHERE` contient la colonne utilisée pour le partitionnement. Pour en savoir plus, consultez Partition Pruning dans la documentation PostgreSQL.

Supprimez les index inutiles

Votre base de données peut contenir des index inutilisés ou rarement utilisés. Si tel est le cas, pensez à les supprimer. Effectuez l'une des actions suivantes :

- Pour en savoir plus sur la recherche des index inutiles, consultez Unused Indexes dans le wiki PostgreSQL.
- Exécutez PG Collector. Ce script SQL rassemble les informations de la base de données et les présente sous forme de rapport HTML. Consultez la section « Unused indexes » (Index inutilisés). Pour plus d'informations, consultez pg-collector dans le référentiel AWS Labs GitHub .

Réglez vos requêtes pour qu'elles utilisent le verrouillage à chemin d'accès rapide

Pour savoir si vos requêtes utilisent le verrouillage à chemin d'accès rapide, interrogez la colonne `fastpath` de la table `pg_locks`. Si vos requêtes n'utilisent pas le verrouillage à chemin d'accès rapide, essayez de réduire le nombre de relations par requête à moins de 16.

Procédez au réglage d'autres événements d'attente

Si `LWLock:lock_manager` est premier ou deuxième dans la liste des attentes les plus fréquentes, vérifiez si les événements d'attente suivants apparaissent également dans la liste :

- `Lock:Relation`
- `Lock:transactionid`
- `Lock:tuple`

S'ils figurent parmi les premiers de la liste, commencez par régler ces événements d'attente. Ces événements peuvent être un moteur pour `LWLock:lock_manager`.

Réduisez les goulots d'étranglement matériels

Un goulot d'étranglement matériel peut se produire, comme une pénurie d'UC ou une saturation de votre bande passante Amazon EBS. Envisagez alors de réduire les goulots d'étranglement matériels. Procédez comme suit :

- Procédez à une augmentation d'échelle de votre classe d'instance.
- Optimisez les requêtes qui sollicitent énormément l'UC et la mémoire.
- Modifiez la logique de votre application.
- Archivez vos données.

Pour en savoir plus sur l'UC, la mémoire et la bande passante réseau EBS, consultez Types d'instances Amazon RDS.

Utilisez une fonction de regroupement de connexions

Si le nombre total de connexions actives dépasse le nombre maximal de vCPU, cela signifie que l'UC requise par les processus du système d'exploitation est supérieure à ce que votre type d'instance peut prendre en charge. Dans ce cas, vous pouvez utiliser ou régler un groupe de connexions. Pour plus d'informations sur le v correspondant CPUs à votre type d'instance, consultez Amazon RDS Instance Types.

Pour en savoir plus sur les groupes de connexions, consultez les ressources suivantes :

- Proxy Amazon RDS
- pgbouncer
- Connection Pools and Data Sources dans la documentation PostgreSQL

Mise à niveau de votre version de RDS for PostgreSQL

Si votre version actuelle de RDS for PostgreSQL est inférieure à 12, procédez à une mise à niveau vers la version 12 ou ultérieure. PostgreSQL versions 12 et ultérieures disposent d'un mécanisme de partition amélioré. Pour en savoir plus la version 12, consultez le document PostgreSQL 12.0 Release Notes. Pour plus d'informations sur la mise à niveau de RDS for PostgreSQL, consultez Mises à niveau du moteur de base de données RDS pour PostgreSQL.

LWLock:pg_stat_statements

L'événement d' LWLockattente :pg_stat_statements se produit lorsque l'extension verrouille de manière exclusive la `pg_stat_statements` table de hachage qui suit les instructions SQL. Cela se produit dans les scénarios suivants :

- Lorsque le nombre d'instructions suivies atteint la valeur de `pg_stat_statements.max` paramètre configurée et qu'il est nécessaire de faire de la place pour d'autres entrées, l'extension effectue un tri sur le nombre d'appels, supprime les 5 % des instructions les moins exécutées et remplit à nouveau le hachage avec les entrées restantes.
- When `pg_stat_statements` effectue une `garbage collection` opération sur le `pgss_query_texts.stat` fichier sur le disque et réécrit le fichier.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'allongement des temps d'attente
- Actions

Versions de moteur prises en charge

Contexte

Comprendre l'extension pg_stat_statements — L'extension pg_stat_statements suit les statistiques d'exécution des instructions SQL dans une table de hachage. L'extension suit les instructions SQL jusqu'à la limite définie par le `pg_stat_statements.max` paramètre. Ce paramètre détermine le nombre maximum d'instructions pouvant être suivies, ce qui correspond au nombre maximum de lignes dans la vue pg_stat_statements.

Persistance des statistiques des instructions : l'extension conserve les statistiques des instructions lors des redémarrages de l'instance en :

- Écrire des données dans un fichier nommé pg_stat_statements.stat
- Utilisation du paramètre pg_stat_statements.save pour contrôler le comportement de persistance

Lorsque pg_stat_statements.save est défini sur :

- activé (par défaut) : les statistiques sont enregistrées à l'arrêt et rechargées au démarrage du serveur
- off : les statistiques ne sont ni enregistrées à l'arrêt ni rechargées au démarrage du serveur

Stockage du texte des requêtes : l'extension stocke le texte des requêtes suivies dans un fichier nomméepgss_query_texts.stat. Ce fichier peut atteindre le double de la taille moyenne de toutes les instructions SQL suivies avant la collecte des déchets. L'extension nécessite un verrouillage exclusif de la table de hachage pendant les opérations de nettoyage et de pgss_query_texts.stat réécriture du fichier.

Processus de désallocation des relevés — Lorsque le nombre de relevés suivis atteint la pg_stat_statements.max limite et que de nouveaux relevés doivent être suivis, l'extension :

- Prend un verrou exclusif (:pg_stat_statementsLWLock) sur la table de hachage.
- Charge les données existantes dans la mémoire locale.
- Effectue un tri rapide en fonction du nombre d'appels.
- Supprime les instructions les moins appelées (5 % inférieurs).
- Reremplit la table de hachage avec les entrées restantes.

Surveillance de la désallocation des instructions — Dans PostgreSQL 14 et versions ultérieures, vous pouvez surveiller la désallocation des instructions à l'aide de la vue pg_stat_statements_info. Cette vue inclut une colonne dealloc qui indique le nombre de fois où des instructions ont été désallouées pour faire de la place à de nouvelles.

Si la désallocation des instructions se produit fréquemment, cela entraînera une collecte plus fréquente du pgss_query_texts.stat fichier sur le disque.

Causes probables de l'allongement des temps d'attente

Les causes typiques de l'augmentation des temps d'LWLock:pg_stat_statementsattente sont les suivantes :

- Augmentation du nombre de requêtes uniques utilisées par l'application.
- La valeur du pg_stat_statements.max paramètre est faible par rapport au nombre de requêtes uniques utilisées.

Actions

Nous vous recommandons différentes actions en fonction des causes de votre événement d'attente. Vous pouvez identifier les `LWLock:pg_stat_statements` événements à l'aide d'Amazon RDS Performance Insights ou en interrogeant la vue. `pg_stat_activity`

Ajustez les `pg_stat_statements` paramètres suivants pour contrôler le comportement de suivi et réduire:les instructions LWLock pg_stat_ attendent les événements.

Rubriques

- Désactiver le paramètre pg_stat_statements.track
- Augmenter le paramètre pg_stat_statements.max
- Désactiver le paramètre pg_stat_statements.track_utility

Désactiver le paramètre pg_stat_statements.track

Si l'événement:pg_stat_statements LWLock wait a un impact négatif sur les performances de la base de données et qu'une solution rapide est requise avant de poursuivre l'analyse de la vue `pg_stat_statements` afin d'identifier la cause première, le paramètre `pg_stat_statements.track` peut être désactivé en le définissant sur. `none` Cela désactivera la collecte des statistiques des relevés.

Augmenter le paramètre pg_stat_statements.max

Pour réduire la désallocation et minimiser la collecte de déchets du `pgss_query_texts.stat` fichier sur le disque, augmentez la valeur du `pg_stat_statements.max` paramètre. La valeur par défaut est `5,000`.

> **Note**
>
> Le `pg_stat_statements.max` paramètre est statique. Vous devez redémarrer votre instance de base de données pour appliquer les modifications apportées à ce paramètre.

Désactiver le paramètre pg_stat_statements.track_utility

Vous pouvez analyser la vue pg_stat_statements pour déterminer les commandes utilitaires consommant le plus de ressources suivies. `pg_stat_statements`

Le `pg_stat_statements.track_utility` paramètre contrôle si le module suit les commandes utilitaires, qui incluent toutes les commandes sauf SELECT, INSERT, UPDATE, DELETE et MERGE. Par défaut, ce paramètre est défini sur on.

Par exemple, lorsque votre application utilise de nombreuses requêtes de point de sauvegarde, qui sont intrinsèquement uniques, cela peut augmenter la désallocation des instructions. Pour résoudre ce problème, vous pouvez désactiver le `pg_stat_statements.track_utility` paramètre pour arrêter le suivi des requêtes `pg_stat_statements` de point de sauvegarde.

> **Note**
>
> Le `pg_stat_statements.track_utility` paramètre est un paramètre dynamique. Vous pouvez modifier sa valeur sans redémarrer votre instance de base de données.

Example Exemple de requêtes de point de sauvegarde uniques dans pg_stat_statements

```
                    query                         |       queryid
--------------------------------------------------+---------------------
 SAVEPOINT JDBC_SAVEPOINT_495701                  | -7249565344517699703
 SAVEPOINT JDBC_SAVEPOINT_1320                    | -1572997038849006629
 SAVEPOINT JDBC_SAVEPOINT_26739                   |  54791337410474486
 SAVEPOINT JDBC_SAVEPOINT_1294466                 |  8170064357463507593
 ROLLBACK TO SAVEPOINT JDBC_SAVEPOINT_65016       | -33608214779996400
 SAVEPOINT JDBC_SAVEPOINT_14185                   | -2175035613806809562
 SAVEPOINT JDBC_SAVEPOINT_45837                   | -6201592986750645383
 SAVEPOINT JDBC_SAVEPOINT_1324                    |  6388797791882029332
```

PostgreSQL 17 introduit plusieurs améliorations pour le suivi des commandes utilitaires :

- Les noms des points de sauvegarde sont désormais affichés sous forme de constantes.
- La transaction globale IDs (GIDs) des commandes de validation en deux phases est désormais affichée sous forme de constantes.
- Les noms des instructions DEALLOCATE sont affichés sous forme de constantes.
- Les paramètres CALL sont désormais affichés sous forme de constantes.

Timeout:PgSleep

L'événement `Timeout:PgSleep` se produit lorsqu'un processus serveur a appelé la fonction `pg_sleep` et attend l'expiration du délai de mise en veille.

Rubriques

- Versions de moteur prises en charge
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Causes probables de l'augmentation du nombre d'événements d'attente

Cet événement d'attente se produit lorsqu'une application, une fonction stockée ou un utilisateur émet une instruction SQL qui appelle l'une des fonctions suivantes :

- `pg_sleep`
- `pg_sleep_for`
- `pg_sleep_until`

Les fonctions précédentes retardent l'exécution jusqu'à ce que le nombre de secondes spécifié se soit écoulé. Par exemple, `SELECT pg_sleep(1)` marque une pause d'une seconde. Pour en savoir plus, consultez Delaying Execution dans la documentation PostgreSQL.

Actions

Identifiez l'instruction qui exécutait la fonction `pg_sleep`. Déterminez si l'utilisation de la fonction est appropriée.

Timeout:VacuumDelay

L'événement `Timeout:VacuumDelay` indique que la limite de coût des E/S du processus vacuum a été dépassée et que le processus vacuum a été mis en veille. Les opérations vacuum

s'arrêtent pendant la durée spécifiée dans le paramètre de délai de coût correspondant, puis le processus reprend. Pour la commande vacuum manuelle, le délai est spécifié dans le paramètre `vacuum_cost_delay`. Pour le démon autovacuum, le délai est spécifié dans `autovacuum_vacuum_cost_delay parameter.`

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de l'augmentation du nombre d'événements d'attente
- Actions

Versions de moteur prises en charge

Ces informations sur les événements d'attente sont prises en charge pour toutes les versions de RDS for PostgreSQL.

Contexte

PostgreSQL possède à la fois un démon autovacuum et une commande vacuum manuelle. Le processus autovacuum est « activé » par défaut pour les instances de base de données RDS for PostgreSQL. La commande vacuum manuelle est utilisée selon les besoins, par exemple pour purger les tables des tuples morts ou générer de nouvelles statistiques.

Lorsque le processus vacuum est en cours, PostgreSQL utilise un compteur interne pour suivre les coûts estimés au fur et à mesure que le système effectue les diverses opérations d'E/S. Quand le compteur atteint la valeur spécifiée par le paramètre de limite de coût, le processus exécutant l'opération est en veille pendant la brève durée spécifiée dans le paramètre de délai de coût. Il réinitialise ensuite le compteur et poursuit les opérations.

Le processus vacuum comporte des paramètres qui peuvent être utilisés pour réguler la consommation de ressources. Le processus autovacuum et la commande vacuum manuelle ont leurs propres paramètres pour définir la valeur limite de coût. Ils ont également leurs propres paramètres pour spécifier un délai de coût, c'est-à-dire le temps alloué pour mettre en veille le processus vacuum quand la limite est atteinte. De cette manière, le paramètre de délai de coût fonctionne comme un mécanisme de limitation de la consommation de ressources. Les listes suivantes contiennent les descriptions de ces paramètres.

Paramètres affectant la limitation du démon autovacuum

- `autovacuum_vacuum_cost_limit` – Spécifie la valeur limite de coût à utiliser dans les opérations vacuum automatiques. L'augmentation de la valeur de ce paramètre permet au processus vacuum d'utiliser davantage de ressources et réduit l'événement d'attente `Timeout:VacuumDelay`.
- `autovacuum_vacuum_cost_delay` – Spécifie la valeur de délai de coût à utiliser dans les opérations vacuum automatiques. La valeur par défaut est de 2 millisecondes. La définition du paramètre de délai sur 0 désactive le mécanisme de limitation, si bien que l'événement d'attente `Timeout:VacuumDelay` n'apparaîtra pas.

Pour plus d'informations, veuillez consulter Action Vacuum automatique dans la documentation PostgreSQL.

Paramètres affectant la limitation du processus vacuum manuel

- `vacuum_cost_limit` – Le seuil à partir duquel le processus vacuum est mis en veille. Par défaut, cette limite est de 200. Ce nombre représente les estimations des coûts cumulés pour les E/S supplémentaires requises par les diverses ressources. L'augmentation de cette valeur réduit le nombre d'événements d'attente `Timeout:VacuumDelay`.
- `vacuum_cost_delay` – La durée pendant laquelle le processus vacuum est en veille quand la limite de coût du processus vacuum est atteinte. La valeur par défaut est 0, ce qui signifie que cette fonctionnalité est désactivée. Vous pouvez définir une valeur entière pour spécifier le nombre de millisecondes nécessaires à l'activation de cette fonctionnalité, mais nous vous recommandons de conserver la valeur par défaut.

Pour plus d'informations sur le paramètre `vacuum_cost_delay`, consultez Consommation des ressources dans la documentation PostgreSQL.

Pour en savoir plus sur la configuration et l'utilisation d'autovacuum avec RDS for PostgreSQL, consultez .

Causes probables de l'augmentation du nombre d'événements d'attente

Le paramètre `Timeout:VacuumDelay` est affecté par l'équilibre entre les valeurs des paramètres de limite de coût (`vacuum_cost_limit`, `autovacuum_vacuum_cost_limit`) et les paramètres de délai de coût (`vacuum_cost_delay`, `autovacuum_vacuum_cost_delay`) qui contrôlent la durée de veille du processus vacuum. L'augmentation d'une valeur de paramètre de limite de coût permet

d'utiliser davantage de ressources pour le processus vacuum avant qu'il soit mis en veille. Cela se traduit par une diminution des événements d'attente `Timeout:VacuumDelay`. L'augmentation de l'un ou l'autre des paramètres de délai entraîne une hausse de la fréquence et de la durée de l'événement d'attente `Timeout:VacuumDelay`.

Le réglage du paramètre `autovacuum_max_workers` peut également augmenter les nombres de `Timeout:VacuumDelay`. Chaque processus de travail autovacuum supplémentaire contribue au contre-mécanisme interne, de sorte que la limite peut être atteinte plus rapidement qu'avec un seul processus de travail autovacuum. Comme la limite de coût est atteinte plus rapidement, le délai de coût entre en vigueur plus fréquemment, ce qui se traduit par un plus grand nombre d'événements d'attente `Timeout:VacuumDelay`. Pour plus d'informations, consultez autovacuum_max_workers dans la documentation PostgreSQL.

Les objets volumineux, de 500 Go ou plus, déclenchent également cet événement d'attente, car le traitement des objets volumineux par le processus vacuum peut prendre un certain temps.

Actions

Si les opérations vacuum se terminent comme prévu, aucune correction n'est requise. En d'autres termes, cet événement d'attente n'indique pas nécessairement un problème. Il indique que le processus vacuum est mis en veille pendant la période spécifiée dans le paramètre de délai, afin que les ressources puissent être affectées à d'autres processus qui doivent être achevés.

Si vous souhaitez terminer plus rapidement les opérations vacuum, vous pouvez réduire les paramètres de délai. Cela raccourcit le temps de veille du processus vacuum.

Optimisation RDSpour Postgre SQL Aurora Postgre d'Amazon DevOps Guru

DevOpsGuru Proactive Insights détecte les conditions sur vos instances RDS de SQL base de données Postgre) susceptibles de provoquer des problèmes, et vous en informe avant qu'ils ne surviennent. Des informations proactives peuvent vous avertir en cas d'interruption prolongée de la connexion aux transactions. Pour plus d'informations sur la résolution des problèmes d'inactivité prolongée dans les connexions transactionnelles, voir La base de données a une connexion de longue durée à l'état Transaction inactive

DevOpsGuru peut effectuer les opérations suivantes :

- Éviter de nombreux problèmes courants liés aux bases de données en recoupant la configuration de votre base de données par rapport aux paramètres courants recommandés.
- Vous alerter face à des problèmes critiques dans votre flotte qui, s'ils ne sont pas vérifiés, peuvent entraîner des problèmes plus importants ultérieurement.
- Vous alerter face à des problèmes nouvellement découverts.

Chaque insight proactif contient une analyse de la cause du problème et des recommandations d'actions correctives.

Pour plus d'informations sur Amazon DevOps Guru pour AmazonRDS, consultezAnalyse des anomalies de performance avec Amazon DevOps Guru pour Amazon RDS.

La base de données a une connexion de longue durée à l'état Transaction inactive

Une connexion à la base de données est à l'état `idle in transaction` depuis plus de 1 800 secondes.

Rubriques

- Versions de moteur prises en charge
- Contexte
- Causes probables de ce problème
- Actions
- Métriques pertinentes

Versions de moteur prises en charge

Ces informations d'information sont prises en charge pour toutes les versions de RDSfor Postgre SQL .

Contexte

Une transaction à l'état `idle in transaction` peut contenir des verrous qui bloquent d'autres requêtes. Elle peut également empêcher `VACUUM` (y compris autovacuum) de nettoyer les lignes inactives, ce qui entraînerait le gonflement des index ou des tables ou le renvoi à la ligne des identifiants de transactions.

Causes probables de ce problème

Une transaction initiée dans une session interactive avec BEGIN START TRANSACTION ou non terminée à l'aide d'une END commande COMMITROLLBACK, ou. Cela entraîne le passage de la transaction à l'état `idle in transaction`.

Actions

Vous pouvez trouver les transactions inactives en exécutant la requête `pg_stat_activity`.

Dans votre SQL client, exécutez la requête suivante pour répertorier toutes les connexions dans leur `idle in transaction` état et les classer par durée :

```
SELECT now() - state_change as idle_in_transaction_duration, now() - xact_start as
 xact_duration,*
FROM  pg_stat_activity
WHERE state  = 'idle in transaction'
AND   xact_start is not null
ORDER BY 1 DESC;
```

Nous vous recommandons différentes actions en fonction des causes de votre insight.

Rubriques

- Arrêt de la transaction
- Interruption de la connexion
- Configuration du paramètre idle_in_transaction_session_timeout
- Vérifiez le AUTOCOMMIT statut

- Vérification de la logique de transaction dans le code de votre application

Arrêt de la transaction

Lorsque vous initiez une transaction dans une session interactive avec BEGIN ou STARTTRANSACTION, elle passe à `idle in transaction` l'état. Il reste dans cet état jusqu'à ce que vous mettiez fin à la transaction en émettant une END commande COMMITROLLBACK, ou que vous déconnectiez complètement la connexion pour annuler la transaction.

Interruption de la connexion

Mettez fin à la connexion avec une transaction inactive à l'aide de la requête suivante :

```
SELECT pg_terminate_backend(pid);
```

pid est l'ID de processus de la connexion.

Configuration du paramètre idle_in_transaction_session_timeout

Définissez le paramètre `idle_in_transaction_session_timeout` dans le nouveau groupe de paramètres. La configuration de ce paramètre présente l'avantage de ne pas nécessiter d'intervention manuelle pour mettre fin à la longue période d'inactivité de la transaction. Pour plus d'informations sur ce paramètre, consultez la SQL documentation Postgre.

Le message suivant sera affiché dans le fichier SQL journal Postgre après la fin de la connexion, lorsqu'une transaction reste dans l'état idle_in_transaction pendant une durée supérieure à la durée spécifiée.

```
FATAL: terminating connection due to idle in transaction timeout
```

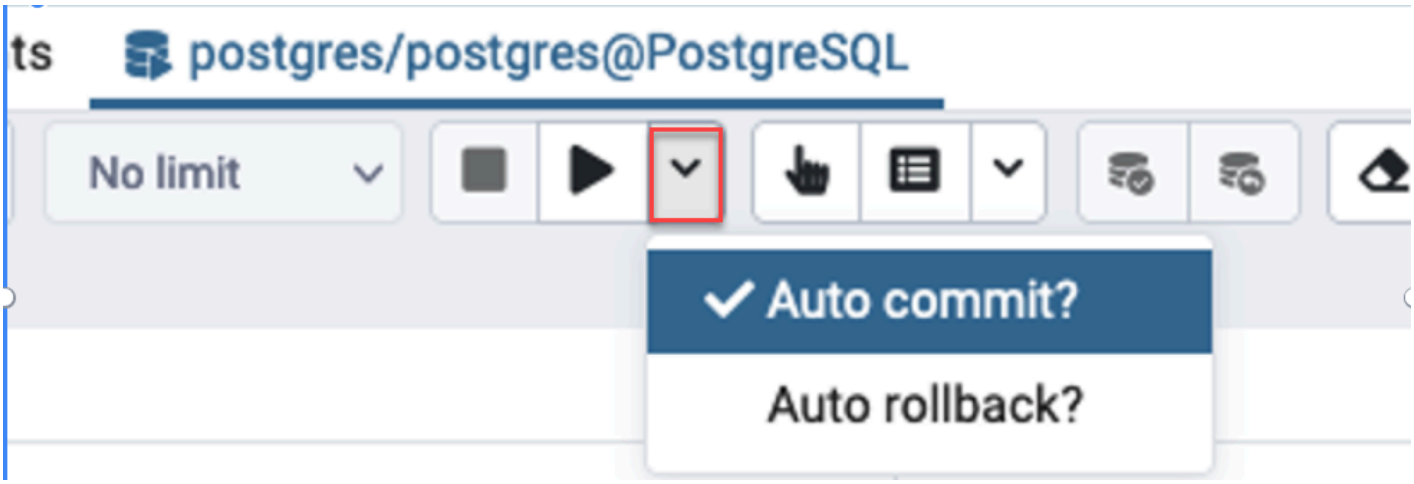
Vérifiez le AUTOCOMMIT statut

AUTOCOMMITest activé par défaut. Mais s'il est désactivé accidentellement dans le client, veillez à le réactiver.

- Dans votre client psql, exécutez la commande suivante :

```
postgres=> \set AUTOCOMMIT on
```

- Dans pgadmin, activez-le en choisissant l'AUTOCOMMITOption dans la flèche vers le bas.



Vérification de la logique de transaction dans le code de votre application

Examinez la logique de votre application pour détecter d'éventuels problèmes. Procédez comme suit :

- Vérifiez si la validation JDBC automatique est définie sur true dans votre application. Pensez également à utiliser des commandes `COMMIT` explicites dans votre code.
- Vérifiez votre logique de gestion des erreurs pour voir si elle clôture une transaction après des erreurs.
- Vérifiez si votre application met du temps à traiter les lignes renvoyées par une requête lorsque la transaction est ouverte. Si tel est le cas, pensez à coder l'application pour clôturer la transaction avant de traiter les lignes.
- Vérifiez si une transaction contient de nombreuses opérations de longue durée. Si tel est le cas, divisez une transaction individuelle en plusieurs transactions.

Métriques pertinentes

Les métriques PI suivantes sont liées à cet insight :

- idle_in_transaction_count : nombre de sessions à l'état `idle in transaction`.
- idle_in_transaction_max_time : durée de la transaction la plus longue à l'état `idle in transaction`.

Utilisation des extensions PostgreSQL avec Amazon RDS for PostgreSQL

Vous pouvez étendre les fonctionnalités de PostgreSQL en installant divers extensions et modules. Par exemple, pour utiliser des données spatiales, vous pouvez installer et utiliser l'extension PostGIS. Pour de plus amples informations, veuillez consulter Gestion des données spatiales avec l'GISextension Post. Par exemple, si vous souhaitez améliorer la saisie de données pour des tables très volumineuses, vous pouvez envisager de partitionner vos données en utilisant l'extension `pg_partman`. Pour en savoir plus, consultez la section Gestion des partitions PostgreSQL avec l'extension pg_partman.

> **Note**
>
> RDS pour PostgreSQL prend en charge les extensions de langage fiables pour PostgreSQL par le biais de l'extension, que vous pouvez ajouter à votre instance `pg_tle` de base de données. En utilisant cette extension, les développeurs peuvent créer leurs propres extensions PostgreSQL dans un environnement sûr qui simplifie les exigences d'installation et de configuration. Pour en savoir plus sur les versions de RDS pour PostgreSQL `pg_tle` prenant en charge les extensions et pour plus d'informations, consultez. Utilisation de Trusted Language Extensions pour PostgreSQL

Dans certains cas, plutôt que d'installer une extension, vous pouvez ajouter un module spécifique à la liste de `shared_preload_libraries` dans votre groupe de paramètres de base de données personnalisé de votre instance de base de données RDS for PostgreSQL. Généralement, le groupe de paramètres du cluster de bases de données par défaut charge uniquement le `pg_stat_statements`, mais plusieurs autres modules peuvent être ajoutés à la liste. Par exemple, vous pouvez ajouter une fonctionnalité de planification en ajoutant le module `pg_cron`, comme indiqué dans Planification de la maintenance avec l'extension PostgreSQL pg_cron. Autre exemple, vous pouvez enregistrer les plans d'exécution des requêtes en chargeant le module `auto_explain`. Pour en savoir plus, consultez la section Enregistrement des plans d'exécution des requêtes dans le centre de AWS connaissances.

Selon votre version de RDS for PostgreSQL, l'installation d'une extension peut exiger des autorisations `rds_superuser`, comme suit :

- Pour RDS for PostgreSQL version 12 et versions antérieures, l'installation d'extensions exige des privilèges `rds_superuser`.
- Pour RDS for PostgreSQL version 13 et versions ultérieures, les utilisateurs (rôles) disposant d'autorisations de création sur une instance de base de données donnée peuvent installer et utiliser n'importe quelle extension approuvée. Pour obtenir la liste des extensions approuvées, consultez Extensions SQL fiables de Postgre.

Vous pouvez également spécifier précisément quelles extensions peuvent être installées sur votre instance de base de données RDS for PostgreSQL, en les répertoriant dans le paramètre `rds.allowed_extensions`. Pour de plus amples informations, veuillez consulter Restreindre l'installation des extensions Postgre SQL.

Pour en savoir sur le rôle `rds_superuser`, veuillez consulter Comprendre les rôles et les autorisations PostgreSQL.

Rubriques

- Utilisation des fonctions de l'extension orafce
- Utilisation de la prise en charge RDS déléguée des extensions Amazon pour Postgre SQL
- Gestion des partitions PostgreSQL avec l'extension pg_partman
- Utilisation pgAudit pour enregistrer l'activité de la base de données
- Planification de la maintenance avec l'extension PostgreSQL pg_cron
- Utilisation de pglogical pour synchroniser les données entre les instances
- Utilisation de pgactive pour prendre en charge la réplication active-active
- Réduction du ballonnement des tables et des index avec l'extension pg_repack
- Mise à niveau et utilisation de l' PLV8extension
- Utilisation de PL/Rust pour écrire des fonctions PostgreSQL dans le langage Rust
- Gestion des données spatiales avec l'GISextension Post

Utilisation des fonctions de l'extension orafce

L'extension orafce fournit des fonctions et des opérateurs qui émulent un sous-ensemble de fonctions et de packages à partir d'une base de données Oracle. L'extension Oracle facilite le portage d'une application Oracle vers Postgre. SQL RDSpour Postgre, SQL les versions 9.6.6 et supérieures supportent cette extension. Pour plus d'informations sur Oracle, voir orace on. GitHub

> **Note**
>
> RDSfor Postgre SQL ne prend pas en charge le `utl_file` package qui fait partie de l'extension Oracle. La raison en est que les fonctions du schéma `utl_file` fournissent des opérations de lecture et d'écriture sur les fichiers de texte des systèmes d'exploitation, ce qui nécessite un accès de super-utilisateur à l'hôte sous-jacent. En tant que service géré, RDS Postgre SQL ne fournit pas d'accès à l'hôte.

Pour utiliser l'extension orafce

1. Connectez-vous à l'instance de base de données avec le nom d'utilisateur principal que vous avez utilisé pour créer l'instance de base de données.

   Si vous souhaitez activer orafce pour une base de données différente dans la même instance de base de données, utilisez la commande psql `/c dbname`. À l'aide de cette commande, vous passez de base de données primaire après avoir initié la connexion.
2. Activez l'extension orafce avec l'instruction `CREATE EXTENSION`.

   ```
   CREATE EXTENSION orafce;
   ```
3. Transférez la propriété du schéma oracle au rôle rds_superuser avec la déclaration `ALTER SCHEMA`.

   ```
   ALTER SCHEMA oracle OWNER TO rds_superuser;
   ```

   Si vous souhaitez voir la liste des propriétaires du schéma oracle, utilisez la commande psql `\dn`.

Utilisation de la prise en charge RDS déléguée des extensions Amazon pour Postgre SQL

Grâce au support RDS délégué des extensions Amazon pour PostgreSQL, vous pouvez déléguer la gestion des extensions à un utilisateur qui n'a pas besoin d'être un`rds_superuser`. Avec cette prise en charge déléguée des extensions, un nouveau rôle appelé `rds_extension` est créé et vous devez l'attribuer à un utilisateur pour gérer les autres extensions. Ce rôle peut créer, mettre à jour et supprimer des extensions.

Vous pouvez spécifier les extensions qui peuvent être installées sur votre RDS instance de base de données en les répertoriant dans le `rds.allowed_extensions` paramètre. Pour plus d'informations, consultez Utilisation des SQL extensions Postgre avec Amazon RDS pour SQL Postgre.

Vous pouvez restreindre la liste des extensions disponibles qui peuvent être gérées par l'utilisateur à l'aide du `rds.allowed_delegated_extensions` paramètre `rds_extension` role using.

La prise en charge déléguée des extensions est disponible dans les versions suivantes :

- Toutes les versions supérieures
- 16.4 et versions ultérieures 16
- 15.8 et versions supérieures 15
- Versions 14.13 et supérieures 14
- Versions 13.16 et supérieures 13
- 12.20 et versions supérieures 12

Rubriques

- Activation de la prise en charge déléguée des extensions à un utilisateur
- Configuration utilisée pour la prise en charge RDS déléguée des extensions pour Postgre SQL
- Désactiver le support pour l'extension déléguée
- Avantages de l'utilisation du support d'extension RDS délégué d'Amazon
- Limitation de la prise en charge RDS déléguée des extensions Amazon pour Postgre SQL
- Autorisations requises pour certaines extensions
- Considérations de sécurité

- Supprimer la cascade d'extensions désactivée
- Exemples d'extensions pouvant être ajoutées à l'aide de la prise en charge déléguée des extensions

Activation de la prise en charge déléguée des extensions à un utilisateur

Vous devez effectuer les opérations suivantes pour permettre à un utilisateur de prendre en charge les extensions déléguées :

1. Accorder un **`rds_extension`** rôle à un utilisateur — Connectez-vous à la base de données en tant que `rds_superuser` et exécutez la commande suivante :

   ```
   Postgres => grant rds_extension to user_name;
   ```

2. Définir la liste des extensions disponibles pour les utilisateurs délégués à gérer : vous `rds.allowed_delegated_extensions` permet de spécifier un sous-ensemble des extensions disponibles à l'aide `rds.allowed_extensions` du paramètre de cluster de base de données. Vous pouvez effectuer cette opération à l'un des niveaux suivants :

   - Dans le cluster ou le groupe de paramètres d'instance, via le AWS Management Console ouAPI. Pour de plus amples informations, veuillez consulter Groupes de paramètres pour (Amazon RDS).
   - Utilisez la commande suivante au niveau de la base de données :

     ```
     alter database database_name set rds.allowed_delegated_extensions =
      'extension_name_1,
                        extension_name_2,...extension_name_n';
     ```

   - Utilisez la commande suivante au niveau de l'utilisateur :

     ```
     alter user user_name set rds.allowed_delegated_extensions = 'extension_name_1,
                        extension_name_2,...extension_name_n';
     ```

   > **Note**
   >
   > Il n'est pas nécessaire de redémarrer la base de données après avoir modifié le paramètre `rds.allowed_delegated_extensions` dynamique.

3. Autoriser l'accès à l'utilisateur délégué aux objets créés lors du processus de création de l'extension : certaines extensions créent des objets qui nécessitent l'octroi d'autorisations supplémentaires avant que l'utilisateur ayant le `rds_extension` rôle puisse y accéder. Ils `rds_superuser` doivent accorder à l'utilisateur délégué l'accès à ces objets. L'une des options consiste à utiliser un déclencheur d'événement pour accorder automatiquement l'autorisation à l'utilisateur délégué. Pour plus d'informations, reportez-vous à l'exemple de déclencheur d'événements dansDésactiver le support pour l'extension déléguée.

Configuration utilisée pour la prise en charge RDS déléguée des extensions pour Postgre SQL

Nom de configuration	Description	Valeur par défaut	Remarques	Qui peut modifier ou accorder l'autorisation
`rds.allowed_delegated_extensions`	Ce paramètre limite les extensions qu'un rôle rds_extension peut gérer dans une base de données. Il doit s'agir d'un sous-ensemble de rds.allowed_extensions.	chaîne vide	• Par défaut, ce paramètre est une chaîne vide, ce qui signifie qu'aucune extension n'a été déléguée aux utilisateurs avec`rds_extension` . • Toute extension prise en charge peut être ajoutée si l'utilisateur est autorisé à le faire. Pour ce faire, définissez le `rds.allowed_delegated_exten`	rds_superutilisateur

Nom de configuration	Description	Valeur par défaut	Remarques	Qui peut modifier ou accorder l'autorisation
			`sions` paramètre sur une chaîne de noms d'extension séparés par des virgules. En ajoutant une liste d'extensions à ce paramètre, vous identifiez explicitement les extensions que l'utilisateur ayant le `rds_extension` rôle peut installer. • Lorsqu'il est défini sur`*`, cela signifie que toutes les extensions répertoriées `rds_allowed_extensions` sont déléguées aux utilisateurs ayant `rds_extension` un rôle. Pour en savoir plus sur la configuration	

Nom de configuration	Description	Valeur par défaut	Remarques	Qui peut modifier ou accorder l'autorisation
			de ce paramètre, consultezActivation de la prise en charge déléguée des extensions à un utilisateur.	
`rds.allowed_extensions`	Ce paramètre permet au client de limiter les extensions qui peuvent être installées dans l'RDSinstance de base de données. Pour plus d'informations, consultez Restreindre l'installation des extensions Postgre SQL	"*"	Par défaut, ce paramètre est défini sur RDS « * », ce qui signifie que toutes les extensions prises en charge par Postgre SQL et Aurora Postgre SQL sont autorisées à être créées par les utilisateurs disposant des privilèges nécessaires. Vide signifie qu'aucune extension ne peut être installée dans l'RDSinstance de base de données.	administrateur

Nom de configuration	Description	Valeur par défaut	Remarques	Qui peut modifier ou accorder l'autorisation
`rds.delegated_extension_allow_drascac`	Ce paramètre contrôle la capacité de l'utilisateur à supprimer l'extension `rds_extension` à l'aide d'une option en cascade.	off	Par défaut, la propriété `rds.delegated_extension_allow_drop_cascade` a la valeur `off`. Cela signifie que les utilisateurs ne `rds_extension` sont pas autorisés à supprimer une extension à l'aide de l'option en cascade. Pour accorder cette capacité, le `rds.delegated_extension_allow_drop_cascade` paramètre doit être défini suron.	rds_superutilisateur

Désactiver le support pour l'extension déléguée

Éteindre partiellement

Les utilisateurs délégués ne peuvent pas créer de nouvelles extensions mais peuvent toujours mettre à jour les extensions existantes.

- `rds.allowed_delegated_extensions`Rétablissez la valeur par défaut dans le groupe de paramètres du cluster de base de données.
- Utilisez la commande suivante au niveau de la base de données :

```
alter database database_name reset rds.allowed_delegated_extensions;
```

- Utilisez la commande suivante au niveau de l'utilisateur :

```
alter user user_name reset rds.allowed_delegated_extensions;
```

Éteindre complètement

La révocation du `rds_extension` rôle d'un utilisateur rétablira les autorisations standard de l'utilisateur. L'utilisateur ne peut plus créer, mettre à jour ou supprimer des extensions.

```
postgres => revoke rds_extension from user_name;
```

Exemple de déclencheur d'événement

Si vous souhaitez autoriser un utilisateur délégué `rds_extension` à utiliser des extensions qui nécessitent de définir des autorisations sur ses objets créés lors de la création de l'extension, vous pouvez personnaliser l'exemple de déclencheur d'événement ci-dessous et ajouter uniquement les extensions pour lesquelles vous souhaitez que les utilisateurs délégués aient accès à toutes les fonctionnalités. Ce déclencheur d'événement peut être créé sur template1 (le modèle par défaut). Par conséquent, toutes les bases de données créées à partir de template1 auront ce déclencheur d'événement. Lorsqu'un utilisateur délégué installe l'extension, ce déclencheur octroie automatiquement la propriété aux objets créés par l'extension.

```
CREATE OR REPLACE FUNCTION create_ext()

  RETURNS event_trigger AS $$

DECLARE

  schemaname TEXT;
  databaseowner TEXT;

  r RECORD;
```

```
BEGIN

  IF tg_tag = 'CREATE EXTENSION' and current_user != 'rds_superuser' THEN
    RAISE NOTICE 'SECURITY INVOKER';
    RAISE NOTICE 'user: %', current_user;
    FOR r IN SELECT * FROM pg_event_trigger_ddl_commands()
    LOOP
        CONTINUE WHEN r.command_tag != 'CREATE EXTENSION' OR r.object_type !=
 'extension';

        schemaname = (
            SELECT n.nspname
            FROM pg_catalog.pg_extension AS e
            INNER JOIN pg_catalog.pg_namespace AS n
            ON e.extnamespace = n.oid
            WHERE e.oid = r.objid
        );

        databaseowner = (
            SELECT pg_catalog.pg_get_userbyid(d.datdba)
            FROM pg_catalog.pg_database d
            WHERE d.datname = current_database()
        );
        RAISE NOTICE 'Record for event trigger %, objid: %,tag: %, current_user: %,
 schema: %, database_owenr: %', r.object_identity, r.objid, tg_tag, current_user,
 schemaname, databaseowner;
        IF r.object_identity = 'address_standardizer_data_us' THEN
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE %I.us_gaz TO
 %I WITH GRANT OPTION;', schemaname, databaseowner);
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE %I.us_lex TO
 %I WITH GRANT OPTION;', schemaname, databaseowner);
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE %I.us_rules
 TO %I WITH GRANT OPTION;', schemaname, databaseowner);
        ELSIF r.object_identity = 'dict_int' THEN
            EXECUTE format('ALTER TEXT SEARCH DICTIONARY %I.intdict OWNER TO %I;',
 schemaname, databaseowner);
        ELSIF r.object_identity = 'pg_partman' THEN
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE
 %I.part_config TO %I WITH GRANT OPTION;', schemaname, databaseowner);
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE
 %I.part_config_sub TO %I WITH GRANT OPTION;', schemaname, databaseowner);
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON TABLE
 %I.custom_time_partitions TO %I WITH GRANT OPTION;', schemaname, databaseowner);
```

```
        ELSIF r.object_identity = 'postgis_topology' THEN
            EXECUTE format('GRANT SELECT, UPDATE, INSERT, DELETE ON ALL TABLES IN
 SCHEMA topology TO %I WITH GRANT OPTION;', databaseowner);
            EXECUTE format('GRANT USAGE, SELECT ON ALL SEQUENCES IN SCHEMA topology TO
 %I WITH GRANT OPTION;', databaseowner);
            EXECUTE format('GRANT EXECUTE ON ALL FUNCTIONS IN SCHEMA topology TO %I
 WITH GRANT OPTION;', databaseowner);
            EXECUTE format('GRANT USAGE ON SCHEMA topology TO %I WITH GRANT OPTION;',
 databaseowner);
        END IF;
    END LOOP;
  END IF;
END;
$$ LANGUAGE plpgsql SECURITY DEFINER;

CREATE EVENT TRIGGER log_create_ext ON ddl_command_end EXECUTE PROCEDURE create_ext();
```

Avantages de l'utilisation du support d'extension RDS délégué d'Amazon

En utilisant le support RDS délégué des extensions Amazon pour PostgreSQL, vous déléguez en toute sécurité la gestion des extensions à des utilisateurs qui n'ont pas le `rds_superuser` rôle. Cette fonctionnalité offre les avantages suivants :

- Vous pouvez facilement déléguer la gestion des extensions aux utilisateurs de votre choix.
- Cela ne nécessite pas de `rds_superuser` rôle.
- Permet de prendre en charge différents ensembles d'extensions pour différentes bases de données dans le même cluster de bases de données.

Limitation de la prise en charge RDS déléguée des extensions Amazon pour Postgre SQL

- Les objets créés pendant le processus de création de l'extension peuvent nécessiter des privilèges supplémentaires pour que l'extension fonctionne correctement.
- Certaines extensions ne peuvent pas être gérées par l'utilisateur de l'extension déléguée par défaut, notamment les suivantes : `log_fdw` `pg_cronpg_tle`,`pgactive`,,`pglogical`,`postgis_raster`,`postgis_tiger_geocoder`,`postgis_to`

Autorisations requises pour certaines extensions

Pour créer, utiliser ou mettre à jour les extensions suivantes, l'utilisateur délégué doit disposer des privilèges nécessaires sur les fonctions, tables et schémas suivants.

Exter s néce nt des droits de propr ou des autor ions	Fonction	Tables	Schema	Dictionnaire de recherche de texte	Comment
addr tanda er_da s	none	us_gaz, us_lex, us_lex, i.US_Rules	none	none	none
amch	bt_index_check, bt_index_ parent_check	none	none	none	none
dict_i	none	none	none	interdit	none
pg_p n	none	partitions temporelles personnalisées, part_config, part_config_sub	none	none	none
pg_st tatem	none	none	none	none	none

Exter s nécer nt des droits de propr ou des autor ions	Fonction	Tables	Schema	Dictionnaire de recherche de texte	Comment
Publi GIS	enveloppe st_tile	spatial_ref_sys	none	none	none
postg aster	none	none	none	none	none
postg opolo	none	topologie, couche	topologie	none	l'utilisateur délégué doit être le propriéta ire de la base de données
log_f	créer_tab le_foreig n_pour_log_file	none	none	none	none
rds_t	role_pass word_encr yption_type	none	none	none	none
postg iger_ oder	none	geocode_s ettings_default, geocode_s ettings	tigre	none	none

Exter s nécer nt des droits de propr ou des autor ions	Fonction	Tables	Schema	Dictionnaire de recherche de texte	Comment
pg_fr acem	pg_freespace	none	none	none	none
pg_vi lity	pg_visibility	none	none	none	none

Considérations de sécurité

N'oubliez pas qu'un utilisateur doté d'un `rds_extension` rôle pourra gérer les extensions sur toutes les bases de données sur lesquelles il dispose du privilège de connexion. Si l'intention est d'avoir une extension déléguée de gestion des utilisateurs sur une seule base de données, il est recommandé de révoquer tous les privilèges publics sur chaque base de données, puis d'accorder explicitement le privilège de connexion pour cette base de données spécifique à l'utilisateur délégué.

Plusieurs extensions peuvent permettre à un utilisateur d'accéder aux informations de plusieurs bases de données. Assurez-vous que les utilisateurs que vous accordez `rds_extension` disposent de capacités interbases de données avant d'ajouter ces extensions à`rds.allowed_delegated_extensions`. Par exemple, `postgres_fdw` et `dblink` fournir des fonctionnalités permettant d'interroger des bases de données sur la même instance ou sur des instances distantes. `log_fdw`lit les fichiers journaux du moteur Postgres, qui concernent toutes les bases de données de l'instance et peuvent contenir des requêtes lentes ou des messages d'erreur provenant de plusieurs bases de données. `pg_cron`permet d'exécuter des tâches d'arrière-plan planifiées sur l'instance de base de données et peut configurer des tâches pour qu'elles s'exécutent dans une autre base de données.

Supprimer la cascade d'extensions désactivée

La possibilité de supprimer l'option d'extension avec cascade par un utilisateur ayant le `rds_extension` rôle est contrôlée par un `rds.delegated_extension_allow_drop_cascade` paramètre. Par défaut, la propriété `rds-delegated_extension_allow_drop_cascade` a la valeur `off`. Cela signifie que les utilisateurs dotés du `rds_extension` rôle ne sont pas autorisés à supprimer une extension à l'aide de l'option en cascade, comme indiqué dans la requête ci-dessous.

```
DROP EXTENSION CASCADE;
```

Car cela supprimera automatiquement les objets qui dépendent de l'extension, et à leur tour tous les objets qui dépendent de ces objets. Toute tentative d'utilisation de l'option en cascade provoquera une erreur.

Pour accorder cette capacité, le `rds.delegated_extension_allow_drop_cascade` paramètre doit être défini suron.

La modification du paramètre `rds.delegated_extension_allow_drop_cascade` dynamique ne nécessite pas le redémarrage de la base de données. Vous pouvez le faire à l'un des niveaux suivants :

- Dans le cluster ou le groupe de paramètres d'instance, via le AWS Management Console ouAPI.
- À l'aide de la commande suivante au niveau de la base de données :

  ```
  alter database database_name set rds.delegated_extension_allow_drop_cascade = 'on';
  ```

- À l'aide de la commande suivante au niveau de l'utilisateur :

  ```
  alter role tenant_user set rds.delegated_extension_allow_drop_cascade = 'on';
  ```

Exemples d'extensions pouvant être ajoutées à l'aide de la prise en charge déléguée des extensions

- `rds_tools`

  ```
  extension_test_db=> create extension rds_tools;
  CREATE EXTENSION
  extension_test_db=> SELECT * from rds_tools.role_password_encryption_type() where
   rolname = 'pg_read_server_files';
  ```

```
ERROR: permission denied for function role_password_encryption_type
```

- amcheck

```
extension_test_db=> CREATE TABLE amcheck_test (id int);
CREATE TABLE
extension_test_db=> INSERT INTO amcheck_test VALUES (generate_series(1,100000));
INSERT 0 100000
extension_test_db=> CREATE INDEX amcheck_test_btree_idx ON amcheck_test USING btree
 (id);
CREATE INDEX
extension_test_db=> create extension amcheck;
CREATE EXTENSION
extension_test_db=> SELECT bt_index_check('amcheck_test_btree_idx'::regclass);
ERROR: permission denied for function bt_index_check
extension_test_db=> SELECT bt_index_parent_check('amcheck_test_btree_idx'::regclass);
ERROR: permission denied for function bt_index_parent_check
```

- pg_freespacemap

```
extension_test_db=> create extension pg_freespacemap;
CREATE EXTENSION
extension_test_db=> SELECT * FROM pg_freespace('pg_authid');
ERROR: permission denied for function pg_freespace
extension_test_db=> SELECT * FROM pg_freespace('pg_authid',0);
ERROR: permission denied for function pg_freespace
```

- pg_visibility

```
extension_test_db=> create extension pg_visibility;
CREATE EXTENSION
extension_test_db=> select * from pg_visibility('pg_database'::regclass);
ERROR: permission denied for function pg_visibility
```

- postgres_fdw

```
extension_test_db=> create extension postgres_fdw;
CREATE EXTENSION
extension_test_db=> create server myserver foreign data wrapper postgres_fdw options
 (host 'foo', dbname 'foodb', port '5432');
ERROR: permission denied for foreign-data wrapper postgres_fdw
```

Gestion des partitions PostgreSQL avec l'extension pg_partman

Le partitionnement de table PostgreSQL fournit un cadre à des fins de traitement hautes performances des entrées de données et des rapports. Utilisez le partitionnement pour les bases de données nécessitant une saisie très rapide de grandes quantités de données. Le partitionnement permet également d'interroger plus rapidement les tables volumineuses. Le partitionnement permet de maintenir les données sans affecter l'instance de base de données car il nécessite moins de ressources d'I/O.

Le partitionnement vous permet de diviser les données en morceaux de taille personnalisée à des fins de traitement. Par exemple, vous pouvez choisir de partitionner des données chronologiques pour des plages telles que les plages horaires, quotidiennes, hebdomadaires, mensuelles, trimestrielles, annuelles, personnalisées ou toute autre combinaison de celles-ci. Pour un exemple de données chronologiques, si vous partitionnez la table par heure, chaque partition contiendra une heure de données. Si vous partitionnez la table chronologique par jour, chaque partition contiendra un jour de données, etc. La clé de partition contrôle la taille d'une partition.

Lorsque vous utilisez une commande SQL `INSERT` ou `UPDATE` sur une table partitionnée, le moteur de base de données achemine les données vers la partition qui convient. Les partitions de table PostgreSQL qui stockent les données sont des tables enfants de la table principale.

Lors de la lecture d'une requête de base de données, l'optimiseur PostgreSQL examine la clause `WHERE` de la requête et, si possible, dirige l'analyse de base de données vers les seules partitions pertinentes.

À partir de la version 10, PostgreSQL utilise le partitionnement déclaratif pour implémenter le partitionnement de table. Cette technique est également connue sous le nom de partitionnement PostgreSQL natif. Avant PostgreSQL version 10, il fallait utiliser des déclencheurs pour implémenter des partitions.

Le partitionnement de table PostgreSQL offre les fonctionnalités suivantes :

- Création de nouvelles partitions à tout moment.
- Plages de partitions variables.
- Partitions détachables et ré-attachables à l'aide d'instructions DDL (data definition language).

  Par exemple, les partitions détachables sont utiles pour supprimer les données historiques de la partition principale, tout en les conservant à des fins d'analyse.

- Les nouvelles partitions héritent des propriétés de la table de base de données parent, et notamment :
  - Index
  - Clés primaires devant inclure la colonne de clé de partition
  - Clés étrangères
  - Contraintes de validation
  - Références
- Création d'index pour la table complète ou chaque partition spécifique.

Vous ne pouvez pas modifier le schéma d'une partition individuelle. Vous pouvez cependant modifier la table parent (par exemple, ajouter une nouvelle colonne), qui se propage aux partitions.

Rubriques

- Présentation de l'extension PostgreSQL pg_partman
- Activation de l'extension pg_partman
- Configuration des partitions à l'aide de la fonction create_parent
- Configuration de la maintenance des partitions à l'aide de la fonction run_maintenance_proc

Présentation de l'extension PostgreSQL pg_partman

Vous pouvez utiliser l'extension `pg_partman` PostgreSQL pour automatiser la création et la maintenance des partitions de table. Pour plus d'informations générales, consultez PG Partition Manager dans la documentation `pg_partman`.

> **Note**
>
> L'extension `pg_partman` est prise en charge sur RDS PostgreSQL versions 12.5 et ultérieures.

Plutôt que de créer manuellement chaque partition, vous configurez `pg_partman` avec les paramètres suivants :

- Table à partitionner
- Type de partition

- Clé de partition
- Granularité de partition
- Options de pré-création et de gestion des partitions

Après avoir créé une table partitionnée PostgreSQL, vous l'enregistrez auprès de `pg_partman` en appelant la fonction `create_parent`. Cela crée les partitions nécessaires en fonction des paramètres passés dans la fonction.

L'extension `pg_partman` propose également la fonction `run_maintenance_proc` que vous pouvez appeler sur une base planifiée pour gérer automatiquement les partitions. Programmez cette fonction de manière à ce qu'elle s'exécute périodiquement (par exemple, toutes les heures) pour vous assurer que les partitions appropriées sont créées, si besoin. Vous pouvez également vous assurer que les partitions sont automatiquement supprimées.

Activation de l'extension pg_partman

En présence de plusieurs bases de données au sein de la même instance de base de données pour laquelle vous souhaitez gérer les partitions, activez l'extension `pg_partman` séparément pour chaque base de données. Pour activer l'extension `pg_partman` pour une base de données spécifique, créez le schéma de maintenance de partition, puis créez l'extension `pg_partman` comme suit.

```
CREATE SCHEMA partman;
CREATE EXTENSION pg_partman WITH SCHEMA partman;
```

> **Note**
>
> Pour créer l'extension `pg_partman`, assurez-vous que vous disposez des privilèges `rds_superuser`.

Si vous recevez une erreur similaire à la suivante, accordez les privilèges `rds_superuser` au compte ou utilisez votre compte de super-utilisateur.

```
ERROR: permission denied to create extension "pg_partman"
HINT: Must be superuser to create this extension.
```

Pour accorder des privilèges `rds_superuser`, connectez-vous avec votre compte de super-utilisateur et exécutez la commande suivante :

```
GRANT rds_superuser TO user-or-role;
```

Pour les exemples illustrant l'utilisation de l'extension pg_partman, nous utilisons l'exemple de table et de partition de base de données ci-dessous. Cette base de données utilise une table partitionnée basée sur un horodatage. Un schéma `data_mart` contient une table nommée `events` avec une colonne nommée `created_at`. Les paramètres suivants sont inclus dans la table `events` :

- Clés primaires `event_id` et `created_at`, qui doivent utiliser la colonne pour guider la partition.
- Contrainte de vérification `ck_valid_operation` pour appliquer des valeurs pour une colonne de table `operation`.
- Deux clés étrangères, l'une (`fk_orga_membership`)) pointant vers la table externe `organization` et l'autre (`fk_parent_event_id`) correspondant à un clé étrangère auto-référencée.
- Deux index, l'un (`idx_org_id`) correspondant à la clé étrangère et l'autre (`idx_event_type`) au type d'événement.

Les instructions DDL suivantes créent ces objets, qui sont automatiquement inclus dans chaque partition.

```
CREATE SCHEMA data_mart;
CREATE TABLE data_mart.organization ( org_id BIGSERIAL,
        org_name TEXT,
        CONSTRAINT pk_organization PRIMARY KEY (org_id)
    );

CREATE TABLE data_mart.events(
        event_id        BIGSERIAL,
        operation       CHAR(1),
        value           FLOAT(24),
        parent_event_id BIGINT,
        event_type      VARCHAR(25),
        org_id          BIGSERIAL,
        created_at      timestamp,
        CONSTRAINT pk_data_mart_event PRIMARY KEY (event_id, created_at),
        CONSTRAINT ck_valid_operation CHECK (operation = 'C' OR operation = 'D'),
        CONSTRAINT fk_orga_membership
```

```
            FOREIGN KEY(org_id)
            REFERENCES data_mart.organization (org_id),
        CONSTRAINT fk_parent_event_id
            FOREIGN KEY(parent_event_id, created_at)
            REFERENCES data_mart.events (event_id,created_at)
    ) PARTITION BY RANGE (created_at);

CREATE INDEX idx_org_id     ON  data_mart.events(org_id);
CREATE INDEX idx_event_type ON  data_mart.events(event_type);
```

Configuration des partitions à l'aide de la fonction create_parent

Après avoir activé l'extension `pg_partman`, utilisez la fonction `create_parent` pour configurer les partitions dans le schéma de maintenance des partitions. L'exemple suivant utilise l'exemple de table `events` créé dans Activation de l'extension pg_partman. Appelez la fonction `create_parent` comme suit.

```
SELECT partman.create_parent(
 p_parent_table => 'data_mart.events',
 p_control      => 'created_at',
 p_type         => 'range',
 p_interval     => '1 day',
 p_premake      => 30);
```

Les paramètres sont les suivants :

- `p_parent_table` – Table parent partitionnée. Cette table doit être présente et pleinement qualifiée, y compris le schéma.
- `p_control` – Colonne sur laquelle le partitionnement doit être basé. Le type de données doit être un entier ou une valeur basée sur le temps.
- `p_type` – Le type est `'range'` ou `'list'`.
- `p_interval` – Intervalle de temps ou plage d'entiers pour chaque partition. Les exemples de valeurs incluent `1 day` `1 hour`, et ainsi de suite.
- `p_premake` – Nombre de partitions à créer à l'avance pour prendre en charge les nouvelles insertions.

Pour une description complète de la fonction `create_parent`, consultez Fonctions de création dans la documentation `pg_partman`.

Configuration de la maintenance des partitions à l'aide de la fonction run_maintenance_proc

Vous pouvez exécuter des opérations de maintenance des partitions pour créer automatiquement de nouvelles partitions, détacher des partitions ou supprimer d'anciennes partitions. La maintenance des partitions repose sur la fonction `run_maintenance_proc` de l'extension `pg_partman`, et l'extension `pg_cron`, qui lance un planificateur interne. Le planificateur `pg_cron` exécute automatiquement les instructions SQL, fonctions et procédures définies dans vos bases de données.

L'exemple suivant utilise l'exemple de table `events` créé dans Activation de l'extension pg_partman pour définir l'exécution automatique des opérations de maintenance des partitions. Au préalable, ajoutez `pg_cron` au paramètre `shared_preload_libraries` dans le groupe de paramètres de l'instance de base de données.

```
CREATE EXTENSION pg_cron;

UPDATE partman.part_config
SET infinite_time_partitions = true,
    retention = '3 months',
    retention_keep_table=true
WHERE parent_table = 'data_mart.events';
SELECT cron.schedule('@hourly', $$CALL partman.run_maintenance_proc()$$);
```

Vous trouverez ci-dessous une step-by-step explication de l'exemple précédent :

1. Modifiez le groupe de paramètres associé à votre instance de base de données et ajoutez `pg_cron` à la valeur du paramètre `shared_preload_libraries`. Pour prendre effet, cette modification implique un redémarrage de l'instance de base de données. Pour plus d'informations, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).
2. Exécutez la commande `CREATE EXTENSION pg_cron;` à l'aide d'un compte disposant des autorisations `rds_superuser`. Cela permet d'activer l'extension `pg_cron`. Pour plus d'informations, consultez Planification de la maintenance avec l'extension PostgreSQL pg_cron.
3. Exécutez la commande `UPDATE partman.part_config` pour ajuster les paramètres `pg_partman` de la table `data_mart.events`.
4. Exécutez la commande `SET . . .` pour configurer la table `data_mart.events` avec les clauses suivantes :

a. `infinite_time_partitions = true,` – Configure la table pour créer automatiquement de nouvelles partitions sans aucune limite.

b. `retention = '3 months',` – Configure la table pour présenter une rétention maximale de trois mois.

c. `retention_keep_table=true` – configure la table de telle sorte qu'au terme de la période de rétention, la table ne soit pas supprimée automatiquement. Les partitions antérieures à la période de rétention sont uniquement détachées de la table parent.

5. Exécutez la commande `SELECT cron.schedule . . .` pour faire un appel de fonction `pg_cron`. Cet appel définit la fréquence à laquelle le planificateur exécute la procédure de maintenance `pg_partman`, `partman.run_maintenance_proc`. Pour cet exemple, la procédure s'exécute toutes les heures.

Pour une description complète de la fonction `run_maintenance_proc`, consultez Fonctions de maintenance dans la documentation `pg_partman`.

Utilisation pgAudit pour enregistrer l'activité de la base de données

Les institutions financières, les agences gouvernementales et de nombreux secteurs doivent tenir des journaux d'audit pour se conformer aux exigences réglementaires. En utilisant l'extension Postgre SQL Audit (pgAudit) avec votre instance de RDS pour Postgre SQL, vous pouvez capturer les enregistrements détaillés généralement nécessaires aux auditeurs ou pour répondre aux exigences réglementaires. Par exemple, vous pouvez configurer l' pgAudit extension pour suivre les modifications apportées à des bases de données et à des tables spécifiques, pour enregistrer l'utilisateur qui a effectué les modifications et pour de nombreux autres détails.

L' pgAudit extension s'appuie sur les fonctionnalités de l'infrastructure de SQL journalisation native de Postgre en étendant les messages de journal de manière plus détaillée. En d'autres termes, vous utilisez la même approche pour consulter votre journal d'audit que pour consulter les messages du journal. Pour plus d'informations sur la SQL journalisation Postgre, consultezFichiers journaux de base de données RDS for PostgreSQL.

L' pgAudit extension supprime les données sensibles telles que les mots de passe en texte clair des journaux. Si votre pour SQL instance de base de données Postgre est configuré pour enregistrer les instructions du langage de manipulation des données (DML) comme indiqué dans la sectionActiver la journalisation des requêtes pour votre instance de base de données RDS for PostgreSQL., vous pouvez éviter le problème du mot de passe en texte clair en utilisant l'extension SQL Postgre Audit.

Vous pouvez configurer l'audit sur vos instances de base de données avec une grande précision. Vous pouvez auditer toutes les bases de données et tous les utilisateurs. Vous pouvez également choisir de n'auditer que certaines bases de données, certains utilisateurs et d'autres objets. Vous pouvez également exclure explicitement certains utilisateurs et certaines bases de données de l'audit. Pour de plus amples informations, veuillez consulter Exclusion d'utilisateurs ou de bases de données de la journalisation d'audit.

Compte tenu de la quantité de détails qui peuvent être capturés, nous vous recommandons, si vous l'utilisezpgAudit, de surveiller votre consommation de stockage.

L' pgAudit extension est prise en charge sur toutes les RDSpour les SQL versions Postgre. Pour obtenir la liste des pgAudit versions prises en charge par les versions disponibles RDS pour Postgre, consultez la section SQL Versions d'extension pour Amazon RDS pour Postgre SQL dans les notes de mise à jour d'Amazon RDS pour Postgre SQL.

Rubriques

- Configuration de l' pgAudit extension

- Audit d'objets de base de données
- Exclusion d'utilisateurs ou de bases de données de la journalisation d'audit
- Référence pour l'pgAuditextension

Configuration de l' pgAudit extension

Pour configurer l' pgAudit extension sur votre , vous devez d'abord l'ajouter pgAudit aux bibliothèques partagées sur le groupe de paramètres de base de données personnalisé RDS pour votre instance de base de données Postgre SQL. Pour plus d'informations sur la création d'un groupe de paramètres de cluster de bases de données, consultez Groupes de paramètres pour (Amazon RDS). Ensuite, vous installez l' pgAudit extension. Enfin, vous spécifiez les bases de données et les objets que vous souhaitez auditer. Les procédures de cette section vous guident. Vous pouvez utiliser le plugin AWS Management Console ou le AWS CLI.

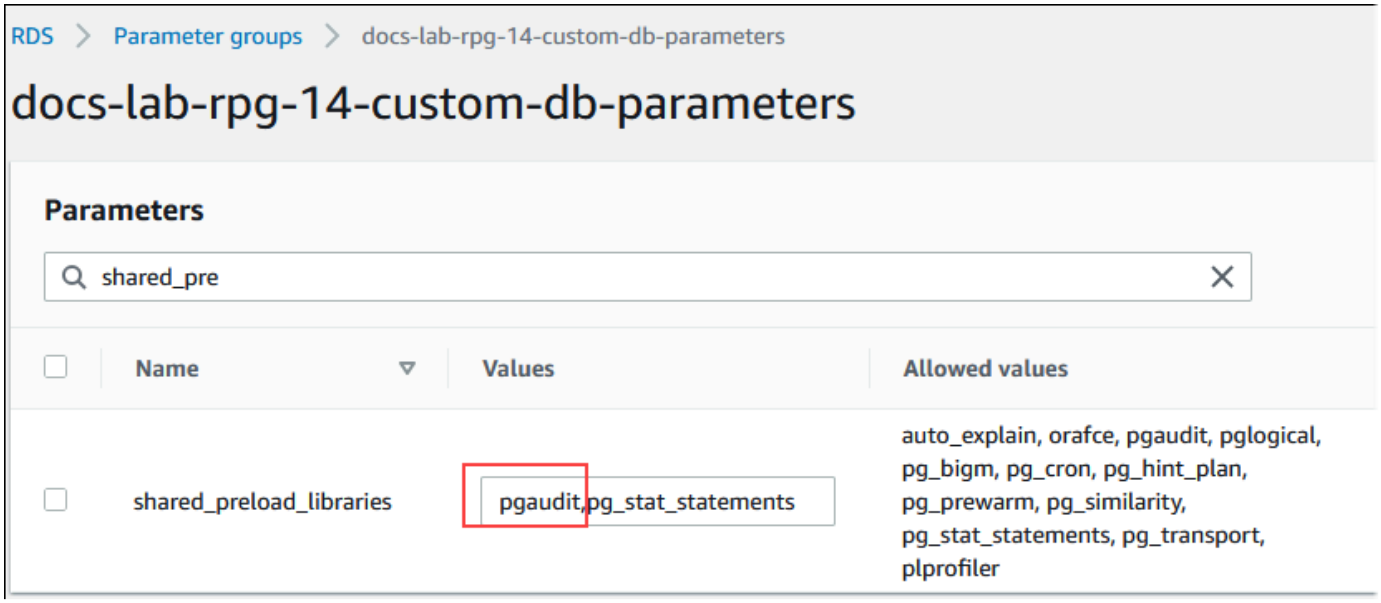
Vous devez disposer d'autorisations en tant que rôle `rds_superuser` pour effectuer toutes ces tâches.

Les étapes suivantes supposent que votre pour SQL instance de base de données Postgre est associé à un groupe de paramètres de base de données de de base de données personnalisé.

Console

Pour configurer l' pgAudit extension

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez l' de SQL base de données Postgre.
3. Ouvrez l'onglet Configuration de votre . RDSpour une SQL instance de base de données Postgre. Parmi les détails de l'instance, trouvez le lien Groupe de paramètres.
4. Cliquez sur le lien pour ouvrir les paramètres personnalisés associés à votre . RDSpour une SQL instance de base de données Postgre.
5. Dans le champ de recherche Parameters (Paramètres), tapez `shared_pre` pour trouver le paramètre `shared_preload_libraries`.
6. Choisissez Edit parameters (Modifier les paramètres) pour accéder aux valeurs des propriétés.
7. Ajoutez `pgaudit` à la liste dans le champ Values (Valeurs). Utilisez une virgule pour séparer les éléments de la liste de valeurs.



8. Redémarrez l' pour l'SQLinstance de base de données Postgre afin que votre modification du `shared_preload_libraries` paramètre prenne effet.
9. Lorsque l'instance est disponible, vérifiez qu'pgAuditelle a été initialisée. Utilisez-le `psql` pour vous connecter à l' pour l'SQLinstance de base de données Postgre, puis exécutez la commande suivante.

```
SHOW shared_preload_libraries;
shared_preload_libraries
--------------------------
rdsutils,pgaudit
(1 row)
```

10. Une pgAudit fois initialisé, vous pouvez désormais créer l'extension. Vous devez créer l'extension après avoir initialisé la bibliothèque, car l'`pgaudit`extension installe des déclencheurs d'événements pour auditer les instructions du langage de définition des données (DDL).

```
CREATE EXTENSION pgaudit;
```

11. Fermez la session `psql`.

```
labdb=> \q
```

12. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.

13. Trouvez le paramètre `pgaudit.log` dans la liste et définissez la valeur appropriée pour votre cas d'utilisation. Par exemple, la définition du paramètre `pgaudit.log` en `write` comme indiqué dans l'image suivante permet de capturer des insertions, des mises à jour, des suppressions et d'autres types de modifications dans le journal.



Vous pouvez également choisir l'une des valeurs suivantes pour le paramètre `pgaudit.log`.

- none – La valeur par défaut. Aucune modification de base de données n'est journalisée.
- all – Journalise tout (lecture, écriture, fonction, rôle, ddl, divers).
- ddl — Enregistre toutes les instructions du langage de définition de données (DDL) qui ne sont pas incluses dans la `ROLE` classe.
- function – Journalise les appels de fonction et les blocs `DO`.
- misc – Journalise diverses commandes, telles que `DISCARD`, `FETCH`, `CHECKPOINT`, `VACUUM` et `SET`.
- read – Journalise `SELECT` et `COPY` lorsque la source est une relation (comme une table) ou une requête.
- role – Journalise les instructions relatives aux rôles et privilèges, telles que `GRANT`, `REVOKE`, `CREATE ROLE`, `ALTER ROLE` et `DROP ROLE`.
- write – Journalise `INSERT`, `UPDATE`, `DELETE`, `TRUNCATE` et `COPY` lorsque la destination est une relation (table).

14. Sélectionnez Enregistrer les modifications.
15. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
16. Choisissez l'instance d' de SQL base de données Postgre dans la liste des bases de données.

AWS CLI

Pour configurer pgAudit

Pour effectuer une configuration pgAudit à l'aide du AWS CLI, vous appelez l'modify-db-parameter-groupopération pour modifier les paramètres du journal d'audit dans votre groupe de paramètres personnalisé, comme indiqué dans la procédure suivante.

1. Utilisez ce qui suit AWS CLI commande à ajouter `pgaudit` au `shared_preload_libraries` paramètre.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=shared_preload_libraries,ParameterValue=pgaudit,ApplyMethod=pending-
reboot" \
   --region aws-region
```

2. Utilisez ce qui suit AWS CLI commande pour redémarrer l' pour SQL instance de base de données Postgre afin que la bibliothèque pgaudit soit initialisée.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

3. Lorsque l'instance est disponible, vous pouvez vérifier que `pgaudit` a été initialisé. Utilisez-le `psql` pour vous connecter à l' pour l'SQLinstance de base de données Postgre, puis exécutez la commande suivante.

```
SHOW shared_preload_libraries;
shared_preload_libraries
--------------------------
rdsutils,pgaudit
(1 row)
```

   Une pgAudit fois initialisé, vous pouvez désormais créer l'extension.

```
CREATE EXTENSION pgaudit;
```

4. Fermez la `psql` session afin de pouvoir utiliser le AWS CLI.

```
labdb=> \q
```

5. Utilisez ce qui suit AWS CLI commande pour spécifier les classes d'instructions qui doivent être enregistrées par journalisation d'audit de session. L'exemple définit le paramètre `pgaudit.log` sur `write`, qui capture les insertions, les mises à jour et les suppressions dans le journal.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=pgaudit.log,ParameterValue=write,ApplyMethod=pending-reboot" \
   --region aws-region
```

Vous pouvez également choisir l'une des valeurs suivantes pour le paramètre `pgaudit.log`.

- none – La valeur par défaut. Aucune modification de base de données n'est journalisée.
- all – Journalise tout (lecture, écriture, fonction, rôle, ddl, divers).
- ddl — Enregistre toutes les instructions du langage de définition de données (DDL) qui ne sont pas incluses dans la `ROLE` classe.
- function – Journalise les appels de fonction et les blocs `DO`.
- misc – Journalise diverses commandes, telles que `DISCARD`, `FETCH`, `CHECKPOINT`, `VACUUM` et `SET`.
- read – Journalise `SELECT` et `COPY` lorsque la source est une relation (comme une table) ou une requête.
- role – Journalise les instructions relatives aux rôles et privilèges, telles que `GRANT`, `REVOKE`, `CREATE ROLE`, `ALTER ROLE` et `DROP ROLE`.
- write – Journalise `INSERT`, `UPDATE`, `DELETE`, `TRUNCATE` et `COPY` lorsque la destination est une relation (table).

Redémarrez l' pour SQL instance de base de données Postgre en procédant comme suit AWS CLI commande.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

Audit d'objets de base de données

Une fois pgAudit configuré sur votre pour l'SQLinstance de base de données Postgre et configuré selon vos besoins, des informations plus détaillées sont capturées dans le journal SQL Postgre. Par exemple, alors que la configuration de SQL journalisation Postgre par défaut identifie la date et l'heure auxquelles une modification a été apportée dans une table de base de données, avec l'pgAuditextension, l'entrée du journal peut inclure le schéma, l'utilisateur qui a effectué la modification et d'autres détails en fonction de la façon dont les paramètres de l'extension sont configurés. Vous pouvez configurer l'audit pour suivre les modifications de différentes manières.

- Pour chaque session, par utilisateur. Au niveau de la session, vous pouvez capturer le texte de commande complet.
- Pour chaque objet, par utilisateur et par base de données.

La fonctionnalité d'audit des objets est activée lorsque vous créez le rôle `rds_pgaudit` sur votre système, puis que vous ajoutez ce rôle au paramètre `pgaudit.role` dans votre groupe de paramètres personnalisé. Par défaut, le paramètre `pgaudit.role` n'est pas défini et la seule valeur autorisée est `rds_pgaudit`. Les étapes suivantes supposent que `pgaudit` a été initialisé et que vous avez créé l'extension `pgaudit` en suivant la procédure décrite dans Configuration de l' pgAudit extension.

```
2022-10-07 23:36:51 UTC:52.95.4.10(14410):postgres@labdb:[1374]:LOG: statement: SELECT feedback, s.sentiment,s.confidence
FROM support,aws_comprehend.detect_sentiment(feedback, 'en') s
ORDER BY s.confidence DESC;
2022-10-07 23:36:51 UTC:52.95.4.10(14410):postgres@labdb:[1374]:LOG: AUDIT: SESSION,2,1,READ,SELECT,TABLE,public.support,"SELECT
feedback, s.sentiment,s.confidence
FROM support,aws_comprehend.detect_sentiment(feedback, 'en') s
ORDER BY s.confidence DESC;",<none>
2022-10-07 23:36:51 UTC:52.95.4.10(14410):postgres@labdb:[1374]:LOG: QUERY STATISTICS
2022-10-07 23:36:51 UTC:52.95.4.10(14410):postgres@labdb:[1374]:DETAIL: ! system usage stats:
! 0.009494 s user, 0.007442 s system, 0.141985 s elapsed
! [0.022327 s user, 0.007442 s system total]
```

Comme le montre cet exemple, la ligne « LOG AUDIT : : SESSION » fournit des informations sur la table et son schéma, entre autres détails.

Configurer l'audit d'objets

1. Utilisez-le `psql` pour vous connecter à l'. RDSpour une SQL instance de base de données Postgre.

   ```
   psql --host=your-instance-name.aws-region.rds.amazonaws.com --port=5432 --
   username=postgrespostgres --password --dbname=labdb
   ```

2. Créez un rôle de base de données appelé `rds_pgaudit` à l'aide de la commande suivante.

```
labdb=> CREATE ROLE rds_pgaudit;
CREATE ROLE
labdb=>
```

3. Fermez la session `psql`.

```
labdb=> \q
```

Au cours des prochaines étapes, utilisez AWS CLI pour modifier les paramètres du journal d'audit dans votre groupe de paramètres personnalisé.

4. Utilisez ce qui suit AWS CLI commande pour définir le `pgaudit.role` paramètre sur`rds_pgaudit`. Par défaut, ce paramètre est vide et `rds_pgaudit` est la seule valeur autorisée.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=pgaudit.role,ParameterValue=rds_pgaudit,ApplyMethod=pending-reboot"
 \
   --region aws-region
```

5. Utilisez ce qui suit AWS CLI commande pour redémarrer l' pour SQL instance de base de données Postgre afin que les modifications apportées aux paramètres prennent effet.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

6. Exécutez la commande suivante pour confirmer que `pgaudit.role` est défini sur `rds_pgaudit`.

```
SHOW pgaudit.role;
pgaudit.role
-----------------
rds_pgaudit
```

Pour tester la pgAudit journalisation, vous pouvez exécuter plusieurs exemples de commandes que vous souhaitez auditer. Par exemple, vous pouvez exécuter les commandes suivantes.

```
CREATE TABLE t1 (id int);
GRANT SELECT ON t1 TO rds_pgaudit;
SELECT * FROM t1;
id
----
(0 rows)
```

Les journaux de base de données doivent contenir une entrée similaire à ce qui suit.

```
...
2017-06-12 19:09:49 UTC:...:rds_test@postgres:[11701]:LOG: AUDIT:
OBJECT,1,1,READ,SELECT,TABLE,public.t1,select * from t1;
...
```

Pour obtenir des informations sur l'affichage des journaux, veuillez consulter Surveillance des fichiers journaux RDSAmazon.

Pour en savoir plus sur l' pgAudit extension, reportez-vous à pgAuditla section suivante GitHub.

Exclusion d'utilisateurs ou de bases de données de la journalisation d'audit

Comme indiqué dans la sectionFichiers journaux de base de données RDS for PostgreSQL, les SQL journaux Postgre consomment de l'espace de stockage. L'utilisation de l' pgAudit extension augmente le volume de données collectées dans vos journaux à des degrés divers, en fonction des modifications que vous suivez. Il se peut que vous n'ayez pas besoin d'auditer chaque utilisateur ou base de données de votre . RDSpour une SQL instance de base de données Postgre.

Pour minimiser les impacts sur votre stockage et éviter de capturer inutilement des enregistrements d'audit, vous pouvez exclure les utilisateurs et les bases de données de l'audit. Vous pouvez également modifier la journalisation au cours d'une session donnée. Les exemples suivants montrent comment procéder.

> **Note**
>
> Les paramètres au niveau de la session ont priorité sur les paramètres du groupe de paramètres du de base de données RDS pour Postgre SQL. Si vous ne souhaitez pas

que les utilisateurs de base de données contournent vos paramètres de configuration de journalisation des audits, veillez à modifier leurs autorisations.

Supposons que votre instance de RDS pour Postgre soit configuré pour auditer le même niveau d'activité pour tous les utilisateurs et SQL bases de données. Vous pouvez ensuite décider de ne pas auditer l'utilisateur `myuser`. Vous pouvez désactiver l'audit pour à l'`myuser`aide de la SQL commande suivante.

```
ALTER USER myuser SET pgaudit.log TO 'NONE';
```

Vous pouvez ensuite utiliser la requête suivante pour vérifier la colonne `user_specific_settings` pour `pgaudit.log` afin de confirmer que le paramètre est défini sur `NONE`.

```
SELECT
    usename AS user_name,
    useconfig AS user_specific_settings
FROM
    pg_user
WHERE
    usename = 'myuser';
```

Vous devez voir la sortie suivante.

```
 user_name | user_specific_settings
-----------+-----------------------
 myuser    | {pgaudit.log=NONE}
(1 row)
```

Vous pouvez désactiver la journalisation pour un utilisateur donné au cours de sa session avec la base de données à l'aide de la commande suivante.

```
ALTER USER myuser IN DATABASE mydatabase SET pgaudit.log TO 'none';
```

Utilisez la requête suivante pour vérifier la colonne des paramètres du fichier pgaudit.log pour une combinaison utilisateur et base de données spécifique.

```
SELECT
    usename AS "user_name",
```

```
    datname AS "database_name",
    pg_catalog.array_to_string(setconfig, E'\n') AS "settings"
FROM
    pg_catalog.pg_db_role_setting s
    LEFT JOIN pg_catalog.pg_database d ON d.oid = setdatabase
    LEFT JOIN pg_catalog.pg_user r ON r.usesysid = setrole
WHERE
    usename = 'myuser'
    AND datname = 'mydatabase'
ORDER BY
    1,
    2;
```

Vous voyez des résultats similaires à ce qui suit.

```
  user_name | database_name |     settings
-----------+---------------+-----------------
 myuser    | mydatabase    | pgaudit.log=none
(1 row)
```

Après avoir désactivé l'audit pour `myuser`, vous décidez de ne pas suivre les modifications apportées à `mydatabase`. Vous pouvez désactiver l'audit pour cette base de données spécifique à l'aide de la commande suivante.

```
ALTER DATABASE mydatabase SET pgaudit.log to 'NONE';
```

Utilisez ensuite la requête suivante pour vérifier la colonne database_specific_settings afin de confirmer que pgaudit.log est défini sur. NONE

```
SELECT
a.datname AS database_name,
b.setconfig AS database_specific_settings
FROM
pg_database a
FULL JOIN pg_db_role_setting b ON a.oid = b.setdatabase
WHERE
a.datname = 'mydatabase';
```

Vous devez voir la sortie suivante.

```
 database_name | database_specific_settings
```

```
---------------+---------------------------
 mydatabase    | {pgaudit.log=NONE}
(1 row)
```

Pour rétablir les paramètres par défaut pour myuser, utilisez la commande suivante :

```
ALTER USER myuser RESET pgaudit.log;
```

Pour rétablir les paramètres par défaut pour une base de données, utilisez la commande suivante.

```
ALTER DATABASE mydatabase RESET pgaudit.log;
```

Pour rétablir les paramètres par défaut pour l'utilisateur et la base de données, utilisez la commande suivante.

```
ALTER USER myuser IN DATABASE mydatabase RESET pgaudit.log;
```

Vous pouvez également capturer des événements spécifiques dans le journal en définissant `pgaudit.log` pour l'une des autres valeurs autorisées pour le paramètre `pgaudit.log`. Pour de plus amples informations, veuillez consulter Liste des paramètres autorisés pour le paramètre `pgaudit.log`.

```
ALTER USER myuser SET pgaudit.log TO 'read';
ALTER DATABASE mydatabase SET pgaudit.log TO 'function';
ALTER USER myuser IN DATABASE mydatabase SET pgaudit.log TO 'read,function'
```

Référence pour l'pgAuditextension

Vous pouvez spécifier le niveau de détail que vous souhaitez pour votre journal d'audit en modifiant un ou plusieurs des paramètres répertoriés dans cette section.

pgAudit Comportement de contrôle

Vous pouvez contrôler la journalisation d'audit en modifiant un ou plusieurs des paramètres répertoriés dans la table suivante.

Paramètre	Description
`pgaudit.log`	Spécifie quelles classes d'instructions seront journalisées par la journalisation de l'audit de session. Les valeurs autorisées

Paramètre	Description
	incluent ddl, function, misc, read, role, write, none, all. Pour de plus amples informations, veuillez consulter Liste des paramètres autorisés pour le paramètre `pgaudit.log` .
`pgaudit.log_catalog`	Lorsque cette option est activée (définie sur 1), cela ajoute des instructions à la piste d'audit si toutes les relations d'une instruction se trouvent dans pg_catalog.
`pgaudit.log_level`	Spécifie le niveau de journal qui sera utilisé pour les entrées de journal. Valeurs autorisées : debug5, debug4, debug3, debug2, debug1, info, notice, warning, log
`pgaudit.log_parameter`	Lorsque cette option est activée (définie sur 1), les paramètres transmis avec l'instruction sont capturés dans le journal d'audit.
`pgaudit.log_relation`	Lorsqu'il est activé (défini sur 1), le journal d'audit de la session crée une entrée de journal distincte pour chaque relation (TABLEVIEW,, etc.) référencée dans une DML instruction SELECT or.
`pgaudit.log_statement_once`	Spécifie si la journalisation inclura le texte de l'instruction et les paramètres avec la première entrée de journal pour une combinaison instruction/sous-instruction ou avec chaque entrée.
`pgaudit.role`	Spécifie le rôle principal à utiliser pour la journalisation de l'audit des objets. La seule entrée autorisée est `rds_pgaudit` .

Liste des paramètres autorisés pour le paramètre **`pgaudit.log`**

Valeur	Description
none	Il s’agit de l’option par défaut. Aucune modification de base de données n'est journalisée.
Tout	Journalise tout (lecture, écriture, fonction, rôle, ddl, divers).

Valeur	Description
ddl	Enregistre toutes les instructions du langage de définition de données (DDL) qui ne sont pas incluses dans la `ROLE` classe.
fonction	Journalise les appels de fonction et les blocs `DO`.
Misc	Journalise diverses commandes, telles que `DISCARD`, `FETCH`, `CHECKPOINT` , `VACUUM` et `SET`.
lire	Journalise `SELECT` et `COPY` lorsque la source est une relation (comme une table) ou une requête.
rôle	Journalise les instructions relatives aux rôles et privilèges, telles que `GRANT`, `REVOKE`, `CREATE ROLE`, `ALTER ROLE` et `DROP ROLE`.
write	Journalise `INSERT`, `UPDATE`, `DELETE`, `TRUNCATE` et `COPY` lorsque la destination est une relation (table).

Pour journaliser plusieurs types d'événements avec l'audit de session, utilisez une liste séparée par des virgules. Pour journaliser tous les types d'événements, définissez `pgaudit.log` à la valeur `ALL`. Redémarrez l'instance de base de données pour appliquer les modifications.

Avec les audits d'objet, vous pouvez affiner la journalisation d'audit pour que celle-ci fonctionne avec des relations spécifiques. Par exemple, vous pouvez spécifier que vous souhaitez une journalisation d'audit pour les opérations `READ` sur une ou plusieurs tables.

Planification de la maintenance avec l'extension PostgreSQL pg_cron

Vous pouvez utiliser l'extension `pg_cron` PostgreSQL pour planifier des commandes de maintenance dans une base de données PostgreSQL. Pour plus d'informations concernant l'extension, consultez la section What is pg_cron? (Qu'est-ce que pg_cron ?) dans la documentation pg_cron.

L'extension `pg_cron` est prise en charge sur RDS for PostgreSQL versions 12.5 et ultérieures du moteur.

Pour en savoir plus sur l'utilisation de `pg_cron`, consultez la section Schedule jobs with pg_cron on your RDS for PostgreSQL or your Aurora PostgreSQL-Compatible Edition databases (Planifier des tâches avec pg_cron sur votre RDS pour PostgreSQL ou sur vos bases de données Aurora Édition compatible avec PostgreSQL).

Rubriques

- Configuration de l'extension pg_cron
- Octroi d'autorisations utilisateurs de la base de données pour l'utilisation de pg_cron
- Planification des tâches pg_cron
- Référence pour l'extension pg_cron

Configuration de l'extension pg_cron

Configurez l'extension `pg_cron` comme suit :

1. Modifiez le groupe de paramètres personnalisé employé avec votre instance de base de données PostgreSQL en ajoutant `pg_cron` à la valeur du paramètre `shared_preload_libraries`.
   - Si votre instance de base de données RDS for PostgreSQL utilise le paramètre `rds.allowed_extensions` pour lister explicitement les extensions qui peuvent être installées, vous devez ajouter l'extension `pg_cron` à la liste. Seules certaines versions de RDS for PostgreSQL prennent en charge le paramètre `rds.allowed_extensions`. Par défaut, toutes les extensions disponibles sont autorisées. Pour de plus amples informations, veuillez consulter Restreindre l'installation des extensions Postgre SQL.

   Redémarrez l'instance de la base de données PostgreSQL pour que les modifications du groupe de paramètres prennent effet. Pour en savoir plus sur l'utilisation des groupes de paramètres,

consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

2. Après le redémarrage de l'instance de base de données PostgreSQL, exécutez la commande suivante à l'aide d'un compte disposant d'autorisations `rds_superuser`. Par exemple, si vous avez utilisé les paramètres par défaut lors de la création de votre instance de base de données RDS for PostgreSQL, connectez-vous en tant qu'utilisateur `postgres` et créez l'extension.

   ```
   CREATE EXTENSION pg_cron;
   ```

   Le planificateur `pg_cron` est défini dans la base de données PostgreSQL par défaut nommée `postgres`. Les objets `pg_cron` sont créés dans cette base de données `postgres` et toutes les actions de planification s'y exécutent.

3. Vous pouvez utiliser les paramètres par défaut ou planifier des tâches à exécuter dans d'autres bases de données de votre instance de base de données PostgreSQL. Pour planifier des tâches dans d'autres bases de données de votre instance de base de données PostgreSQL, veuillez consulter l'exemple disponible dans Planification d'une tâche cron pour une base de données autre que la base de données par défaut.

Octroi d'autorisations utilisateurs de la base de données pour l'utilisation de pg_cron

L'installation de l'extension `pg_cron` requiert les privilèges `rds_superuser`. Toutefois, les autorisations d'utiliser le `pg_cron` peuvent être accordées (par un membre du groupe/rôle `rds_superuser`) à d'autres utilisateurs de la base de données, afin qu'ils puissent planifier leurs propres tâches. Nous vous recommandons de n'accorder des autorisations au schéma `cron` qu'en cas de besoin, si cela améliore les opérations dans votre environnement de production.

Pour accorder à un utilisateur de base de données des autorisations dans le schéma `cron`, exécutez la commande suivante :

```
postgres=> GRANT USAGE ON SCHEMA cron TO db-user;
```

Cela donne *`db-user`* l'autorisation d'accéder au `cron` schéma pour planifier des tâches cron pour les objets auxquels ils sont autorisés à accéder. Si l'utilisateur de la base de données ne dispose pas des autorisations nécessaires, la tâche échoue après avoir validé le message d'erreur dans le fichier `postgresql.log`, comme indiqué ci-dessous :

```
2020-12-08 16:41:00 UTC::@:[30647]:ERROR: permission denied for table table-name
```

```
2020-12-08 16:41:00 UTC::@:[27071]:LOG: background worker "pg_cron" (PID 30647) exited
 with exit code 1
```

En d'autres termes, assurez-vous que les utilisateurs de base de données dotés d'autorisations sur le `cron` schéma disposent également d'autorisations sur les objets (tables, schémas, etc.) qu'ils prévoient de planifier.

Les détails de la tâche cron et de son succès ou de son échec sont également enregistrés dans le `cron.job_run_details` tableau. Pour de plus amples informations, veuillez consulter Tableaux pour planifier les tâches et capturer leur statut .

Planification des tâches pg_cron

Les sections suivantes montrent comment vous pouvez planifier diverses tâches de gestion à l'aide de tâches `pg_cron`.

> **Note**
>
> Lorsque vous créez des tâches `pg_cron`, vérifiez que le paramètre `max_worker_processes` est supérieur au nombre de `cron.max_running_jobs`. Une tâche `pg_cron` échoue si elle manque de processus de travail en arrière-plan. Le nombre de tâches `pg_cron` par défaut est de 5. Pour de plus amples informations, veuillez consulter Paramètres de gestion de l'extension pg_cron.

Rubriques

- Vidage d'une table
- Purge de la table Historique pg_cron
- Journalisation des erreurs dans le fichier postgresql.log uniquement
- Planification d'une tâche cron pour une base de données autre que la base de données par défaut

Vidage d'une table

Autovacuum gère la maintenance dans la plupart des cas. Toutefois, vous pouvez vider une table spécifique quand bon vous semble.

Voir aussi, .

L'exemple suivant montre comment utiliser la fonction `cron.schedule` pour configurer une tâche de manière à ce qu'elle utilise `VACUUM FREEZE` sur une table spécifique tous les jours à 22:00 (GMT).

```
SELECT cron.schedule('manual vacuum', '0 22 * * *', 'VACUUM FREEZE pgbench_accounts');
 schedule
----------
1
(1 row)
```

Une fois l'exemple précédent exécuté, vous pouvez vérifier l'historique dans la table `cron.job_run_details` comme suit.

```
postgres=> SELECT * FROM cron.job_run_details;
jobid  | runid | job_pid | database | username | command                        |
 status    | return_message | start_time                    | end_time
-------+-------+---------+----------+----------+--------------------------------
+-----------+----------------+--------------------------------
+--------------------------------
 1     | 1     | 3395    | postgres | adminuser| vacuum freeze pgbench_accounts
 | succeeded | VACUUM         | 2020-12-04 21:10:00.050386+00 | 2020-12-04
 21:10:00.072028+00
(1 row)
```

Vous trouverez ci-dessous une requête de la `cron.job_run_details` table pour voir les tâches ayant échoué.

```
postgres=> SELECT * FROM cron.job_run_details WHERE status = 'failed';
jobid | runid | job_pid | database | username | command                        | status
 | return_message                                  | start_time                    |
 end_time
------+-------+---------+----------+----------+--------------------------------+--------
+-------------------------------------------------+--------------------------------
+--------------------------------
 5    | 4     | 30339   | postgres | adminuser| vacuum freeze pgbench_account | failed
 | ERROR: relation "pgbench_account" does not exist | 2020-12-04 21:48:00.015145+00 |
 2020-12-04 21:48:00.029567+00
(1 row)
```

Pour de plus amples informations, veuillez consulter Tableaux pour planifier les tâches et capturer leur statut .

Purge de la table Historique pg_cron

La table `cron.job_run_details` contient un historique des tâches cron et celui-ci peut considérablement s'étoffer au fil du temps. Nous vous recommandons de planifier une tâche afin de purger cette table. Par exemple, conserver les entrées d'une semaine peut s'avérer suffisant à des fins de dépannage.

L'exemple suivant utilise la fonction cron.schedule pour planifier une tâche qui s'exécute tous les jours à minuit afin de purger la table `cron.job_run_details`. Cette tâche ne conserve que les entrées des sept derniers jours. Utilisez votre compte `rds_superuser` pour planifier la tâche comme suit :

```
SELECT cron.schedule('0 0 * * *', $$DELETE
    FROM cron.job_run_details
    WHERE end_time < now() - interval '7 days'$$);
```

Pour de plus amples informations, veuillez consulter Tableaux pour planifier les tâches et capturer leur statut .

Journalisation des erreurs dans le fichier postgresql.log uniquement

Pour empêcher les écritures dans la table `cron.job_run_details`, modifiez le groupe de paramètres associé à l'instance de base de données PostgreSQL et désactivez le paramètre `cron.log_run`. L'extension `pg_cron` n'écrit plus dans la table et consigne uniquement des erreurs dans le fichier `postgresql.log`. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Utilisez la commande suivante pour vérifier la valeur du paramètre `cron.log_run`.

```
postgres=> SHOW cron.log_run;
```

Pour de plus amples informations, veuillez consulter Paramètres de gestion de l'extension pg_cron.

Planification d'une tâche cron pour une base de données autre que la base de données par défaut

Toutes les métadonnées de `pg_cron` sont conservées dans la base de données par défaut PostgreSQL nommée `postgres`. Des exécutants étant utilisés en arrière-plan pour exécuter les tâches de maintenance cron, vous pouvez planifier une tâche dans n'importe quelle base de données de l'instance de base de données PostgreSQL.

> **Note**
>
> Seuls les utilisateurs disposant d'`rds_superuser`un rôle ou de `rds_superuser` privilèges peuvent répertorier toutes les tâches cron de la base de données. Les autres utilisateurs peuvent uniquement consulter leurs propres tâches dans le `cron.job` tableau.

1. Dans la base de données cron, planifiez la tâche comme vous le faites normalement à l'aide de la fonction cron.schedule.

```
postgres=> SELECT cron.schedule('database1 manual vacuum', '29 03 * * *', 'vacuum
 freeze test_table');
```

2. En tant qu'utilisateur ayant le rôle `rds_superuser`, mettez à jour la colonne de base de données correspondant à la tâche que vous venez de créer de manière à l'exécuter dans une autre base de données de votre instance de base de données PostgreSQL.

```
postgres=> UPDATE cron.job SET database = 'database1' WHERE jobid = 106;
```

3. Procédez à une vérification en interrogeant la table `cron.job`.

```
postgres=> SELECT * FROM cron.job;
jobid | schedule    | command                          | nodename  | nodeport |
 database | username  | active | jobname
------+-------------+----------------------------------+-----------+----------
+----------+-----------+--------+-------------------------
106   | 29 03 * * * | vacuum freeze test_table       | localhost | 8192     |
 database1| adminuser | t      | database1 manual vacuum
  1   | 59 23 * * * | vacuum freeze pgbench_accounts | localhost | 8192     |
 postgres | adminuser | t      | manual vacuum
(2 rows)
```

> **Note**
>
> Dans certains cas, vous pouvez ajouter une tâche cron que vous avez l'intention d'exécuter sur une base de données différente. Dans de tels cas, le tâche peut essayer de s'exécuter dans la base de données par défaut (`postgres`) avant la mise à jour de la colonne de

base de données correcte. Si le nom d'utilisateur dispose d'autorisations, la tâche s'exécute correctement dans la base de données par défaut.

Référence pour l'extension pg_cron

Vous pouvez utiliser les paramètres, fonctions et tables suivants avec l'extension `pg_cron`. Pour plus d'informations, consultez la section Qu'est-ce que pg_cron dans la documentation pg_cron.

Rubriques

- Paramètres de gestion de l'extension pg_cron
- Référence de fonction : cron.schedule
- Référence de fonction : cron.unschedule
- Tableaux pour planifier les tâches et capturer leur statut

Paramètres de gestion de l'extension pg_cron

La liste ci-dessous répertorie les paramètres permettant de contrôler le comportement de l'extension `pg_cron`.

Paramètre	Description
cron.database_name	Base de données dans laquelle les métadonnées `pg_cron` sont conservées.
`cron.host`	Nom d'hôte permettant de se connecter à PostgreSQL. Vous ne pouvez pas modifier cette valeur.
`cron.log_run`	Enregistrez chaque tâche qui s'exécute dans la table `job_run_details` . Les valeurs sont on ou `off`. Pour plus d'informations, consultez Tableaux pour planifier les tâches et capturer leur statut .
`cron.log_statement`	Enregistre toutes les instructions cron avant leur exécution. Les valeurs sont on ou `off`.

Paramètre	Description
`cron.max_running_jobs`	Nombre maximal de tâches pouvant être exécutées simultanément.
`cron.use_background_workers`	Utilisez des exécutants en arrière-plan plutôt que des sessions client. Vous ne pouvez pas modifier cette valeur.

Utilisez la commande SQL suivante pour afficher ces paramètres et leurs valeurs.

```
postgres=> SELECT name, setting, short_desc FROM pg_settings WHERE name LIKE 'cron.%'
 ORDER BY name;
```

Référence de fonction : cron.schedule

Cette fonction planifie une tâche cron. Cette tâche est initialement planifiée dans la base de données `postgres` par défaut. La fonction renvoie une valeur `bigint` correspondant à l'identifiant de la tâche. Pour planifier l'exécution de tâches dans d'autres bases de données de votre instance de base de données PostgreSQL, consultez l'exemple disponible dans Planification d'une tâche cron pour une base de données autre que la base de données par défaut.

La fonction présente deux formats de syntaxe.

Syntaxe

```
cron.schedule (job_name,
    schedule,
    command
);

cron.schedule (schedule,
    command
);
```

Paramètres

Paramètre	Description
`job_name`	Nom de la tâche cron.
`schedule`	Texte indiquant la planification de la tâche cron. Le format correspond au format cron standard.
`command`	Texte de la commande à exécuter.

Exemples

```
postgres=> SELECT cron.schedule ('test','0 10 * * *', 'VACUUM pgbench_history');
 schedule
----------
      145
(1 row)

postgres=> SELECT cron.schedule ('0 15 * * *', 'VACUUM pgbench_accounts');
 schedule
----------
      146
(1 row)
```

Référence de fonction : cron.unschedule

Cette fonction supprime une tâche cron. Vous pouvez spécifier soit le `job_name` ou le `job_id`. Une politique assure que vous soyez le propriétaire pouvant supprimer la planification de la tâche. La fonction renvoie une valeur booléenne indiquant la réussite ou l'échec.

La fonction a les formats de syntaxe suivants.

Syntaxe

```
cron.unschedule (job_id);

cron.unschedule (job_name);
```

Paramètres

Paramètre	Description
`job_id`	Identifiant de tâche renvoyé par la fonction `cron.schedule` lors de la planification de la tâche cron.
`job_name`	Nom d'une tâche cron planifiée avec la fonction `cron.schedule` .

Exemples

```
postgres=> SELECT cron.unschedule(108);
 unschedule
------------
 t
(1 row)

postgres=> SELECT cron.unschedule('test');
 unschedule
------------
 t
(1 row)
```

Tableaux pour planifier les tâches et capturer leur statut

Les tables suivantes sont utilisées pour planifier les tâches cron et enregistrer la façon dont elles ont été accomplies.

Tableau	Description
`cron.job`	Contient les métadonnées relatives à chaque tâche planifiée. La plupart des interactions avec cette table doivent être effectuées à l'aide des fonctions `cron.schedule` et `cron.unschedule` .

Tableau	Description
	Important Nous vous recommandons de ne pas accorder de privilèges de mise à jour ou d'insertion directement à cette table. Ce faisant, l'utilisateur pourrait mettre à jour la colonne `username` à exécuter en tant que `rds-superuser` .
`cron.job_run_details`	Contient des informations historiques sur l'exécution de tâches planifiées antérieures. Ces informations sont utiles pour examiner l'état, les messages renvoyés et les heures de début et de fin d'exécution de la tâche. **Note** Pour éviter que cette table évolue indéfiniment, purgez-la de manière régulière. Pour obtenir un exemple, veuillez consulter Purge de la table Historique pg_cron.

Utilisation de pglogical pour synchroniser les données entre les instances

Toutes les versions Postgre SQL prennent en charge l'`pglogical`extension. L'extension pglogical est antérieure à la fonctionnalité de réplication logique fonctionnellement similaire introduite par Postgre SQL dans la version 10. Pour de plus amples informations, veuillez consulter Réplication logique pour Amazon RDS for PostgreSQL.

L'`pglogical`extension prend en charge la réplication logique entre deux ou plusieurs . RDSpour les SQL instances de base de données Postgre. Il prend également en charge la réplication entre les différentes SQL versions de Postgre et entre les bases de données exécutées sur des instances de SQL base RDS de données Postgre et des clusters de bases de données Aurora PostgreSQL. L'extension `pglogical` utilise un modèle de publication et d'abonnement pour répliquer les changements apportés aux tables et aux autres objets, tels que les séquences, d'un serveur de publication à un abonné. Elle s'appuie sur un emplacement de réplication pour assurer la synchronisation des changements d'un nœud de serveur de publication à un nœud abonné, défini comme suit.

- Le nœud éditeur est le pour l'SQLinstance de base de données Postgre qui est la source des données à répliquer sur d'autres nœuds. Le nœud de serveur de publication définit les tables à répliquer dans un ensemble de publication.
- Le nœud abonné est le pour l'SQLinstance de base de données Postgre qui reçoit les WAL mises à jour de l'éditeur. L'abonné crée un abonnement pour se connecter à l'éditeur et obtenir les WAL données décodées. Lorsque l'abonné crée l'abonnement, l'emplacement de réplication est créé sur le nœud de serveur de publication.

Vous trouverez ci-après des informations sur la configuration de l'extension `pglogical`.

Rubriques

- Exigences et limites de l'extension pglogique
- Configuration de l'extension pglogical
- Configuration de la réplication logique pour l'instance de base de données RDS for PostgreSQL
- Rétablissement de la réplication logique après une mise à niveau majeure
- Gestion des emplacements de réplication logiques pour SQL
- Référence des paramètres de l'extension pglogical

Exigences et limites de l'extension pglogique

Toutes les versions actuellement disponibles d' prennent en charge l'`pglogical`extension.

Le nœud de serveur de publication et le nœud abonné doivent tous deux être configurés pour la réplication logique.

Les tables que vous souhaitez répliquer d'un éditeur à un abonné doivent porter les mêmes noms et le même schéma. Ces tables doivent également contenir les mêmes colonnes, et les colonnes doivent utiliser les mêmes types de données. Les tables des serveurs de publication et des abonnés doivent avoir les mêmes clés primaires. Nous vous recommandons de n'utiliser PRIMARY KEY que le comme contrainte unique.

Les tables du nœud abonné peuvent avoir des contraintes plus permissives que celles du nœud éditeur en ce qui concerne les CHECK contraintes et NOT NULL les contraintes.

L'`pglogical`extension fournit des fonctionnalités telles que la réplication bidirectionnelle qui ne sont pas prises en charge par la fonctionnalité de réplication logique intégrée à Postgre SQL (version 10 et supérieure). Pour plus d'informations, consultez Réplication SQL bidirectionnelle Postgre à l'aide de pglogical.

Configuration de l'extension pglogical

Pour configurer l'`pglogical`extension sur votre , vous devez l'ajouter `pglogical` aux bibliothèques partagées sur le groupe de paramètres de base de données personnalisé pour votre instance de base de données RDS pour Postgre SQL. Vous devez également définir la valeur du paramètre `rds.logical_replication` sur `1`, pour activer le décodage logique. Enfin, vous créez l'extension dans la base de données. Vous pouvez utiliser le AWS Management Console ou AWS CLI pour ces tâches.

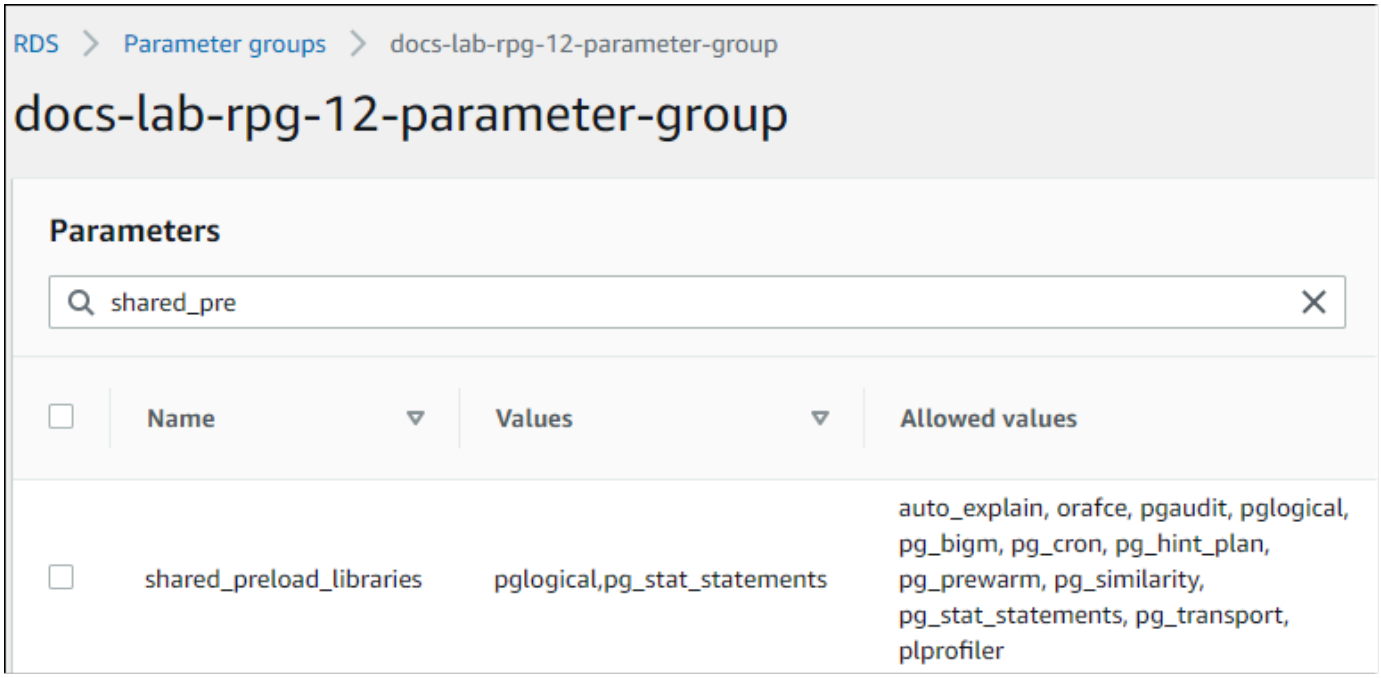
Vous devez disposer d'autorisations en tant que rôle `rds_superuser` pour effectuer ces tâches.

Les étapes suivantes supposent que votre instance de RDS pour Postgre SQL est associé à un groupe de paramètres de base de données de de base de données personnalisé. Pour plus d'informations sur la création d'un groupe de paramètres de cluster de bases de données, consultez Groupes de paramètres pour (Amazon RDS).

Console

Pour configurer l'extension pglogical

1. Connectez-vous à la RDS console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez l' de SQL base de données Postgre.
3. Ouvrez l'onglet Configuration de votre . RDSpour une SQL instance de base de données Postgre. Parmi les détails de l'instance, trouvez le lien Groupe de paramètres.
4. Cliquez sur le lien pour ouvrir les paramètres personnalisés associés à votre . RDSpour une SQL instance de base de données Postgre.
5. Dans le champ de recherche Parameters (Paramètres), tapez `shared_pre` pour trouver le paramètre `shared_preload_libraries`.
6. Choisissez Edit parameters (Modifier les paramètres) pour accéder aux valeurs des propriétés.
7. Ajoutez `pglogical` à la liste dans le champ Values (Valeurs). Utilisez une virgule pour séparer les éléments de la liste de valeurs.



8. Recherchez le paramètre `rds.logical_replication` et définissez-le sur `1`, pour activer la réplication logique.
9. Redémarrez l' pour l'SQLinstance de base de données Postgre afin que vos modifications prennent effet.

10. Lorsque l'instance est disponible, vous pouvez utiliser `psql` (oupgAdmin) pour vous connecter à l'. RDSpour une SQL instance de base de données Postgre.

```
psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=labdb
```

11. Pour vérifier que pglogical est initialisé, exécutez la commande suivante.

```
SHOW shared_preload_libraries;
shared_preload_libraries
--------------------------
rdsutils,pglogical
(1 row)
```

12. Vérifiez le paramètre qui active le décodage logique, comme suit.

```
SHOW wal_level;
wal_level
-----------
 logical
(1 row)
```

13. Créez l'extension, comme suit.

```
CREATE EXTENSION pglogical;
EXTENSION CREATED
```

14. Sélectionnez Enregistrer les modifications.
15. Ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
16. Choisissez l'instance d' dans la liste des bases de données pour la sélectionner, puis choisissez Redémarrer dans le menu Actions.

AWS CLI

Pour configurer l'extension pglogical

Pour configurer pglogical à l'aide de AWS CLI, vous devez appeler l'modify-db-parameter-groupopération pour modifier certains paramètres de votre groupe de paramètres personnalisé, comme indiqué dans la procédure suivante.

1. Utilisez la AWS CLI commande suivante pour `pglogical` ajouter au `shared_preload_libraries` paramètre.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=shared_preload_libraries,ParameterValue=pglogical,ApplyMethod=pending-
reboot" \
   --region aws-region
```

2. Utilisez la AWS CLI commande suivante pour définir sur `rds.logical_replication` 1 afin d'activer la fonctionnalité de décodage logique pour l'. RDSpour une SQL instance de base de données Postgre.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=rds.logical_replication,ParameterValue=1,ApplyMethod=pending-
reboot" \
   --region aws-region
```

3. Utilisez la AWS CLI commande suivante pour redémarrer l' pour SQL instance de base de données Postgre afin que la bibliothèque pglogical soit initialisée.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

4. Lorsque l'instance est disponible, utilisez-la `psql` pour vous connecter à l'. RDSpour une SQL instance de base de données Postgre.

```
psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=labdb
```

5. Créez l'extension, comme suit.

```
CREATE EXTENSION pglogical;
EXTENSION CREATED
```

6. Redémarrez l' pour SQL instance de base de données Postgre à l'aide de la commande suivante AWS CLI .

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

Configuration de la réplication logique pour l'instance de base de données RDS for PostgreSQL

La procédure suivante vous montre comment démarrer la réplication logique entre deux instances de base de données RDS for PostgreSQL. Les étapes supposent que la source (serveur de publication) et la cible (abonné) ont toutes deux l'extension `pglogical` configurée comme indiqué dans le document Configuration de l'extension pglogical.

> **Note**
>
> Le nœud `node_name` d'un abonné ne peut pas commencer par `rds`.

Pour créer le nœud de serveur de publication et définir les tables à répliquer

Ces étapes supposent que votre instance de base de données RDS for PostgreSQL possède une base de données qui contient une ou plusieurs tables que vous voulez répliquer vers un autre nœud. Vous devez recréer la structure de la table du serveur de publication sur l'abonné, donc d'abord, récupérer la structure de la table si nécessaire. Vous pouvez le faire en utilisant la métacommande `psql \d tablename` et en créant ensuite la même table sur l'instance de l'abonné. La procédure suivante crée un exemple de table sur le serveur de publication (source) à des fins de démonstration.

1. Utilisez `psql` pour vous connecter à l'instance qui possède la table que vous voulez utiliser comme source pour les abonnés.

   ```
   psql --host=source-instance.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password --dbname=labdb
   ```

   Si vous ne disposez pas d'une table existante que vous souhaitez répliquer, vous pouvez créer un exemple de table comme suit.

   a. Créez un exemple de table en utilisant l'instruction SQL suivante.

```
CREATE TABLE docs_lab_table (a int PRIMARY KEY);
```

b. Remplissez la table avec les données générées en utilisant l'instruction SQL suivante.

```
INSERT INTO docs_lab_table VALUES (generate_series(1,5000));
INSERT 0 5000
```

c. Vérifiez que les données existent dans la table à l'aide de l'instruction SQL suivante.

```
SELECT count(*) FROM docs_lab_table;
```

2. Identifiez cette instance de base de données RDS for PostgreSQL comme le nœud de serveur de publication, comme suit.

```
SELECT pglogical.create_node(
    node_name := 'docs_lab_provider',
    dsn := 'host=source-instance.aws-region.rds.amazonaws.com port=5432
 dbname=labdb');
 create_node
-------------
   3410995529
(1 row)
```

3. Ajoutez la table que vous souhaitez répliquer à l'ensemble de réplication par défaut. Pour plus d'informations sur les ensembles de réplication, consultez Replication sets (Ensembles de réplication) dans la documentation pglogical.

```
SELECT pglogical.replication_set_add_table('default', 'docs_lab_table', 'true',
 NULL, NULL);
 replication_set_add_table
  --------------------------
  t
  (1 row)
```

La configuration du nœud de serveur de publication est terminée. Vous pouvez maintenant configurer le nœud abonné pour recevoir les mises à jour du serveur de publication.

Pour configurer le nœud abonné et créer un abonnement pour recevoir des mises à jour

Ces étapes supposent que l'instance de base de données RDS for PostgreSQL a été configurée avec l'extension `pglogical`. Pour de plus amples informations, veuillez consulter Configuration de l'extension pglogical.

1. Utilisez `psql` pour vous connecter à l'instance qui doit recevoir les mises à jour du serveur de publication.

```
psql --host=target-instance.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=labdb
```

2. Sur l'instance de base de données RDS for PostgreSQL de l'abonné, créez la même table que celle qui existe sur le serveur de publication. Pour cet exemple, la table est `docs_lab_table`. Vous pouvez créer la table comme suit.

```
CREATE TABLE docs_lab_table (a int PRIMARY KEY);
```

3. Vérifiez que cette table est vide.

```
SELECT count(*) FROM docs_lab_table;
 count
-------
  0
(1 row)
```

4. Identifiez cette instance de base de données RDS for PostgreSQL comme le nœud abonné, comme suit.

```
SELECT pglogical.create_node(
    node_name := 'docs_lab_target',
    dsn := 'host=target-instance.aws-region.rds.amazonaws.com port=5432
 sslmode=require dbname=labdb user=postgres password=********');
 create_node
-------------
   2182738256
(1 row)
```

5. Créez l'abonnement.

```
SELECT pglogical.create_subscription(
   subscription_name := 'docs_lab_subscription',
```

```
   provider_dsn := 'host=source-instance.aws-region.rds.amazonaws.com port=5432
 sslmode=require dbname=labdb user=postgres password=*******',
   replication_sets := ARRAY['default'],
   synchronize_data := true,
   forward_origins := '{}' );
 create_subscription
---------------------
1038357190
(1 row)
```

Lorsque vous terminez cette étape, les données de la table du serveur de publication sont créées dans la table de l'abonné. Vous pouvez le vérifier en utilisant la requête SQL suivante.

```
SELECT count(*) FROM docs_lab_table;
 count
-------
  5000
(1 row)
```

À partir de ce moment, les modifications apportées à la table sur le serveur de publication sont répliquées sur la table sur l'abonné.

Rétablissement de la réplication logique après une mise à niveau majeure

Avant de pouvoir effectuer une mise à niveau de version majeure d'un pour une SQL instance de base de données Postgre configurée en tant que nœud d'éditeur pour la réplication logique, vous devez supprimer tous les emplacements de réplication, même ceux qui ne sont pas actifs. Nous vous recommandons de détourner temporairement les transactions de base de données du nœud éditeur, de supprimer les emplacements de réplication, de de mettre à niveau l'SQLinstance de base de données RDS pour Postgre, puis de rétablir et de redémarrer la réplication.

Les emplacements de réplication sont hébergés uniquement sur le nœud de serveur de publication. Dans un scénario de réplication logique, le nœud d'SQLabonné RDS destiné à Postgre ne comporte aucun emplacement à supprimer, mais il ne peut pas être mis à niveau vers une version majeure tant qu'il est désigné comme nœud d'abonné avec un abonnement auprès de l'éditeur. Avant de mettre à niveau le nœud RDS réservé aux SQL abonnés Postgre, supprimez l'abonnement et le nœud. Pour plus d'informations, consultezGestion des emplacements de réplication logiques pour SQL.

Détermination de la perturbation de la réplication logique

Vous pouvez déterminer que le processus de réplication a été interrompu en interrogeant le nœud de serveur de publication ou le nœud abonné, comme suit.

Pour vérifier le nœud de serveur de publication

- Utilisez `psql` pour vous connecter au nœud de serveur de publication, puis interrogez la fonction `pg_replication_slots`. Notez la valeur dans la colonne active. Normalement, cela renvoie la valeur `t` (true), ce qui montre que la réplication est active. Si la requête renvoie la valeur `f` (false), cela indique que la réplication vers l'abonné a cessé.

```
SELECT slot_name,plugin,slot_type,active FROM pg_replication_slots;
                    slot_name              |      plugin      | slot_type | active
-------------------------------------------+------------------+-----------+--------
 pgl_labdb_docs_labcb4fa94_docs_lab3de412c | pglogical_output | logical   | f
(1 row)
```

Pour vérifier le nœud abonné

Sur le nœud abonné, vous pouvez vérifier l'état de la réplication de trois manières différentes.

- Parcourez les SQL journaux Postgre sur le nœud abonné pour trouver les messages d'échec. Le journal identifie l'échec avec des messages qui incluent le code de sortie 1, comme indiqué ci-dessous.

```
2022-07-06 16:17:03 UTC::@:[7361]:LOG: background worker "pglogical apply
 16404:2880255011" (PID 14610) exited with exit code 1
2022-07-06 16:19:44 UTC::@:[7361]:LOG: background worker "pglogical apply
 16404:2880255011" (PID 21783) exited with exit code 1
```

- Interrogez la fonction `pg_replication_origin`. Connectez-vous à la base de données sur le nœud abonné en utilisant `psql` et interrogez la fonction `pg_replication_origin`, comme suit.

```
SELECT * FROM pg_replication_origin;
 roident | roname
---------+--------
(0 rows)
```

L'ensemble de résultats vide signifie que la réplication a été perturbée. Normalement, vous obtenez un résultat qui ressemble au suivant.

```
 roident |                      roname
---------+---------------------------------------------------
       1 | pgl_labdb_docs_labcb4fa94_docs_lab3de412c
(1 row)
```

- Interrogez la fonction `pglogical.show_subscription_status` comme indiqué dans l'exemple suivant.

```
SELECT subscription_name,status,slot_name FROM pglogical.show_subscription_status();
     subscription_name | status |              slot_name
---====----------------+--------+-----------------------------------
 docs_lab_subscription | down   | pgl_labdb_docs_labcb4fa94_docs_lab3de412c
(1 row)
```

Cette sortie montre que la réplication a été perturbée. Son statut est down. Normalement, la sortie indique le statut `replicating`.

Si votre processus de réplication logique a été perturbé, vous pouvez rétablir la réplication en suivant les étapes suivantes.

Pour rétablir la réplication logique entre les nœuds de serveur de publication et abonné.

Pour rétablir la réplication, vous devez d'abord déconnecter l'abonné du nœud de serveur de publication, puis rétablir l'abonnement, comme indiqué dans les étapes suivantes.

1. Connectez-vous au nœud abonné à l'aide de `psql`, comme suit.

```
psql --host=222222222222.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=labdb
```

2. Désactivez l'abonnement en utilisant la fonction `pglogical.alter_subscription_disable`.

```
SELECT pglogical.alter_subscription_disable('docs_lab_subscription',true);
 alter_subscription_disable
----------------------------
 t
```

```
(1 row)
```

3. Obtenez l'identifiant du nœud de serveur de publication en interrogeant `pg_replication_origin`, comme suit.

```
SELECT * FROM pg_replication_origin;
 roident |               roname
---------+--------------------------------------
       1 | pgl_labdb_docs_labcb4fa94_docs_lab3de412c
(1 row)
```

4. Utilisez la réponse de l'étape précédente avec la commande `pg_replication_origin_create` pour attribuer l'identifiant qui peut être utilisé par l'abonnement lorsqu'il est rétabli.

```
SELECT pg_replication_origin_create('pgl_labdb_docs_labcb4fa94_docs_lab3de412c');
  pg_replication_origin_create
------------------------------
                             1
(1 row)
```

5. Activez l'abonnement en transmettant son nom avec un statut `true`, comme indiqué dans l'exemple suivant.

```
SELECT pglogical.alter_subscription_enable('docs_lab_subscription',true);
  alter_subscription_enable
---------------------------
 t
(1 row)
```

Vérifiez le statut du nœud. Son statut doit être `replicating`, tel qu'indiqué dans cet exemple.

```
SELECT subscription_name,status,slot_name
  FROM pglogical.show_subscription_status();
            subscription_name |   status    |              slot_name
------------------------------+-------------+------------------------------------
 docs_lab_subscription         | replicating |
 pgl_labdb_docs_lab98f517b_docs_lab3de412c
(1 row)
```

Vérifiez le statut de l'emplacement de réplication de l'abonné sur le nœud de serveur de publication. La colonne `active` de l'emplacement doit retourner `t` (true), indiquant que la réplication a été rétablie.

```
SELECT slot_name,plugin,slot_type,active
  FROM pg_replication_slots;
                   slot_name              |      plugin      | slot_type | active
------------------------------------------+------------------+-----------+--------
 pgl_labdb_docs_lab98f517b_docs_lab3de412c | pglogical_output | logical   | t
(1 row)
```

Gestion des emplacements de réplication logiques pour SQL

Avant de pouvoir effectuer une mise à niveau de version majeure sur une pour une SQL instance de base de données Postgre qui sert de nœud d'éditeur dans un scénario de réplication logique, vous devez supprimer les emplacements de réplication de l'instance. Le processus de pré-vérification des mises à niveau de versions majeures vous informe que la mise à niveau ne peut pas avoir lieu tant que les emplacements ne sont pas supprimés.

Pour supprimer des emplacements de votre RDS SQL instance de base de données Postgre, supprimez d'abord l'abonnement, puis supprimez l'emplacement.

Pour identifier les emplacements de réplication qui ont été créés à l'aide de l'extension `pglogical`, connectez-vous à chaque base de données et obtenez le nom des nœuds. Lorsque vous interrogez le nœud abonné, vous obtenez à la fois le nœud de serveur de publication et le nœud abonné dans la sortie, comme le montre cet exemple.

```
SELECT * FROM pglogical.node;
node_id   |     node_name
------------+------------------
 2182738256 | docs_lab_target
 3410995529 | docs_lab_provider
(2 rows)
```

Vous pouvez obtenir les détails de l'abonnement avec la requête suivante.

```
SELECT sub_name,sub_slot_name,sub_target
  FROM pglogical.subscription;
 sub_name |         sub_slot_name          | sub_target
```

```
----------+---------------------------------+------------
  docs_lab_subscription     | pgl_labdb_docs_labcb4fa94_docs_lab3de412c | 2182738256
(1 row)
```

Vous pouvez maintenant supprimer l'abonnement, comme suit.

```
SELECT pglogical.drop_subscription(subscription_name := 'docs_lab_subscription');
 drop_subscription
-------------------
                 1
(1 row)
```

Après avoir supprimé l'abonnement, vous pouvez supprimer le nœud.

```
SELECT pglogical.drop_node(node_name := 'docs-lab-subscriber');
 drop_node
-----------
 t
(1 row)
```

Vous pouvez vérifier que le nœud n'existe plus, comme suit.

```
SELECT * FROM pglogical.node;
 node_id | node_name
---------+-----------
(0 rows)
```

Référence des paramètres de l'extension pglogical

Dans le tableau, vous pouvez trouver les paramètres associés à l'extension `pglogical`. Les paramètres tels que `pglogical.conflict_log_level` et `pglogical.conflict_resolution` sont utilisés pour gérer les conflits de mise à jour. Des conflits peuvent survenir lorsque des modifications sont apportées localement aux mêmes tables qui sont abonnées aux modifications du serveur de publication. Des conflits peuvent également se produire au cours de divers scénarios, tels que la réplication bidirectionnelle ou lorsque plusieurs abonnés se répliquent à partir du même serveur de publication. Pour plus d'informations, consultez Réplication SQL bidirectionnelle Postgre à l'aide de pglogical.

Paramètre	Description
pglogical.batch_inserts	Insertions de lots si possible Non défini par défaut. Remplacez par « 1 » pour activer, par « 0 » pour désactiver.
pglogical.conflict_log_level	Définit le niveau de journalisation à utiliser pour la journalisation des conflits résolus. Les valeurs de chaîne prises en charge sont debug5, debug4, debug3, debug2, debug1, info, notice, warning, error, log, fatal, panic.
pglogical.conflict_resolution	Définit la méthode à utiliser pour résoudre les conflits lorsque ceux-ci sont résolubles. Les valeurs de chaîne prises en charge sont error, apply_remote, keep_local, last_update_wins, first_update_wins.
pglogical.extra_connection_options	Options de connexion à ajouter à toutes les connexions de nœuds de pairs.
pglogical.synchronous_commit	valeur de validation synchrone spécifique pglogical
pglogical.use_spi	Utilisez SPI (interface de programmation du serveur) plutôt que le bas niveau API pour appliquer les modifications. Définissez sur « 1 » pour activer, sur « 0 » pour désactiver. Pour plus d'informations à ce sujetSPI, consultez la section Interface de programmation du serveur dans la SQL documentation Postgre.

Utilisation de pgactive pour prendre en charge la réplication active-active

L'extension `pgactive` utilise la réplication active-active pour prendre en charge et coordonner les opérations d'écriture sur plusieurs bases de données RDS for PostgreSQL. Amazon RDS for PostgreSQL `pgactive` prend en charge l'extension sur les versions suivantes :

- RDS pour PostgreSQL 17.0 et toutes les versions supérieures
- RDS pour PostgreSQL 16.1 et versions ultérieures 16
- RDS pour PostgreSQL 15.4-R2 et versions ultérieures (15 versions)
- RDS pour PostgreSQL 14.10 et versions ultérieures 14
- RDS pour PostgreSQL 13.13 et versions ultérieures 13
- RDS pour PostgreSQL 12.17 et versions ultérieures 12
- RDS pour PostgreSQL 11.22

> **Note**
>
> Lorsque des opérations d'écriture se produisent sur plusieurs bases de données dans une configuration de réplication, des conflits peuvent survenir. Pour de plus amples informations, consultez Gestion des conflits de la réplication active-active.

Rubriques

- Limitations liées à l'extension pgactive
- Initialisation de la capacité d'extension pgactive
- Configuration de la réplication active-active pour des instances de base de données RDS for PostgreSQL
- Mesurer le délai de réplication entre les membres actifs
- Configuration des paramètres de l'extension pgactive
- Comprendre les conflits actifs-actifs
- Comprendre le schéma pgactive
- référence des fonctions pgactive
- Gestion des conflits de la réplication active-active
- Gestion des séquences dans une réplication active-active

Limitations liées à l'extension pgactive

- Toutes les tables nécessitent une clé primaire, faute de quoi, les mises à jour et les suppressions ne sont pas autorisées. Les valeurs de la colonne Primary Key (Clé primaire) ne doivent pas être mises à jour.
- Les séquences peuvent présenter des écarts et parfois ne suivre aucun ordre. Les séquences ne sont pas répliquées. Pour de plus amples informations, veuillez consulter Gestion des séquences dans une réplication active-active.
- Les objets volumineux et DDL ne sont pas répliqués.
- Les index uniques secondaires peuvent causer des divergences de données.
- Le classement doit être identique sur tous les nœuds du groupe.
- L'équilibrage de charge entre les nœuds est un anti-modèle.
- Les transactions volumineuses peuvent occasionner un retard de réplication.

Initialisation de la capacité d'extension pgactive

Pour initialiser la capacité d'extension `pgactive` sur votre instance de base de données RDS for PostgreSQL, définissez la valeur du paramètre `rds.enable_pgactive` sur `1`, puis créez l'extension dans la base de données. Les paramètres `rds.logical_replication` et `track_commit_timestamp` sont alors automatiquement activés et la valeur de `wal_level` est définie sur `logical`.

Vous devez disposer d'autorisations en tant que rôle `rds_superuser` pour effectuer ces tâches.

Vous pouvez utiliser le AWS Management Console ou le AWS CLI pour créer le RDS requis pour les instances de base de données PostgreSQL. Les étapes suivantes partent du principe que votre instance de base de données RDS for PostgreSQL est associée à un groupe de paramètres de base de données personnalisés. Pour obtenir des informations sur la création d'un groupe de paramètres de base de données personnalisé, consultez Groupes de paramètres pour (Amazon RDS).

console

Pour initialiser la capacité d'extension pgactive

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.

2. Dans le volet de navigation, sélectionnez votre instance de base de données RDS for PostgreSQL.
3. Ouvrez l'onglet Configuration pour votre instance de base de données RDS for PostgreSQL. Dans les détails de l'instance, recherchez le lien Groupe de paramètres d'instance de base de données.
4. Cliquez sur le lien pour ouvrir les paramètres personnalisés associés à votre instance de base de données RDS for PostgreSQL.
5. Recherchez le paramètre `rds.enable_pgactive` et définissez-le sur `1` pour initialiser la fonctionnalité `pgactive`.
6. Sélectionnez Enregistrer les modifications.
7. Dans le panneau de navigation de la console Amazon RDS, sélectionnez Bases de données.
8. Sélectionnez votre instance de base de données RDS for PostgreSQL, puis choisissez Redémarrer dans le menu Actions.
9. Confirmez le redémarrage de l'instance de base de données pour que vos modifications prennent effet.
10. Une fois l'instance de base de données disponible, vous pouvez utiliser `psql` ou tout autre client PostgreSQL pour vous connecter à l'instance de base de données RDS for PostgreSQL.

    L'exemple suivant suppose que votre instance de base de données RDS pour PostgreSQL possède une base de données par défaut nommée. *`postgres`*

    ```
    psql --host=mydb.111122223333.aws-region.rds.amazonaws.com --port=5432 --
    username=postgres --password=PASSWORD --dbname=postgres
    ```

11. Pour vérifier que pgactive est initialisé, exécutez la commande suivante.

    ```
    postgres=>SELECT setting ~ 'pgactive'
    FROM pg_catalog.pg_settings
    WHERE name = 'shared_preload_libraries';
    ```

    Si `pgactive` se trouve dans `shared_preload_libraries`, la commande précédente renvoie ceci :

    ```
    ?column?
    ----------
    ```

```
 t
```

AWS CLI

Pour initialiser la capacité d'extension pgactive

Pour initialiser l'`pgactive`utilisation de AWS CLI, appelez l'modify-db-parameter-groupopération pour modifier certains paramètres de votre groupe de paramètres personnalisé, comme indiqué dans la procédure suivante.

1. Utilisez la AWS CLI commande suivante pour définir sur afin `rds.enable_pgactive` d'1initialiser la `pgactive` fonctionnalité de l'instance de base de données RDS pour PostgreSQL.

```
postgres=>aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=rds.enable_pgactive,ParameterValue=1,ApplyMethod=pending-reboot" \
   --region aws-region
```

2. Utilisez la AWS CLI commande suivante pour redémarrer l'instance de base de données RDS pour PostgreSQL afin que `pgactive` la bibliothèque soit initialisée.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

3. Lorsque l'instance est disponible, utilisez `psql` pour vous connecter à l'instance de base de données RDS for PostgreSQL.

```
psql --host=mydb.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=master user --password=PASSWORD --dbname=postgres
```

4. Pour vérifier que pgactive est initialisé, exécutez la commande suivante.

```
postgres=>SELECT setting ~ 'pgactive'
FROM pg_catalog.pg_settings
WHERE name = 'shared_preload_libraries';
```

Si `pgactive` se trouve dans `shared_preload_libraries`, la commande précédente renvoie ceci :

```
?column?
----------
 t
```

Configuration de la réplication active-active pour des instances de base de données RDS for PostgreSQL

La procédure suivante explique comment démarrer une réplication active-active entre deux , le cas échéant. `pgactive` Pour suivre l'exemple de haute disponibilité multirégionale, vous devez déployer des instances Amazon RDS for PostgreSQL dans deux régions différentes et configurer l'appairage de VPC. Pour en savoir plus, consultez Appairage de VPC.

> **Note**
>
> L'envoi de trafic entre plusieurs régions peut induire des coûts supplémentaires.

Ces étapes supposent que l'instance de base de données RDS pour PostgreSQL a été activée avec l'extension. `pgactive` Pour de plus amples informations, veuillez consulter Initialisation de la capacité d'extension pgactive.

Pour configurer la première instance de base de données RDS for PostgreSQL avec l'extension **`pgactive`**

L'exemple suivant illustre la façon dont le groupe `pgactive` est créé et présente les autres étapes nécessaires à la création de l'extension `pgactive` sur l'instance de base de données RDS for PostgreSQL.

1. Utilisez `psql` ou un autre outil client pour vous connecter à votre première instance de base de données RDS for PostgreSQL.

```
psql --host=firstinstance.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password=PASSWORD --dbname=postgres
```

2. Créez une base de données sur l'instance RDS for PostgreSQL à l'aide de la commande suivante :

```
postgres=> CREATE DATABASE app;
```

3. Faites basculer la connexion sur la nouvelle base de données à l'aide de la commande suivante :

```
\c app
```

4. Créez et remplissez une table d'exemple à l'aide des instructions SQL suivantes :

   a. Créez un exemple de table en utilisant l'instruction SQL suivante.

```
app=> CREATE SCHEMA inventory;
CREATE TABLE inventory.products (
id int PRIMARY KEY, product_name text NOT NULL,
created_at timestamptz NOT NULL DEFAULT CURRENT_TIMESTAMP);
```

   b. Remplissez la table avec des données d'exemple à l'aide de l'instruction SQL suivante.

```
app=> INSERT INTO inventory.products (id, product_name)
VALUES (1, 'soap'), (2, 'shampoo'), (3, 'conditioner');
```

   c. Vérifiez que les données existent dans la table à l'aide de l'instruction SQL suivante.

```
 app=>SELECT count(*) FROM inventory.products;

 count
-------
 3
```

5. Créez l'extension `pgactive` sur la base de données existante.

```
app=> CREATE EXTENSION pgactive;
```

6. Pour créer et initialiser le groupe pgactive en toute sécurité, utilisez les commandes suivantes :

```
app=>
-- connection info for endpoint1
CREATE SERVER pgactive_server_endpoint1
    FOREIGN DATA WRAPPER pgactive_fdw
    OPTIONS (host '<endpoint1>', dbname 'app');
```

```
CREATE USER MAPPING FOR postgres
    SERVER pgactive_server_endpoint1
    OPTIONS (user 'postgres', password '<password>');
      -- connection info for endpoint2
CREATE SERVER pgactive_server_endpoint2
    FOREIGN DATA WRAPPER pgactive_fdw
    OPTIONS (host '<endpoint2>', dbname 'app');
CREATE USER MAPPING FOR postgres
    SERVER pgactive_server_endpoint2
    OPTIONS (user 'postgres', password '<password>');
```

Vous pouvez maintenant initialiser le groupe de réplication et ajouter cette première instance :

```
SELECT pgactive.pgactive_create_group(
    node_name := 'endpoint1-app',
    node_dsn := 'user_mapping=postgres
 pgactive_foreign_server=pgactive_server_endpoint1'

);
```

Utilisez les commandes suivantes comme méthode alternative mais moins sécurisée pour créer et initialiser le groupe pgactive :

```
app=> SELECT pgactive.pgactive_create_group(
    node_name := 'node1-app',
    node_dsn := 'dbname=app host=firstinstance.111122223333.aws-
region.rds.amazonaws.com user=postgres password=PASSWORD');
```

node1-app est le nom que vous attribuez pour identifier de manière unique un nœud dans le groupe `pgactive`.

> **Note**
>
> Pour pouvoir effectuer cette étape sur une instance de base de données accessible au public, vous devez activer le paramètre `rds.custom_dns_resolution` en le définissant sur 1.

7. Pour vérifier si l'instance de base de données est prête, utilisez la commande suivante :

```
app=> SELECT pgactive.pgactive_wait_for_node_ready();
```

Si la commande aboutit, vous obtenez la sortie suivante :

```
pgactive_wait_for_node_ready
-----------------------------
(1 row)
```

Pour configurer la deuxième instance RDS for PostgreSQL et la joindre au groupe **`pgactive`**

L'exemple suivant illustre la façon dont une instance de base de données RDS for PostgreSQL est jointe au groupe `pgactive` et présente les autres étapes nécessaires à la création de l'extension `pgactive` sur l'instance de base de données.

Ces étapes partent du principe que d'autres instances de base de données RDS for PostgreSQL ont été configurées avec l'extension `pgactive`. Pour de plus amples informations, veuillez consulter Initialisation de la capacité d'extension pgactive.

1. Utilisez `psql` pour vous connecter à l'instance qui doit recevoir les mises à jour du serveur de publication.

```
psql --host=secondinstance.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password=PASSWORD --dbname=postgres
```

2. Créez une base de données sur la deuxième instance de base de données RDS for PostgreSQL à l'aide de la commande suivante :

```
postgres=> CREATE DATABASE app;
```

3. Faites basculer la connexion sur la nouvelle base de données à l'aide de la commande suivante :

```
\c app
```

4. Créez l'extension `pgactive` sur la base de données existante.

```
app=> CREATE EXTENSION pgactive;
```

5. Joignez la deuxième instance de base de données du cluster de bases de RDS pour PostgreSQL au groupe de manière plus sécurisée à `pgactive` l'aide des commandes suivantes :

```
-- connection info for endpoint1
CREATE SERVER pgactive_server_endpoint1
    FOREIGN DATA WRAPPER pgactive_fdw
    OPTIONS (host '<endpoint1>', dbname 'app');
CREATE USER MAPPING FOR postgres
    SERVER pgactive_server_endpoint1
    OPTIONS (user 'postgres', password '<password>');

-- connection info for endpoint2
CREATE SERVER pgactive_server_endpoint2
    FOREIGN DATA WRAPPER pgactive_fdw
    OPTIONS (host '<endpoint2>', dbname 'app');
CREATE USER MAPPING FOR postgres
    SERVER pgactive_server_endpoint2
    OPTIONS (user 'postgres', password '<password>');
```

```
SELECT pgactive.pgactive_join_group(
    node_name := 'endpoint2-app',
    node_dsn := 'user_mapping=postgres
 pgactive_foreign_server=pgactive_server_endpoint2',
    join_using_dsn := 'user_mapping=postgres
 pgactive_foreign_server=pgactive_server_endpoint1'
);
```

Utilisez les commandes suivantes comme méthode alternative, mais moins sécurisée, pour joindre la deuxième instance de base de données du (RDS pour PostgreSQL) au groupe. `pgactive`

```
app=> SELECT pgactive.pgactive_join_group(
node_name := 'node2-app',
node_dsn := 'dbname=app host=secondinstance.111122223333.aws-
region.rds.amazonaws.com user=postgres password=PASSWORD',
```

```
join_using_dsn := 'dbname=app host=firstinstance.111122223333.aws-
region.rds.amazonaws.com user=postgres password=PASSWORD');
```

node2-app est le nom que vous attribuez pour identifier de manière unique un nœud dans le groupe `pgactive`.

6. Pour vérifier si l'instance de base de données est prête, utilisez la commande suivante :

```
app=> SELECT pgactive.pgactive_wait_for_node_ready();
```

Si la commande aboutit, vous obtenez la sortie suivante :

```
pgactive_wait_for_node_ready
-----------------------------
(1 row)
```

Si la première base de données RDS for PostgreSQL est relativement volumineuse, `pgactive.pgactive_wait_for_node_ready()` émet le rapport de progression de l'opération de restauration. La sortie ressemble à ce qui suit:

```
NOTICE:  restoring database 'app', 6% of 7483 MB complete
NOTICE:  restoring database 'app', 42% of 7483 MB complete
NOTICE:  restoring database 'app', 77% of 7483 MB complete
NOTICE:  restoring database 'app', 98% of 7483 MB complete
NOTICE:  successfully restored database 'app' from node node1-app in
 00:04:12.274956
 pgactive_wait_for_node_ready
-----------------------------
(1 row)
```

À partir de cet instant, `pgactive` synchronise les données entre les deux instances de base de données.

7. Vous pouvez utiliser la commande suivante pour vérifier si la base de données de la deuxième instance de base de données contient les données :

```
app=> SELECT count(*) FROM inventory.products;
```

Si les données ont bien été synchronisées, vous obtenez la sortie suivante :

```
 count
-------
 3
```

8. Exécutez la commande suivante pour insérer de nouvelles valeurs :

```
app=> INSERT INTO inventory.products (id, product_name) VALUES (4, 'lotion');
```

9. Connectez-vous à la base de données de la première instance de base de données et exécutez la requête suivante :

```
app=> SELECT count(*) FROM inventory.products;
```

Si la réplication active-active est initialisée, vous obtenez une sortie de ce type :

```
count
-------
 4
```

Pour détacher et supprimer une instance de base de données du groupe **`pgactive`**

Vous pouvez détacher et supprimer une instance de base de données du groupe `pgactive` en suivant ces étapes :

1. Vous pouvez détacher la deuxième instance de base de données de la première à l'aide de la commande suivante :

```
app=> SELECT * FROM pgactive.pgactive_detach_nodes(ARRAY['node2-app']);
```

2. Supprimez l'extension `pgactive` de la deuxième instance de base de données à l'aide de la commande suivante :

```
app=> SELECT * FROM pgactive.pgactive_remove();
```

Pour supprimer l'extension de force :

```
app=> SELECT * FROM pgactive.pgactive_remove(true);
```

3. Supprimez l'extension à l'aide de la commande suivante :

```
app=> DROP EXTENSION pgactive;
```

Mesurer le délai de réplication entre les membres actifs

Vous pouvez utiliser la requête suivante pour visualiser le délai de réplication entre les `pgactive` membres. Exécutez cette requête sur chaque `pgactive` nœud pour obtenir une vue d'ensemble complète.

```
app=> SELECT * FROM pgactive.pgactive_get_replication_lag_info();
#-[ RECORD 1 ]--------+-----------------------------------------
#node_name            | node2-app
#node_sysid           | 7481018224801653637
#application_name     | pgactive:7481018224801653637:send
#slot_name            | pgactive_16385_7481018224801653637_0_16385__
#active               | t
#active_pid           | 783486
#pending_wal_decoding | 0
#pending_wal_to_apply | 0
#restart_lsn          | 0/2108150
#confirmed_flush_lsn  | 0/2154690
#sent_lsn             | 0/2154690
#write_lsn            | 0/2154690
#flush_lsn            | 0/2154690
#replay_lsn           | 0/2154690
#-[ RECORD 2 ]--------+-----------------------------------------
#node_name            | node1-app
#node_sysid           | 7481018033434600853
#application_name     | pgactive:7481018033434600853:send
#slot_name            | pgactive_16385_7481018033434600853_0_16385__
#active               | t
#active_pid           | 783488
#pending_wal_decoding | 0
#pending_wal_to_apply | 0
#restart_lsn          | 0/20F5AD0
#confirmed_flush_lsn  | 0/214EF68
#sent_lsn             | 0/214EF68
#write_lsn            | 0/214EF68
#flush_lsn            | 0/214EF68
```

```
#replay_lsn           | 0/214EF68
```

Surveillez au minimum les diagnostics suivants :

actif

Configurer des alertes lorsqu'il est actif est défini sur false, ce qui indique que le slot n'est pas actuellement utilisé (l'instance d'abonné s'est déconnectée de l'éditeur).

pending_wal_decoding

Dans la réplication logique de PostgreSQL, les fichiers WAL sont stockés au format binaire. L'éditeur doit décoder ces modifications WAL et les convertir en modifications logiques (telles que des opérations d'insertion, de mise à jour ou de suppression).

La métrique pending_wal_decoding indique le nombre de fichiers WAL en attente de décodage côté éditeur.

Ce nombre peut augmenter en raison des facteurs suivants :

- Lorsque l'abonné n'est pas connecté, le statut actif devient faux et le paramètre pending_wal_decoding augmente
- Le slot est actif, mais l'éditeur ne parvient pas à suivre le volume de modifications du WAL

pending_wal_to_apply

La métrique pending_wal_apply indique le nombre de fichiers WAL en attente d'être appliqués côté abonné.

Plusieurs facteurs peuvent empêcher l'abonné d'appliquer les modifications et provoquer un scénario de saturation du disque :

- Différences de schéma : par exemple, lorsque vous avez des modifications dans le flux WAL pour une table nommée sample, mais que cette table n'existe pas du côté abonné
- Les valeurs des colonnes clés primaires ont été mises à jour
- Les index uniques secondaires peuvent entraîner une divergence des données

Configuration des paramètres de l'extension pgactive

Vous pouvez utiliser la requête suivante pour afficher tous les paramètres associés à l'extension `pgactive`.

```
app=> SELECT * FROM pg_settings WHERE name LIKE 'pgactive.%';
```

Vous pouvez configurer l'`pgactive`extension à l'aide de différents paramètres. Ces paramètres peuvent être définis par le biais de l'interface CLI AWS Management Console ou de l'interface AWS CLI.

Principaux paramètres de l'extension pgactive

Le tableau suivant fournit une référence pour les principaux paramètres de l'`pgactive`extension :

Paramètre	Unit	Par défaut	Description
`pgactive.conflict_logging_include_tuples`	`boolean`	–	Enregistre les informations complètes du tuple pour l'`pgactive`extension. **Note** Un redémarrage du serveur est nécessaire pour que les modifications prennent effet.
`pgactive.log_confl`	`boolean`	–	Détermine si l'`pgactive`extension enregistre

Paramètre	Unit	Par défaut	Description
`icts_to_table`			les conflits détectés dans la `pgactive.pgactive_conflict_history` table. Pour plus d'informations, consultez la section Enregistrement des conflits pour plus de détails. **Note** Un redémarrage du serveur est nécessaire pour que les modifications prennent effet.

Paramètre	Unit	Par défaut	Description
`pgactive.log_conflicts_to_logfile`	`boolean`	–	Détermine si l'`pgactive`extension enregistre les conflits détectés dans le fichier journal PostgreSQL. Pour plus d'informations, consultez la section Enregistrement des conflits pour plus de détails. **Note** Un redémarrage du serveur est nécessaire pour que les modifications prennent effet.

Paramètre	Unit	Par défaut	Description
`pgactive.synchronous_commit`	`boolean`	off	Détermine le comportement de validation pour pgactive apply workers. Lorsque cette option est désactivée (désactivée), les applications exécutent des validations asynchrones, ce qui améliore le débit de PostgreSQL lors des opérations d'application, mais retarde les confirmations de rediffusion en amont. Le paramétrer sur `off` est toujours sûr et n'entraînera pas de perte ou de saut de transaction. Ce paramètre affecte uniquement le calendrier des vidanges de disque sur le

Paramètre	Unit	Par défaut	Description
			nœud en aval et le moment où les confirmations sont envoyées en amont. Le système retarde l'envoi des confirmations de rediffusion jusqu'à ce que les validations soient vidées sur le disque par le biais d'opérations indépendantes, telles que des points de contrôle ou des travaux périodiques. Toutefois, si l'amont contient l'aval dans la liste`synchronous_standby_names` , le paramétrage de ce paramètre `off` entraîne un délai plus long pour que les validations synchrones en amont mettent plus de temps à signaler le

Paramètre	Unit	Par défaut	Description
			succès au client. Dans ce cas, définissez le paramètre suron. **Note** Même lorsque ce paramètre est défini sur un nœud répertorié dans `synchronous_standby_names`, des conflits de réplication peuvent toujours se produire dans les configurations active-active. on Cela est dû au fait

Paramètre	Unit	Par défaut	Description
			que le système ne dispose pas de verrouillage entre nœuds et de gestion globale des instantanés, ce qui permet à des transactions simultanées sur différents nœuds de modifier le même tuple. De plus, les transactions ne commencent la réplication

Paramètre	Unit	Par défaut	Description
			qu'après avoir été validées sur le nœud en amont. L'activation de la validation synchrone ne transforme pas l'extension pgactive en un système toujours cohérent.

Paramètre	Unit	Par défaut	Description
`pgactive.temp_dump_directory`	`string`	–	Définit le chemin de stockage temporaire requis pour les opérations de clonage de base de données lors de la configuration initiale. Ce répertoire doit être accessible en écriture par l'utilisateur postgres et disposer d'un espace de stockage suffisant pour contenir un vidage complet de la base de données. Le système utilise cet emplacement uniquement lors de la configuration initiale de la base de données avec des opérations de copie logique. Ce paramètre n'est pas utilisé par

Paramètre	Unit	Par défaut	Description
			lepgactive_init_copy command.

Paramètre	Unit	Par défaut	Description
`pgactive.max_ddl_lock_delay`	`milliseconds`	`-1`	Spécifie le temps d'attente maximal pour le verrouillage DDL avant l'abandon forcé de transactions d'écriture simultanées. La valeur par défaut est `-1`, qui adopte la valeur définie dans `max_standby_streaming_delay` . Ce paramètre accepte les unités de temps. Par exemple, vous pouvez le régler sur 10 secondes pendant 10 secondes. Pendant cette période d'attente, le système tente d'acquérir des verrous DDL en attendant que les transactions d'écriture en cours soient validées ou

Paramètre	Unit	Par défaut	Description
			annulées. Pour plus d'informations, consultez la section Verrouillage DDL.

Paramètre	Unit	Par défaut	Description
`pgactive.ddl_lock_timeout`	`milliseconds`	`-1`	Spécifie le temps d'attente d'une tentative de verrouillage DDL pour obtenir le verrou. La valeur par défaut est `-1`, qui utilise la valeur spécifiée dans lock_timeout. Vous pouvez définir ce paramètre à l'aide d'unités de temps telles que 10 s pendant 10 secondes. Ce temporisateur contrôle uniquement le délai d'attente pour obtenir un verrou DDL. Une fois que le système obtient le verrou et commence l'opération DDL, le temporisateur s'arrête. Ce paramètre ne limite pas la durée totale pendant laquelle

Paramètre	Unit	Par défaut	Description
			un verrou DDL peut être maintenu ni le temps global de fonctionnement du DDL. Pour contrôler la durée totale de l'opération, utilisez `statement_timeout` plutôt. Pour plus d'informations, consultez la section Verrouillage DDL.

Paramètre	Unit	Par défaut	Description
`pgactive.debug_trace_ddl_locks_level`	`boolean`	–	Remplace le niveau du journal de débogage par défaut pour les opérations de verrouillage DDL dans l'extension. `pgactive` Lorsqu'il est configuré, ce paramètre entraîne l'émission de messages relatifs au verrouillage DDL au niveau du débogage du LOG au lieu de leur niveau par défaut. Utilisez ce paramètre pour surveiller l'activité de verrouillage DDL sans activer les niveaux verbeux `DEBUG1` ou de `DEBUG2` journalisation sur l'ensemble de votre serveur. Niveaux de log disponibles, par

Paramètre	Unit	Par défaut	Description
			ordre croissant de verbosité : • none : les messages de verrouillage DDL apparaissent uniquement aux niveaux de journalisation du serveur DEBUG1 et aux niveaux inférieurs. • statement - Ajoute une sortie LOG pour les tentatives d'acquisition de verrous DDL. • acquire_release - Enregistre les événements d'acquisition, de libération, de déclinaison et les applications de nœuds homologues des verrous DDL distants.

Paramètre	Unit	Par défaut	Description
			• pairs - Fournit des informations supplémentaires sur les négociations de verrouillage DDL entre les nœuds homologues. • debug - Enregistre toutes les activités liées au verrouillage DDL au niveau LOG. Pour plus d'informations sur les options de surveillance, consultez la section Surveillance des verrous DDL globaux. **Note** Les modifications apportées à ce paramètre

Paramètre	Unit	Par défaut	Description
			prennent effet lorsque vous rechargez la configuration. Il n'est pas nécessaire de redémarrer le serveur.

Paramètres d'extension pgactive supplémentaires

Le tableau suivant présente les options de configuration internes les moins fréquemment utilisées et disponibles pour l'`pgactive`extension.

Paramètre	Unit	Par défaut	Description
`pgactive.debug_apply_delay`	`integer`	–	Définit un délai d'application (en millisecondes) pour les connexions configurées dont la saisie ne comporte pas de délai d'application explicite . `pgactive.pgactive_`

Paramètre	Unit	Par défaut	Description
			`connections` Ce délai est défini lors de la création ou de la jointure d'un nœud, et pgactive ne rejoue pas une transaction sur des nœuds homologues avant qu'au moins le nombre de millisecondes spécifié ne se soit écoulé depuis sa validation. Principalement utilisé pour simuler des réseaux à latence élevée dans des environnements de test afin de faciliter la création de conflits. Par exemple, avec un délai de 500 ms sur les nœuds A et B, vous

Paramètre	Unit	Par défaut	Description
			disposez d'au moins 500 ms pour effectuer une insertion conflictuelle sur le nœud B après avoir inséré une valeur sur le nœud A. **Note** Nécessite le rechargement du serveur ou le redémarrage d'Apply Workers pour prendre effet.

Paramètre	Unit	Par défaut	Description
pgactive.connectability_check_duration	integer	–	Spécifie la durée (en secondes) pendant laquelle un utilisateur de base de données tente d'établir des connexions en cas d'échec. Le travailleur effectue une tentative de connexion par seconde jusqu'à ce qu'il réussisse ou atteigne ce délai d'expiration. Ce paramètre est utile lorsque le moteur de base de données démarre avant que le travailleur ne soit prêt à établir des connexions.

Paramètre	Unit	Par défaut	Description
`pgactive.skip_ddl_replication`	`boolean`	on	Contrôle la manière dont les modifications DDL sont répliquées ou gérées dans Amazon RDS lorsque cette option est activée. `pgactive` Lorsqu'il est défini suron, le nœud traite les modifications DDL comme un nœud non passif. Les exigences suivantes s'appliquent lors de l'utilisation de ce paramètre : • Les nouveaux nœuds ne peuvent pas rejoindre un groupe pgactive si leur `skip_ddl_replication` valeur est différente de

Paramètre	Unit	Par défaut	Description
			celle du nœud en amont. • Les nœuds existants ne peuvent pas démarrer pgactive workers si la valeur de leurs paramètres ne correspon d pas à celle du nœud en amont. • Tous les membres de pgactive doivent utiliser la même valeur de paramètre. Vous pouvez modifier ce paramètre de deux manières avec les privilège s de super utilisateur : globalement, localement (au niveau de la session).

Paramètre	Unit	Par défaut	Description
			Note Une modification incorrecte de ce paramètre peut interrompre vos configurations de réplication.

Paramètre	Unit	Par défaut	Description
`pgactive.do_not_replicate`	`boolean`	–	Ce paramètre est réservé à un usage interne. Lorsque vous définissez ce paramètre dans une transaction, les modifications ne sont pas répliquées sur les autres nœuds de votre cluster de base de données. **Note** Une modification incorrecte de ce paramètre peut interrompre vos configurations de réplication.

Paramètre	Unit	Par défaut	Description
`pgactive.discard_mismatched_row_attributes`	`boolean`	–	Ce paramètre est destiné à un usage spécialisé uniquement. Nous recommandons d'utiliser ce paramètre uniquement pour résoudre des problèmes de réplication spécifiques. Utilisez ce paramètre lorsque : • Le flux de réplication entrant contient des lignes comportant plus de colonnes que votre table locale. • Ces lignes distantes contiennent des valeurs non nulles.

Paramètre	Unit	Par défaut	Description
			Ce paramètre remplace le message d'erreur suivant et permet à des données de diverger afin de permettre la poursuite de la réplication : `cannot right-pad mismatched attributes; attno %u is missing in local table and remote row has non-null, non-dropped value for this attribute` **Note** Une modification incorrecte de ce paramètre peut interromp

Paramètre	Unit	Par défaut	Description
			re vos configurations de réplication.

Paramètre	Unit	Par défaut	Description
pgactive.debug_trace_replay	boolean	–	Lorsqu'il est défini suron, il émet un message de journal pour chaque action à distance appliquée en aval au processus des travailleurs. Les journaux incluent : • Type de modification • Nom de la table affectée • Nombre de modifications depuis le début de la transaction • Transaction : validation (LSN) • Horodatage de validation • Identifiant du nœud en amont • Identifiant du nœud de

Paramètre	Unit	Par défaut	Description
			transfert (le cas échéant) Les journaux capturent également les commandes DDL en file d'attente et les listes déroulant es. para> Par défaut, les journaux n'incluent pas le contenu des champs de ligne. Pour inclure des valeurs de ligne dans les journaux, vous devez recompiler avec les indicateurs suivants activés : • ENCART DÉTAILLÉ • MISE À JOUR DÉTAILLÉE • VERBOSE_DELETE

Paramètre	Unit	Par défaut	Description
			Note L'activation de ce paramètre de journalisation peut avoir un impact sur les performances. Nous vous recommandons de l'activer uniquement lorsque cela est nécessaire pour le dépannage. Les modifications apportées à ce paramètre prennent effet lorsque vous

Paramètre	Unit	Par défaut	Description
			rechargez la configuration. Il n'est pas nécessaire de redémarrer le serveur.

Paramètre	Unit	Par défaut	Description
pgactive.extra_apply_connection_options		–	Vous pouvez configurer les paramètres de connexion pour toutes les connexions de nœuds homologues avec des nœuds pgactive. Ces paramètres contrôlent les paramètres tels que les modes keepalives et SSL. Par défaut, pgactive utilise les paramètres de connexion suivants : • connect_timeout=30 • Kepalives=1 • keepalives_idle=20 • keepalives_interval = 20 • keepalives_count=5 Pour remplacer les paramètres par défaut,

Paramètre	Unit	Par défaut	Description
			utilisez la commande similaire suivante : `pgactive.extra_apply_connection_options = 'keepalives=0'` Les chaînes de connexion des nœuds individuels ont priorité sur ces paramètres et sur les options de connexion intégrées de pgactive. Pour plus d'informations sur les formats de chaînes de connexion, consultez les chaînes de connexion libpq. Nous vous recommandons de conserver les paramètres keepalive par

Paramètre	Unit	Par défaut	Description
			défaut activés. Désactivez Keepalives uniquement si vous rencontrez des problèmes lors de transactions importantes effectuées sur des réseaux peu fiables. **Note** Nous vous recommandons de conserver les paramètres keepalive par défaut activés. Désactivez Keepalives uniquement si vous rencontrez des problèmes

Paramètre	Unit	Par défaut	Description
			lors de transactions importantes effectuées sur des réseaux peu fiables. Les modifications apportées à ce paramètre prennent effet lorsque vous rechargez la configuration. Il n'est pas nécessaire de redémarrer le serveur.

Paramètre	Unit	Par défaut	Description
`pgactive.init_node_parallel_jobs (int)`		–	Spécifie le nombre de tâches parallèles que `pg_dump` et `pg_restore` peuvent utiliser lors des jointures de nœuds logiques avec la `pgactive.pgactive_join_group` fonction. Les modifications apportées à ce paramètre prennent effet lorsque vous rechargez la configuration. Il n'est pas nécessaire de redémarrer le serveur.

Paramètre	Unit	Par défaut	Description
`pgactive.max_nodes`	`int`	4	Spécifie le nombre maximum de nœuds autorisés dans un groupe d'extension pgactive. La valeur par défaut est de 4 nœuds. Vous devez tenir compte des points suivants lorsque vous définissez la valeur de ce paramètre : • Tous les nœuds d'un groupe d'extensions pgactive doivent utiliser la même valeur de paramètre. • Un nouveau nœud ne peut pas se joindre si la valeur de ses paramètres est différente de celle du nœud en amont.

Paramètre	Unit	Par défaut	Description
			• Les nœuds existants ne peuvent pas démarrer les programmes d'extension pgactive si la valeur de leurs paramètres est différente de celle du nœud en amont. • Les groupes plus importants nécessitent des efforts de surveillance et de maintenance supplémentaires, donc évaluez judicieusement la valeur de ce paramètre. Vous pouvez définir ce paramètre de deux manières : dans le fichier de configuration, à l'aide de la `ALTER SYSTEM SET` commande

Paramètre	Unit	Par défaut	Description
			La valeur par défaut de ce paramètre est4, ce qui signifie qu'il peut y avoir un maximum de 4 nœuds autorisés dans le groupe d'`pgactive`extensions à tout moment. **Note** La modification prend effet après le redémarrage du serveur.

Paramètre	Unit	Par défaut	Description
`pgactive.permit_node_identifier_getter_function_creation`	`boolean`	–	Ce paramètre est destiné à un usage interne uniquement. Lorsqu'elle est activée, `pgactive` l'extension permet de créer la fonction pgactive node identifier getter.

Comprendre les conflits actifs-actifs

Lorsque vous utilisez pgactive en mode actif-actif, l'écriture dans les mêmes tables à partir de plusieurs nœuds peut créer des conflits de données. Alors que certains systèmes de clustering utilisent des verrous distribués pour empêcher les accès simultanés, pgactive adopte une approche optimiste qui convient mieux aux applications distribuées géographiquement.

Certains systèmes de clustering de bases de données empêchent l'accès simultané aux données en utilisant des verrous distribués. Bien que cette approche fonctionne lorsque les serveurs sont à proximité, elle ne prend pas en charge les applications distribuées géographiquement car elle nécessite une latence extrêmement faible pour de bonnes performances. Au lieu d'utiliser des verrous distribués (approche pessimiste), l'extension pgactive utilise une approche optimiste. Cela signifie que :

- Vous aide à éviter les conflits dans la mesure du possible.
- Permet à certains types de conflits de se produire.
- Permet de résoudre les conflits lorsque des conflits surviennent.

Cette approche vous donne plus de flexibilité lors de la création d'applications distribuées.

Comment se produisent les conflits

Les conflits entre nœuds proviennent de séquences d'événements qui ne pourraient pas se produire si toutes les transactions impliquées se produisaient simultanément sur le même nœud. Comme les nœuds n'échangent des modifications qu'une fois les transactions validées, chaque transaction est valide individuellement sur le nœud sur lequel elle a été validée, mais ne le serait pas si elle était exécutée sur un autre nœud ayant effectué un autre travail dans l'intervalle. Comme pgactive apply rejoue essentiellement la transaction sur les autres nœuds, l'opération de réexécution peut échouer en cas de conflit entre une transaction appliquée et une transaction validée sur le nœud récepteur.

La raison pour laquelle la plupart des conflits ne peuvent pas se produire lorsque toutes les transactions sont exécutées sur un seul nœud est que PostgreSQL dispose de mécanismes de communication entre transactions pour les empêcher, notamment :

- Des index UNIQUES
- SEQUENCEs
- Verrouillage des lignes et des relations
- Suivi des dépendances SÉRIALISABLE

Tous ces mécanismes sont des moyens de communiquer entre les transactions afin d'éviter les problèmes de simultanéité indésirables.

pgactive atteint une faible latence et gère bien les partitions réseau car il n'utilise pas de gestionnaire de transactions distribué ni de gestionnaire de verrouillage. Cependant, cela signifie que les transactions sur différents nœuds s'exécutent de manière totalement isolée les unes des autres. Bien que l'isolation améliore généralement la cohérence de la base de données, dans ce cas, vous devez réduire l'isolation pour éviter les conflits.

Types de conflits

Les conflits qui peuvent survenir incluent :

Rubriques

- Conflits de clé primaire ou unique
- Conflits entre INSERTION ET INSERTION
- INSERTsqui violent plusieurs contraintes UNIQUES
- Conflits de mise à jour/mise à jour
- METTRE À JOUR les conflits sur la CLÉ PRIMAIRE

- UPDATEsqui violent plusieurs contraintes UNIQUES
- Conflits de mise à jour/suppression
- Conflits d'INSERTION ET DE MISE
- Conflits SUPPRIMER/SUPPRIMER
- Conflits de contraintes de clé étrangère
- Conflits de contraintes d'exclusion
- Conflits de données mondiaux
- Conflits de verrouillage et abandons les blocages
- Conflits divergents

Conflits de clé primaire ou unique

Les conflits de lignes se produisent lorsque plusieurs opérations tentent de modifier la même clé de ligne d'une manière impossible sur un seul nœud. Ces conflits constituent le type de conflit de données le plus courant.

pgactive résout les conflits détectés par le biais de last-update-wins la gestion ou de votre gestionnaire de conflits personnalisé.

Les conflits de lignes incluent :

- INSÉRER ou INSÉRER
- INSÉRER ou METTRE À JOUR
- METTRE À JOUR ou SUPPRIMER
- INSÉRER ou SUPPRIMER
- SUPPRIMER ou SUPPRIMER
- INSÉRER ou SUPPRIMER

Conflits entre INSERTION ET INSERTION

Ce conflit le plus courant se produit lorsque, INSERTs sur deux nœuds différents, créez un tuple avec les mêmes valeurs de CLÉ PRIMAIRE (ou des valeurs de contrainte UNIQUE identiques lorsqu'aucune CLÉ PRIMAIRE n'existe).

pgactivelink résout les conflits INSERT en utilisant l'horodatage de l'hôte d'origine pour conserver le tuple le plus récent. Vous pouvez remplacer ce comportement par défaut à l'aide de votre

gestionnaire de conflits personnalisé. Bien que ce processus ne nécessite aucune action particulière de l'administrateur, sachez que pgactivelink ignore l'une des opérations INSERT sur tous les nœuds. Aucune fusion automatique de données ne se produit à moins que votre gestionnaire personnalisé ne l'implémente.

Le pgactivelink ne peut résoudre que les conflits impliquant une seule violation de contrainte. Si un INSERT enfreint plusieurs contraintes UNIQUE, vous devez mettre en œuvre des stratégies de résolution de conflits supplémentaires.

INSERTsqui violent plusieurs contraintes UNIQUES

Un conflit INSERT/INSERT peut violer plusieurs contraintes UNIQUE, y compris la CLÉ PRIMAIRE. pgactivelink ne peut gérer que les conflits impliquant une seule contrainte UNIQUE. Lorsque les conflits violent plusieurs contraintes UNIQUE, le programme d'application échoue et renvoie le message d'erreur suivant :

```
multiple unique constraints violated by remotely INSERTed tuple.
```

Dans les anciennes versions, cette situation générait plutôt une erreur de « conflit d'unicité divergente ».

Pour résoudre ces conflits, vous devez effectuer une action manuelle. SUPPRIMEZ les tuples locaux en conflit ou METTEZ-les à jour pour supprimer les conflits avec le nouveau tuple distant. Sachez que vous devrez peut-être résoudre plusieurs tuples en conflit. À l'heure actuelle, pgactivelink ne fournit aucune fonctionnalité intégrée permettant d'ignorer, de supprimer ou de fusionner les tuples qui violent plusieurs contraintes uniques.

> **Note**
>
> Pour plus d'informations, consultez la section UPDATEs relative à la violation de plusieurs contraintes UNIQUE.

Conflits de mise à jour/mise à jour

Ce conflit se produit lorsque deux nœuds modifient simultanément le même tuple sans changer sa CLÉ PRIMAIRE. pgactivelink résout ces conflits en utilisant la last-update-wins logique ou votre gestionnaire de conflits personnalisé, s'il est défini. Une CLÉ PRIMAIRE est essentielle pour l'appariement des tuples et la résolution des conflits. Pour les tables sans CLÉ PRIMAIRE, pgactivelink rejette les opérations UPDATE avec l'erreur suivante :

```
Cannot run UPDATE or DELETE on table (tablename) because it does not have a
primary key.
```

METTRE À JOUR les conflits sur la CLÉ PRIMAIRE

pgactive a des limites lors de la gestion des mises à jour de PRIMARY KEY. Bien que vous puissiez effectuer une opération de mise à jour sur une CLÉ PRIMAIRE, pgactive ne peut pas résoudre automatiquement les conflits en utilisant last-update-wins la logique de ces opérations. Vous devez vous assurer que les mises à jour de votre CLÉ PRIMAIRE n'entrent pas en conflit avec les valeurs existantes. Si des conflits surviennent lors des mises à jour de la CLÉ PRIMAIRE, ils deviennent des conflits divergents qui nécessitent une intervention manuelle de votre part. Pour plus d'informations sur la gestion de ces situations, consultezConflits divergents.

UPDATEsqui violent plusieurs contraintes UNIQUES

pgactivelink ne peut pas appliquer de résolution de last-update-wins conflit lorsqu'une mise à jour entrante viole plusieurs contraintes UNIQUES ou valeurs PRIMARY KEY. Ce comportement est similaire à celui des opérations INSERT comportant plusieurs violations de contraintes. Ces situations créent des conflits divergents qui nécessitent une intervention manuelle de votre part. Pour de plus amples informations, veuillez consulter Conflits divergents.

Conflits de mise à jour/suppression

Ces conflits se produisent lorsqu'un nœud UPDATEs par ligne et qu'un autre nœud le DELETEs fait simultanément. Dans ce cas, un conflit UPDATE/DELETE se produit lors de la rediffusion. La solution consiste à ignorer toute mise à jour qui arrive après un DELETE, sauf indication contraire de votre gestionnaire de conflits personnalisé.

pgactivelink a besoin d'une CLÉ PRIMAIRE pour faire correspondre les tuples et résoudre les conflits. Pour les tables dépourvues de CLÉ PRIMAIRE, il rejette les opérations DELETE avec l'erreur suivante :

```
Cannot run UPDATE or DELETE on table (tablename) because it does not have a
primary key.
```

> **Note**
>
> pgactivelink ne peut pas faire la distinction entre les conflits. UPDATE/DELETE and INSERT/UPDATE Dans les deux cas, une mise à jour affecte une ligne inexistante. En raison de la réplication asynchrone et de l'absence d'ordre de réexécution entre les nœuds, pgactivelink

ne peut pas déterminer si la mise à jour concerne une nouvelle ligne (INSERT non encore reçu) ou une ligne supprimée. Dans les deux scénarios, pgactivelink ignore la mise à jour.

Conflits d'INSERTION ET DE MISE

Ce conflit peut se produire dans des environnements à nœuds multiples. Cela se produit lorsqu' INSERTs un nœud par ligne, un deuxième nœud UPDATEs et un troisième nœud reçoivent la mise à jour avant l'INSERT d'origine. Par défaut, pgactivelink résout ces conflits en supprimant la mise à jour, sauf indication contraire de votre déclencheur de conflit personnalisé. Sachez que cette méthode de résolution peut entraîner des incohérences de données entre les nœuds. Pour plus d'informations sur des scénarios similaires et leur gestion, consultezConflits de mise à jour/suppression.

Conflits SUPPRIMER/SUPPRIMER

Ce conflit se produit lorsque deux nœuds différents suppriment simultanément le même tuple. pgactivelink considère que ces conflits sont inoffensifs car les deux opérations DELETE ont le même résultat final. Dans ce scénario, pgactivelink ignore en toute sécurité l'une des opérations DELETE sans affecter la cohérence des données.

Conflits de contraintes de clé étrangère

Les contraintes FOREIGN KEY peuvent provoquer des conflits lors de l'application de transactions distantes à des données locales existantes. Ces conflits se produisent généralement lorsque les transactions sont appliquées dans un ordre différent de leur ordre logique sur les nœuds d'origine.

Par défaut, pgactive applique les modifications avec session_replication_role as`replica`, ce qui permet de contourner les vérifications par clé étrangère lors de la réplication. Dans les configurations active-active, cela peut entraîner des violations de clés étrangères. La plupart des violations sont temporaires et sont résolues une fois que la réplication a été rattrapée. Cependant, des clés étrangères peuvent se produire parce que pgactive ne prend pas en charge le verrouillage des lignes entre nœuds.

Ce comportement est inhérent aux systèmes actifs-actifs asynchrones tolérants aux partitions. Par exemple, le nœud A peut insérer une nouvelle ligne enfant tandis que le nœud B supprime simultanément sa ligne parent. Le système ne peut pas empêcher ce type de modification simultanée entre les nœuds.

Pour minimiser les conflits de clés étrangères, nous recommandons ce qui suit :

- Limitez les relations de clé étrangère aux entités étroitement liées.
- Modifiez les entités associées à partir d'un seul nœud lorsque cela est possible.
- Choisissez des entités qui nécessitent rarement des modifications.
- Mettez en œuvre un contrôle de simultanéité au niveau de l'application pour les modifications.

Conflits de contraintes d'exclusion

pgactive link ne supporte pas les contraintes d'exclusion et restreint leur création.

> **Note**
>
> Si vous convertissez une base de données autonome existante en base de données pgactivelink, supprimez manuellement toutes les contraintes d'exclusion.

Dans un système asynchrone distribué, il n'est pas possible de garantir qu'aucun ensemble de lignes ne viole la contrainte. Cela est dû au fait que toutes les transactions sur les différents nœuds sont totalement isolées. Les contraintes d'exclusion peuvent entraîner des blocages de rediffusion, dans lesquels la rediffusion ne peut pas progresser d'un nœud à l'autre en raison de violations des contraintes d'exclusion.

Si vous forcez pgactive Link à créer une contrainte d'exclusion, ou si vous ne supprimez pas les contraintes existantes lors de la conversion d'une base de données autonome en pgactive Link, la réplication risque de s'interrompre. Pour rétablir la progression de la réplication, supprimez ou modifiez les tuples locaux qui entrent en conflit avec un tuple distant entrant afin que la transaction distante puisse être appliquée.

Conflits de données mondiaux

Lorsque vous utilisez pgactivelink, des conflits peuvent survenir lorsque les nœuds possèdent des données globales PostgreSQL différentes à l'échelle du système, telles que des rôles. Ces conflits peuvent entraîner la réussite et la validation d'opérations (principalement DDL) sur un nœud, mais ne pas s'appliquer aux autres nœuds.

Si un utilisateur existe sur un nœud mais pas sur un autre, des problèmes de réplication peuvent survenir :

- Node1 a un nom d'utilisateur`fred`, mais cet utilisateur n'existe pas sur Node2
- Lors de `fred` la création d'une table sur Node1, la table est répliquée `fred` en tant que propriétaire

- Lorsque cette commande DDL est appliquée à Node2, elle échoue car l'utilisateur `fred` n'existe pas
- Cet échec génère une erreur dans les journaux PostgreSQL sur Node2 et incrémente le compteur `pgactive.pgactive_stats.nr_rollbacks`

Résolution : créez l'utilisateur `fred` sur Node2. L'utilisateur n'a pas besoin d'autorisations identiques mais doit exister sur les deux nœuds.

Si une table existe sur un nœud mais pas sur un autre, les opérations de modification des données échoueront :

- Node1 possède une table nommée `foo` qui n'existe pas sur Node2
- Toute opération DML sur la `foo` table du Node1 échouera une fois répliquée sur Node2

Résolution : créez la table `foo` sur Node2 avec la même structure.

> **Note**
>
> pgactivelink ne réplique actuellement pas les commandes CREATE USER ni les opérations DDL. La réplication DDL est prévue pour une future version.

Conflits de verrouillage et abandons les blocages

Comme les processus pgactive apply fonctionnent comme des sessions utilisateur normales, ils suivent les règles standard de verrouillage des lignes et des tables. Cela peut amener les processus d'application de pgactivelink à attendre les verrous détenus par les transactions des utilisateurs ou par d'autres processus d'application.

Les types de verrous suivants peuvent affecter les processus d'application :

- Verrouillage explicite au niveau de la table (LOCK TABLE...) par les sessions utilisateur
- Verrouillage explicite au niveau des lignes (SELECT... (POUR LA MISE À JOUR/POUR LE PARTAGE) par sessions utilisateur
- Verrouillage à partir de clés étrangères
- Verrouillage implicite dû à une ligne UPDATEs INSERTs DELETEs, ou à une activité locale ou à une application depuis d'autres serveurs

Des blocages peuvent survenir entre :

- Un processus d'application pgactivelink et une transaction utilisateur
- Deux processus d'application

Lorsque des blocages se produisent, le détecteur de blocage de PostgreSQL met fin à l'une des transactions problématiques. Si le processus du worker d'application pgactivelink est arrêté, il réessaie automatiquement et réussit généralement.

> **Note**
>
> - Ces problèmes sont temporaires et ne nécessitent généralement pas l'intervention de l'administrateur. Si un processus d'application est bloqué pendant une période prolongée par le verrouillage d'une session utilisateur inactive, vous pouvez mettre fin à la session utilisateur pour reprendre la réplication. Cette situation est similaire à celle où un utilisateur détient un long verrou qui affecte une autre session utilisateur.
> - Pour identifier les délais de rediffusion liés au verrouillage, activez cette fonctionnalité `log_lock_waits` dans PostgreSQL.

Conflits divergents

Des conflits divergents se produisent lorsque des données qui devraient être identiques entre les nœuds diffèrent de façon inattendue. Bien que ces conflits ne devraient pas se produire, ils ne peuvent pas tous être évités de manière fiable dans le cadre de la mise en œuvre actuelle.

> **Note**
>
> La modification de la CLÉ PRIMAIRE d'une ligne peut provoquer des conflits divergents si un autre nœud modifie la clé de la même ligne avant que tous les nœuds n'aient traité la modification. Évitez de modifier les clés primaires ou limitez les modifications à un nœud désigné. Pour de plus amples informations, veuillez consulter the section called “METTRE À JOUR les conflits sur la CLÉ PRIMAIRE ”.

Les conflits divergents impliquant des données de ligne nécessitent généralement l'intervention de l'administrateur. Pour résoudre ces conflits, vous devez ajuster manuellement les données d'un nœud

pour qu'elles correspondent à celles d'un autre tout en désactivant temporairement la réplication à l'aide `pgactive.pgactive_do_not_replicate` de. Ces conflits ne devraient pas se produire lorsque vous utilisez pgactive comme indiqué et que vous évitez les paramètres ou les fonctions marqués comme dangereux.

En tant qu'administrateur, vous devez résoudre ces conflits manuellement. Selon le type de conflit, vous devrez utiliser des options avancées telles que`pgactive.pgactive_do_not_replicate`. Utilisez ces options avec prudence, car une mauvaise utilisation peut aggraver la situation. En raison de la variété des conflits possibles, nous ne pouvons pas fournir d'instructions de résolution universelles.

Des conflits divergents se produisent lorsque des données qui devraient être identiques sur différents nœuds diffèrent de façon inattendue. Bien que ces conflits ne devraient pas se produire, ils ne peuvent pas tous être évités de manière fiable dans le cadre de la mise en œuvre actuelle.

Éviter ou tolérer les conflits

Dans la plupart des cas, vous pouvez utiliser une conception d'application appropriée pour éviter les conflits ou faire en sorte que votre application tolère les conflits.

Les conflits se produisent uniquement lorsque des opérations simultanées ont lieu sur plusieurs nœuds. Pour éviter les conflits :

- Écrire sur un seul nœud
- Écrire dans des sous-ensembles de base de données indépendants sur chaque nœud (par exemple, attribuer à chaque nœud un schéma distinct)

Pour les conflits INSERT et INSERT, utilisez des séquences globales pour éviter complètement les conflits.

Si les conflits ne sont pas acceptables pour votre cas d'utilisation, envisagez d'implémenter le verrouillage distribué au niveau de l'application. Souvent, la meilleure approche consiste à concevoir votre application pour qu'elle fonctionne avec les mécanismes de résolution de conflits de pgactive plutôt que d'essayer de prévenir tous les conflits. Pour de plus amples informations, veuillez consulter the section called “Types de conflits”.

Enregistrement des conflits

pgactivelink enregistre les incidents de conflit dans le `pgactive.pgactive_conflict_history` tableau pour vous aider à diagnostiquer et à gérer les conflits actifs-actifs.

L'enregistrement des conflits dans cette table ne se produit que lorsque vous définissez `pgactive.log_conflicts_to_table` le paramètre sur true. L'extension pgactive enregistre également les conflits dans le fichier journal de PostgreSQL lorsque log_min_messages est défini sur ou, quel que soit le paramètre. `LOG` `lower` `pgactive.log_conflicts_to_table`

Utilisez le tableau de l'historique des conflits pour :

- Mesurez la fréquence à laquelle votre application crée des conflits
- Identifiez les endroits où les conflits se produisent
- Améliorez votre application pour réduire les taux de conflit
- Détectez les cas où la résolution des conflits ne produit pas les résultats souhaités
- Déterminez les domaines dans lesquels vous avez besoin de déclencheurs de conflits définis par l'utilisateur ou de modifications de conception d'applications

Pour les conflits de lignes, vous pouvez éventuellement enregistrer les valeurs des lignes. Ceci est contrôlé par le `pgactive.log_conflicts_to_table` réglage. Remarque :

- Il s'agit d'une option globale à l'échelle de la base de données
- Il n'existe aucun contrôle par table sur la journalisation des valeurs des lignes
- Aucune limite n'est appliquée aux numéros de champs, aux éléments du tableau ou à la longueur des champs
- L'activation de cette fonctionnalité peut être déconseillée si vous travaillez avec des lignes de plusieurs mégaoctets susceptibles de déclencher des conflits

Étant donné que la table d'historique des conflits contient les données de chaque table de la base de données (chacune avec des schémas potentiellement différents), les valeurs des lignes enregistrées sont stockées sous forme de champs JSON. Le JSON est créé en utilisant`row_to_json`, comme si vous l'appeliez directement depuis SQL. PostgreSQL ne fournit pas de fonction, vous aurez donc besoin `json_to_row` d'un code spécifique à la table (PL/pgSQL, PL/Python, PL/Perlin, etc.) pour reconstruire un tuple de type composite à partir du JSON enregistré.

> **Note**
>
> Support des conflits définis par l'utilisateur dans le cadre d'une future fonctionnalité d'extension.

Comprendre le schéma pgactive

Le schéma pgactive gère la réplication active-active dans RDS pour PostgreSQL. Ce schéma contient des tables qui stockent les informations de configuration et d'état de la réplication.

> **Note**
>
> Le schéma pgactive évolue et est sujet à modification. Ne modifiez pas directement les données de ces tables.

Les tables clés du schéma pgactive sont les suivantes :

- `pgactive_nodes`— Stocke les informations relatives aux nœuds du groupe de réplication actif-actif.
- `pgactive_connections`— Stocke les détails de connexion pour chaque nœud.

pgactive_nodes

Le pgactive_nodes stocke des informations sur les nœuds participant au groupe de réplication actif-actif.

Colonne	Type	Classement (Collation)	Nullable	Par défaut
node_sysid	text	–	non nul	–
node_timeline	oid	–	non nul	–
node_dboid	oid	–	non nul	–
statut_nœud	char	–	non nul	–
nom_nœud	text	–	non nul	–
node_dsn	text	–	non nul	–
node_init_from_dsn	text	–	non nul	–

Colonne	Type	Classement (Collation)	Nullable	Par défaut
nœud en lecture seule	boolean	–	–	false
node_seq_id	smallint	–	non nul	–

node_sysid

ID unique pour un nœud, généré pendant `pgactive_create_group` ou `pgactive_join_group`

statut_nœud

État de préparation du nœud :

- b - début de la configuration
- i - initialisation
- c - rattrapage
- o - création de slots sortants
- r - prêt
- k - tué

> **Note**
>
> Cette colonne n'indique pas si un nœud est connecté ou déconnecté.

nom_nœud

Nom de nœud unique fourni par l'utilisateur.

node_dsn

Chaîne de connexion ou nom de mappage utilisateur

node_init_from_dsn

DSN à partir duquel ce nœud a été créé.

pgactive_connection

Le pgactive_connections stocke les détails de connexion pour chaque nœud.

Colonne	Type	Classement (Collation)	Nullable	Par défaut	
conn_sysid	text	none	non nul	none	
conn_timeline	oid	none	non nul	none	
conn_dboid	oid	none	non nul	none	
conn_dsn	text	none	non nul	none	
conn_apply_delay	entier	none	none	none	
conn_replication_sets	text	none	none	none	

conn_sysid

Identifiant de nœud pour le nœud auquel cette entrée fait référence.

conn_dsn

Identique à pgactive.pgactive_nodesnode_dsn.

conn_apply_delay

Si cette option est définie, il faut attendre quelques millisecondes avant d'appliquer chaque transaction depuis le nœud distant. Principalement pour le débogage. Si la valeur est nulle, la valeur globale par défaut s'applique.

Utilisation d'ensembles de réplication

Les ensembles de réplication déterminent les tables à inclure ou à exclure des opérations de réplication. Par défaut, toutes les tables sont répliquées sauf indication contraire à l'aide des fonctions suivantes :

- `pgactive_exclude_table_replication_set()`- Exclut les tables spécifiées de la réplication
- `pgactive_include_table_replication_set()`- Inclut les tables spécifiées dans la réplication

> **Note**
>
> Avant de configurer des ensembles de réplication, tenez compte des points suivants :
>
> - Vous pouvez configurer l'inclusion ou l'exclusion de tables uniquement après l'exécution`pgactive_create_group()`, mais avant`pgactive_join_group()`.
> - Après utilisation`pgactive_exclude_table_replication_set()`, vous ne pouvez plus utiliser`pgactive_include_table_replication_set()`.
> - Après utilisation`pgactive_include_table_replication_set()`, vous ne pouvez plus utiliser`pgactive_exclude_table_replication_set()`.

Le système gère les tables nouvellement créées différemment en fonction de votre configuration initiale :

- Si vous avez exclu des tables : toutes les nouvelles tables créées par la suite `pgactive_join_group()` sont automatiquement incluses dans la réplication
- Si vous avez inclus des tables : toutes les nouvelles tables créées par la suite `pgactive_join_group()` sont automatiquement exclues de la réplication.

Pour afficher la configuration du jeu de réplication pour une table spécifique, utilisez la `pgactive.pgactive_get_table_replication_sets()` fonction.

référence des fonctions pgactive

Vous trouverez ci-dessous une liste des fonctions pgactive avec leurs paramètres, leurs valeurs renvoyées, ainsi que des notes d'utilisation pratiques pour vous aider à les utiliser efficacement :

get_last_applied_xact_info

Récupère les dernières informations de transaction appliquées pour un nœud spécifié.

Arguments

- sysid (texte) - OID de chronologie

- dboid (OID)

Type de retour

Il enregistre les éléments suivants :

- last_applied_xact_id (OID)
- last_applied_xact_committs (horodatage avec fuseau horaire)
- last_applied_xact_at (horodatage avec fuseau horaire)

Notes d’utilisation

Utilisez cette fonction pour récupérer les dernières informations de transaction appliquées pour un nœud spécifié.

pgactive_apply_pause

Suspend le processus d'application de la réplication.

Arguments

Aucun

Type de retour

boolean

Notes d’utilisation

Appelez cette fonction pour suspendre le processus d'application de la réplication.

pgactive_apply_resume

Reprend le processus d'application de la réplication.

Arguments

Aucun

Type de retour

void

Notes d’utilisation

Appelez cette fonction pour reprendre le processus d'application de la réplication.

pgactive_is_apply_paused

Vérifie si l'application de réplication est actuellement suspendue.

Arguments

Aucun

Type de retour

boolean

Notes d’utilisation

Utilisez cette fonction pour vérifier si l'application de réplication est actuellement suspendue.

pgactive_create_group

Crée un groupe pgactive en convertissant une base de données autonome en nœud initial.

Arguments

- nom_nœud (texte)
- node_dsn (texte)
- apply_delay integer DEFAULT NULL : :integer - replication_sets text [] TABLEAU PAR DÉFAUT ['default' : :text]

Type de retour

void

Notes d’utilisation

Crée un groupe pgactive en convertissant une base de données autonome en nœud initial. La fonction effectue des vérifications d'intégrité avant de transformer le nœud en nœud pgactive. Avant d'utiliser cette fonction, assurez-vous que votre cluster PostgreSQL est `max_worker_processes` suffisamment disponible pour prendre en charge les travailleurs en arrière-plan avec pgactive.

pgactive_detach_nodes

Supprime les nœuds spécifiés du groupe pgactive.

Arguments

- p_nodes (texte [])

Type de retour

void

Notes d’utilisation

Utilisez cette fonction pour supprimer les nœuds spécifiés du groupe pgactive.

pgactive_exclude_table_replication_set

Exclut une table spécifique de la réplication.

Arguments

- p_relation (regclass)

Type de retour

void

Notes d’utilisation

Utilisez cette fonction pour exclure une table spécifique de la réplication.

pgactive_get_replication_lag_info

Récupère des informations détaillées sur le délai de réplication, notamment les détails du nœud, l'état du WAL et les valeurs LSN.

Arguments

Aucun

Type de retour

SETOF record - node_name text - node_sysid text - application_name text - slot_name text - active boolean - active_pid integer - pending_wal_decoding bigint - Taille approximative du WAL en octets à décoder sur le nœud expéditeur - pending_wal_to_apply bigint - Taille approximative du WAL en octets à appliquer au nœud récepteur - restart_lsn pg_lsn pg_lsn - confirmed_flush_lsn pg_lsn - sent_lsn pg_lsn - write_lsn pg_lsn - flush_lsn pg_lsn - replay_lsn pg_lsn

Notes d’utilisation

Appelez cette fonction pour récupérer les informations relatives au délai de réplication, notamment les détails du nœud, l'état du WAL et les valeurs LSN.

pgactive_get_stats

Récupère les statistiques de réplication passive.

Arguments

Aucun

Type de retour

Enregistrement SETOF - rep_node_id oid - rilocalid oid - riremoteid text - nr_commit bigint - nr_rollback bigint - nr_insert bigint - nr_insert_conflict bigint - nr_update bigint - nr_update_conflict bigint - nr_delete bigint - nr_delete_conflict bigint - nr_disconnect bigint

Notes d’utilisation

Utilisez cette fonction pour récupérer les statistiques de réplication de pgactive.

pgactive_get_table_replication_sets

Obtient la configuration de l'ensemble de réplication pour une relation spécifique.

Arguments

- relation (regclass)

Type de retour

Registre SETOF

Notes d’utilisation

Appelez cette fonction pour obtenir la configuration du jeu de réplication pour une relation spécifique.

pgactive_include_table_replication_set

Inclut une table spécifique dans la réplication.

Arguments

- p_relation (regclass)

Type de retour

void

Notes d’utilisation

Utilisez cette fonction pour inclure une table spécifique dans la réplication.

pgactive_join_group

Ajoute un nœud à un groupe pgactive existant.

Arguments

- nom_nœud (texte)
- node_dsn (texte)
- join_using_dsn (texte)
- apply_delay (entier, facultatif)
- replication_sets (text [], par défaut : ['default'])
- bypass_collation_check (booléen, valeur par défaut : false)
- bypass_node_identifier_creation (booléen, valeur par défaut : false)
- bypass_user_tables_check (booléen, valeur par défaut : false)

Type de retour

void

Notes d’utilisation

Appelez cette fonction pour ajouter un nœud à un groupe pgactive existant. Assurez-vous que votre cluster PostgreSQL possède suffisamment de max_worker_processes pour les travailleurs en arrière-plan de pgactive.

pgactive_remove

Supprime tous les composants pgactive du nœud local.

Arguments

- force (booléen, valeur par défaut : false)

Type de retour

void

Notes d’utilisation

Appelez cette fonction pour supprimer tous les composants pgactive du nœud local.

pgactive_snowflake_id_nextval

Génère des valeurs de séquence uniques spécifiques au nœud.

Arguments

- regclass

Type de retour

bigint

Notes d’utilisation

Utilisez cette fonction pour générer des valeurs de séquence uniques spécifiques au nœud.

pgactive_update_node_conninfo

Met à jour les informations de connexion pour un nœud pgactive.

Arguments

- node_name_to_update (texte)
- node_dsn_to_update (texte)

Type de retour

void

Notes d’utilisation

Utilisez cette fonction pour mettre à jour les informations de connexion d'un nœud pgactive.

pgactive_wait_for_node_ready

Surveille la progression des opérations de création ou de jonction de groupes.

Arguments

- délai d'attente (entier, valeur par défaut : 0)
- progress_interval (entier, par défaut : 60)

Type de retour

void

Notes d’utilisation

Appelez cette fonction pour suivre la progression des opérations de création ou de jonction de groupes.

Gestion des conflits de la réplication active-active

L'extension `pgactive` fonctionne par base de données et non par cluster. Chaque instance de base de données qui utilise `pgactive` est une instance indépendante et peut accepter les modifications de données de n'importe quelle source. Lorsqu'une modification est envoyée à une instance de base de données, PostgreSQL la valide localement, puis utilise `pgactive` pour répliquer la modification de manière asynchrone sur les autres instances de base de données. Lorsque deux instances de base de données PostgreSQL mettent à jour le même enregistrement à peu près au même moment, un conflit peut survenir.

L'extension `pgactive` propose des mécanismes de détection et de résolution automatique des conflits. Il suit l'horodatage du moment où la transaction a été validée sur les deux instances de base de données et applique automatiquement la modification avec l'horodatage le plus récent. L'extension `pgactive` journalise également la survenance d'un conflit dans la table `pgactive.pgactive_conflict_history`.

Ils `pgactive.pgactive_conflict_history` continueront de croître. Vous souhaiterez peut-être définir une politique de purge. Cela peut être fait en supprimant régulièrement certains enregistrements ou en définissant un schéma de partitionnement pour cette relation (puis en détachant, supprimant, tronquant les partitions qui vous intéressent). Pour mettre en œuvre régulièrement la politique de purge, l'une des options consiste à utiliser l'`pg_cron`extension. Consultez les informations suivantes concernant un exemple de table d'`pg_cron`historique, Planification de la maintenance avec l'extension PostgreSQL pg_cron.

Gestion des séquences dans une réplication active-active

Une instance de base de données RDS for PostgreSQL dotée de l'extension `pgactive` utilise deux mécanismes de séquence différents pour générer des valeurs uniques.

Séquences globales

Pour utiliser une séquence globale, créez une séquence locale avec l'instruction `CREATE SEQUENCE`. Utilisez `pgactive.pgactive_snowflake_id_nextval(seqname)` plutôt que `usingnextval(seqname)` pour obtenir la prochaine valeur unique de la séquence.

L'exemple suivant crée une séquence globale :

```
app=> CREATE TABLE gstest (
      id bigint primary key,
      parrot text
    );
```

```
app=>CREATE SEQUENCE gstest_id_seq OWNED BY gstest.id;
```

```
app=> ALTER TABLE gstest \
      ALTER COLUMN id SET DEFAULT \
      pgactive.pgactive_snowflake_id_nextval('gstest_id_seq');
```

Séquences partitionnées

Dans les séquences fractionnées ou partitionnées, une séquence PostgreSQL normale est utilisée sur chaque nœud. Chaque séquence s'incrémente de la même valeur et débute à des décalages différents. Par exemple, avec un pas de 100, le nœud 1 génère la séquence 101, 201, 301, etc. et le nœud 2 génère la séquence 102, 202, 302, etc. Ce mécanisme fonctionne bien même si les nœuds ne peuvent pas communiquer pendant de longues périodes. Cependant, il impose au concepteur de spécifier un nombre maximal de nœuds au moment d'établir le schéma et nécessite une configuration par nœud. Les erreurs peuvent facilement engendrer un chevauchement de séquences.

Il est relativement simple de configurer cette approche avec `pgactive` en créant la séquence souhaitée sur un nœud comme suit :

```

CREATE TABLE some_table (generated_value bigint primary key);
```

```
app=> CREATE SEQUENCE some_seq INCREMENT 100 OWNED BY some_table.generated_value;
```

```
app=> ALTER TABLE some_table ALTER COLUMN generated_value SET DEFAULT
 nextval('some_seq');
```

Appelez ensuite `setval` sur chaque nœud pour donner une valeur de départ de décalage différente comme suit.

```
app=>
-- On node 1
SELECT setval('some_seq', 1);

-- On node 2
SELECT setval('some_seq', 2);
```

Réduction du ballonnement des tables et des index avec l'extension pg_repack

Vous pouvez utiliser l'`pg_repack`extension pour supprimer la surcharge des tables et des index comme alternative à. `VACUUM FULL` Cette extension est prise en charge sur RDS les SQL versions 9.6.3 et supérieures de Postgre. Pour plus d'informations sur l'`pg_repack`extension et le réemballage complet de la table, consultez la documentation du GitHub projet.

Contrairement à cela`VACUUM FULL`, l'`pg_repack`extension ne nécessite un lock exclusif (AccessExclusiveLock) que pendant une courte période lors de l'opération de reconstruction de la table dans les cas suivants :

- Création initiale de la table de journal — Une table de journal est créée pour enregistrer les modifications survenues lors de la copie initiale des données, comme indiqué dans l'exemple suivant :

```
postgres=>\dt+ repack.log_*
List of relations
-[ RECORD 1 ]-+----------
Schema        | repack
Name          | log_16490
Type          | table
Owner         | postgres
Persistence   | permanent
Access method | heap
Size          | 65 MB
Description   |
```

- swap-and-dropPhase finale.

Pour le reste de l'opération de reconstruction, il suffit de `ACCESS SHARE` verrouiller la table d'origine pour copier des lignes de celle-ci vers la nouvelle table. Cela permet aux DELETE opérations INSERTUPDATE, et de se dérouler comme d'habitude.

Recommandations

Les recommandations suivantes s'appliquent lorsque vous supprimez le bloat des tables et des index à l'aide de l'`pg_repack`extension :

- Effectuez le reconditionnement en dehors des heures ouvrables ou pendant une période de maintenance afin de minimiser son impact sur les performances des autres activités de base de données.
- Surveillez de près les sessions bloquantes pendant l'activité de reconstruction et assurez-vous qu'aucune activité sur la table d'origine ne risque de bloquer`pg_repack`, en particulier pendant la swap-and-drop phase finale lorsqu'un verrouillage exclusif de la table d'origine est nécessaire. Pour plus d'informations, consultez la section Identification de ce qui bloque une requête.

  Lorsque vous voyez une session bloquante, vous pouvez y mettre fin à l'aide de la commande suivante après mûre réflexion. Cela permet de continuer `pg_repack` à terminer la reconstruction :

  ```
  SELECT pg_terminate_backend(pid);
  ```

- Lors de l'application des modifications accumulées à partir de la table des `pg_repack's` journaux sur des systèmes présentant un taux de transactions très élevé, le processus d'application risque de ne pas être en mesure de suivre le rythme des modifications. Dans de tels cas, `pg_repack` il ne serait pas en mesure de terminer le processus de candidature. Pour de plus amples informations, veuillez consulter Surveillance de la nouvelle table lors du reconditionnement. Si les index sont très volumineux, une autre solution consiste à effectuer un reconditionnement des index uniquement. Cela permet également aux cycles VACUUM de nettoyage des index de se terminer plus rapidement.

  Vous pouvez ignorer la phase de nettoyage de l'index à l'aide VACUUM du manuel depuis Postgre SQL version 12, et elle est automatiquement ignorée lors de l'aspiration automatique d'urgence depuis Postgre version 14. SQL Cela permet de VACUUM terminer plus rapidement sans supprimer le gonflement de l'index et n'est destiné qu'aux situations d'urgence telles que la prévention de l'VACUUMencapsulation. Pour plus d'informations, consultez la section Éviter le gonflement des index dans le guide de l'utilisateur Amazon Aurora.

Prérequis

- La table doit avoir PRIMARY KEY ou non une UNIQUE contrainte nulle.
- La version de l'extension doit être la même pour le client et le serveur.
- Assurez-vous que la taille de l'RDSinstance est `FreeStorageSpace` supérieure à la taille totale de la table sans le gonflement. Par exemple, considérez que la taille totale de la table, index TOAST et index inclus, est de 2 To, et que le volume total de la table est de 1 To. La valeur requise `FreeStorageSpace` doit être supérieure à la valeur renvoyée par le calcul suivant :

```
2TB (Table size) - 1TB (Table bloat) = 1TB
```

Vous pouvez utiliser la requête suivante pour vérifier la taille totale de la table et l'utiliser `pgstattuple` pour en déduire le gonflement. Pour plus d'informations, consultez la section Diagnostic du gonflement des tables et des index dans le guide de l'utilisateur Amazon Aurora

```
SELECT pg_size_pretty(pg_total_relation_size('table_name')) AS total_table_size;
```

Cet espace est récupéré une fois l'activité terminée.

- Assurez-vous que l'RDSinstance dispose d'une capacité de calcul et d'E/S suffisante pour gérer l'opération de reconditionnement. Vous pouvez envisager d'augmenter la classe d'instance pour un équilibre optimal des performances.

Pour utiliser l'**`pg_repack`**extension

1. Installez l'`pg_repack`extension sur votre SQL instance de base de données RDS for Postgre en exécutant la commande suivante.

   ```
   CREATE EXTENSION pg_repack;
   ```

2. Exécutez les commandes suivantes pour accorder un accès en écriture aux tables de journaux temporaires créées par`pg_repack`.

   ```
   ALTER DEFAULT PRIVILEGES IN SCHEMA repack GRANT INSERT ON TABLES TO PUBLIC;
   ALTER DEFAULT PRIVILEGES IN SCHEMA repack GRANT USAGE, SELECT ON SEQUENCES TO
    PUBLIC;
   ```

3. Connectez-vous à la base de données à l'aide de l'utilitaire `pg_repack` client. Utilisez un compte qui possède les privilèges `rds_superuser`. Par exemple, supposons que le rôle `rds_test` a les privilèges `rds_superuser`. La syntaxe suivante s'applique `pg_repack` aux tables complètes, y compris tous les index de table de la `postgres` base de données.

   ```
   pg_repack -h db-instance-name.111122223333.aws-region.rds.amazonaws.com -U rds_test
    -k postgres
   ```

> **Note**
>
> Vous devez vous connecter à l'aide de l'option -k. L'option -a n'est pas prise en charge.

La réponse du `pg_repack` client fournit des informations sur les tables de l'instance de base de données qui sont reconditionnées.

```
INFO: repacking table "pgbench_tellers"
INFO: repacking table "pgbench_accounts"
INFO: repacking table "pgbench_branches"
```

4. La syntaxe suivante réemballe une seule table, `orders` y compris les index de la base de données. `postgres`

```
pg_repack -h db-instance-name.111122223333.aws-region.rds.amazonaws.com -U rds_test
 --table orders -k postgres
```

La syntaxe suivante réemballe uniquement les index de la `orders` table dans `postgres` la base de données.

```
pg_repack -h db-instance-name.111122223333.aws-region.rds.amazonaws.com -U rds_test
 --table orders --only-indexes -k postgres
```

Surveillance de la nouvelle table lors du reconditionnement

- La taille de la base de données est augmentée de la taille totale de la table moins le surchargement, jusqu'à la swap-and-drop phase de reconditionnement. Vous pouvez surveiller le taux de croissance de la taille de la base de données, calculer la vitesse du reconditionnement et estimer approximativement le temps nécessaire pour terminer le transfert de données initial.

  Par exemple, considérez que la taille totale de la table est de 2 To, la taille de la base de données de 4 To et la charge totale de la table de 1 To. La valeur de taille totale de la base de données renvoyée par le calcul à la fin de l'opération de reconditionnement est la suivante :

  `2TB (Table size) + 4 TB (Database size) - 1TB (Table bloat) = 5TB`

Vous pouvez estimer approximativement la vitesse de l'opération de reconditionnement en échantillonnant le taux de croissance en octets entre deux points dans le temps. Si le taux de croissance est de 1 Go par minute, l'opération initiale de création de table peut prendre environ 1 000 minutes ou 16,6 heures. Outre la création initiale de la table, vous `pg_repack` devez également appliquer les modifications accumulées. Le temps nécessaire dépend du taux d'application des modifications en cours et des modifications cumulées.

> **Note**
>
> Vous pouvez utiliser `pgstattuple` l'extension pour calculer le gonflement dans le tableau. Pour plus d'informations, consultez pgstattuple.

- Le nombre de lignes de la table de `pg_repack's` journal, dans le schéma de reconditionnement, représente le volume de modifications en attente d'être appliquées à la nouvelle table après le chargement initial.

  Vous pouvez consulter la table des `pg_repack's` journaux `pg_stat_all_tables` pour surveiller les modifications appliquées à la nouvelle table. `pg_stat_all_tables.n_live_tup`indique le nombre d'enregistrements en attente d'être appliqués à la nouvelle table. Pour plus d'informations, consultez pg_stat_all_tables.

```
postgres=>SELECT relname,n_live_tup FROM pg_stat_all_tables WHERE schemaname =
 'repack' AND relname ILIKE '%log%';

-[ RECORD 1 ]---------
relname    | log_16490
n_live_tup | 2000000
```

- Vous pouvez utiliser l'`pg_stat_statements`extension pour connaître le temps nécessaire à chaque étape de l'opération de reconditionnement. Cela est utile pour préparer l'application de la même opération de reconditionnement dans un environnement de production. Vous pouvez ajuster la `LIMIT` clause pour étendre davantage la sortie.

```
postgres=>SELECT
     SUBSTR(query, 1, 100) query,
     round((round(total_exec_time::numeric, 6) / 1000 / 60),4)
 total_exec_time_in_minutes
 FROM
```

```
    pg_stat_statements
 WHERE
    query ILIKE '%repack%'
 ORDER BY
    total_exec_time DESC LIMIT 5;

 query                                                                    |
 total_exec_time_in_minutes
--------------------------------------------------------------------------
+---------------------------
 CREATE UNIQUE INDEX index_16493 ON repack.table_16490 USING btree (a) |
        6.8627
 INSERT INTO repack.table_16490 SELECT a FROM ONLY public.t1           |
        6.4150
 SELECT repack.repack_apply($1, $2, $3, $4, $5, $6)                    |
        0.5395
 SELECT repack.repack_drop($1, $2)                                     |
        0.0004
 SELECT repack.repack_swap($1)                                         |
        0.0004
(5 rows)
```

Le reconditionnement est une out-of-place opération complète, de sorte que la table d'origine n'est pas affectée et nous ne prévoyons aucun problème inattendu nécessitant la restauration de la table d'origine. Si le reconditionnement échoue de façon inattendue, vous devez rechercher la cause de l'erreur et la résoudre.

Une fois le problème résolu, supprimez et recréez l'`pg_repack`extension dans la base de données où se trouve la table, puis recommencez l'`pg_repack`étape. En outre, la disponibilité des ressources informatiques et l'accessibilité simultanée de la table jouent un rôle crucial dans la réalisation en temps voulu de l'opération de reconditionnement.

Mise à niveau et utilisation de l' PLV8extension

PLV8 est une extension de langage Javascript fiable pour PostgreSQL. Vous pouvez l'utiliser pour des procédures stockées, des déclencheurs et tout autre code procédural pouvant être appelé depuis SQL. Cette extension de langage est prise en charge par toutes les versions actuelles de PostgreSQL.

Si vous utilisez PLV8et mettez à niveau PostgreSQL vers une PLV8 nouvelle version, vous profitez immédiatement de la nouvelle extension. Procédez comme suit pour synchroniser les métadonnées de votre catalogue avec la nouvelle version de PLV8. Ces étapes sont facultatives, mais nous vous recommandons vivement de les compléter afin d'éviter des avertissements de décalage des métadonnées.

Le processus de mise à niveau supprime toutes vos PLV8 fonctions existantes. Nous vous recommandons donc de créer un instantané de votre instance de base de données RDS for PostgreSQL avant la mise à niveau. Pour de plus amples informations, veuillez consulter Création d'un instantané de base de données pour une instance de base de données mono-AZ pour Amazon RDS.

> **Important**
>
> À partir de la version 18 de PostgreSQL, Amazon RDS for PostgreSQL déconseillera les extensions et PostgreSQL. `plcoffee` `plls` Nous vous recommandons de cesser d'utiliser CoffeeScript et LiveScript dans vos applications afin de disposer d'un chemin de mise à niveau pour les futures mises à niveau des versions du moteur.

Pour synchroniser les métadonnées de votre catalogue avec une nouvelle version de PLV8

1. Vérifiez que vous devez mettre à jour. Pour ce faire, exécutez la commande suivante tout en étant connecté à votre instance.

   ```
   SELECT * FROM pg_available_extensions WHERE name IN ('plv8','plls','plcoffee');
   ```

   Si les résultats contiennent des valeurs pour une version installée avec un numéro inférieur à celui de la version par défaut, poursuivez cette procédure pour mettre à jour vos extensions. Par exemple, l'ensemble de résultats suivant indique que vous devez procéder à la mise à jour.

   ```
   name    | default_version | installed_version |                    comment
   ```

```
--------+----------------+------------------
+-------------------------------------------------
plls    | 2.1.0          | 1.5.3            | PL/LiveScript (v8) trusted
 procedural language
plcoffee| 2.1.0          | 1.5.3            | PL/CoffeeScript (v8) trusted
 procedural language
plv8    | 2.1.0          | 1.5.3            | PL/JavaScript (v8) trusted
 procedural language
(3 rows)
```

2. Créez un instantané de votre instance de base de données RDS for PostgreSQL si vous ne l'avez pas encore fait. Vous pouvez poursuivre avec les étapes suivantes tandis que l'instantané est en cours de création.
3. Déterminez le nombre de PLV8 fonctions de votre instance de base de données afin de vérifier qu'elles sont toutes en place après la mise à niveau. Par exemple, la requête SQL suivante renvoie le nombre de fonctions écrites en pvl8, plcoffee et plls.

```
SELECT proname, nspname, lanname
FROM pg_proc p, pg_language l, pg_namespace n
WHERE p.prolang = l.oid
AND n.oid = p.pronamespace
AND lanname IN ('plv8','plcoffee','plls');
```

4. Utiliser pg_dump pour créer un fichier de vidage schema-only. Par exemple, créez un fichier sur votre ordinateur client dans le répertoire `/tmp`.

```
./pg_dump -Fc --schema-only -U master postgres >/tmp/test.dmp
```

Cet exemple utilise les options suivantes :

- `-Fc` : format personnalisé
- --schema-only : supprime uniquement les commandes nécessaires à la création du schéma (les fonctions dans ce cas)
- `-U` : le nom de l'utilisateur principal RDS
- `database` : le nom de base de données dans votre instance de base de données

Pour plus d'informations sur pg_dump, veuillez consulter la section pg_dump de la documentation PostgreSQL.

5. Extrayez la déclaration DDL « CREATE FUNCTION » présente dans le fichier de vidage. L'exemple suivant utilise la commande `grep` pour extraire l'instruction DDL qui crée les fonctions et les enregistre dans un fichier. Vous l'utiliserez dans les étapes suivantes pour recréer les fonctions.

   ```
   ./pg_restore -l /tmp/test.dmp | grep FUNCTION > /tmp/function_list
   ```

   Pour plus d'informations sur pg_restore, veuillez consulter la section pg_restore de la documentation PostgreSQL.

6. Supprimez les fonctions et les extensions. L'exemple suivant supprime tous les objets PLV8 basés. L'option cascade garantit que les objets dépendants sont supprimés.

   ```
   DROP EXTENSION plv8 CASCADE;
   ```

   Si votre instance PostgreSQL contient des objets basés sur plcoffee ou plls, répétez l'étape pour ces extensions.

7. Créez les extensions. L'exemple suivant crée les extensions plv8, plcoffee et plls.

   ```
   CREATE EXTENSION plv8;
   CREATE EXTENSION plcoffee;
   CREATE EXTENSION plls;
   ```

8. Créez les fonctions à l'aide du fichier de vidage et du fichier « pilote ».

   L'exemple suivant recrée les fonctions que vous avez extraites précédemment.

   ```
   ./pg_restore -U master -d postgres -Fc -L /tmp/function_list /tmp/test.dmp
   ```

9. Vérifiez que toutes vos fonctions ont été recréées à l'aide de la requête suivante.

   ```
   SELECT * FROM pg_available_extensions WHERE name IN ('plv8','plls','plcoffee');
   ```

   La PLV8 version 2 ajoute la ligne supplémentaire suivante à votre jeu de résultats :

   ```
       proname     |  nspname   | lanname
   ----------------+------------+----------
    plv8_version   | pg_catalog | plv8
   ```

Utilisation de PL/Rust pour écrire des fonctions PostgreSQL dans le langage Rust

PL/Rust is a trusted Rust language extension for PostgreSQL. You can use it for stored procedures, functions, and other procedural code that's callable from SQL. The PL/Rustl'extension de langue est disponible dans les versions suivantes :

- RDS pour PostgreSQL 17.1 et versions ultérieures 17
- RDS pour PostgreSQL 16.1 et versions ultérieures 16
- RDS for PostgreSQL 15.2-R2 et versions 15 ultérieures
- RDS for PostgreSQL 14.9 et versions 14 ultérieures
- RDS for PostgreSQL 13.12 et versions 13 ultérieures

Pour plus d'informations, voir PL/Rust on. GitHub

Rubriques

- Configuration de PL/Rust
- Création de fonctions avec PL/Rust
- Utilisation de caisses avec PL/Rust
- Limites de PL/Rust

Configuration de PL/Rust

Pour installer l'extension plrust sur votre instance de base de données, ajoutez plrust au paramètre `shared_preload_libraries` dans le groupe de paramètres de la base de données associé à votre instance de base de données. Une fois l'extension plrust installée, vous pouvez créer des fonctions.

Pour modifier le paramètre `shared_preload_libraries`, votre instance de base de données doit être associée à un groupe de paramètres personnalisé. Pour obtenir des informations sur la création d'un groupe de paramètres de base de données personnalisé, consultez Groupes de paramètres pour (Amazon RDS).

Vous pouvez installer l'extension plrust en utilisant le AWS Management Console ou le AWS CLI.

Les étapes suivantes supposent que votre instance de base de données est associée à un groupe de paramètres de cluster de bases de données personnalisé.

console

Installer l'extension plrust dans le paramètre **`shared_preload_libraries`**

Effectuez les étapes suivantes à l'aide d'un compte membre du groupe (rôle) `rds_superuser`.

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de votre instance de base de données pour afficher ses détails.
4. Ouvrez l'onglet Configuration de votre instance de base de données et recherchez le lien du groupe de paramètres de l'instance de base de données.
5. Cliquez sur le lien pour ouvrir les paramètres personnalisés associés à votre instance de base de données.
6. Dans le champ de recherche Parameters (Paramètres), tapez `shared_pre` pour trouver le paramètre **`shared_preload_libraries`**.
7. Choisissez Edit parameters (Modifier les paramètres) pour accéder aux valeurs des propriétés.
8. Ajoutez plrust à la liste dans le champ Valeurs. Utilisez une virgule pour séparer les éléments de la liste de valeurs.
9. Redémarrez l'instance de base de données pour que la modification apportée au paramètre `shared_preload_libraries` prenne effet. Le redémarrage initial peut nécessiter plus de temps.
10. Lorsque l'instance est disponible, vérifiez que plrust a été initialisé. Utilisez `psql` pour vous connecter à l'instance de base de données, puis exécutez la commande suivante.

    ```
    SHOW shared_preload_libraries;
    ```

    Votre sortie doit ressembler à ce qui suit :

    ```
    shared_preload_libraries
    --------------------------
    rdsutils,plrust
    (1 row)
    ```

AWS CLI

Installez l'extension plrust dans le paramètre shared_preload_libraries

Effectuez les étapes suivantes à l'aide d'un compte membre du groupe (rôle) `rds_superuser`.

1. Utilisez la modify-db-parameter-group AWS CLI commande pour ajouter plrust au `shared_preload_libraries` paramètre.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=shared_preload_libraries,ParameterValue=plrust,ApplyMethod=pending-
reboot" \
   --region aws-region
```

2. Utilisez la reboot-db-instance AWS CLI commande pour redémarrer l'instance de base de données et initialiser la bibliothèque plrust. Le redémarrage initial peut nécessiter plus de temps.

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

3. Lorsque l'instance est disponible, vous pouvez vérifier que plrust a été initialisé. Utilisez `psql` pour vous connecter à l'instance de base de données, puis exécutez la commande suivante.

```
SHOW shared_preload_libraries;
```

   Votre sortie doit ressembler à ce qui suit :

```
shared_preload_libraries
--------------------------
rdsutils,plrust
(1 row)
```

Création de fonctions avec PL/Rust

PL/Rust compile la fonction sous forme de bibliothèque dynamique, la charger et l'exécute.

La fonction Rust suivante filtre les multiples d'un tableau.

```
postgres=> CREATE LANGUAGE plrust;
CREATE EXTENSION
```

```
CREATE OR REPLACE FUNCTION filter_multiples(a BIGINT[], multiple BIGINT) RETURNS
 BIGINT[]
    IMMUTABLE STRICT
    LANGUAGE PLRUST AS
$$
    Ok(Some(a.into_iter().filter(|x| x.unwrap() % multiple != 0).collect()))
$$;

WITH gen_values AS (
SELECT ARRAY(SELECT * FROM generate_series(1,100)) as arr)
SELECT filter_multiples(arr, 3)
from gen_values;
```

Utilisation de caisses avec PL/Rust

Dans RDS pour PostgreSQL versions 16.3-R2 et supérieures, 15.7-R2 et versions supérieures 15, 14.12-R2 et versions supérieures 14 versions, et 13.15-R2 et versions supérieures 13 versions, PL/Rust prend en charge des caisses supplémentaires :

- `url`
- `regex`
- `serde`
- `serde_json`

Dans RDS pour PostgreSQL versions 15.5-R2 et supérieures, 14.10-R2 et versions 14 supérieures, et 13.13-R2 et versions 13 supérieures, PL/Rust prend en charge deux caisses supplémentaires :

- `croaring-rs`
- `num-bigint`

À partir des versions 15.4, 14.9 et 13.12 d'Amazon RDS pour PostgreSQL, PL/Rust prend en charge les caisses suivantes :

- `aes`
- `ctr`

- `rand`

Seules les fonctionnalités par défaut sont prises en charge pour ces caisses. Les nouvelles versions de RDS for PostgreSQL peuvent contenir des versions mises à jour de caisses, et les anciennes versions de caisses peuvent ne plus être prises en charge.

Suivez les bonnes pratiques pour effectuer une mise à niveau de version majeure afin de tester si vos fonctions PL/Rust sont compatibles avec la nouvelle version majeure. Pour plus d'informations, consultez le blog Bonnes pratiques pour la mise à niveau d'Amazon RDS vers les versions majeures et mineures de PostgreSQL (langue française non garantie) et Mise à niveau du moteur de base de données PostgreSQL pour Amazon RDS dans le Guide de l'utilisateur Amazon RDS.

Des exemples d'utilisation des dépendances lors de la création d'une fonction PL/Rust sont disponibles dans Utiliser les dépendances (langue française non garantie).

Limites de PL/Rust

Par défaut, les utilisateurs de la base de données ne peuvent pas utiliserPL/Rust. To provide access to PL/Rust, se connecter en tant qu'utilisateur avec le privilège rds_superuser et exécuter la commande suivante :

```
postgres=> GRANT USAGE ON LANGUAGE PLRUST TO user;
```

Gestion des données spatiales avec l'GISextension Post

Post GIS est une extension de Postgre SQL pour le stockage et la gestion des informations spatiales. Pour en savoir plus sur PostGIS, consultez Post GIS .net.

À partir de la version 10.5, Postgre SQL prend en charge la bibliothèque libprotobuf 1.3.0 utilisée par Post GIS pour travailler avec les données de tuiles vectorielles des boîtes de cartes.

La configuration de l'GISextension Post nécessite `rds_superuser` des privilèges. Nous vous recommandons de créer un utilisateur (rôle) pour gérer l'GISextension Post et vos données spatiales. L'GISextension Post et ses composants associés ajoutent des milliers de fonctions à PostgreSQL. Envisagez de créer l'GISextension Post dans son propre schéma si cela convient à votre cas d'utilisation. L'exemple suivant montre comment installer l'extension dans sa propre base de données, mais cela n'est pas nécessaire.

Rubriques

- Étape 1 : créer un utilisateur (rôle) pour gérer l'GISextension Post
- Étape 2 : Chargez les GIS extensions Post
- Étape 3 : Transférer la propriété des schémas d'extension
- Étape 4 : Transférer la propriété des GIS tables Post
- Étape 5 : Testez les extensions
- Étape 6 : mise à niveau de l'GISextension Post
- Versions GIS postérieures à l'extension
- Mise à niveau du Post GIS 2 vers le Post GIS 3

Étape 1 : créer un utilisateur (rôle) pour gérer l'GISextension Post

Tout d'abord, connectez-vous à votre SQL instance de base de données RDS for Postgre en tant qu'utilisateur disposant de `rds_superuser` privilèges. Si vous avez conservé le nom par défaut lors de la configuration de votre instance, vous vous connectez en tant que `postgres`.

```
psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --username=postgres
 --password
```

Créez un rôle distinct (utilisateur) pour administrer l'GISextension Post.

```
postgres=>  CREATE ROLE gis_admin LOGIN PASSWORD 'change_me';
```

```
CREATE ROLE
```

Accordez des privilèges `rds_superuser` à ce rôle, pour lui permettre d'installer l'extension.

```
postgres=> GRANT rds_superuser TO gis_admin;
GRANT
```

Créez une base de données à utiliser pour vos GIS artefacts Post. Cette étape est facultative. Vous pouvez également créer un schéma dans votre base de données utilisateur pour les GIS extensions Post, mais cela n'est pas non plus obligatoire.

```
postgres=> CREATE DATABASE lab_gis;
CREATE DATABASE
```

Accordez à `gis_admin` tous les privilèges sur la base de données `lab_gis`.

```
postgres=> GRANT ALL PRIVILEGES ON DATABASE lab_gis TO gis_admin;
GRANT
```

Quittez la session et reconnectez-vous à votre SQL instance de base de données RDS for Postgre en tant que. `gis_admin`

```
postgres=> psql --host=111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=gis_admin --password --dbname=lab_gis
Password for user gis_admin:...
lab_gis=>
```

Continuez à configurer l'extension comme indiqué dans les étapes suivantes.

Étape 2 : Chargez les GIS extensions Post

L'GISextension Post inclut plusieurs extensions connexes qui fonctionnent ensemble pour fournir des fonctionnalités géospatiales. En fonction de votre cas d'utilisation, vous n'aurez peut-être pas besoin de toutes les extensions créées dans cette étape.

Utilisez `CREATE EXTENSION` des instructions pour charger les GIS extensions Post.

```
CREATE EXTENSION postgis;
CREATE EXTENSION
CREATE EXTENSION postgis_raster;
```

```
CREATE EXTENSION
CREATE EXTENSION fuzzystrmatch;
CREATE EXTENSION
CREATE EXTENSION postgis_tiger_geocoder;
CREATE EXTENSION
CREATE EXTENSION postgis_topology;
CREATE EXTENSION
CREATE EXTENSION address_standardizer_data_us;
CREATE EXTENSION
```

Vous pouvez vérifier les résultats en exécutant la SQL requête présentée dans l'exemple suivant, qui répertorie les extensions et leurs propriétaires.

```
SELECT n.nspname AS "Name",
  pg_catalog.pg_get_userbyid(n.nspowner) AS "Owner"
  FROM pg_catalog.pg_namespace n
  WHERE n.nspname !~ '^pg_' AND n.nspname <> 'information_schema'
  ORDER BY 1;
List of schemas
     Name     |   Owner
--------------+-----------
 public       | postgres
 tiger        | rdsadmin
 tiger_data   | rdsadmin
 topology     | rdsadmin
(4 rows)
```

Étape 3 : Transférer la propriété des schémas d'extension

Utilisez les ALTER SCHEMA instructions pour transférer la propriété des schémas au `gis_admin` rôle.

```
ALTER SCHEMA tiger OWNER TO gis_admin;
ALTER SCHEMA
ALTER SCHEMA tiger_data OWNER TO gis_admin;
ALTER SCHEMA
ALTER SCHEMA topology OWNER TO gis_admin;
ALTER SCHEMA
```

Vous pouvez confirmer le changement de propriétaire en exécutant la SQL requête suivante. Vous pouvez également utiliser la méta-commande \dn à partir de la ligne de commande psql.

```
SELECT n.nspname AS "Name",
  pg_catalog.pg_get_userbyid(n.nspowner) AS "Owner"
  FROM pg_catalog.pg_namespace n
  WHERE n.nspname !~ '^pg_' AND n.nspname <> 'information_schema'
  ORDER BY 1;

       List of schemas
     Name     |     Owner
--------------+---------------
 public       | postgres
 tiger        | gis_admin
 tiger_data   | gis_admin
 topology     | gis_admin
(4 rows)
```

Étape 4 : Transférer la propriété des GIS tables Post

> **Note**
>
> Ne changez pas de propriétaire des GIS fonctions de poste. Le bon fonctionnement et les futures mises à niveau de Post GIS nécessitent que ces fonctions conservent leur propriété d'origine. Pour plus d'informations sur les GIS autorisations de publication, consultez la section SQLSécurité de Postgre.

Utilisez la fonction suivante pour transférer la propriété des GIS tables Post au `gis_admin` rôle. Exécutez l'instruction suivante à partir de l'invite psql pour créer la fonction.

```
CREATE FUNCTION exec(text) returns text language plpgsql volatile AS $f$ BEGIN EXECUTE
 $1; RETURN $1; END; $f$;
CREATE FUNCTION
```

Ensuite, exécutez cette requête pour exécuter la fonction `exec` qui à son tour exécute les instructions et modifie les autorisations.

```
SELECT exec('ALTER TABLE ' || quote_ident(s.nspname) || '.' || quote_ident(s.relname)
 || ' OWNER TO gis_admin;')
  FROM (
    SELECT nspname, relname
    FROM pg_class c JOIN pg_namespace n ON (c.relnamespace = n.oid)
```

```
    WHERE nspname in ('tiger','topology') AND
    relkind IN ('r','S','v') ORDER BY relkind = 'S')
s;
```

Étape 5 : Testez les extensions

Pour éviter d'avoir à spécifier le nom du schéma, ajoutez le schéma `tiger` à votre chemin de recherche en utilisant la commande suivante.

```
SET search_path=public,tiger;
SET
```

Testez le `tiger` schéma à l'aide de l'SELECTinstruction suivante.

```
SELECT address, streetname, streettypeabbrev, zip
 FROM normalize_address('1 Devonshire Place, Boston, MA 02109') AS na;
address | streetname | streettypeabbrev |  zip
---------+------------+------------------+-------
       1 | Devonshire | Pl               | 02109
(1 row)
```

Pour en savoir plus sur cette extension, consultez Tiger Geocoder dans la GIS documentation du Post.

Testez l'accès au schéma `topology` en utilisant l'instruction `SELECT` suivante. Cela appelle la fonction `createtopology` qui enregistre un nouvel objet topologique (my_new_topo) avec l'identifiant de référence spatiale spécifié (26986) et la tolérance par défaut (0.5). Pour en savoir plus, consultez CreateTopologyla GIS documentation du Post.

```
SELECT topology.createtopology('my_new_topo',26986,0.5);
 createtopology
----------------
              1
(1 row)
```

Étape 6 : mise à niveau de l'GISextension Post

Chaque nouvelle version de Postgre SQL prend en charge une ou plusieurs versions de l'GISextension Post compatibles avec cette version. La mise à niveau SQL du moteur Postgre vers

une nouvelle version ne met pas automatiquement à niveau l'GISextension Post. Avant de mettre à niveau le SQL moteur Postgre, vous mettez généralement Post GIS à niveau vers la dernière version disponible pour la version actuelle de PostgreSQL. Pour plus de détails, consultez Versions GIS postérieures à l'extension.

Après la mise à niveau SQL du moteur Postgre, vous devez à nouveau mettre à niveau l'GISextension Post, vers la version prise en charge pour la version du SQL moteur Postgre récemment mise à niveau. Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL.

Vous pouvez vérifier à tout moment les mises à jour des versions d'GISextension Post disponibles sur votre pour une SQL instance de base de données Postgre. Pour ce faire, exécutez la commande suivante. Cette fonction est disponible avec Post GIS 2.5.0 et les versions supérieures.

```
SELECT postGIS_extensions_upgrade();
```

Si votre application ne prend pas en charge la dernière GIS version de Post, vous pouvez installer une ancienne version de Post GIS disponible dans votre version principale comme suit.

```
CREATE EXTENSION postgis VERSION "2.5.5";
```

Si vous souhaitez effectuer une mise à niveau vers une GIS version Post spécifique à partir d'une ancienne version, vous pouvez également utiliser la commande suivante.

```
ALTER EXTENSION postgis UPDATE TO "2.5.5";
```

Selon la version à partir de laquelle vous effectuez la mise à niveau, vous devrez peut-être utiliser à nouveau cette fonction. Le résultat de la première exécution de la fonction détermine si une mise à niveau supplémentaire est nécessaire. C'est le cas par exemple pour la mise à niveau du Post GIS 2 vers le Post GIS 3. Pour de plus amples informations, veuillez consulter Mise à niveau du Post GIS 2 vers le Post GIS 3.

Si vous avez mis à niveau cette extension pour préparer une mise à niveau majeure du SQL moteur Postgre, vous pouvez poursuivre les autres tâches préliminaires. Pour de plus amples informations, veuillez consulter Comment effectuer une mise à niveau de version majeure de RDS pour PostgreSQL.

Versions GIS postérieures à l'extension

Nous vous recommandons d'installer les versions de toutes les extensions telles que PostGIS, telles que répertoriées dans la section Postgre. SQL Les versions des extensions pour Amazon RDS pour Postgre sont disponibles SQL dans les notes de SQLmise à jour RDS d'Amazon pour Postgre. Pour obtenir une liste des versions qui sont disponibles dans votre version, utilisez la commande suivante.

```
SELECT * FROM pg_available_extension_versions WHERE name='postgis';
```

Vous trouverez des informations sur les versions dans les sections suivantes des notes de SQL mise à jour RDS d'Amazon pour Postgre :

- Extensions Postgre SQL version 16 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 15 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 14 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 13 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 12 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 11 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 10 prises en charge sur Amazon RDS
- Extensions Postgre SQL version 9.6.x prises en charge sur Amazon RDS

Mise à niveau du Post GIS 2 vers le Post GIS 3

À partir de la version 3.0, la fonctionnalité Post GIS raster est désormais une extension distincte,`postgis_raster`. Cette extension dispose de son propre chemin d'installation et de mise à niveau. Cela supprime des dizaines de fonctions, de types de données et d'autres artefacts nécessaires au traitement des images matricielles de l'extension `postgis` de base. Cela signifie que si votre cas d'utilisation ne nécessite pas de traitement matriciel, vous n'avez pas besoin d'installer l'extension `postgis_raster`.

Dans l'exemple de mise à niveau suivant, la première commande de mise à niveau extrait la fonctionnalité raster dans l'extension `postgis_raster`. Une deuxième commande de mise à niveau est alors nécessaire pour mettre à niveau `postgis_raster` vers la nouvelle version.

Pour passer du poste GIS 2 au poste GIS 3

1. Identifiez la version par défaut de Post GIS disponible pour la SQL version Postgre sur votre cluster de . RDSpour une SQL instance de base de données Postgre. Pour ce faire, exécutez la requête suivante.

```
SELECT * FROM pg_available_extensions
    WHERE default_version > installed_version;
  name   | default_version | installed_version |                          comment
---------+-----------------+-------------------
+-------------------------------------------------------------
 postgis | 3.1.4           | 2.3.7             | PostGIS geometry and geography
 spatial types and functions
(1 row)
```

2. Identifiez les versions de Post GIS installées dans chaque base de données sur . votre SQL instance RDS de base de données Postgre for. En d'autres termes, interrogez chaque base de données utilisateur comme suit.

```
SELECT
    e.extname AS "Name",
    e.extversion AS "Version",
    n.nspname AS "Schema",
    c.description AS "Description"
FROM
    pg_catalog.pg_extension e
    LEFT JOIN pg_catalog.pg_namespace n ON n.oid = e.extnamespace
    LEFT JOIN pg_catalog.pg_description c ON c.objoid = e.oid
    AND c.classoid = 'pg_catalog.pg_extension'::pg_catalog.regclass
WHERE
    e.extname LIKE '%postgis%'
ORDER BY
    1;
  Name   | Version | Schema |                            Description
---------+---------+--------
+-----------------------------------------------------------------------
 postgis | 2.3.7   | public | PostGIS geometry, geography, and raster spatial types
 and functions
(1 row)
```

Cette incompatibilité entre la version par défaut (Post GIS 3.1.4) et la version installée (Post GIS 2.3.7) signifie que vous devez mettre à jour l'extension Post. GIS

```
ALTER EXTENSION postgis UPDATE;
ALTER EXTENSION
WARNING: unpackaging raster
WARNING: PostGIS Raster functionality has been unpackaged
```

3. Exécutez la requête suivante pour vérifier que la fonctionnalité raster est maintenant dans son propre package.

```
SELECT
    probin,
    count(*)
FROM
    pg_proc
WHERE
    probin LIKE '%postgis%'
GROUP BY
    probin;
          probin          | count
--------------------------+-------
 $libdir/rtpostgis-2.3    | 107
 $libdir/postgis-3        | 487
(2 rows)
```

Le résultat montre qu'il y a toujours une différence entre les versions. Les GIS fonctions Post sont en version 3 (postgis-3), tandis que les fonctions raster (rtpostgis) sont en version 2 (rtpostgis-2.3). Pour terminer la mise à niveau, vous exécutez à nouveau la commande de mise à niveau, comme suit.

```
postgres=> SELECT postgis_extensions_upgrade();
```

Vous pouvez ignorer les messages d'avertissement, il n'y a aucun risque. Exécutez à nouveau la requête suivante pour vérifier que la mise à niveau est terminée. La mise à niveau est terminée lorsque Post GIS et toutes les extensions associées ne sont pas marquées comme nécessitant une mise à niveau.

```
SELECT postgis_full_version();
```

4. Utilisez la requête suivante pour voir le processus de mise à niveau terminé et les extensions packagées séparément, et vérifiez que leurs versions correspondent.

```
SELECT
    e.extname AS "Name",
    e.extversion AS "Version",
    n.nspname AS "Schema",
    c.description AS "Description"
FROM
    pg_catalog.pg_extension e
    LEFT JOIN pg_catalog.pg_namespace n ON n.oid = e.extnamespace
    LEFT JOIN pg_catalog.pg_description c ON c.objoid = e.oid
        AND c.classoid = 'pg_catalog.pg_extension'::pg_catalog.regclass
WHERE
    e.extname LIKE '%postgis%'
ORDER BY
    1;
      Name      | Version | Schema |                            Description
----------------+---------+--------
+------------------------------------------------------------------------
 postgis        | 3.1.5   | public | PostGIS geometry, geography, and raster
 spatial types and functions
 postgis_raster | 3.1.5   | public | PostGIS raster types and functions
(2 rows)
```

Le résultat indique que l'extension Post GIS 2 a été mise à niveau vers Post GIS 3, et que les deux extensions, `postgis` ainsi que l'`postgis_raster`extension désormais séparée, sont la version 3.1.5.

Une fois cette mise à niveau terminée, si vous ne prévoyez pas d'utiliser la fonctionnalité raster, vous pouvez abandonner l'extension comme suit.

```
DROP EXTENSION postgis_raster;
```

Utilisation des wrappers de données étrangers pris en charge pour SQL

Un wrapper de données étrangères (FDW) est un type spécifique d'extension qui permet d'accéder à des données externes. Par exemple, l'`oracle_fdw`extension permet à votre pour le SQL cluster de base de données Postgre de fonctionner avec des bases de données Oracle. Autre exemple, en utilisant l'`postgres_fdw`extension SQL native Postgre, vous pouvez accéder aux données stockées dans des instances de SQL base de données Postgre externes à votre instance de base RDS de données PostgreSQL.

Vous trouverez ci-dessous des informations sur plusieurs wrappers de données SQL étrangers Postgre pris en charge.

Rubriques

- Utilisation de l'extension log_fdw pour accéder au journal de base de données en utilisant SQL
- Utilisation de l'extension postgres_fdw pour accéder à des données externes
- Utilisation de Mes SQL bases de données à l'aide de l'extension mysql_fdw
- Utilisation des bases de données Oracle avec l'extension oracle_fdw
- Utilisation des bases de données SQL du serveur à l'aide de l'extension tds_fdw

Utilisation de l'extension log_fdw pour accéder au journal de base de données en utilisant SQL

Le pour l'SQLinstance de base de données Postgre prend en charge l'`log_fdw`extension, que vous pouvez utiliser pour accéder au journal de votre moteur de base de données à l'aide d'une SQL interface. L'extension `log_fdw` fournit deux fonctions qui facilitent la création de tables source pour les journaux de base de données :

- `list_postgres_log_files` – Répertorie les fichiers dans le répertoire du journal de base de données et indique la taille des fichiers en octets.
- `create_foreign_table_for_log_file(table_name text, server_name text, log_file_name text)` – Crée un tableau source pour le fichier spécifié dans la base de données actuelle.

Toutes les fonctions créées par `log_fdw` appartiennent à `rds_superuser`. Les membres du rôle `rds_superuser` peuvent accorder l'accès à ces fonctions à d'autres utilisateurs de base de données.

Par défaut, les fichiers journaux sont générés par Amazon RDS au format `stderr` (erreur standard), tel que spécifié dans le `log_destination` paramètre. Il n'y a que deux options pour ce paramètre, `stderr` et `csvlog` (valeurs séparées par des virgules,CSV). Si vous ajoutez l'`csvlog`option au paramètre, RDS génère à la fois des `csvlog` journaux `stderr` et des journaux. Cela peut affecter la capacité de stockage de votre cluster de base de données. Vous devez donc connaître les autres paramètres qui affectent la gestion des journaux. Pour de plus amples informations, veuillez consulter Définition de la destination du journal (`stderr`, `csvlog`).

L'un des avantages de la génération de journaux `csvlog` est que l'extension `log_fdw` vous permet de créer des tables externes dont les données sont soigneusement réparties en plusieurs colonnes. Pour ce faire, votre instance doit être associée à un groupe de paramètres de base de données personnalisé afin que vous puissiez modifier le paramètre de `log_destination`. Pour plus d'informations sur la manière de procéder, consultez Utilisation de paramètres sur votre instance de base de données RDS for PostgreSQL.

L'exemple suivant suppose que le paramètre `log_destination` comprend le champ `cvslog`.

Pour utiliser l'extension log_fdw

1. Installez l'extension `log_fdw`.

```
postgres=> CREATE EXTENSION log_fdw;
CREATE EXTENSION
```

2. Créez le serveur de journal en tant qu'encapsuleur de données externes.

```
postgres=> CREATE SERVER log_server FOREIGN DATA WRAPPER log_fdw;
CREATE SERVER
```

3. Sélectionnez l'ensemble des fichiers journaux d'une liste.

```
postgres=> SELECT * FROM list_postgres_log_files() ORDER BY 1;
```

   Voici un exemple de réponse.

```
          file_name           | file_size_bytes
```

```
------------------------------+-----------------
 postgresql.log.2023-08-09-22.csv |            1111
 postgresql.log.2023-08-09-23.csv |            1172
 postgresql.log.2023-08-10-00.csv |            1744
 postgresql.log.2023-08-10-01.csv |            1102
(4 rows)
```

4. Créez une table avec une seule colonne « log_entry » pour le fichier sélectionné.

```
postgres=> SELECT create_foreign_table_for_log_file('my_postgres_error_log',
     'log_server', 'postgresql.log.2023-08-09-22.csv');
```

La réponse ne fournit aucun détail autre que l'existence de la table.

```
-----------------------------------
(1 row)
```

5. Sélectionnez un exemple de fichier journal. Le code suivant récupère l'heure du journal et la description du message d'erreur.

```
postgres=> SELECT log_time, message FROM my_postgres_error_log ORDER BY 1;
```

Voici un exemple de réponse.

```
             log_time             |                                  message
----------------------------------
+-------------------------------------------------------------------------------
Tue Aug 09 15:45:18.172 2023 PDT | ending log output to stderr
Tue Aug 09 15:45:18.175 2023 PDT | database system was interrupted; last known up
 at 2023-08-09 22:43:34 UTC
Tue Aug 09 15:45:18.223 2023 PDT | checkpoint record is at 0/90002E0
Tue Aug 09 15:45:18.223 2023 PDT | redo record is at 0/90002A8; shutdown FALSE
Tue Aug 09 15:45:18.223 2023 PDT | next transaction ID: 0/1879; next OID: 24578
Tue Aug 09 15:45:18.223 2023 PDT | next MultiXactId: 1; next MultiXactOffset: 0
Tue Aug 09 15:45:18.223 2023 PDT | oldest unfrozen transaction ID: 1822, in
 database 1
(7 rows)
```

Utilisation de l'extension postgres_fdw pour accéder à des données externes

Vous pouvez accéder aux données d'un tableau sur un serveur de bases de données distant à l'aide de l'extension postgres_fdw. Si vous configurez une connexion distante à partir de votre instance de base de données PostgreSQL, l'accès à votre réplica en lecture est également disponible.

Pour utiliser postgres_fdw pour accéder à un serveur de bases de données distant

1. Installez l'extension postgres_fdw.

```
CREATE EXTENSION postgres_fdw;
```

2. Créez un serveur de données externes à l'aide de CREATE SERVER.

```
CREATE SERVER foreign_server
FOREIGN DATA WRAPPER postgres_fdw
OPTIONS (host 'xxx.xx.xxx.xx', port '5432', dbname 'foreign_db');
```

3. Créez un mappage utilisateur pour identifier le rôle à utiliser sur le serveur distant.

> **Important**
>
> Pour supprimer le mot de passe afin qu'il n'apparaisse pas dans les journaux, définissez-le `log_statement=none` au niveau de la session. Le réglage au niveau du paramètre ne supprime pas le mot de passe.

```
CREATE USER MAPPING FOR local_user
SERVER foreign_server
OPTIONS (user 'foreign_user', password 'password');
```

4. Créez une table mappée à la table sur le serveur distant.

```
CREATE FOREIGN TABLE foreign_table (
        id integer NOT NULL,
        data text)
SERVER foreign_server
OPTIONS (schema_name 'some_schema', table_name 'some_table');
```

Utilisation de Mes SQL bases de données à l'aide de l'extension mysql_fw

Pour accéder à une base de données SQL compatible My depuis votre instance de RDS pour Postgre SQL, vous pouvez installer et utiliser l'extension. `mysql_fdw` Ce wrapper de données étrangères vous permet de travailler avec RDS MySQL, Aurora MySQL, MariaDB et d'autres bases de données compatibles avec My. SQL La connexion entre le pour l'SQLinstance de base de données Postgre et la SQL base de données My Database est cryptée au mieux, en fonction des configurations du client et du serveur. Cependant, vous pouvez imposer le chiffrement si vous le souhaitez. Pour de plus amples informations, veuillez consulter Utilisation du chiffrement en transit avec l'extension.

L'`mysql_fdw`extension est prise en charge sur pour Postgre SQL versions 14.2, 13.6 et supérieures. Il prend en charge les sélections, les insertions, les mises à jour et les suppressions RDS depuis une SQL base de données Postgre vers des tables d'une instance de base de données SQL compatible My.

Rubriques

- Configuration de votre SQL base de données afin d'utiliser l'extension mysql_fdw
- Exemple : utilisation d'une SQL base de données for My depuis SQL
- Utilisation du chiffrement en transit avec l'extension

Configuration de votre SQL base de données afin d'utiliser l'extension mysql_fdw

La configuration de l'`mysql_fdw`extension sur votre instance de pour l'SQLinstance de base de données Postgre implique de charger l'extension dans votre instance de de base de données, puis de créer le point de connexion vers l'instance My SQL DB. Pour cette tâche, vous devez disposer des informations suivantes concernant l'instance My SQL DB :

- Nom d'hôte ou point de terminaison. Pour un pour une instance My SQL DB, vous pouvez trouver le point de terminaison à l'aide de la console. Sélectionnez l'onglet Connectivité et sécurité et regardez dans la section « Point de terminaison et port ».
- Numéro de port. Le numéro de port par défaut pour My SQL est 3306.
- Nom du moteur de la base de données. L'identifiant de la base de données.

Vous devez également fournir un accès au groupe de sécurité ou à la liste de contrôle d'accès (ACL) pour le SQL port My, 3306. Le cluster de pour l'instance de SQL base de données Postgre

et l'instance RDS pour My SQL DB doivent tous deux accéder au port 3306. Si l'accès n'est pas configuré correctement, lorsque vous essayez de vous connecter à la table My SQL -compatible, un message d'erreur semblable au suivant s'affiche :

```
ERROR: failed to connect to MySQL: Can't connect to MySQL server on 'hostname.aws-
region.rds.amazonaws.com:3306' (110)
```

Dans la procédure suivante, vous (en tant que compte `rds_superuser`) créez le serveur externe. Vous accordez ensuite l'accès au serveur externe à des utilisateurs spécifiques. Ces utilisateurs créent ensuite leurs propres mappages vers les comptes My SQL user appropriés pour travailler avec l'instance My SQL DB.

Pour utiliser mysql_fdw pour accéder à un serveur My database SQL

1. Connectez-vous à votre SQL instance de base de données Postgre à l'aide d'un compte doté du `rds_superuser` rôle. Si vous avez accepté les valeurs par défaut lorsque vous avez créé votre pour l'SQLinstance de base de données Postgre, le nom d'utilisateur est `postgres` et vous pouvez vous connecter à l'aide de l'outil de ligne de `psql` commande comme suit :

   ```
   psql --host=your-DB-instance.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password
   ```

2. Installez l'extension `mysql_fdw` comme suit :

   ```
   postgres=> CREATE EXTENSION mysql_fdw;
   CREATE EXTENSION
   ```

Une fois l'extension installée sur votre pour SQL instance de base de données Postgre, vous configurez le serveur étranger qui fournit la connexion à une base de données MySQL.

Pour créer le serveur externe

Effectuez ces tâches sur le pour l'instance de base de données PostgreSQL. Les étapes supposent que vous êtes connecté en tant qu'utilisateur avec des privilèges `rds_superuser`, tels que `postgres`.

1. Créez un serveur étranger dans le pour l'instance de base de données Postgre SQL :

```
postgres=> CREATE SERVER mysql-db FOREIGN DATA WRAPPER mysql_fdw OPTIONS (host 'db-
name.111122223333.aws-region.rds.amazonaws.com', port '3306');
CREATE SERVER
```

2. Accordez aux utilisateurs appropriés l'accès au serveur externe. Il doit s'agir d'utilisateurs non administrateurs, c'est-à-dire d'utilisateurs sans rôle `rds_superuser`.

```
postgres=> GRANT USAGE ON FOREIGN SERVER mysql-db to user1;
GRANT
```

SQLLes utilisateurs de Postgre créent et gèrent leurs propres connexions à la SQL base de données My via le serveur étranger.

Exemple : utilisation d'une SQL base de données for My depuis SQL

Supposons que vous disposiez d'une table simple sur une instance de base de données de base de SQL données Postgre. Vos SQL utilisateurs d' pour Postgre souhaitent interroger (`SELECT`), `INSERTUPDATE`, et `DELETE` les éléments de cette table. Supposons que l'`mysql_fdw`extension ait été créée sur votre SQL instance de base de données RDS for Postgre, comme indiqué dans la procédure précédente. Après vous être connecté à l'SQLinstance de base de données RDS for Postgre en tant qu'utilisateur disposant de `rds_superuser` privilèges, vous pouvez suivre les étapes suivantes.

1. Sur l'instance de de SQL base de données Postgre, créez un serveur étranger :

```
test=> CREATE SERVER mysqldb FOREIGN DATA WRAPPER mysql_fdw OPTIONS (host 'your-
DB.aws-region.rds.amazonaws.com', port '3306');
CREATE SERVER
```

2. Accordez l'utilisation à un utilisateur dépourvu d'autorisations `rds_superuser`, par exemple `user1` :

```
test=> GRANT USAGE ON FOREIGN SERVER mysqldb TO user1;
GRANT
```

3. Connect en tant que *`user1`*, puis créez un mappage vers le My SQL user :

```
test=> CREATE USER MAPPING FOR user1 SERVER mysqldb OPTIONS (username 'myuser',
 password 'mypassword');
```

```
CREATE USER MAPPING
```

4. Créez une table étrangère liée à la SQL table Ma table :

```
test=> CREATE FOREIGN TABLE mytab (a int, b text) SERVER mysqldb OPTIONS (dbname
 'test', table_name '');
CREATE FOREIGN TABLE
```

5. Exécutez une requête simple dans la table externe :

```
test=> SELECT * FROM mytab;
a |   b
---+-------
1 | apple
(1 row)
```

6. Vous pouvez ajouter, modifier et supprimer des données dans le SQL tableau Mon. Par exemple :

```
test=> INSERT INTO mytab values (2, 'mango');
INSERT 0 1
```

Exécutez à nouveau la requête SELECT pour voir les résultats :

```
test=> SELECT * FROM mytab ORDER BY 1;
 a |   b
---+-------
1 | apple
2 | mango
(2 rows)
```

Utilisation du chiffrement en transit avec l'extension

La connexion à My SQL depuis pour Postgre SQL utilise le chiffrement en transit (TLS/SSL) par défaut. Toutefois, la connexion redevient non chiffrée lorsque la configuration du client et du serveur diffère. Vous pouvez appliquer le chiffrement à toutes les connexions sortantes en spécifiant l'REQUIRE SSLoption RDS pour Mes comptes SQL utilisateurs. Cette même approche fonctionne également pour les comptes utilisateur MariaDB et SQL Aurora My.

Pour Mes comptes SQL utilisateurs configurés pour`REQUIRE SSL`, la tentative de connexion échoue si une connexion sécurisée ne peut pas être établie.

Pour appliquer le chiffrement aux comptes utilisateur existants SQL de Ma base de données, vous pouvez utiliser la `ALTER USER` commande. La syntaxe varie en fonction de la SQL version My, comme indiqué dans le tableau suivant. Pour plus d'informations, reportez-vous ALTERUSERà Mon manuel SQL de référence.

SQL5,7 mai, SQL 8,0 mai	Mon SQL 5.6
`ALTER USER 'user'@'%' REQUIRE SSL;`	`GRANT USAGE ON *.* to 'user'@'%' REQUIRE SSL;`

Pour plus d'informations sur l'extension `mysql_fdw`, consultez la documentation mysql_fdw.

Utilisation des bases de données Oracle avec l'extension oracle_fdw

Pour accéder à une base de données Oracle depuis votre instance de base de données RDS for PostgreSQL, vous pouvez installer et utiliser l'extension `oracle_fdw`. Cette extension est un encapsuleur de données externes pour les bases de données Oracle. Pour en savoir plus sur cette extension, veuillez consulter la documentation oracle_fdw.

L'extension `oracle_fdw` est prise en charge sur RDS pour PostgreSQL 12.7, 13.3 et les versions ultérieures.

Rubriques

- Activation de l'extension oracle_fdw
- Exemple : utilisation d'un serveur externe lié à une base de données Amazon RDS for Oracle Database
- Utilisation du chiffrement en transit
- Comprendre la vue et les autorisations pg_user_mappings

Activation de l'extension oracle_fdw

Pour utiliser l'extension oracle_fdw, suivez la procédure suivante.

Pour activer l'extension oracle_fdw

- Exécutez la commande suivante en utilisant un compte disposant d'autorisations `rds_superuser`.

```
CREATE EXTENSION oracle_fdw;
```

Exemple : utilisation d'un serveur externe lié à une base de données Amazon RDS for Oracle Database

Les exemples suivants démontrent l'utilisation d'un serveur externe lié à une base de données Amazon RDS for Oracle.

Pour créer un serveur externe lié à une base de données RDS for Oracle

1. Notez ce qui suit sur l'instance de base de données RDS for Oracle :
   - Point de terminaison
   - Port
   - Nom de base de données
2. Créez un serveur externe.

```
test=> CREATE SERVER oradb FOREIGN DATA WRAPPER oracle_fdw OPTIONS (dbserver
 '//endpoint:port/DB_name');
CREATE SERVER
```

3. Accordez l'utilisation à un utilisateur dépourvu d'autorisations `rds_superuser`, par exemple `user1`.

```
test=> GRANT USAGE ON FOREIGN SERVER oradb TO user1;
GRANT
```

4. Connectez-vous en tant que `user1` et créez un mappage à un utilisateur Oracle.

```
test=> CREATE USER MAPPING FOR user1 SERVER oradb OPTIONS (user 'oracleuser',
 password 'mypassword');
CREATE USER MAPPING
```

5. Créez une table externe liée à une table Oracle.

```
test=> CREATE FOREIGN TABLE mytab (a int) SERVER oradb OPTIONS (table 'MYTABLE');
CREATE FOREIGN TABLE
```

6. Interrogez la table externe.

```
test=>  SELECT * FROM mytab;
a
---
1
(1 row)
```

Si la requête signale l'erreur suivante, vérifiez votre groupe de sécurité et votre liste de contrôle d'accès (ACL) pour vous assurer que les deux instances peuvent communiquer.

```
ERROR: connection for foreign table "mytab" cannot be established
DETAIL: ORA-12170: TNS:Connect timeout occurred
```

Utilisation du chiffrement en transit

PostgreSQL-to-Oracle le chiffrement en transit repose sur une combinaison de paramètres de configuration du client et du serveur. Pour un exemple d'utilisation d'Oracle 21c, consultez A propos de la négociation du chiffrement et de l'intégrité dans la documentation Oracle. Le client utilisé pour oracle_fdw sur Amazon RDS est configuré `ACCEPTED` avec, ce qui signifie que le chiffrement dépend de la configuration du serveur de base de données Oracle et qu'il utilise la bibliothèque de sécurité Oracle (libnnz) pour le chiffrement.

Si votre base de données se trouve sur RDS for Oracle, consultez la section Oracle native network encryption (Chiffrement réseau natif Oracle) pour configurer le chiffrement.

Comprendre la vue et les autorisations pg_user_mappings

Le catalogue PostgreSQL `pg_user_mapping` stocke le mappage d'un utilisateur RDS for PostgreSQL vers l'utilisateur d'un serveur de données externe (distant). L'accès au catalogue est restreint, mais vous utilisez la vue `pg_user_mappings` pour visualiser les mappages. Dans ce qui suit, vous trouverez un exemple qui présente comment les autorisations s'appliquent avec un exemple de base de données Oracle, mais ces informations s'appliquent plus généralement à tout encapsuleur de données externes.

Dans la sortie suivante, vous pouvez trouver des rôles et des autorisations mappés à trois exemples d'utilisateurs différents. Les utilisateurs `rdssu1` et `rdssu2` sont membres du rôle `rds_superuser`, et `user1` ne l'est pas. L'exemple utilise la métacommande `psql` `\du` pour lister les rôles existants.

```
test=>  \du
                                                          List of roles
    Role name    |                         Attributes                         |
 Member of
-----------------+------------------------------------------------------------
+-----------------------------------------------------
 rdssu1          |                                                            |
 {rds_superuser}
 rdssu2          |                                                            |
 {rds_superuser}
 user1           |                                                            | {}
```

Tous les utilisateurs, y compris ceux qui disposent de privilèges `rds_superuser`, sont autorisés à voir leurs propres mappages d'utilisateurs (`umoptions`) dans la table `pg_user_mappings`. Comme le montre l'exemple suivant, lorsque `rdssu1` tente d'obtenir tous les mappages d'utilisateurs, une erreur s'affiche en dépit des privilèges `rds_superuser` de `rdssu1` :

```
test=> SELECT * FROM pg_user_mapping;
ERROR: permission denied for table pg_user_mapping
```

Voici quelques exemples.

```
test=> SET SESSION AUTHORIZATION rdssu1;
SET
test=> SELECT * FROM pg_user_mappings;
 umid  | srvid | srvname | umuser | usename    |            umoptions
-------+-------+---------+--------+------------+---------------------------------
 16414 | 16411 | oradb   |  16412 | user1      |
 16423 | 16411 | oradb   |  16421 | rdssu1     | {user=oracleuser,password=mypwd}
 16424 | 16411 | oradb   |  16422 | rdssu2     |
 (3 rows)

test=> SET SESSION AUTHORIZATION rdssu2;
SET
test=> SELECT * FROM pg_user_mappings;
 umid  | srvid | srvname | umuser | usename    |            umoptions
-------+-------+---------+--------+------------+---------------------------------
 16414 | 16411 | oradb   |  16412 | user1      |
```

```
 16423 | 16411 | oradb   |  16421 | rdssu1     |
 16424 | 16411 | oradb   |  16422 | rdssu2     | {user=oracleuser,password=mypwd}
 (3 rows)

test=> SET SESSION AUTHORIZATION user1;
SET
test=> SELECT * FROM pg_user_mappings;
 umid  | srvid | srvname | umuser | usename    |           umoptions
-------+-------+---------+--------+------------+-------------------------------
 16414 | 16411 | oradb   |  16412 | user1      | {user=oracleuser,password=mypwd}
 16423 | 16411 | oradb   |  16421 | rdssu1     |
 16424 | 16411 | oradb   |  16422 | rdssu2     |
 (3 rows)
```

En raison des différences dans l'implémentation de `information_schema._pg_user_mappings` et de `pg_catalog.pg_user_mappings`, un `rds_superuser` créé manuellement nécessite des autorisations supplémentaires pour afficher les mots de passe dans `pg_catalog.pg_user_mappings`.

Un `rds_superuser` n'a besoin d'aucune autorisation supplémentaire pour afficher les mots de passe dans `information_schema._pg_user_mappings`.

Les utilisateurs qui n'ont pas le rôle `rds_superuser` peuvent afficher les mots de passe dans `pg_user_mappings` uniquement dans les conditions suivantes :

- L'utilisateur actif est celui faisant l'objet du mappage. Il possède le serveur ou détient le privilège `USAGE` sur celui-ci.
- L'utilisateur actuel est le propriétaire du serveur et le mappage est pour `PUBLIC`.

Utilisation des bases de données SQL du serveur à l'aide de l'extension tds_fdw

Vous pouvez utiliser l'SQL`tds_fdw`extension Postgre pour accéder aux bases de données qui prennent en charge le protocole tabular data stream (TDS), telles que les bases de données Sybase et Microsoft SQL Server. Ce wrapper de données étrangères vous permet de vous connecter depuis votre aux bases de données qui utilisent le TDS protocole, y compris Amazon pour RDS Microsoft Server. SQL Pour plus d'informations, consultez la documentation de tds-fdw/tds_fdw sur. GitHub

L'`tds_fdw`extension est prise en charge sur Amazon RDS pour les SQL versions 14.2, 13.6 et supérieures de Postgre.

Configuration de votre SQL base de données Aurora Postgre pour utiliser l'extension tds_fdw

Dans les procédures suivantes, vous trouverez un exemple de configuration et d'utilisation du cluster de base de `tds_fdw` données . Avant de pouvoir vous connecter à une base de données SQL Server en utilisant`tds_fdw`, vous devez obtenir les informations suivantes pour l'instance :

- Nom d'hôte ou point de terminaison. Pour une instance de base de données RDS for SQL Server, vous pouvez trouver le point de terminaison à l'aide de la console. Sélectionnez l'onglet Connectivité et sécurité et regardez dans la section « Point de terminaison et port ».
- Numéro de port. Le numéro de port par défaut pour Microsoft SQL Server est 1433.
- Nom du moteur de la base de données. L'identifiant de la base de données.

Vous devez également fournir un accès au groupe de sécurité ou à la liste de contrôle d'accès (ACL) pour le port 1433 SQL du serveur. Le pour l'instance de base de données Postgre et l'SQLinstance de base de données RDS pour le SQL serveur doivent tous deux accéder au port 1433. Si l'accès n'est pas configuré correctement, le message d'erreur suivant s'affiche lorsque vous essayez d'interroger le Microsoft SQL Server :

```
ERROR: DB-Library error: DB #: 20009, DB Msg: Unable to connect:
Adaptive Server is unavailable or does not exist (mssql2019.aws-
region.rds.amazonaws.com), OS #: 0, OS Msg: Success, Level: 9
```

Pour utiliser tds_fdw pour se connecter à une base de données du serveur SQL

1. Connectez-vous à votre instance de base de données Postgre L'SQLinstance à l'aide d'un compte ayant le `rds_superuser` rôle suivant :

   ```
   psql --host=your-DB-instance.aws-region.rds.amazonaws.com --port=5432 --
   username=test --password
   ```

2. Installez l'extension `tds_fdw` :

   ```
   test=> CREATE EXTENSION tds_fdw;
   CREATE EXTENSION
   ```

Une fois l'extension installée sur votre pour SQL instance de base de données Postgre, vous configurez le serveur étranger.

Pour créer le serveur externe

Effectuez ces tâches sur le pour l'SQLinstance de base de données Postgre à l'aide d'un compte doté `rds_superuser` de privilèges.

1. Créez un serveur étranger dans le pour l'instance de base de données Postgre SQL :

```
test=> CREATE SERVER sqlserverdb FOREIGN DATA WRAPPER tds_fdw OPTIONS
 (servername 'mssql2019.aws-region.rds.amazonaws.com', port '1433', database
 'tds_fdw_testing');
CREATE SERVER
```

   Pour accéder à ASCII des données non SQLServer secondaires, créez un lien serveur avec l'option character_set dans le :

```
test=> CREATE SERVER sqlserverdb FOREIGN DATA WRAPPER tds_fdw OPTIONS (servername
 'mssql2019.aws-region.rds.amazonaws.com', port '1433', database 'tds_fdw_testing',
 character_set 'UTF-8');
CREATE SERVER
```

2. Accordez des autorisations à un utilisateur qui n'a pas de privilèges de rôle `rds_superuser`, par exemple, `user1` :

```
test=> GRANT USAGE ON FOREIGN SERVER sqlserverdb TO user1;
```

3. Connectez-vous en tant qu'utilisateur1 et créez un mappage vers un utilisateur SQL du serveur :

```
test=> CREATE USER MAPPING FOR user1 SERVER sqlserverdb OPTIONS (username
 'sqlserveruser', password 'password');
CREATE USER MAPPING
```

4. Créez une table étrangère liée à une table SQL de serveur :

```
test=> CREATE FOREIGN TABLE mytab (a int) SERVER sqlserverdb OPTIONS (table
 'MYTABLE');
CREATE FOREIGN TABLE
```

5. Interrogez la table externe :

```
test=> SELECT * FROM mytab;
 a
---
 1
(1 row)
```

Utilisation du chiffrement en transit pour la connexion

La connexion entre et le SQL serveur utilise le chiffrement en transit (TLS/SSL) en fonction de la configuration de la base de données du SQL serveur. Si le SQL serveur n'est pas configuré pour le chiffrement, le RDS SQL client Postgre qui fait la demande à la base de données du SQL serveur revient à une version non chiffrée.

Vous pouvez appliquer le chiffrement pour la connexion aux RDS instances de base de données SQL du serveur en définissant le `rds.force_ssl` paramètre. Pour savoir comment procéder, consultez Forcer les connexions à votre instance de base de données à utiliser SSL. Pour plus d'informations sur la TLS configurationSSL/pour RDS for SQL Server, consultez la section Utilisation SSL avec une instance de base de données Microsoft SQL Server.

Utilisation de Trusted Language Extensions pour PostgreSQL

Trusted Language Extensions pour PostgreSQL est un kit de développement open source permettant de créer des extensions PostgreSQL. Il vous permet de créer des extensions PostgreSQL à hautes performances et de les exécuter en toute sécurité sur votre instance de base de données RDS for PostgreSQL. En utilisant Trusted Language Extensions (TLE) pour PostgreSQL, vous pouvez créer des extensions PostgreSQL qui suivent l'approche documentée pour étendre les fonctionnalités de PostgreSQL. Pour plus d'informations, consultez Packaging Related Objects into an Extension (Empaquetage d'objets associés dans une extension) dans la documentation PostgreSQL.

L'un des principaux avantages de TLE est que vous pouvez l'utiliser dans des environnements qui ne donnent pas accès au système de fichiers sous-jacent à l'instance PostgreSQL. Auparavant, l'installation d'une nouvelle extension nécessitait l'accès au système de fichiers. TLE supprime cette contrainte. Il fournit un environnement de développement permettant de créer de nouvelles extensions pour n'importe quelle base de données PostgreSQL, y compris celles qui s'exécutent sur vos instances de base de données RDS for PostgreSQL.

TLE est conçu pour empêcher l'accès à des ressources dangereuses pour les extensions que vous créez à l'aide de TLE. Son environnement d'exécution limite l'impact de tout défaut d'extension à une seule connexion de base de données. TLE permet également aux administrateurs de base de données de contrôler précisément qui peut installer les extensions et fournit un modèle d'autorisations pour les exécuter.

TLE est pris en charge sur les versions suivantes de RDS for PostgreSQL :

- Version 17.1 et versions supérieures 17 versions
- Version 16.1 et supérieure 16 versions
- Version 15.2 et supérieure 15 versions
- Version 14.5 et supérieure 14 versions
- Version 13.12 et versions supérieures 13 versions

Le runtime et l'environnement de développement Trusted Language Extensions sont fournis sous la forme de l'extension PostgreSQL `pg_tle`, version 1.0.1. Il prend en charge la création d'extensions en JavaScript Perl, Tcl, PL/pgSQL et SQL. Vous installez l'extension `pg_tle` dans votre instance de base de données RDS for PostgreSQL de la même manière que vous installez les autres extensions PostgreSQL. Une fois le kit `pg_tle` configuré, les développeurs peuvent l'utiliser pour créer de nouvelles extensions PostgreSQL, appelées extensions TLE.

Dans les rubriques suivantes, vous apprendrez comment configurer le kit Trusted Language Extensions et comment commencer à créer vos propres extensions TLE.

Rubriques

- Terminologie
- Conditions requises pour utiliser des extensions linguistiques fiables pour Postgre SQL
- Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL
- Vue d'ensemble des extensions linguistiques fiables pour Postgre SQL
- Création d'TLEextensions pour SQL
- Supprimer vos TLE extensions d'une base de données
- Désinstaller les extensions linguistiques fiables pour Postgre SQL
- Utilisation des hooks PostgreSQL avec vos extensions TLE
- Utilisation de types de données personnalisés dans TLE
- Référence fonctionnelle pour les extensions linguistiques fiables pour Postgre SQL
- Référence des hooks pour Trusted Language Extensions pour PostgreSQL

Terminologie

Pour vous aider à mieux comprendre Trusted Language Extensions, consultez le glossaire suivant des termes utilisés dans cette rubrique.

Extensions linguistiques fiables pour Postgre SQL

: Trusted Language Extensions for Postgre SQL est le nom officiel du kit de développement open source fourni sous forme d'`pg_tle`extension. Il peut être utilisé sur n'importe quel SQL système Postgre. Pour plus d'informations, consultez aws/pg_tle on. GitHub

Trusted Language Extensions

: Trusted Language Extensions est le nom abrégé de Trusted Language Extensions for PostgreSQL. Ce nom abrégé et son abréviation (TLE) sont également utilisés dans cette documentation.

langage approuvé

: Un langage approuvé est un langage de programmation ou de script doté d'attributs de sécurité spécifiques. Par exemple, les langages approuvés limitent généralement l'accès au système

de fichiers et limitent l'utilisation de propriétés réseau spécifiées. Le kit TLE de développement est conçu pour prendre en charge les langages fiables. Postgre SQL prend en charge plusieurs langages différents utilisés pour créer des extensions fiables ou non fiables. Pour un exemple, consultez Trusted and Untrusted PL/Perl dans la documentation Postgre. SQL Lorsque vous créez une extension à l'aide du kit Trusted Language Extensions, l'extension utilise intrinsèquement des mécanismes linguistiques approuvés.

TLEextension

Une TLEextension est une SQL extension Postgre créée à l'aide du kit de développement Trusted Language Extensions (TLE).

Conditions requises pour utiliser des extensions linguistiques fiables pour Postgre SQL

Les conditions requises pour configurer et utiliser le kit de TLE développement sont les suivantes.

- Versions versions Postgre — Les extensions linguistiques fiables sont prises en charge sur pour Postgre SQL versions 13.12 et supérieures 13 versions, 14.5 et supérieures 14 versions, et 15.2 et versions supérieures uniquement.
  - Mises à niveau du moteur de base de données RDS pour PostgreSQL.
  - Si vous ne possédez pas encore d'instance de base de données Amazon du cluster de RDS bases exécutant PostgreSQL, vous pouvez en créer une. Pour en savoir plus, consultez RDSpour une SQL instance de base de données Postgre, voirCréation et connexion à une instance de base de données PostgreSQL.
- Nécessite les privilèges **`rds_superuser`** – Pour installer et configurer l'extension `pg_tle`, votre rôle d'utilisateur de base de données doit disposer des autorisations du rôle `rds_superuser`. Par défaut, ce rôle est accordé à l'`postgres`utilisateur qui crée le . RDSpour une SQL instance de base de données Postgre.
- Nécessite un groupe de paramètres de base de données personnalisé : votre pour SQL instance de base de données Postgre doit être configuré avec un groupe de paramètres de base de données personnalisé.
  - Si votre pour SQL instance de base de données Postgre n'est pas configuré avec un groupe de paramètres de base de données personnalisé, vous devez en créer un et l'associer à . votre SQL instance RDS de base de données Postgre for. Pour un bref résumé des étapes, consultez Création et application d'un groupe de paramètres de base de données personnalisé.

- Si votre pour SQL instance de base de données Postgre est déjà configuré à l'aide d'un groupe de paramètres de base de données personnalisé, vous pouvez configurer des extensions de langage fiables. Pour plus de détails, consultez Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL.

Création et application d'un groupe de paramètres de base de données personnalisé

Procédez comme suit pour créer un groupe de paramètres de base de données personnalisé et configurer votre pour que l'SQLinstance de base de données Postgre l'utilise.

Console

Pour créer un groupe de paramètres de base de données personnalisé et l'utiliser avec votre pour Postgre SQL

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Choisissez Groupes de paramètres dans le RDS menu Amazon.
3. Choisissez Créer un groupe de paramètres.
4. Dans la page Parameter group details (Détails des groupes de paramètres), entrez les informations suivantes.
   - Pour Parameter group family (Famille de groupes de paramètres), choisissez postgres14.
   - Pour Type, choisissez DB Parameter Group (Groupe de paramètres de base de données).
   - Pour Group name (Nom du groupe), attribuez un nom significatif à votre groupe de paramètres dans le contexte de vos opérations.
   - Pour Description, entrez une description utile afin que les autres membres de votre équipe puissent la trouver facilement.
5. Sélectionnez Create (Créer). Votre groupe de paramètres de base de données personnalisé est créé dans votre Région AWS. Vous pouvez désormais modifier votre pour que l'SQLinstance de base de données Postgre l'utilise en suivant les étapes suivantes.
6. Choisissez Databases dans le RDS menu Amazon.
7. Choisissez le pour l'SQLinstance de base de données Postgre que vous souhaitez utiliser TLE parmi ceux répertoriés, puis choisissez Modifier.
8. Dans la page Modify DB instance settings (Modifier les paramètres d'instance de base de données), recherchez Database options (Options de base de données) dans la section

Additional configuration (Configuration supplémentaire) et choisissez votre groupe de paramètres de base de données personnalisé dans le sélecteur.

9. Choisissez Continue (Continuer) pour enregistrer la modification.
10. Choisissez Appliquer immédiatement afin de continuer à configurer le pour l'utilisation de l'SQLinstance de base de données Postgre. TLE

Pour continuer à configurer votre système pour Trusted Language Extensions, consultez Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL.

Pour plus d'informations sur l'utilisation de groupes de paramètres de base de données, consultez Groupes de paramètres de base de données pour les instances de base de données RDSAmazon.

AWS CLI

Vous pouvez éviter de spécifier l'`--region`argument lorsque vous utilisez des CLI commandes en configurant votre AWS CLI avec votre valeur par défaut Région AWS. Pour plus d'informations, consultez la section Principes de base de la configuration dans le AWS Command Line Interface Guide de l'utilisateur.

Pour créer un groupe de paramètres de base de données personnalisé et l'utiliser avec votre pour Postgre SQL

1. Utilisez le create-db-parameter-group AWS CLI commande pour créer un groupe de paramètres de base de données personnalisé basé sur pour votre Région AWS.

   Dans Linux, macOS, ou Unix:

   ```
   aws rds create-db-parameter-group \
     --region aws-region \
     --db-parameter-group-name custom-params-for-pg-tle \
     --db-parameter-group-family postgres14 \
     --description "My custom DB parameter group for Trusted Language Extensions"
   ```

   Dans Windows:

   ```
   aws rds create-db-parameter-group ^
     --region aws-region ^
     --db-parameter-group-name custom-params-for-pg-tle ^
     --db-parameter-group-family postgres14 ^
   ```

```
  --description "My custom DB parameter group for Trusted Language Extensions"
```

Votre groupe de paramètres de base de données personnalisé est disponible dans votre Région AWS, vous pouvez donc modifier pour que l'SQLinstance de base de données Postgre l'utilise.

2. Utilisez le modify-db-instance AWS CLI commande pour appliquer votre groupe de paramètres de base de données personnalisé à . votre SQL instance RDS de base de données Postgre for. Cette commande redémarre immédiatement l'instance active.

   Dans Linux, macOS, ou Unix:

   ```
   aws rds modify-db-instance \
     --region aws-region \
     --db-instance-identifier your-instance-name \
     --db-parameter-group-name custom-params-for-pg-tle \
     --apply-immediately
   ```

   Dans Windows:

   ```
   aws rds modify-db-instance ^
     --region aws-region ^
     --db-instance-identifier your-instance-name ^
     --db-parameter-group-name custom-params-for-pg-tle ^
     --apply-immediately
   ```

Pour continuer à configurer votre système pour Trusted Language Extensions, consultez Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL.

Pour plus d'informations, consultez Groupes de paramètres pour (Amazon RDS).

Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL

Les étapes suivantes supposent que votre instance de RDS pour Postgre SQL est associé à un groupe de paramètres de base de données de de base de données personnalisé. Vous pouvez utiliser le plugin AWS Management Console ou le AWS CLI pour ces étapes.

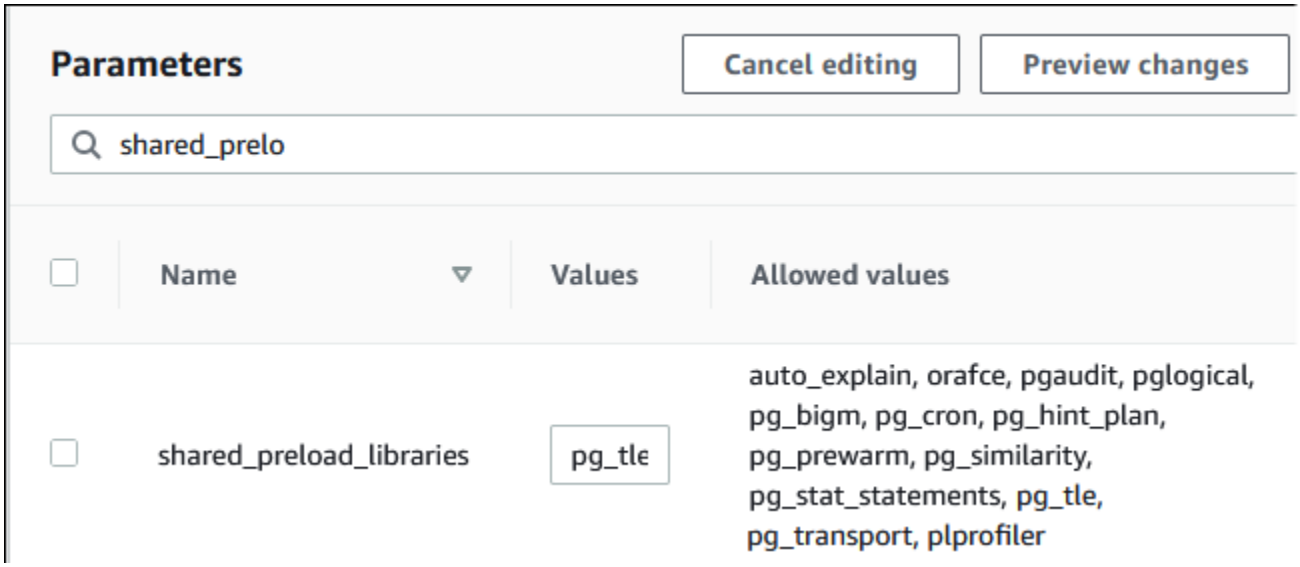
Lorsque vous configurez des extensions linguistiques fiables dans votre pour une SQL instance de base de données Postgre, vous les installez dans une base de données spécifique à l'usage des utilisateurs de base de données autorisés sur cette base de données.

Console

Pour configurer Trusted Language Extensions

Effectuez les étapes suivantes à l'aide d'un compte membre du groupe (rôle) `rds_superuser`.

1. Connectez-vous au AWS Management Console et ouvrez la RDS console Amazon à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le volet de navigation, choisissez l' de SQL base de données Postgre.
3. Ouvrez l'onglet Configuration de votre . RDSpour une SQL instance de base de données Postgre. Parmi les détails de l'instance, trouvez le lien Groupe de paramètres.
4. Cliquez sur le lien pour ouvrir les paramètres personnalisés associés à votre . RDSpour une SQL instance de base de données Postgre.
5. Dans le champ de recherche Parameters (Paramètres), tapez `shared_pre` pour trouver le paramètre `shared_preload_libraries`.
6. Choisissez Edit parameters (Modifier les paramètres) pour accéder aux valeurs des propriétés.
7. Ajoutez `pg_tle` à la liste dans le champ Values (Valeurs). Utilisez une virgule pour séparer les éléments de la liste de valeurs.



8. Redémarrez l' pour l'SQLinstance de base de données Postgre afin que votre modification du `shared_preload_libraries` paramètre prenne effet.
9. Lorsque l'instance est disponible, vérifiez que `pg_tle` a été initialisé. Utilisez-le `psql` pour vous connecter à l' pour l'SQLinstance de base de données Postgre, puis exécutez la commande suivante.

```
SHOW shared_preload_libraries;
shared_preload_libraries
-------------------------
rdsutils,pg_tle
(1 row)
```

10. Une fois l'extension `pg_tle` initialisée, vous pouvez maintenant créer l'extension.

```
CREATE EXTENSION pg_tle;
```

Vous pouvez vérifier que l'extension est installée en utilisant la métacommande `psql` suivante.

```
labdb=> \dx
                          List of installed extensions
  Name   | Version |   Schema   |                Description
---------+---------+------------+-------------------------------------------
 pg_tle  | 1.0.1   | pgtle      | Trusted-Language Extensions for PostgreSQL
 plpgsql | 1.0     | pg_catalog | PL/pgSQL procedural language
```

11. Accordez le `pgtle_admin` rôle au nom d'utilisateur principal que vous avez créé pour votre instance de votre SQL instance de base de données Postgre lors de sa configuration. Si vous avez accepté la valeur par défaut, il s'agit de `postgres`.

```
labdb=> GRANT pgtle_admin TO postgres;
GRANT ROLE
```

Vous pouvez vérifier que l'octroi a eu lieu à l'aide de la métacommande `psql`, comme illustré dans l'exemple suivant. Seuls les rôles `pgtle_admin` et `postgres` sont affichés dans la sortie. Pour plus d'informations, consultez Comprendre le rôle rds_superuser.

```
labdb=> \du
                          List of roles
    Role name    |           Attributes            |              Member of
-----------------+---------------------------------
+-----------------------------------
pgtle_admin     | Cannot login                     | {}
postgres        | Create role, Create DB          +| {rds_superuser,pgtle_admin}
                | Password valid until infinity    |...
```

12. Fermez la session `psql` à l'aide de la métacommande `\q`.

```
\q
```

Pour commencer à créer des TLE extensions, consultezExemple : création d'une extension de langue fiable à l'aide de SQL.

AWS CLI

Vous pouvez éviter de spécifier l'`--region`argument lorsque vous utilisez des CLI commandes en configurant votre AWS CLI avec votre valeur par défaut Région AWS. Pour plus d'informations, consultez la section Principes de base de la configuration dans le AWS Command Line Interface Guide de l'utilisateur.

Pour configurer Trusted Language Extensions

1. Utilisez le modify-db-parameter-group AWS CLI commande à ajouter `pg_tle` au `shared_preload_libraries` paramètre.

```
aws rds modify-db-parameter-group \
   --db-parameter-group-name custom-param-group-name \
   --parameters
 "ParameterName=shared_preload_libraries,ParameterValue=pg_tle,ApplyMethod=pending-
reboot" \
   --region aws-region
```

2. Utilisez le reboot-db-instance AWS CLI commande pour redémarrer l' pour SQL instance de base de données Postgre et initialiser la bibliothèque. `pg_tle`

```
aws rds reboot-db-instance \
    --db-instance-identifier your-instance \
    --region aws-region
```

3. Lorsque l'instance est disponible, vous pouvez vérifier que `pg_tle` a été initialisé. Utilisez-le `psql` pour vous connecter à l' pour l'SQLinstance de base de données Postgre, puis exécutez la commande suivante.

```
SHOW shared_preload_libraries;
shared_preload_libraries
--------------------------
rdsutils,pg_tle
```

```
(1 row)
```

Une fois `pg_tle` initialisé, vous pouvez maintenant créer l'extension.

```
CREATE EXTENSION pg_tle;
```

4. Accordez le `pgtle_admin` rôle au nom d'utilisateur principal que vous avez créé pour votre instance de votre SQL instance de base de données Postgre lors de sa configuration. Si vous avez accepté la valeur par défaut, il s'agit de `postgres`.

```
GRANT pgtle_admin TO postgres;
GRANT ROLE
```

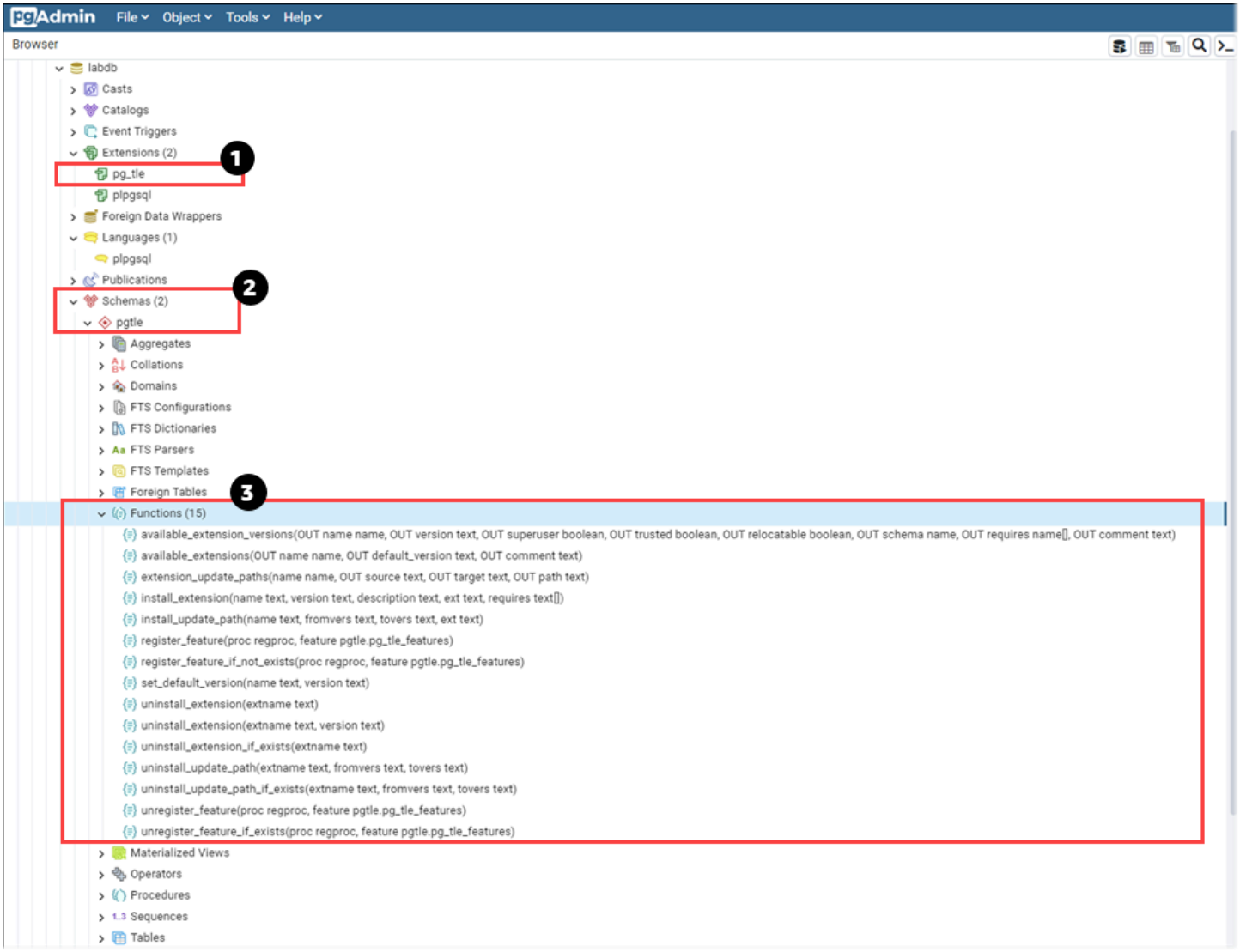
5. Fermez la session `psql` comme suit.

```
labdb=> \q
```

Pour commencer à créer des TLE extensions, consultezExemple : création d'une extension de langue fiable à l'aide de SQL.

Vue d'ensemble des extensions linguistiques fiables pour Postgre SQL

Trusted Language Extensions for Postgre SQL est une SQL extension Postgre que vous installez dans votre pour une SQL instance de base de données Postgre de la même manière que vous configurez les autres extensions Postgre. SQL Dans l'image suivante d'un exemple de base de données dans l'outil pgAdmin client, vous pouvez visualiser certains des composants composant l'`pg_tle`extension.



Vous pouvez voir les détails suivants.

1. Le kit de SQL développement Trusted Language Extensions (TLE) pour Postgre est fourni sous forme d'`pg_tle`extension. En tant que tel, `pg_tle` est ajouté aux extensions disponibles pour la base de données dans laquelle il est installé.
2. TLEpossède son propre schéma,`pgtle`. Ce schéma contient des fonctions d'assistance (3) pour installer et gérer les extensions que vous créez.
3. TLEfournit plus d'une douzaine de fonctions d'assistance pour l'installation, l'enregistrement et la gestion de vos extensions. Pour en savoir plus sur ces fonctions, consultez Référence fonctionnelle pour les extensions linguistiques fiables pour Postgre SQL.

L'extension `pg_tle` comprend les autres composants suivants :

- Le rôle **`pgtle_admin`** : le rôle `pgtle_admin` est créé lors de l'installation de l'extension `pg_tle`. Ce rôle est privilégié et doit être traité comme tel. Nous vous recommandons vivement de respecter le principe du moindre privilège lorsque vous accordez le rôle `pgtle_admin` aux utilisateurs de base de données. En d'autres termes, accordez le `pgtle_admin` rôle uniquement aux utilisateurs de base de données autorisés à créer, installer et gérer de nouvelles TLE extensions, telles que`postgres`.
- La **`pgtle.feature_info`** table — La `pgtle.feature_info` table est une table protégée qui contient des informations sur vos crochetsTLEs, ainsi que sur les procédures stockées personnalisées et les fonctions qu'ils utilisent. Si vous disposez de privilèges `pgtle_admin`, vous pouvez utiliser les fonctions Trusted Language Extensions suivantes pour ajouter et mettre à jour ces informations dans la table.
  - pgtle.register_feature
  - pgtle.register_feature_if_not_exists
  - pgtle.unregister_feature
  - pgtle.unregister_feature_if_exists

Création d'TLEextensions pour SQL

Vous pouvez installer toutes les extensions que vous créez TLE dans n'importe quel pour SQL instance de base de données Postgre sur laquelle l'`pg_tle`extension est installée. L'`pg_tle`extension est limitée à la base de SQL données Postgre dans laquelle elle est installée. Les extensions que vous créez à l'aide TLE sont limitées à la même base de données.

Utilisez les différentes `pgtle` fonctions pour installer le code qui constitue votre TLE extension. Les fonctions Trusted Language Extensions suivantes nécessitent toutes le rôle `pgtle_admin`.

- pgtle.install_extension
- pgtle.install_update_path
- pgtle.register_feature
- pgtle.register_feature_if_not_exists
- pgtle.set_default_version
- pgtle.uninstall_extension(name)
- pgtle.uninstall_extension(name, version)
- pgtle.uninstall_extension_if_exists

- pgtle.uninstall_update_path
- pgtle.uninstall_update_path_if_exists
- pgtle.unregister_feature
- pgtle.unregister_feature_if_exists

Exemple : création d'une extension de langue fiable à l'aide de SQL

L'exemple suivant montre comment créer une TLE extension nommée contenant quelques SQL fonctions `pg_distance` permettant de calculer des distances à l'aide de différentes formules. Dans la liste, vous trouverez la fonction de calcul de la distance de Manhattan et la fonction de calcul de la distance euclidienne. Pour plus d'informations sur la différence entre ces formules, consultez Taxicab geometry (Géométrie de Manhattan) et Euclidean geometry (Géométrie euclidienne) sur Wikipedia.

Vous pouvez utiliser cet exemple dans votre propre pour une SQL instance de base de données Postgre si l'`pg_tle`extension est configurée comme indiqué dans. Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL

> **Note**
>
> Vous devez disposer des privilèges du rôle `pgtle_admin` pour suivre cette procédure.

Pour créer l'TLEextension d'exemple

Les étapes suivantes utilisent un exemple de base de données nommé `labdb`. Cette base de données appartient à l'utilisateur principal `postgres`. Le rôle `postgres` possède également les autorisations du rôle `pgtle_admin`.

1. Utilisez-le `psql` pour vous connecter à . RDSpour une SQL instance de base de données Postgre.

   ```
   psql --host=db-instance-123456789012.aws-region.rds.amazonaws.com
   --port=5432 --username=postgres --password --dbname=labdb
   ```

2. Créez une TLE extension nommée `pg_distance` en copiant le code suivant et en le collant dans votre console de `psql` session.

   ```
   SELECT pgtle.install_extension
   ```

```
(
 'pg_distance',
 '0.1',
  'Distance functions for two points',
$_pg_tle_$
    CREATE FUNCTION dist(x1 float8, y1 float8, x2 float8, y2 float8, norm int)
    RETURNS float8
    AS $$
      SELECT (abs(x2 - x1) ^ norm + abs(y2 - y1) ^ norm) ^ (1::float8 / norm);
    $$ LANGUAGE SQL;

    CREATE FUNCTION manhattan_dist(x1 float8, y1 float8, x2 float8, y2 float8)
    RETURNS float8
    AS $$
      SELECT dist(x1, y1, x2, y2, 1);
    $$ LANGUAGE SQL;

    CREATE FUNCTION euclidean_dist(x1 float8, y1 float8, x2 float8, y2 float8)
    RETURNS float8
    AS $$
      SELECT dist(x1, y1, x2, y2, 2);
    $$ LANGUAGE SQL;
$_pg_tle_$
);
```

Vous devez voir la sortie suivante.

```
install_extension
---------------
 t
(1 row)
```

Les artefacts qui constituent l'extension `pg_distance` sont désormais installés dans votre base de données. Ces artefacts incluent le fichier de contrôle et le code de l'extension, qui sont des éléments qui doivent être présents pour que l'extension puisse être créée à l'aide de la commande `CREATE EXTENSION`. En d'autres termes, il vous reste à créer l'extension pour mettre ses fonctions à la disposition des utilisateurs de la base de données.

3. Pour créer cette extension, utilisez la commande `CREATE EXTENSION` comme vous le feriez pour toute autre extension. Comme pour les autres extensions, l'utilisateur de la base de données doit disposer des autorisations `CREATE` dans la base de données.

```
CREATE EXTENSION pg_distance;
```

4. Pour tester l'`pg_distance`TLEextension, vous pouvez l'utiliser pour calculer la distance de Manhattan entre quatre points.

   ```
   labdb=> SELECT manhattan_dist(1, 1, 5, 5);
   8
   ```

   Pour calculer la distance euclidienne entre le même ensemble de points, vous pouvez utiliser ce qui suit.

   ```
   labdb=> SELECT euclidean_dist(1, 1, 5, 5);
   5.656854249492381
   ```

L'extension `pg_distance` charge les fonctions dans la base de données et les met à la disposition de tous les utilisateurs dotés d'autorisations sur la base de données.

Modifier votre TLE extension

Pour améliorer les performances des requêtes pour les fonctions incluses dans cette TLE extension, ajoutez les deux SQL attributs Postgre suivants à leurs spécifications.

- `IMMUTABLE` : l'attribut `IMMUTABLE` garantit que l'optimiseur de requêtes peut utiliser des optimisations pour améliorer les temps de réponse aux requêtes. Pour plus d'informations, consultez la section Catégories de volatilité des fonctions dans la SQL documentation Postgre.
- `PARALLEL SAFE`— L'`PARALLEL SAFE`attribut est un autre attribut qui permet à Postgre SQL d'exécuter la fonction en mode parallèle. Pour plus d'informations, consultez CREATEFUNCTIONla SQL documentation Postgre.

Dans l'exemple suivant, vous pouvez voir comment la `pgtle.install_update_path` fonction est utilisée pour ajouter ces attributs à chaque fonction afin de créer une version `0.2` de l'`pg_distance`TLEextension. Pour de plus amples informations sur cette fonction, veuillez consulter pgtle.install_update_path. Vous devez avoir le rôle `pgtle_admin` pour effectuer cette tâche.

Pour mettre à jour une TLE extension existante et spécifier la version par défaut

1. Connectez-vous à pour SQL instance de base de données Postgre à l'aide `psql` d'un autre outil client, tel que. pgAdmin

```
psql --host=db-instance-123456789012.aws-region.rds.amazonaws.com
--port=5432 --username=postgres --password --dbname=labdb
```

2. Modifiez l'TLEextension existante en copiant le code suivant et en le collant dans votre console de `psql` session.

```
SELECT pgtle.install_update_path
(
 'pg_distance',
 '0.1',
 '0.2',
$_pg_tle_$
    CREATE OR REPLACE FUNCTION dist(x1 float8, y1 float8, x2 float8, y2 float8,
 norm int)
    RETURNS float8
    AS $$
      SELECT (abs(x2 - x1) ^ norm + abs(y2 - y1) ^ norm) ^ (1::float8 / norm);
    $$ LANGUAGE SQL IMMUTABLE PARALLEL SAFE;

    CREATE OR REPLACE FUNCTION manhattan_dist(x1 float8, y1 float8, x2 float8, y2
 float8)
    RETURNS float8
    AS $$
      SELECT dist(x1, y1, x2, y2, 1);
    $$ LANGUAGE SQL IMMUTABLE PARALLEL SAFE;

    CREATE OR REPLACE FUNCTION euclidean_dist(x1 float8, y1 float8, x2 float8, y2
 float8)
    RETURNS float8
    AS $$
      SELECT dist(x1, y1, x2, y2, 2);
    $$ LANGUAGE SQL IMMUTABLE PARALLEL SAFE;
$_pg_tle_$
);
```

Vous voyez une réponse similaire à la suivante.

```
install_update_path
--------------------
 t
(1 row)
```

Vous pouvez faire de cette version de l'extension la version par défaut, afin que les utilisateurs de base de données n'aient pas à spécifier de version lorsqu'ils créent ou mettent à jour l'extension dans leur base de données.

3. Pour indiquer que la version modifiée (version 0.2) de votre TLE extension est la version par défaut, utilisez la `pgtle.set_default_version` fonction illustrée dans l'exemple suivant.

```
SELECT pgtle.set_default_version('pg_distance', '0.2');
```

Pour de plus amples informations sur cette fonction, veuillez consulter pgtle.set_default_version.

4. Une fois le code en place, vous pouvez mettre à jour l'TLEextension installée de la manière habituelle, en utilisant `ALTER EXTENSION ... UPDATE` la commande, comme indiqué ici :

```
ALTER EXTENSION pg_distance UPDATE;
```

Supprimer vos TLE extensions d'une base de données

Vous pouvez supprimer vos TLE extensions en utilisant la `DROP EXTENSION` commande de la même manière que pour les autres SQL extensions Postgre. La suppression de l'extension ne supprime pas les fichiers d'installation qui composent l'extension et qui permettent aux utilisateurs de la recréer. Pour supprimer l'extension et ses fichiers d'installation, effectuez la procédure en deux étapes suivante.

Pour supprimer l'TLEextension et supprimer ses fichiers d'installation

1. Utilisez `psql` un autre outil client pour vous connecter à l'. RDSpour une SQL instance de base de données Postgre.

```
psql --host=.111122223333.aws-region.rds.amazonaws.com --port=5432 --
username=postgres --password --dbname=dbname
```

2. Supprimez l'extension comme vous le feriez pour n'importe quelle SQL extension Postgre.

```
DROP EXTENSION your-TLE-extension
```

Par exemple, si vous créez l'extension `pg_distance` comme détaillé dans Exemple : création d'une extension de langue fiable à l'aide de SQL, vous pouvez la supprimer comme suit.

```
DROP EXTENSION pg_distance;
```

Vous voyez une sortie confirmant que l'extension a été supprimée, comme suit.

```
DROP EXTENSION
```

À ce stade, l'extension n'est plus active dans la base de données. Cependant, ses fichiers d'installation et son fichier de contrôle sont toujours disponibles dans la base de données, ce qui permet aux utilisateurs de la base de données de recréer l'extension s'ils le souhaitent.

- Si vous souhaitez conserver les fichiers d'extension intacts afin que les utilisateurs de la base de données puissent créer votre TLE extension, vous pouvez vous arrêter ici.
- Pour supprimer tous les fichiers qui composent l'extension, passez à l'étape suivante.

3. Pour supprimer tous les fichiers d'installation de votre extension, utilisez la fonction `pgtle.uninstall_extension`. Cette fonction supprime tous les fichiers de code et de contrôle relatifs à votre extension.

```
SELECT pgtle.uninstall_extension('your-tle-extension-name');
```

Par exemple, pour supprimer tous les fichiers d'installation `pg_distance`, utilisez la commande suivante.

```
SELECT pgtle.uninstall_extension('pg_distance');
 uninstall_extension
---------------------
 t
(1 row)
```

Désinstaller les extensions linguistiques fiables pour Postgre SQL

Si vous ne souhaitez plus créer vos propres TLE extensions en utilisantTLE, vous pouvez supprimer l'`pg_tle`extension et supprimer tous les artefacts. Cette action inclut la suppression de toutes les TLE extensions de la base de données et la suppression du `pgtle` schéma.

Pour supprimer l'extension **`pg_tle`** et son schéma d'une base de données

1. Utilisez `psql` un autre outil client pour vous connecter à l'. RDSpour une SQL instance de base de données Postgre.

   ```
   psql --host=.111122223333.aws-region.rds.amazonaws.com --port=5432 --
   username=postgres --password --dbname=dbname
   ```

2. Supprimez l'extension `pg_tle` de la base de données. Si vos propres TLE extensions sont toujours actives dans la base de données, vous devez également les supprimer. Pour ce faire, vous pouvez utiliser le mot clé `CASCADE`, comme illustré ci-dessous.

   ```
   DROP EXTENSION pg_tle CASCADE;
   ```

   Si l'extension `pg_tle` n'est toujours pas active dans la base de données, vous n'avez pas besoin d'utiliser le mot clé `CASCADE`.

3. Supprimez le schéma `pgtle`. Cette action supprime toutes les fonctions de gestion de la base de données.

   ```
   DROP SCHEMA pgtle CASCADE;
   ```

   La commande renvoie ce qui suit une fois le processus terminé.

   ```
   DROP SCHEMA
   ```

   L'extension `pg_tle`, son schéma et ses fonctions, ainsi que tous les artefacts sont supprimés. Pour créer de nouvelles extensions à l'aide deTLE, recommencez le processus de configuration. Pour de plus amples informations, veuillez consulter Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL.

Utilisation des hooks PostgreSQL avec vos extensions TLE

Un hook est un mécanisme de rappel disponible dans PostgreSQL qui permet aux développeurs d'appeler des fonctions personnalisées ou d'autres routines lors d'opérations de base de données normales. Le kit de développement TLE prend en charge les hooks PostgreSQL afin que vous puissiez intégrer des fonctions personnalisées au comportement PostgreSQL au moment de l'exécution. Par exemple, vous pouvez utiliser un hook pour associer le processus d'authentification à votre propre code personnalisé, ou pour modifier le processus de planification et d'exécution des requêtes en fonction de vos besoins spécifiques.

Vos extensions TLE peuvent utiliser des hooks. Si un hook a une portée globale, il s'applique à toutes les bases de données. Par conséquent, si votre extension TLE utilise un hook global, vous devez créer votre extension TLE dans toutes les bases de données auxquelles vos utilisateurs peuvent accéder.

Lorsque vous utilisez l'extension `pg_tle` pour créer votre propre kit Trusted Language Extensions, vous pouvez utiliser les hooks disponibles à partir d'une API SQL pour développer les fonctions de votre extension. Vous devez enregistrer tous les hooks avec `pg_tle`. Pour certains hooks, vous devrez peut-être également définir différents paramètres de configuration. Par exemple, le hook de vérification `passcode` peut être activé, désactivé ou requis. Pour plus d'informations sur les exigences spécifiques relatives aux hooks `pg_tle` disponibles, consultez Référence des hooks pour Trusted Language Extensions pour PostgreSQL.

Exemple : création d'une extension utilisant un hook PostgreSQL

L'exemple présenté dans cette section utilise un hook PostgreSQL pour vérifier le mot de passe fourni lors d'opérations SQL spécifiques et empêche les utilisateurs de base de données de définir leurs mots de passe sur l'un de ceux contenus dans la table `password_check.bad_passwords`. La table contient les dix choix de mots de passe les plus couramment utilisés, mais les plus faciles à déchiffrer.

Pour configurer cet exemple dans votre instance de base de données RDS for PostgreSQL, vous devez avoir déjà installé Trusted Language Extensions. Pour en savoir plus, consultez Configuration d'extensions linguistiques fiables dans votre de base de données Postgre SQL.

Pour configurer l'exemple de hook de vérification de mot de passe

1. Utilisez `psql` pour vous connecter à Instance de base de données RDS for PostgreSQL.

```
psql --host=db-instance-123456789012.aws-region.rds.amazonaws.com
```

```
--port=5432 --username=postgres --password --dbname=labdb
```

2. Copiez le code à partir de Listing du code du hook de vérification de mot de passe et collez-le dans votre base de données.

```
SELECT pgtle.install_extension (
  'my_password_check_rules',
  '1.0',
  'Do not let users use the 10 most commonly used passwords',
$_pgtle_$
  CREATE SCHEMA password_check;
  REVOKE ALL ON SCHEMA password_check FROM PUBLIC;
  GRANT USAGE ON SCHEMA password_check TO PUBLIC;

  CREATE TABLE password_check.bad_passwords (plaintext) AS
  VALUES
    ('123456'),
    ('password'),
    ('12345678'),
    ('qwerty'),
    ('123456789'),
    ('12345'),
    ('1234'),
    ('111111'),
    ('1234567'),
    ('dragon');
  CREATE UNIQUE INDEX ON password_check.bad_passwords (plaintext);

  CREATE FUNCTION password_check.passcheck_hook(username text, password text,
 password_type pgtle.password_types, valid_until timestamptz, valid_null boolean)
  RETURNS void AS $$
    DECLARE
      invalid bool := false;
    BEGIN
      IF password_type = 'PASSWORD_TYPE_MD5' THEN
        SELECT EXISTS(
          SELECT 1
          FROM password_check.bad_passwords bp
          WHERE ('md5' || md5(bp.plaintext || username)) = password
        ) INTO invalid;
        IF invalid THEN
          RAISE EXCEPTION 'Cannot use passwords from the common password
 dictionary';
```

```
        END IF;
      ELSIF password_type = 'PASSWORD_TYPE_PLAINTEXT' THEN
        SELECT EXISTS(
          SELECT 1
          FROM password_check.bad_passwords bp
          WHERE bp.plaintext = password
        ) INTO invalid;
        IF invalid THEN
          RAISE EXCEPTION 'Cannot use passwords from the common password
 dictionary';
        END IF;
      END IF;
    END
  $$ LANGUAGE plpgsql SECURITY DEFINER;

  GRANT EXECUTE ON FUNCTION password_check.passcheck_hook TO PUBLIC;

  SELECT pgtle.register_feature('password_check.passcheck_hook', 'passcheck');
$_pgtle_$
);
```

Lorsque l'extension a été chargée dans votre base de données, le résultat suivant s'affiche.

```
 install_extension
-------------------
 t
(1 row)
```

3. Toujours connecté à la base de données, vous pouvez maintenant créer l'extension.

```
CREATE EXTENSION my_password_check_rules;
```

4. Vous pouvez confirmer que l'extension a été créée dans la base de données à l'aide de la métacommande `psql` suivante.

```
\dx
                        List of installed extensions
          Name           | Version |   Schema   |
 Description
-------------------------+---------+------------
+-------------------------------------------------------------
```

```
 my_password_check_rules | 1.0     | public     | Prevent use of any of the top-ten
 most common bad passwords
 pg_tle                  | 1.0.1   | pgtle      | Trusted-Language Extensions for
 PostgreSQL
 plpgsql                 | 1.0     | pg_catalog | PL/pgSQL procedural language
(3 rows)
```

5. Ouvrez une autre session de terminal pour travailler avec le AWS CLI. Vous devez modifier votre groupe de paramètres de base de données personnalisé pour activer le hook de vérification de mot de passe. Pour ce faire, utilisez la commande modify-db-parameter-groupCLI comme indiqué dans l'exemple suivant.

```
aws rds modify-db-parameter-group \
    --region aws-region \
    --db-parameter-group-name your-custom-parameter-group \
    --parameters
 "ParameterName=pgtle.enable_password_check,ParameterValue=on,ApplyMethod=immediate"
```

   Lorsque le paramètre est activé avec succès, le résultat suivant s'affiche.

```
(
    "DBParameterGroupName": "docs-lab-parameters-for-tle"
}
```

   L'application de la modification du groupe de paramètres peut prendre quelques minutes. Toutefois, ce paramètre étant dynamique, vous n'avez pas besoin de redémarrer l'instance de base de données RDS for PostgreSQL pour que le paramètre prenne effet.

6. Ouvrez la session `psql` et interrogez la base de données pour vérifier que le hook password_check a été activé.

```
labdb=> SHOW pgtle.enable_password_check;
pgtle.enable_password_check
----------------------------
on
(1 row)
```

Le hook de vérification de mot de passe est désormais actif. Vous pouvez le tester en créant un nouveau rôle et en utilisant l'un des mauvais mots de passe, comme illustré dans l'exemple suivant.

```
CREATE ROLE test_role PASSWORD 'password';
ERROR:  Cannot use passwords from the common password dictionary
CONTEXT:  PL/pgSQL function
 password_check.passcheck_hook(text,text,pgtle.password_types,timestamp with time
 zone,boolean) line 21 at RAISE
SQL statement "SELECT password_check.passcheck_hook(
    $1::pg_catalog.text,
    $2::pg_catalog.text,
    $3::pgtle.password_types,
    $4::pg_catalog.timestamptz,
    $5::pg_catalog.bool)"
```

La sortie a été formatée pour être lisible.

L'exemple suivant montre que le comportement `pgsql` de la métacomcommande interactive `\password` est également affecté par le hook password_check.

```
postgres=> SET password_encryption TO 'md5';
SET
postgres=> \password
Enter new password for user "postgres":*****
Enter it again:*****
ERROR:  Cannot use passwords from the common password dictionary
CONTEXT:  PL/pgSQL function
 password_check.passcheck_hook(text,text,pgtle.password_types,timestamp with time
 zone,boolean) line 12 at RAISE
SQL statement "SELECT password_check.passcheck_hook($1::pg_catalog.text,
 $2::pg_catalog.text, $3::pgtle.password_types, $4::pg_catalog.timestamptz,
 $5::pg_catalog.bool)"
```

Vous pouvez supprimer cette extension TLE et désinstaller ses fichiers sources si vous le souhaitez. Pour de plus amples informations, veuillez consulter Supprimer vos TLE extensions d'une base de données.

Listing du code du hook de vérification de mot de passe

L'exemple de code affiché ici définit la spécification de l'extension TLE `my_password_check_rules`. Lorsque vous copiez ce code et que vous le collez dans votre base de données, le code de l'extension `my_password_check_rules` est chargé dans la base de données et le hook `password_check` est enregistré pour être utilisé par l'extension.

```
SELECT pgtle.install_extension (
```

```
  'my_password_check_rules',
  '1.0',
  'Do not let users use the 10 most commonly used passwords',
$_pgtle_$
  CREATE SCHEMA password_check;
  REVOKE ALL ON SCHEMA password_check FROM PUBLIC;
  GRANT USAGE ON SCHEMA password_check TO PUBLIC;

  CREATE TABLE password_check.bad_passwords (plaintext) AS
  VALUES
    ('123456'),
    ('password'),
    ('12345678'),
    ('qwerty'),
    ('123456789'),
    ('12345'),
    ('1234'),
    ('111111'),
    ('1234567'),
    ('dragon');
  CREATE UNIQUE INDEX ON password_check.bad_passwords (plaintext);

  CREATE FUNCTION password_check.passcheck_hook(username text, password text,
 password_type pgtle.password_types, valid_until timestamptz, valid_null boolean)
  RETURNS void AS $$
    DECLARE
      invalid bool := false;
    BEGIN
      IF password_type = 'PASSWORD_TYPE_MD5' THEN
        SELECT EXISTS(
          SELECT 1
          FROM password_check.bad_passwords bp
          WHERE ('md5' || md5(bp.plaintext || username)) = password
        ) INTO invalid;
        IF invalid THEN
          RAISE EXCEPTION 'Cannot use passwords from the common password dictionary';
        END IF;
      ELSIF password_type = 'PASSWORD_TYPE_PLAINTEXT' THEN
        SELECT EXISTS(
          SELECT 1
          FROM password_check.bad_passwords bp
          WHERE bp.plaintext = password
        ) INTO invalid;
        IF invalid THEN
```

```
          RAISE EXCEPTION 'Cannot use passwords from the common password dictionary';
        END IF;
      END IF;
    END
  $$ LANGUAGE plpgsql SECURITY DEFINER;

  GRANT EXECUTE ON FUNCTION password_check.passcheck_hook TO PUBLIC;

  SELECT pgtle.register_feature('password_check.passcheck_hook', 'passcheck');
$_pgtle_$
);
```

Utilisation de types de données personnalisés dans TLE

Postgre SQL prend en charge les commandes pour enregistrer de nouveaux types de base (également appelés types scalaires) afin de gérer efficacement les structures de données complexes dans votre base de données. Un type de base vous permet de personnaliser la façon dont les données sont stockées en interne et la façon de les convertir vers et depuis une représentation textuelle externe. Ces types de données personnalisés sont utiles pour étendre Postgre SQL afin de prendre en charge des domaines fonctionnels dans lesquels un type intégré tel qu'un nombre ou un texte ne peut pas fournir une sémantique de recherche suffisante.

RDSfor Postgre vous SQL permet de créer des types de données personnalisés dans votre extension de langage fiable et de définir des fonctions qui prennent en charge SQL et indexent les opérations pour ces nouveaux types de données. Les types de données personnalisés sont disponibles pour les versions suivantes :

- RDSpour Postgre SQL 15.4 et versions 15 supérieures
- RDSpour Postgre SQL 14.9 et versions 14 supérieures
- RDSpour Postgre SQL 13.12 et versions 13 supérieures

Pour plus d'informations, consultez Types de base de langage approuvé (langue française non garantie).

Référence fonctionnelle pour les extensions linguistiques fiables pour Postgre SQL

Consultez la documentation de référence suivante sur les fonctions disponibles dans Trusted Language Extensions for PostgreSQL. Utilisez ces fonctions pour installer, enregistrer, mettre à jour

et gérer vos TLEextensions, c'est-à-dire les extensions Postgre SQL que vous développez à l'aide du kit de développement Trusted Language Extensions.

Fonctions

- pgtle.available_extensions
- pgtle.available_extension_versions
- pgtle.extension_update_paths
- pgtle.install_extension
- pgtle.install_update_path
- pgtle.register_feature
- pgtle.register_feature_if_not_exists
- pgtle.set_default_version
- pgtle.uninstall_extension(name)
- pgtle.uninstall_extension(name, version)
- pgtle.uninstall_extension_if_exists
- pgtle.uninstall_update_path
- pgtle.uninstall_update_path_if_exists
- pgtle.unregister_feature
- pgtle.unregister_feature_if_exists

pgtle.available_extensions

La fonction `pgtle.available_extensions` est une fonction à renvoi d'ensemble. Elle renvoie toutes les TLE extensions disponibles dans la base de données. Chaque ligne renvoyée contient des informations sur une seule TLE extension.

Prototype de fonction

```
pgtle.available_extensions()
```

Rôle

Aucune.

Arguments

Aucune.

Sortie

- `name`— Le nom de l'TLEextension.
- `default_version`— Version de l'TLEextension à utiliser lorsqu'elle `CREATE EXTENSION` est appelée sans qu'une version soit spécifiée.
- `description`— Une description plus détaillée de l'TLEextension.

Exemple d'utilisation

```
SELECT * FROM pgtle.available_extensions();
```

pgtle.available_extension_versions

La fonction `available_extension_versions` est une fonction à renvoi d'ensemble. Elle renvoie une liste de toutes les TLE extensions disponibles et de leurs versions. Chaque ligne contient des informations sur une version spécifique de l'TLEextension donnée, notamment si elle nécessite un rôle spécifique.

Prototype de fonction

```
pgtle.available_extension_versions()
```

Rôle

Aucune.

Arguments

Aucune.

Sortie

- `name`— Le nom de l'TLEextension.
- `version`— La version de l'TLEextension.

- `superuser`— Cette valeur s'applique toujours `false` à vos TLE extensions. Les autorisations nécessaires pour créer l'TLEextension ou la mettre à jour sont les mêmes que pour créer d'autres objets dans la base de données donnée.
- `trusted`— Cette valeur correspond toujours `false` à une TLE extension.
- `relocatable`— Cette valeur correspond toujours `false` à une TLE extension.
- `schema`— Spécifie le nom du schéma dans lequel l'TLEextension est installée.
- `requires`— Un tableau contenant les noms des autres extensions nécessaires à cette TLE extension.
- `description`— Une description détaillée de l'TLEextension.

Pour plus d'informations sur les valeurs de sortie, voir Empaqueter des objets associés dans une extension > Fichiers d'extension dans la SQL documentation Postgre.

Exemple d'utilisation

```
SELECT * FROM pgtle.available_extension_versions();
```

pgtle.extension_update_paths

La fonction `extension_update_paths` est une fonction à renvoi d'ensemble. Elle renvoie une liste de tous les chemins de mise à jour possibles pour une TLE extension. Chaque ligne inclut les mises à niveau ou rétrogradations disponibles pour cette TLE extension.

Prototype de fonction

```
pgtle.extension_update_paths(name)
```

Rôle

Aucune.

Arguments

`name`— Le nom de l'TLEextension à partir de laquelle obtenir les chemins de mise à niveau.

Sortie

- `source` : version source d'une mise à jour.
- `target` : version cible d'une mise à jour.

- `path`— Le chemin de mise à niveau utilisé pour mettre à jour une TLE extension d'une `source` `target` version à l'autre, par exemple,`0.1--0.2`.

Exemple d'utilisation

```
SELECT * FROM pgtle.extension_update_paths('your-TLE');
```

pgtle.install_extension

La `install_extension` fonction vous permet d'installer les artefacts qui constituent votre TLE extension dans la base de données, après quoi elle peut être créée à l'aide de la `CREATE EXTENSION` commande.

Prototype de fonction

```
pgtle.install_extension(name text, version text, description text, ext text, requires
 text[] DEFAULT NULL::text[])
```

Rôle

Aucune.

Arguments

- `name`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `version`— La version de l'TLEextension.
- `description`— Une description détaillée de l'TLEextension. Cette description est affichée dans le champ `comment` de `pgtle.available_extensions()`.
- `ext`— Le contenu de l'TLEextension. Cette valeur contient des objets tels que des fonctions.
- `requires`— Paramètre facultatif qui spécifie les dépendances de cette TLE extension. L'extension `pg_tle` est automatiquement ajoutée en tant que dépendance.

La plupart de ces arguments sont identiques à ceux inclus dans un fichier de contrôle d'extension pour installer une SQL extension Postgre sur le système de fichiers d'une instance PostgreSQL. Pour plus d'informations, consultez les fichiers d'extension dans la section Empaqueter des objets associés dans une extension dans la SQL documentation Postgre.

Sortie

Cette fonction renvoie `OK` en cas de réussite ou `NULL` en cas d'erreur.

- `OK`— L'TLEextension a été correctement installée dans la base de données.
- `NULL`— L'TLEextension n'a pas été correctement installée dans la base de données.

Exemple d'utilisation

```
SELECT pgtle.install_extension(
 'pg_tle_test',
 '0.1',
 'My first pg_tle extension',
$_pgtle_$
  CREATE FUNCTION my_test()
  RETURNS INT
  AS $$
    SELECT 42;
  $$ LANGUAGE SQL IMMUTABLE;
$_pgtle_$
);
```

pgtle.install_update_path

La `install_update_path` fonction fournit un chemin de mise à jour entre deux versions différentes d'une TLE extension. Cette fonction permet aux utilisateurs de votre TLE extension de mettre à jour sa version en utilisant la `ALTER EXTENSION ... UPDATE` syntaxe.

Prototype de fonction

```
pgtle.install_update_path(name text, fromvers text, tovers text, ext text)
```

Rôle

`pgtle_admin`

Arguments

- `name`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `fromvers`— La version source de l'TLEextension pour la mise à niveau.

- `tovers`— Version de destination de l'TLEextension pour la mise à niveau.
- `ext` : contenu de la mise à jour. Cette valeur contient des objets tels que des fonctions.

Sortie

Aucune.

Exemple d'utilisation

```
SELECT pgtle.install_update_path('pg_tle_test', '0.1', '0.2',
  $_pgtle_$
    CREATE OR REPLACE FUNCTION my_test()
    RETURNS INT
    AS $$
      SELECT 21;
    $$ LANGUAGE SQL IMMUTABLE;
  $_pgtle_$
);
```

pgtle.register_feature

La `register_feature` fonction ajoute la SQL fonctionnalité interne Postgre spécifiée à la `pgtle.feature_info` table. Les SQL hooks Postgre sont un exemple de fonctionnalité interne de PostgreSQL. Le kit de développement Trusted Language Extensions prend en charge l'utilisation des SQL hooks Postgre. Actuellement, cette fonction prend en charge la fonctionnalité suivante.

- `passcheck`— Enregistre le crochet de vérification des mots de passe avec votre procédure ou fonction qui personnalise le comportement de vérification des mots de passe de SQL Postgre.

Prototype de fonction

```
pgtle.register_feature(proc regproc, feature pg_tle_feature)
```

Rôle

`pgtle_admin`

Arguments

- `proc` : nom d'une procédure ou d'une fonction stockée à utiliser pour la fonctionnalité.

- `feature` : nom de la fonctionnalité `pg_tle` (tel que `passcheck`) à enregistrer avec la fonction.

Sortie

Aucune.

Exemple d'utilisation

```
SELECT pgtle.register_feature('pw_hook', 'passcheck');
```

pgtle.register_feature_if_not_exists

La `pgtle.register_feature_if_not_exists` fonction ajoute la SQL fonctionnalité Postgre spécifiée au `pgtle.feature_info` tableau et identifie l'TLEextension ou l'autre procédure ou fonction qui utilise cette fonctionnalité. Pour plus d'informations sur les hooks et le kit Trusted Language Extensions, consultez Utilisation des hooks PostgreSQL avec vos extensions TLE.

Prototype de fonction

```
pgtle.register_feature_if_not_exists(proc regproc, feature pg_tle_feature)
```

Rôle

`pgtle_admin`

Arguments

- `proc`— Le nom d'une procédure stockée ou d'une fonction contenant la logique (code) à utiliser comme fonctionnalité pour votre TLE extension. Par exemple, le code `pw_hook`.
- `feature`— Le nom de la SQL fonctionnalité Postgre à enregistrer pour la TLE fonction. Actuellement, la seule fonctionnalité disponible est le hook `passcheck`. Pour de plus amples informations, veuillez consulter Crochet de vérification du mot de passe (passcheck).

Sortie

Renvoie `true` après l'enregistrement de la fonction pour l'extension spécifiée. Renvoie `false` si la fonctionnalité est déjà enregistrée.

Exemple d'utilisation

```
SELECT pgtle.register_feature_if_not_exists('pw_hook', 'passcheck');
```

pgtle.set_default_version

La `set_default_version` fonction vous permet de spécifier un `default_version` pour votre TLE extension. Vous pouvez utiliser cette fonction pour définir un chemin de mise à niveau et désigner la version par défaut pour votre TLE extension. Lorsque les utilisateurs de base de données spécifient votre TLE extension dans les `ALTER EXTENSION ... UPDATE` commandes `CREATE EXTENSION` et, cette version de votre TLE extension est créée dans la base de données pour cet utilisateur.

Cette fonction renvoie `true` en cas de réussite. Si l'TLEextension spécifiée dans l'`name`argument n'existe pas, la fonction renvoie une erreur. De même, si `version` l'TLEextension n'existe pas, elle renvoie une erreur.

Prototype de fonction

```
pgtle.set_default_version(name text, version text)
```

Rôle

`pgtle_admin`

Arguments

- `name`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `version`— Version de l'TLEextension pour définir la valeur par défaut.

Sortie

- `true` : lorsque la définition de la version par défaut réussit, la fonction renvoie `true`.
- `ERROR`— Renvoie un message d'erreur si aucune TLE extension portant le nom ou la version spécifiés n'existe.

Exemple d'utilisation

```
SELECT * FROM pgtle.set_default_version('my-extension', '1.1');
```

pgtle.uninstall_extension(name)

La `uninstall_extension` fonction supprime toutes les versions d'une TLE extension d'une base de données. Cette fonction empêche les futurs appels `CREATE EXTENSION` de pour installer l'TLEextension. Si l'TLEextension n'existe pas dans la base de données, une erreur est générée.

La `uninstall_extension` fonction ne supprime pas une TLE extension actuellement active dans la base de données. Pour supprimer une TLE extension actuellement active, vous devez appeler explicitement `DROP EXTENSION` pour la supprimer.

Prototype de fonction

```
pgtle.uninstall_extension(extname text)
```

Rôle

`pgtle_admin`

Arguments

- `extname`— Nom de l'TLEextension à désinstaller. Ce nom est le même que celui utilisé pour charger l'TLEextension `CREATE EXTENSION` à utiliser dans une base de données donnée.

Sortie

Aucune.

Exemple d'utilisation

```
SELECT * FROM pgtle.uninstall_extension('pg_tle_test');
```

pgtle.uninstall_extension(name, version)

La `uninstall_extension(name, version)` fonction supprime la version spécifiée de l'TLEextension de la base de données. Cette fonction empêche `CREATE EXTENSION` et `ALTER EXTENSION` empêche l'installation ou la mise à jour d'une TLE extension vers la

version spécifiée. Cette fonction supprime également tous les chemins de mise à jour pour la version spécifiée de l'TLEextension. Cette fonction ne désinstalle pas l'TLEextension si elle est actuellement active dans la base de données. Vous devez appeler explicitement `DROP EXTENSION` pour supprimer l'TLEextension. Pour désinstaller toutes les versions d'une TLE extension, consultezpgtle.uninstall_extension(name).

Prototype de fonction

```
pgtle.uninstall_extension(extname text, version text)
```

Rôle

`pgtle_admin`

Arguments

- `extname`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `version`— Version de l'TLEextension à désinstaller de la base de données.

Sortie

Aucune.

Exemple d'utilisation

```
SELECT * FROM pgtle.uninstall_extension('pg_tle_test', '0.2');
```

pgtle.uninstall_extension_if_exists

La `uninstall_extension_if_exists` fonction supprime toutes les versions d'une TLE extension d'une base de données donnée. Si l'TLEextension n'existe pas, la fonction revient silencieusement (aucun message d'erreur n'est généré). Si l'extension spécifiée est actuellement active dans une base de données, cette fonction ne la supprime pas. Vous devez appeler explicitement `DROP EXTENSION` pour supprimer l'TLEextension avant d'utiliser cette fonction pour désinstaller ses artefacts.

Prototype de fonction

```
pgtle.uninstall_extension_if_exists(extname text)
```

Rôle

`pgtle_admin`

Arguments

- `extname`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.

Sortie

La fonction `uninstall_extension_if_exists` renvoie `true` après avoir désinstallé l'extension spécifiée. Si l'extension spécifiée n'existe pas, la fonction renvoie `false`.

- `true`— Retourne `true` après la désinstallation de l'TLEextension.
- `false`— Retourne `false` lorsque l'TLEextension n'existe pas dans la base de données.

Exemple d'utilisation

```
SELECT * FROM pgtle.uninstall_extension_if_exists('pg_tle_test');
```

pgtle.uninstall_update_path

La `uninstall_update_path` fonction supprime le chemin de mise à jour spécifique d'une TLE extension. Cela empêche `ALTER EXTENSION ... UPDATE TO` de l'utiliser comme chemin de mise à jour.

Si l'TLEextension est actuellement utilisée par l'une des versions de ce chemin de mise à jour, elle reste dans la base de données.

Si le chemin de mise à jour spécifié n'existe pas, cette fonction génère une erreur.

Prototype de fonction

```
pgtle.uninstall_update_path(extname text, fromvers text, tovers text)
```

Rôle

`pgtle_admin`

Arguments

- `extname`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `fromvers`— Version source de l'TLEextension utilisée dans le chemin de mise à jour.
- `tovers`— Version de destination de l'TLEextension utilisée dans le chemin de mise à jour.

Sortie

Aucune.

Exemple d'utilisation

```
SELECT * FROM pgtle.uninstall_update_path('pg_tle_test', '0.1', '0.2');
```

pgtle.uninstall_update_path_if_exists

La `uninstall_update_path_if_exists` fonction est similaire `uninstall_update_path` en ce sens qu'elle supprime le chemin de mise à jour spécifié d'une TLE extension. Toutefois, si le chemin de mise à jour n'existe pas, cette fonction ne renvoie pas de message d'erreur. Au lieu de cela, la fonction renvoie `false`.

Prototype de fonction

```
pgtle.uninstall_update_path_if_exists(extname text, fromvers text, tovers text)
```

Rôle

`pgtle_admin`

Arguments

- `extname`— Le nom de l'TLEextension. Cette valeur est utilisée lors d'un appel de `CREATE EXTENSION`.
- `fromvers`— Version source de l'TLEextension utilisée dans le chemin de mise à jour.
- `tovers`— Version de destination de l'TLEextension utilisée dans le chemin de mise à jour.

Sortie

- `true`— La fonction a correctement mis à jour le chemin de l'TLEextension.
- `false`— La fonction n'a pas pu mettre à jour le chemin de l'TLEextension.

Exemple d'utilisation

```
SELECT * FROM pgtle.uninstall_update_path_if_exists('pg_tle_test', '0.1', '0.2');
```

pgtle.unregister_feature

La fonction `unregister_feature` permet de supprimer les fonctions qui ont été enregistrées pour utiliser des fonctionnalités `pg_tle`, telles que les hooks. Pour obtenir des informations sur l'enregistrement d'une fonctionnalité, consultez pgtle.register_feature.

Prototype de fonction

```
pgtle.unregister_feature(proc regproc, feature pg_tle_features)
```

Rôle

`pgtle_admin`

Arguments

- `proc` : nom d'une fonction stockée à enregistrer avec une fonctionnalité `pg_tle`.
- `feature` : nom de la fonctionnalité `pg_tle` à enregistrer avec la fonction. Par exemple, `passcheck` est une fonctionnalité qui peut être enregistrée pour être utilisée par les extensions Trusted Language Extensions (TLE) que vous développez. Pour de plus amples informations, veuillez consulter Crochet de vérification du mot de passe (passcheck).

Sortie

Aucune.

Exemple d'utilisation

```
SELECT * FROM pgtle.unregister_feature('pw_hook', 'passcheck');
```

pgtle.unregister_feature_if_exists

La fonction `unregister_feature` permet de supprimer les fonctions qui ont été enregistrées pour utiliser des fonctionnalités `pg_tle`, telles que les hooks. Pour de plus amples informations, veuillez consulter Utilisation des hooks PostgreSQL avec vos extensions TLE. Renvoie `true` après avoir réussi à annuler l'enregistrement de la fonctionnalité. Renvoie `false` si la fonctionnalité n'a pas été enregistrée.

Pour plus d'informations sur l'enregistrement des `pg_tle` fonctionnalités TLE de vos extensions, consultezpgtle.register_feature.

Prototype de fonction

```
pgtle.unregister_feature_if_exists('proc regproc', 'feature pg_tle_features')
```

Rôle

`pgtle_admin`

Arguments

- `proc` : nom de la fonction stockée qui a été enregistrée pour inclure une fonctionnalité `pg_tle`.
- `feature` : nom de la fonctionnalité `pg_tle` qui a été enregistrée avec l'extension de langage approuvé.

Sortie

Renvoie `true` ou `false`, comme suit.

- `true` : la fonction a annulé l'enregistrement de la fonctionnalité à l'extension.
- `false`— La fonction n'a pas pu annuler l'enregistrement de la fonctionnalité dans l'TLEextension.

Exemple d'utilisation

```
SELECT * FROM pgtle.unregister_feature_if_exists('pw_hook', 'passcheck');
```

Référence des hooks pour Trusted Language Extensions pour PostgreSQL

Le kit Trusted Language Extensions pour PostgreSQL prend en charge les hooks PostgreSQL. Un hook est un mécanisme de rappel interne mis à la disposition des développeurs pour étendre les fonctionnalités de base de PostgreSQL. En utilisant des hooks, les développeurs peuvent implémenter leurs propres fonctions ou procédures à utiliser lors de diverses opérations de base de données, modifiant ainsi le comportement de PostgreSQL. Par exemple, vous pouvez utiliser un hook `passcheck` pour personnaliser la façon dont PostgreSQL gère les mots de passe fournis lors de la création ou de la modification de mots de passe pour les utilisateurs (rôles).

Consultez la documentation suivante pour en savoir plus sur le crochet Passcheck disponible pour vos extensions TLE. Pour en savoir plus sur les hooks disponibles, y compris le hook d'authentification client, consultez la section hooks Trusted Language Extensions.

Crochet de vérification du mot de passe (passcheck)

Le crochet `passcheck` permet de personnaliser le comportement de PostgreSQL pendant le processus de vérification du mot de passe pour les commandes SQL et la métacommande `psql`suivantes.

- `CREATE ROLE` *`username`* `...PASSWORD` : pour plus d'informations, consultez CREATE USER (CRÉER UN RÔLE) dans la documentation PostgreSQL.
- `ALTER ROLE` *`username`*`...PASSWORD` : pour plus d'informations, consultez ALTER ROLE (ALTÉRER UN RÔLE) dans la documentation PostgreSQL.
- `\password` *`username`* : cette métacommande `psql` interactive modifie de manière sécurisée le mot de passe de l'utilisateur spécifié en hachant le mot de passe avant d'utiliser la syntaxe `ALTER ROLE ... PASSWORD` de manière transparente. La métacommande est un encapsuleur sécurisé pour la commande `ALTER ROLE ... PASSWORD`, et le crochet s'applique donc au comportement de la métacommande `psql`.

Pour obtenir un exemple, consultez Listing du code du hook de vérification de mot de passe.

Table des matières

- Prototype de fonction
- Arguments
- Configuration

- Notes d’utilisation

Prototype de fonction

```
passcheck_hook(username text, password text, password_type pgtle.password_types,
 valid_until timestamptz, valid_null boolean)
```

Arguments

Une fonction de crochet `passcheck` accepte les arguments suivants.

- `username` : nom (sous forme de texte) du rôle (nom d'utilisateur) qui définit un mot de passe.
- `password` : texte brut ou mot de passe haché. Le mot de passe saisi doit correspondre au type spécifié dans `password_type`.
- `password_type` : spécifiez le format `pgtle.password_type` du mot de passe. Ce format peut être l'une des options suivantes.
  - `PASSWORD_TYPE_PLAINTEXT` : un mot de passe en texte brut.
  - `PASSWORD_TYPE_MD5`— Un mot de passe haché à l'aide de l'algorithme MD5 (message digest 5).
  - `PASSWORD_TYPE_SCRAM_SHA_256` : un mot de passe qui a été haché en utilisant l'algorithme SCRAM-SHA-256.
- `valid_until` : spécifiez l'heure à laquelle le mot de passe devient invalide. Cet argument est facultatif. Si vous utilisez cet argument, spécifiez l'heure comme une valeur `timestamptz`.
- `valid_null` : si cette valeur booléenne est définie sur `true`, l'option `valid_until` est définie sur `NULL`.

Configuration

La fonction `pgtle.enable_password_check` contrôle si le crochet passcheck est actif. Le crochet passcheck a trois paramètres possibles.

- `off` : désactive le crochet de vérification du mot de passe `passcheck`. C’est la valeur par défaut.
- on : active le crochet de vérification du mot de passe `passcode` afin que les mots de passe soient vérifiés dans la table.
- `require` : nécessite la définition d'un crochet de vérification du mot de passe.

Notes d’utilisation

Pour activer ou désactiver le crochet `passcheck`, vous devez modifier le groupe de paramètres de base de données personnalisé pour votre instance de base de données RDS for PostgreSQL.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-parameter-group \
    --region aws-region \
    --db-parameter-group-name your-custom-parameter-group \
    --parameters
 "ParameterName=pgtle.enable_password_check,ParameterValue=on,ApplyMethod=immediate"
```

Dans Windows:

```
aws rds modify-db-parameter-group ^
    --region aws-region ^
    --db-parameter-group-name your-custom-parameter-group ^
    --parameters
 "ParameterName=pgtle.enable_password_check,ParameterValue=on,ApplyMethod=immediate"
```

Exemples de code pour Amazon RDS utilisant AWS SDKs

Les exemples de code suivants montrent comment utiliser Amazon RDS avec un kit de développement AWS logiciel (SDK).

Les principes de base sont des exemples de code qui vous montrent comment effectuer les opérations essentielles au sein d'un service.

Les actions sont des extraits de code de programmes plus larges et doivent être exécutées dans leur contexte. Alors que les actions vous indiquent comment appeler des fonctions de service individuelles, vous pouvez les voir en contexte dans leurs scénarios associés.

Les Scénarios sont des exemples de code qui vous montrent comment accomplir des tâches spécifiques en appelant plusieurs fonctions au sein d'un même service ou combinés à d'autres Services AWS.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Mise en route

Hello Amazon RDS

Les exemples de code suivants montrent comment bien démarrer avec Amazon RDS.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
using System;
using System.Threading.Tasks;
```

```
using Amazon.RDS;
using Amazon.RDS.Model;

namespace RDSActions;

public static class HelloRds
{
    static async Task Main(string[] args)
    {
        var rdsClient = new AmazonRDSClient();

        Console.WriteLine($"Hello Amazon RDS! Following are some of your DB
 instances:");
        Console.WriteLine();

        // You can use await and any of the async methods to get a response.
        // Let's get the first twenty DB instances.
        var response = await rdsClient.DescribeDBInstancesAsync(
            new DescribeDBInstancesRequest()
            {
                MaxRecords = 20 // Must be between 20 and 100.
            });

        foreach (var instance in response.DBInstances)
        {
            Console.WriteLine($"\tDB name: {instance.DBName}");
            Console.WriteLine($"\tArn: {instance.DBInstanceArn}");
            Console.WriteLine($"\tIdentifier: {instance.DBInstanceIdentifier}");
            Console.WriteLine();
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Code pour le CMake fichier CMake Lists.txt.

```
# Set the minimum required version of CMake for this project.
cmake_minimum_required(VERSION 3.13)

# Set the AWS service components used by this project.
set(SERVICE_COMPONENTS rds)

# Set this project's name.
project("hello_rds")

# Set the C++ standard to use to build this target.
# At least C++ 11 is required for the AWS SDK for C++.
set(CMAKE_CXX_STANDARD 11)

# Use the MSVC variable to determine if this is a Windows build.
set(WINDOWS_BUILD ${MSVC})

if (WINDOWS_BUILD) # Set the location where CMake can find the installed
 libraries for the AWS SDK.
    string(REPLACE ";" "/aws-cpp-sdk-all;" SYSTEM_MODULE_PATH
 "${CMAKE_SYSTEM_PREFIX_PATH}/aws-cpp-sdk-all")
    list(APPEND CMAKE_PREFIX_PATH ${SYSTEM_MODULE_PATH})
endif ()

# Find the AWS SDK for C++ package.
find_package(AWSSDK REQUIRED COMPONENTS ${SERVICE_COMPONENTS})

if (WINDOWS_BUILD AND AWSSDK_INSTALL_AS_SHARED_LIBS)
     # Copy relevant AWS SDK for C++ libraries into the current binary directory
 for running and debugging.
```

```
    # set(BIN_SUB_DIR "/Debug") # If you are building from the command line, you
 may need to uncomment this
                                    # and set the proper subdirectory to the
 executables' location.

    AWSSDK_CPY_DYN_LIBS(SERVICE_COMPONENTS ""
 ${CMAKE_CURRENT_BINARY_DIR}${BIN_SUB_DIR})
endif ()

add_executable(${PROJECT_NAME}
        hello_rds.cpp)

target_link_libraries(${PROJECT_NAME}
        ${AWSSDK_LINK_LIBRARIES})
```

Code pour le fichier source hello_rds.cpp.

```
#include <aws/core/Aws.h>
#include <aws/rds/RDSClient.h>
#include <aws/rds/model/DescribeDBInstancesRequest.h>
#include <iostream>

/*
 *  A "Hello Rds" starter application which initializes an Amazon Relational
 Database Service (Amazon RDS) client and
 *  describes the Amazon RDS instances.
 *
 *  main function
 *
 *  Usage: 'hello_rds'
 *
 */

int main(int argc, char **argv) {
    Aws::SDKOptions options;
    // Optionally change the log level for debugging.
//   options.loggingOptions.logLevel = Utils::Logging::LogLevel::Debug;
    Aws::InitAPI(options); // Should only be called once.
    int result = 0;
    {
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
```

```
        // clientConfig.region = "us-east-1";

        Aws::RDS::RDSClient rdsClient(clientConfig);
        Aws::String marker;
        std::vector<Aws::String> instanceDBIDs;

        do {
            Aws::RDS::Model::DescribeDBInstancesRequest request;

            if (!marker.empty()) {
                request.SetMarker(marker);
            }

            Aws::RDS::Model::DescribeDBInstancesOutcome outcome =
                    rdsClient.DescribeDBInstances(request);

            if (outcome.IsSuccess()) {
                for (auto &instance: outcome.GetResult().GetDBInstances()) {
                    instanceDBIDs.push_back(instance.GetDBInstanceIdentifier());
                }
                marker = outcome.GetResult().GetMarker();
            } else {
                result = 1;
                std::cerr << "Error with RDS::DescribeDBInstances. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                break;
            }
        } while (!marker.empty());

        std::cout << instanceDBIDs.size() << " RDS instances found." <<
 std::endl;
        for (auto &instanceDBID: instanceDBIDs) {
            std::cout << "   Instance: " << instanceDBID << std::endl;
        }
    }

    Aws::ShutdownAPI(options); // Should only be called once.
    return result;
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour C++ API.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
package main

import (
 "context"
 "fmt"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/config"
 "github.com/aws/aws-sdk-go-v2/service/rds"
)

// main uses the AWS SDK for Go V2 to create an Amazon Relational Database
 Service (Amazon RDS)
// client and list up to 20 DB instances in your account.
// This example uses the default settings specified in your shared credentials
// and config files.
func main() {
 ctx := context.Background()
 sdkConfig, err := config.LoadDefaultConfig(ctx)
 if err != nil {
  fmt.Println("Couldn't load default configuration. Have you set up your AWS
 account?")
  fmt.Println(err)
  return
 }
 rdsClient := rds.NewFromConfig(sdkConfig)
 const maxInstances = 20
```

```
 fmt.Printf("Let's list up to %v DB instances.\n", maxInstances)
 output, err := rdsClient.DescribeDBInstances(ctx,
  &rds.DescribeDBInstancesInput{MaxRecords: aws.Int32(maxInstances)})
 if err != nil {
  fmt.Printf("Couldn't list DB instances: %v\n", err)
  return
 }
 if len(output.DBInstances) == 0 {
  fmt.Println("No DB instances found.")
 } else {
  for _, instance := range output.DBInstances {
   fmt.Printf("DB instance %v has database %v.\n",
 *instance.DBInstanceIdentifier,
    *instance.DBName)
  }
 }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesResponse;
import software.amazon.awssdk.services.rds.model.DBInstance;
import software.amazon.awssdk.services.rds.model.RdsException;
import java.util.List;

/**
```

```
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DescribeDBInstances {

    public static void main(String[] args) {
        Region region = Region.US_EAST_1;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        describeInstances(rdsClient);
        rdsClient.close();
    }

    public static void describeInstances(RdsClient rdsClient) {
        try {
            DescribeDbInstancesResponse response =
 rdsClient.describeDBInstances();
            List<DBInstance> instanceList = response.dbInstances();
            for (DBInstance instance : instanceList) {
                System.out.println("Instance ARN is: " +
 instance.dbInstanceArn());
                System.out.println("The Engine is " + instance.engine());
                System.out.println("Connection endpoint is" +
 instance.endpoint().address());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
"""
Purpose

Shows how to use the AWS SDK for Python (Boto3) with the Amazon Relational
 Database Service
(Amazon RDS) to list the databases in your account.
"""

import boto3
from botocore.exceptions import ClientError

# Create an RDS client
rds_client = boto3.client("rds")

# Create a paginator for the describe_db_instances operation
paginator = rds_client.get_paginator("describe_db_instances")

try:
    # Use the paginator to get a list of DB instances
    response_iterator = paginator.paginate(
        PaginationConfig={
            "MaxItems": 123,
            "PageSize": 50,  # Adjust PageSize as needed
            "StartingToken": None,
        }
    )

    # Iterate through the pages of the response
    instances_found = False
    for page in response_iterator:
        if "DBInstances" in page and page["DBInstances"]:
            instances_found = True
```

```
            print("Your RDS instances are:")
            for db in page["DBInstances"]:
                print(db["DBInstanceIdentifier"])

    if not instances_found:
        print("No RDS instances found!")

except ClientError as e:
    print(f"Couldn't list RDS instances. Here's why: {e.response['Error']
['Message']}")
```

- Pour plus de détails sur l'API, voir AWS Describe DBInstances in SDK for Python (Boto3) API Reference.

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds'
require 'logger'

# RDSManager is a class responsible for managing RDS operations
# such as listing all RDS DB instances in the current AWS account.
class RDSManager
  def initialize(client)
    @client = client
    @logger = Logger.new($stdout)
  end

  # Lists and prints all RDS DB instances in the current AWS account.
  def list_db_instances
    @logger.info('Listing RDS DB instances')
```

```
    paginator = @client.describe_db_instances
    instances = []

    paginator.each_page do |page|
      instances.concat(page.db_instances)
    end

    if instances.empty?
      @logger.info('No instances found.')
    else
      @logger.info("Found #{instances.count} instance(s):")
      instances.each do |instance|
        @logger.info(" * #{instance.db_instance_identifier}
 (#{instance.db_instance_status})")
      end
    end
  end
end

if $PROGRAM_NAME == __FILE__
  rds_client = Aws::RDS::Client.new(region: 'us-west-2')
  manager = RDSManager.new(rds_client)
  manager.list_db_instances
end
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Ruby API.

Exemples de code

- Exemples de base pour l'utilisation d'Amazon RDS AWS SDKs
  - Hello Amazon RDS
  - Découvrez les bases d'Amazon RDS avec un SDK AWS
  - Actions pour Amazon RDS utilisant AWS SDKs
    - Utilisation CreateDBInstance avec un AWS SDK ou une CLI
    - Utilisation CreateDBParameterGroup avec un AWS SDK ou une CLI
    - Utilisation CreateDBSnapshot avec un AWS SDK ou une CLI

    - Utilisation DeleteDBInstance avec un AWS SDK ou une CLI
    - Utilisation DeleteDBParameterGroup avec un AWS SDK ou une CLI
    - Utilisation DescribeAccountAttributes avec un AWS SDK ou une CLI
    - Utilisation DescribeDBEngineVersions avec un AWS SDK ou une CLI
    - Utilisation DescribeDBInstances avec un AWS SDK ou une CLI
    - Utilisation DescribeDBParameterGroups avec un AWS SDK ou une CLI
    - Utilisation DescribeDBParameters avec un AWS SDK ou une CLI
    - Utilisation DescribeDBSnapshots avec un AWS SDK ou une CLI
    - Utilisation DescribeOrderableDBInstanceOptions avec un AWS SDK ou une CLI
    - Utilisation GenerateRDSAuthToken avec un AWS SDK
    - Utilisation ModifyDBInstance avec un AWS SDK ou une CLI
    - Utilisation ModifyDBParameterGroup avec un AWS SDK ou une CLI
    - Utilisation RebootDBInstance avec un AWS SDK ou une CLI
- Scénarios d'utilisation d'Amazon RDS AWS SDKs
    - Créer un outil de suivi des éléments de travail sans serveur Aurora
- Exemples de solutions sans serveur pour Amazon RDS
    - Connexion à une base de données Amazon RDS dans une fonction Lambda

Exemples de base pour l'utilisation d'Amazon RDS AWS SDKs

Les exemples de code suivants montrent comment utiliser les bases d'Amazon Relational Database Service AWS SDKs avec.

Exemples

- Hello Amazon RDS
- Découvrez les bases d'Amazon RDS avec un SDK AWS
- Actions pour Amazon RDS utilisant AWS SDKs
    - Utilisation CreateDBInstance avec un AWS SDK ou une CLI
    - Utilisation CreateDBParameterGroup avec un AWS SDK ou une CLI
    - Utilisation CreateDBSnapshot avec un AWS SDK ou une CLI
    - Utilisation DeleteDBInstance avec un AWS SDK ou une CLI

- Utilisation DeleteDBParameterGroup avec un AWS SDK ou une CLI
- Utilisation DescribeAccountAttributes avec un AWS SDK ou une CLI
- Utilisation DescribeDBEngineVersions avec un AWS SDK ou une CLI
- Utilisation DescribeDBInstances avec un AWS SDK ou une CLI
- Utilisation DescribeDBParameterGroups avec un AWS SDK ou une CLI
- Utilisation DescribeDBParameters avec un AWS SDK ou une CLI
- Utilisation DescribeDBSnapshots avec un AWS SDK ou une CLI
- Utilisation DescribeOrderableDBInstanceOptions avec un AWS SDK ou une CLI
- Utilisation GenerateRDSAuthToken avec un AWS SDK
- Utilisation ModifyDBInstance avec un AWS SDK ou une CLI
- Utilisation ModifyDBParameterGroup avec un AWS SDK ou une CLI
- Utilisation RebootDBInstance avec un AWS SDK ou une CLI

Hello Amazon RDS

Les exemples de code suivants montrent comment bien démarrer avec Amazon RDS.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
using System;
using System.Threading.Tasks;
using Amazon.RDS;
using Amazon.RDS.Model;

namespace RDSActions;

public static class HelloRds
{
```

```
    static async Task Main(string[] args)
    {
        var rdsClient = new AmazonRDSClient();

        Console.WriteLine($"Hello Amazon RDS! Following are some of your DB
 instances:");
        Console.WriteLine();

        // You can use await and any of the async methods to get a response.
        // Let's get the first twenty DB instances.
        var response = await rdsClient.DescribeDBInstancesAsync(
            new DescribeDBInstancesRequest()
            {
                MaxRecords = 20 // Must be between 20 and 100.
            });

        foreach (var instance in response.DBInstances)
        {
            Console.WriteLine($"\tDB name: {instance.DBName}");
            Console.WriteLine($"\tArn: {instance.DBInstanceArn}");
            Console.WriteLine($"\tIdentifier: {instance.DBInstanceIdentifier}");
            Console.WriteLine();
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Code pour le CMake fichier CMake Lists.txt.

```
# Set the minimum required version of CMake for this project.
cmake_minimum_required(VERSION 3.13)

# Set the AWS service components used by this project.
set(SERVICE_COMPONENTS rds)

# Set this project's name.
project("hello_rds")

# Set the C++ standard to use to build this target.
# At least C++ 11 is required for the AWS SDK for C++.
set(CMAKE_CXX_STANDARD 11)

# Use the MSVC variable to determine if this is a Windows build.
set(WINDOWS_BUILD ${MSVC})

if (WINDOWS_BUILD) # Set the location where CMake can find the installed
 libraries for the AWS SDK.
    string(REPLACE ";" "/aws-cpp-sdk-all;" SYSTEM_MODULE_PATH
 "${CMAKE_SYSTEM_PREFIX_PATH}/aws-cpp-sdk-all")
    list(APPEND CMAKE_PREFIX_PATH ${SYSTEM_MODULE_PATH})
endif ()

# Find the AWS SDK for C++ package.
find_package(AWSSDK REQUIRED COMPONENTS ${SERVICE_COMPONENTS})

if (WINDOWS_BUILD AND AWSSDK_INSTALL_AS_SHARED_LIBS)
     # Copy relevant AWS SDK for C++ libraries into the current binary directory
 for running and debugging.

     # set(BIN_SUB_DIR "/Debug") # If you are building from the command line, you
 may need to uncomment this
                                     # and set the proper subdirectory to the
 executables' location.

     AWSSDK_CPY_DYN_LIBS(SERVICE_COMPONENTS ""
 ${CMAKE_CURRENT_BINARY_DIR}${BIN_SUB_DIR})
endif ()

add_executable(${PROJECT_NAME}
        hello_rds.cpp)

target_link_libraries(${PROJECT_NAME}
```

```
        ${AWSSDK_LINK_LIBRARIES})
```

Code pour le fichier source hello_rds.cpp.

```
#include <aws/core/Aws.h>
#include <aws/rds/RDSClient.h>
#include <aws/rds/model/DescribeDBInstancesRequest.h>
#include <iostream>

/*
 *  A "Hello Rds" starter application which initializes an Amazon Relational
 Database Service (Amazon RDS) client and
 *  describes the Amazon RDS instances.
 *
 *  main function
 *
 *  Usage: 'hello_rds'
 *
 */

int main(int argc, char **argv) {
    Aws::SDKOptions options;
    // Optionally change the log level for debugging.
//   options.loggingOptions.logLevel = Utils::Logging::LogLevel::Debug;
    Aws::InitAPI(options); // Should only be called once.
    int result = 0;
    {
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

        Aws::RDS::RDSClient rdsClient(clientConfig);
        Aws::String marker;
        std::vector<Aws::String> instanceDBIDs;

        do {
            Aws::RDS::Model::DescribeDBInstancesRequest request;

            if (!marker.empty()) {
                request.SetMarker(marker);
            }
```

```
            Aws::RDS::Model::DescribeDBInstancesOutcome outcome =
                    rdsClient.DescribeDBInstances(request);

            if (outcome.IsSuccess()) {
                for (auto &instance: outcome.GetResult().GetDBInstances()) {
                    instanceDBIDs.push_back(instance.GetDBInstanceIdentifier());
                }
                marker = outcome.GetResult().GetMarker();
            } else {
                result = 1;
                std::cerr << "Error with RDS::DescribeDBInstances. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                break;
            }
        } while (!marker.empty());

        std::cout << instanceDBIDs.size() << " RDS instances found." <<
 std::endl;
        for (auto &instanceDBID: instanceDBIDs) {
            std::cout << "   Instance: " << instanceDBID << std::endl;
        }
    }

    Aws::ShutdownAPI(options); // Should only be called once.
    return result;
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour C++ API.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
package main

import (
 "context"
 "fmt"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/config"
 "github.com/aws/aws-sdk-go-v2/service/rds"
)

// main uses the AWS SDK for Go V2 to create an Amazon Relational Database
 Service (Amazon RDS)
// client and list up to 20 DB instances in your account.
// This example uses the default settings specified in your shared credentials
// and config files.
func main() {
 ctx := context.Background()
 sdkConfig, err := config.LoadDefaultConfig(ctx)
 if err != nil {
  fmt.Println("Couldn't load default configuration. Have you set up your AWS
 account?")
  fmt.Println(err)
  return
 }
 rdsClient := rds.NewFromConfig(sdkConfig)
 const maxInstances = 20
 fmt.Printf("Let's list up to %v DB instances.\n", maxInstances)
 output, err := rdsClient.DescribeDBInstances(ctx,
  &rds.DescribeDBInstancesInput{MaxRecords: aws.Int32(maxInstances)})
 if err != nil {
  fmt.Printf("Couldn't list DB instances: %v\n", err)
  return
 }
 if len(output.DBInstances) == 0 {
  fmt.Println("No DB instances found.")
 } else {
  for _, instance := range output.DBInstances {
   fmt.Printf("DB instance %v has database %v.\n",
 *instance.DBInstanceIdentifier,
    *instance.DBName)
  }
```

```
 }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesResponse;
import software.amazon.awssdk.services.rds.model.DBInstance;
import software.amazon.awssdk.services.rds.model.RdsException;
import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DescribeDBInstances {

    public static void main(String[] args) {
        Region region = Region.US_EAST_1;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();
```

```
        describeInstances(rdsClient);
        rdsClient.close();
    }

    public static void describeInstances(RdsClient rdsClient) {
        try {
            DescribeDbInstancesResponse response =
 rdsClient.describeDBInstances();
            List<DBInstance> instanceList = response.dbInstances();
            for (DBInstance instance : instanceList) {
                System.out.println("Instance ARN is: " +
 instance.dbInstanceArn());
                System.out.println("The Engine is " + instance.engine());
                System.out.println("Connection endpoint is" +
 instance.endpoint().address());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
"""
Purpose
```

```
Shows how to use the AWS SDK for Python (Boto3) with the Amazon Relational
 Database Service
(Amazon RDS) to list the databases in your account.
"""

import boto3
from botocore.exceptions import ClientError

# Create an RDS client
rds_client = boto3.client("rds")

# Create a paginator for the describe_db_instances operation
paginator = rds_client.get_paginator("describe_db_instances")

try:
    # Use the paginator to get a list of DB instances
    response_iterator = paginator.paginate(
        PaginationConfig={
            "MaxItems": 123,
            "PageSize": 50,  # Adjust PageSize as needed
            "StartingToken": None,
        }
    )

    # Iterate through the pages of the response
    instances_found = False
    for page in response_iterator:
        if "DBInstances" in page and page["DBInstances"]:
            instances_found = True
            print("Your RDS instances are:")
            for db in page["DBInstances"]:
                print(db["DBInstanceIdentifier"])

    if not instances_found:
        print("No RDS instances found!")

except ClientError as e:
    print(f"Couldn't list RDS instances. Here's why: {e.response['Error']
['Message']}")
```

- Pour plus de détails sur l'API, voir AWS Describe DBInstances in SDK for Python (Boto3) API Reference.

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds'
require 'logger'

# RDSManager is a class responsible for managing RDS operations
# such as listing all RDS DB instances in the current AWS account.
class RDSManager
  def initialize(client)
    @client = client
    @logger = Logger.new($stdout)
  end

  # Lists and prints all RDS DB instances in the current AWS account.
  def list_db_instances
    @logger.info('Listing RDS DB instances')

    paginator = @client.describe_db_instances
    instances = []

    paginator.each_page do |page|
      instances.concat(page.db_instances)
    end

    if instances.empty?
      @logger.info('No instances found.')
    else
      @logger.info("Found #{instances.count} instance(s):")
      instances.each do |instance|
```

```
        @logger.info(" * #{instance.db_instance_identifier}
 (#{instance.db_instance_status})")
      end
    end
  end
end

if $PROGRAM_NAME == __FILE__
  rds_client = Aws::RDS::Client.new(region: 'us-west-2')
  manager = RDSManager.new(rds_client)
  manager.list_db_instances
end
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Ruby API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Découvrez les bases d'Amazon RDS avec un SDK AWS

Les exemples de code suivants montrent comment :

- Créez un groupe de paramètres de bases de données personnalisé et définissez des valeurs pour les paramètres.
- Créez une instance de base de données configurée pour utiliser le groupe de paramètres. L'instance de base de données contient également une base de données.
- Prenez un instantané de l'instance.
- Supprimez l'instance et le groupe de paramètres.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Exécutez un scénario interactif à une invite de commande.

```
/// <summary>
/// Scenario for RDS DB instance example.
/// </summary>
public class RDSInstanceScenario
{
    /*
    Before running this .NET code example, set up your development environment,
 including your credentials.

    This .NET example performs the following tasks:
    1.  Returns a list of the available DB engine families using the
 DescribeDBEngineVersionsAsync method.
    2.  Selects an engine family and creates a custom DB parameter group using
 the CreateDBParameterGroupAsync method.
    3.  Gets the parameter groups using the DescribeDBParameterGroupsAsync
 method.
    4.  Gets parameters in the group using the DescribeDBParameters method.
    5.  Parses and displays parameters in the group.
    6.  Modifies both the auto_increment_offset and auto_increment_increment
 parameters
        using the ModifyDBParameterGroupAsync method.
    7.  Gets and displays the updated parameters using the DescribeDBParameters
 method with a source of "user".
    8.  Gets a list of allowed engine versions using the
 DescribeDBEngineVersionsAsync method.
    9.  Displays and selects from a list of micro instance classes available for
 the selected engine and version.
    10. Creates an RDS DB instance that contains a MySql database and uses the
 parameter group
        using the CreateDBInstanceAsync method.
```

```
   11. Waits for DB instance to be ready using the DescribeDBInstancesAsync
method.
   12. Prints out the connection endpoint string for the new DB instance.
   13. Creates a snapshot of the DB instance using the CreateDBSnapshotAsync
method.
   14. Waits for DB snapshot to be ready using the DescribeDBSnapshots method.
   15. Deletes the DB instance using the DeleteDBInstanceAsync method.
   16. Waits for DB instance to be deleted using the DescribeDbInstances method.
   17. Deletes the parameter group using the DeleteDBParameterGroupAsync.
   */

   private static readonly string sepBar = new('-', 80);
   private static RDSWrapper rdsWrapper = null!;
   private static ILogger logger = null!;
   private static readonly string engine = "mysql";
   static async Task Main(string[] args)
   {
       // Set up dependency injection for the Amazon RDS service.
       using var host = Host.CreateDefaultBuilder(args)
           .ConfigureLogging(logging =>
               logging.AddFilter("System", LogLevel.Debug)
                   .AddFilter<DebugLoggerProvider>("Microsoft",
LogLevel.Information)
                   .AddFilter<ConsoleLoggerProvider>("Microsoft",
LogLevel.Trace))
           .ConfigureServices((_, services) =>
               services.AddAWSService<IAmazonRDS>()
                   .AddTransient<RDSWrapper>()
           )
           .Build();

       logger = LoggerFactory.Create(builder =>
       {
           builder.AddConsole();
       }).CreateLogger<RDSInstanceScenario>();

       rdsWrapper = host.Services.GetRequiredService<RDSWrapper>();

       Console.WriteLine(sepBar);
       Console.WriteLine(
           "Welcome to the Amazon Relational Database Service (Amazon RDS) DB
instance scenario example.");
       Console.WriteLine(sepBar);
```

```
        try
        {
            var parameterGroupFamily = await ChooseParameterGroupFamily();

            var parameterGroup = await
CreateDbParameterGroup(parameterGroupFamily);

            var parameters = await
DescribeParametersInGroup(parameterGroup.DBParameterGroupName,
                new List<string> { "auto_increment_offset",
"auto_increment_increment" });

            await ModifyParameters(parameterGroup.DBParameterGroupName,
parameters);

            await
DescribeUserSourceParameters(parameterGroup.DBParameterGroupName);

            var engineVersionChoice = await
ChooseDbEngineVersion(parameterGroupFamily);

            var instanceChoice = await ChooseDbInstanceClass(engine,
engineVersionChoice.EngineVersion);

            var newInstanceIdentifier = "Example-Instance-" + DateTime.Now.Ticks;

            var newInstance = await CreateRdsNewInstance(parameterGroup, engine,
engineVersionChoice.EngineVersion,
                instanceChoice.DBInstanceClass, newInstanceIdentifier);
            if (newInstance != null)
            {
                DisplayConnectionString(newInstance);

                await CreateSnapshot(newInstance);

                await DeleteRdsInstance(newInstance);
            }

            await DeleteParameterGroup(parameterGroup);

            Console.WriteLine("Scenario complete.");
            Console.WriteLine(sepBar);
        }
        catch (Exception ex)
```

```
        {
            logger.LogError(ex, "There was a problem executing the scenario.");
        }
    }

    /// <summary>
    /// Choose the RDS DB parameter group family from a list of available
options.
    /// </summary>
    /// <returns>The selected parameter group family.</returns>
    public static async Task<string> ChooseParameterGroupFamily()
    {
        Console.WriteLine(sepBar);
        // 1. Get a list of available engines.
        var engines = await rdsWrapper.DescribeDBEngineVersions(engine);

        Console.WriteLine("1. The following is a list of available DB parameter
group families:");
        int i = 1;
        var parameterGroupFamilies = engines.GroupBy(e =>
e.DBParameterGroupFamily).ToList();
        foreach (var parameterGroupFamily in parameterGroupFamilies)
        {
            // List the available parameter group families.
            Console.WriteLine(
                $"\t{i}. Family: {parameterGroupFamily.Key}");
            i++;
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > parameterGroupFamilies.Count)
        {
            Console.WriteLine("Select an available DB parameter group family by
entering a number from the list above:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }
        var parameterGroupFamilyChoice = parameterGroupFamilies[choiceNumber -
1];
        Console.WriteLine(sepBar);
        return parameterGroupFamilyChoice.Key;
    }

    /// <summary>
```

```
   /// Create and get information on a DB parameter group.
   /// </summary>
   /// <param name="dbParameterGroupFamily">The DBParameterGroupFamily for the
new DB parameter group.</param>
   /// <returns>The new DBParameterGroup.</returns>
   public static async Task<DBParameterGroup> CreateDbParameterGroup(string
dbParameterGroupFamily)
   {
       Console.WriteLine(sepBar);
       Console.WriteLine($"2. Create new DB parameter group with family
{dbParameterGroupFamily}:");

       var parameterGroup = await rdsWrapper.CreateDBParameterGroup(
           "ExampleParameterGroup-" + DateTime.Now.Ticks,
           dbParameterGroupFamily, "New example parameter group");

       var groupInfo =
           await rdsWrapper.DescribeDBParameterGroups(parameterGroup
               .DBParameterGroupName);

       Console.WriteLine(
           $"3. New DB parameter group: \n\t{groupInfo[0].Description}, \n\tARN
{groupInfo[0].DBParameterGroupArn}");
       Console.WriteLine(sepBar);
       return parameterGroup;
   }

   /// <summary>
   /// Get and describe parameters from a DBParameterGroup.
   /// </summary>
   /// <param name="parameterGroupName">Name of the DBParameterGroup.</param>
   /// <param name="parameterNames">Optional specific names of parameters to
describe.</param>
   /// <returns>The list of requested parameters.</returns>
   public static async Task<List<Parameter>> DescribeParametersInGroup(string
parameterGroupName, List<string>? parameterNames = null)
   {
       Console.WriteLine(sepBar);
       Console.WriteLine("4. Get some parameters from the group.");
       Console.WriteLine(sepBar);

       var parameters =
           await rdsWrapper.DescribeDBParameters(parameterGroupName);
```

```
        var matchingParameters =
            parameters.Where(p => parameterNames == null ||
parameterNames.Contains(p.ParameterName)).ToList();

        Console.WriteLine("5. Parameter information:");
        matchingParameters.ForEach(p =>
            Console.WriteLine(
                $"\n\tParameter: {p.ParameterName}." +
                $"\n\tDescription: {p.Description}." +
                $"\n\tAllowed Values: {p.AllowedValues}." +
                $"\n\tValue: {p.ParameterValue}."));

        Console.WriteLine(sepBar);

        return matchingParameters;
    }

    /// <summary>
    /// Modify a parameter from a DBParameterGroup.
    /// </summary>
    /// <param name="parameterGroupName">Name of the DBParameterGroup.</param>
    /// <param name="parameters">The parameters to modify.</param>
    /// <returns>Async task.</returns>
    public static async Task ModifyParameters(string parameterGroupName,
List<Parameter> parameters)
    {
        Console.WriteLine(sepBar);
        Console.WriteLine("6. Modify some parameters in the group.");

        foreach (var p in parameters)
        {
            if (p.IsModifiable && p.DataType == "integer")
            {
                int newValue = 0;
                while (newValue == 0)
                {
                    Console.WriteLine(
                        $"Enter a new value for {p.ParameterName} from the
allowed values {p.AllowedValues} ");

                    var choice = Console.ReadLine();
                    Int32.TryParse(choice, out newValue);
                }
```

```
                p.ParameterValue = newValue.ToString();
            }
        }

        await rdsWrapper.ModifyDBParameterGroup(parameterGroupName, parameters);

        Console.WriteLine(sepBar);
    }

    /// <summary>
    /// Describe the user source parameters in the group.
    /// </summary>
    /// <param name="parameterGroupName">Name of the DBParameterGroup.</param>
    /// <returns>Async task.</returns>
    public static async Task DescribeUserSourceParameters(string
parameterGroupName)
    {
        Console.WriteLine(sepBar);
        Console.WriteLine("7. Describe user source parameters in the group.");

        var parameters =
            await rdsWrapper.DescribeDBParameters(parameterGroupName, "user");


        parameters.ForEach(p =>
            Console.WriteLine(
                $"\n\tParameter: {p.ParameterName}." +
                $"\n\tDescription: {p.Description}." +
                $"\n\tAllowed Values: {p.AllowedValues}." +
                $"\n\tValue: {p.ParameterValue}."));

        Console.WriteLine(sepBar);
    }


    /// <summary>
    /// Choose a DB engine version.
    /// </summary>
    /// <param name="dbParameterGroupFamily">DB parameter group family for engine
choice.</param>
    /// <returns>The selected engine version.</returns>
    public static async Task<DBEngineVersion> ChooseDbEngineVersion(string
dbParameterGroupFamily)
    {
```

```
        Console.WriteLine(sepBar);
        // Get a list of allowed engines.
        var allowedEngines =
            await rdsWrapper.DescribeDBEngineVersions(engine,
 dbParameterGroupFamily);

        Console.WriteLine($"Available DB engine versions for parameter group
 family {dbParameterGroupFamily}:");
        int i = 1;
        foreach (var version in allowedEngines)
        {
            Console.WriteLine(
                $"\t{i}. Engine: {version.Engine} Version
 {version.EngineVersion}.");
            i++;
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > allowedEngines.Count)
        {
            Console.WriteLine("8. Select an available DB engine version by
 entering a number from the list above:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }

        var engineChoice = allowedEngines[choiceNumber - 1];
        Console.WriteLine(sepBar);
        return engineChoice;
    }

    /// <summary>
    /// Choose a DB instance class for a particular engine and engine version.
    /// </summary>
    /// <param name="engine">DB engine for DB instance choice.</param>
    /// <param name="engineVersion">DB engine version for DB instance choice.</
param>
    /// <returns>The selected orderable DB instance option.</returns>
    public static async Task<OrderableDBInstanceOption>
 ChooseDbInstanceClass(string engine, string engineVersion)
    {
        Console.WriteLine(sepBar);
        // Get a list of allowed DB instance classes.
        var allowedInstances =
```

```
            await rdsWrapper.DescribeOrderableDBInstanceOptions(engine,
 engineVersion);

        Console.WriteLine($"8. Available micro DB instance classes for engine
 {engine} and version {engineVersion}:");
        int i = 1;

        // Filter to micro instances for this example.
        allowedInstances = allowedInstances
            .Where(i => i.DBInstanceClass.Contains("micro")).ToList();

        foreach (var instance in allowedInstances)
        {
            Console.WriteLine(
                $"\t{i}. Instance class: {instance.DBInstanceClass} (storage type
 {instance.StorageType})");
            i++;
        }

        var choiceNumber = 0;
        while (choiceNumber < 1 || choiceNumber > allowedInstances.Count)
        {
            Console.WriteLine("9. Select an available DB instance class by
 entering a number from the list above:");
            var choice = Console.ReadLine();
            Int32.TryParse(choice, out choiceNumber);
        }

        var instanceChoice = allowedInstances[choiceNumber - 1];
        Console.WriteLine(sepBar);
        return instanceChoice;
    }

    /// <summary>
    /// Create a new RDS DB instance.
    /// </summary>
    /// <param name="parameterGroup">Parameter group to use for the DB
 instance.</param>
    /// <param name="engineName">Engine to use for the DB instance.</param>
    /// <param name="engineVersion">Engine version to use for the DB instance.</
param>
    /// <param name="instanceClass">Instance class to use for the DB instance.</
param>
```

```
   /// <param name="instanceIdentifier">Instance identifier to use for the DB
instance.</param>
   /// <returns>The new DB instance.</returns>
   public static async Task<DBInstance?> CreateRdsNewInstance(DBParameterGroup
parameterGroup,
       string engineName, string engineVersion, string instanceClass, string
instanceIdentifier)
   {
       Console.WriteLine(sepBar);
       Console.WriteLine($"10. Create a new DB instance with identifier
{instanceIdentifier}.");
       bool isInstanceReady = false;
       DBInstance newInstance;
       var instances = await rdsWrapper.DescribeDBInstances();
       isInstanceReady = instances.FirstOrDefault(i =>
           i.DBInstanceIdentifier == instanceIdentifier)?.DBInstanceStatus ==
"available";

       if (isInstanceReady)
       {
           Console.WriteLine("Instance already created.");
           newInstance = instances.First(i => i.DBInstanceIdentifier ==
instanceIdentifier);
       }
       else
       {
           Console.WriteLine("Please enter an admin user name:");
           var username = Console.ReadLine();

           Console.WriteLine("Please enter an admin password:");
           var password = Console.ReadLine();

           newInstance = await rdsWrapper.CreateDBInstance(
               "ExampleInstance",
               instanceIdentifier,
               parameterGroup.DBParameterGroupName,
               engineName,
               engineVersion,
               instanceClass,
               20,
               username,
               password
           );
```

```
            // 11. Wait for the DB instance to be ready.

            Console.WriteLine("11. Waiting for DB instance to be ready...");
            while (!isInstanceReady)
            {
                instances = await
 rdsWrapper.DescribeDBInstances(instanceIdentifier);
                isInstanceReady = instances.FirstOrDefault()?.DBInstanceStatus ==
 "available";
                newInstance = instances.First();
                Thread.Sleep(30000);
            }
        }

        Console.WriteLine(sepBar);
        return newInstance;
    }

    /// <summary>
    /// Display a connection string for an RDS DB instance.
    /// </summary>
    /// <param name="instance">The DB instance to use to get a connection
 string.</param>
    public static void DisplayConnectionString(DBInstance instance)
    {
        Console.WriteLine(sepBar);
        // Display the connection string.
        Console.WriteLine("12. New DB instance connection string: ");
        Console.WriteLine(
            $"\n{engine} -h {instance.Endpoint.Address} -P
 {instance.Endpoint.Port} "
            + $"-u {instance.MasterUsername} -p [YOUR PASSWORD]\n");

        Console.WriteLine(sepBar);
    }

    /// <summary>
    /// Create a snapshot from an RDS DB instance.
    /// </summary>
    /// <param name="instance">DB instance to use when creating a snapshot.</
param>
    /// <returns>The snapshot object.</returns>
    public static async Task<DBSnapshot> CreateSnapshot(DBInstance instance)
    {
```

```
        Console.WriteLine(sepBar);
        // Create a snapshot.
        Console.WriteLine($"13. Creating snapshot from DB instance
{instance.DBInstanceIdentifier}.");
        var snapshot = await
rdsWrapper.CreateDBSnapshot(instance.DBInstanceIdentifier, "ExampleSnapshot-" +
DateTime.Now.Ticks);

        // Wait for the snapshot to be available
        bool isSnapshotReady = false;

        Console.WriteLine($"14. Waiting for snapshot to be ready...");
        while (!isSnapshotReady)
        {
            var snapshots = await
rdsWrapper.DescribeDBSnapshots(instance.DBInstanceIdentifier);
            isSnapshotReady = snapshots.FirstOrDefault()?.Status == "available";
            snapshot = snapshots.First();
            Thread.Sleep(30000);
        }

        Console.WriteLine(
            $"Snapshot {snapshot.DBSnapshotIdentifier} status is
{snapshot.Status}.");
        Console.WriteLine(sepBar);
        return snapshot;
    }

    /// <summary>
    /// Delete an RDS DB instance.
    /// </summary>
    /// <param name="instance">The DB instance to delete.</param>
    /// <returns>Async task.</returns>
    public static async Task DeleteRdsInstance(DBInstance newInstance)
    {
        Console.WriteLine(sepBar);
        // Delete the DB instance.
        Console.WriteLine($"15. Delete the DB instance
{newInstance.DBInstanceIdentifier}.");
        await rdsWrapper.DeleteDBInstance(newInstance.DBInstanceIdentifier);

        // Wait for the DB instance to delete.
        Console.WriteLine($"16. Waiting for the DB instance to delete...");
        bool isInstanceDeleted = false;
```

```
        while (!isInstanceDeleted)
        {
            var instance = await rdsWrapper.DescribeDBInstances();
            isInstanceDeleted = instance.All(i => i.DBInstanceIdentifier !=
 newInstance.DBInstanceIdentifier);
            Thread.Sleep(30000);
        }

        Console.WriteLine("DB instance deleted.");
        Console.WriteLine(sepBar);
    }

    /// <summary>
    /// Delete a DB parameter group.
    /// </summary>
    /// <param name="parameterGroup">The parameter group to delete.</param>
    /// <returns>Async task.</returns>
    public static async Task DeleteParameterGroup(DBParameterGroup
 parameterGroup)
    {
        Console.WriteLine(sepBar);
        // Delete the parameter group.
        Console.WriteLine($"17. Delete the DB parameter group
 {parameterGroup.DBParameterGroupName}.");
        await
 rdsWrapper.DeleteDBParameterGroup(parameterGroup.DBParameterGroupName);

        Console.WriteLine(sepBar);
    }
```

Méthodes d'encapsulation utilisées par le scénario pour les actions d'instance de base de données.

```
/// <summary>
/// Wrapper methods to use Amazon Relational Database Service (Amazon RDS) with
 DB instance operations.
/// </summary>
public partial class RDSWrapper
{
    private readonly IAmazonRDS _amazonRDS;
```

```
    public RDSWrapper(IAmazonRDS amazonRDS)
    {
        _amazonRDS = amazonRDS;
    }


    /// <summary>
    /// Get a list of DB engine versions for a particular DB engine.
    /// </summary>
    /// <param name="engine">Name of the engine.</param>
    /// <param name="dbParameterGroupFamily">Optional parameter group family
name.</param>
    /// <returns>List of DBEngineVersions.</returns>
    public async Task<List<DBEngineVersion>> DescribeDBEngineVersions(string
engine,
        string dbParameterGroupFamily = null)
    {
        var response = await _amazonRDS.DescribeDBEngineVersionsAsync(
            new DescribeDBEngineVersionsRequest()
            {
                Engine = engine,
                DBParameterGroupFamily = dbParameterGroupFamily
            });
        return response.DBEngineVersions;
    }



    /// <summary>
    /// Get a list of orderable DB instance options for a specific
    /// engine and engine version.
    /// </summary>
    /// <param name="engine">Name of the engine.</param>
    /// <param name="engineVersion">Version of the engine.</param>
    /// <returns>List of OrderableDBInstanceOptions.</returns>
    public async Task<List<OrderableDBInstanceOption>>
DescribeOrderableDBInstanceOptions(string engine, string engineVersion)
    {
        // Use a paginator to get a list of DB instance options.
        var results = new List<OrderableDBInstanceOption>();
        var paginateInstanceOptions =
_amazonRDS.Paginators.DescribeOrderableDBInstanceOptions(
            new DescribeOrderableDBInstanceOptionsRequest()
            {
```

```
                Engine = engine,
                EngineVersion = engineVersion,
            });
        // Get the entire list using the paginator.
        await foreach (var instanceOptions in
paginateInstanceOptions.OrderableDBInstanceOptions)
        {
            results.Add(instanceOptions);
        }
        return results;
    }



    /// <summary>
    /// Returns a list of DB instances.
    /// </summary>
    /// <param name="dbInstanceIdentifier">Optional name of a specific DB
instance.</param>
    /// <returns>List of DB instances.</returns>
    public async Task<List<DBInstance>> DescribeDBInstances(string
dbInstanceIdentifier = null)
    {
        var results = new List<DBInstance>();
        var instancesPaginator = _amazonRDS.Paginators.DescribeDBInstances(
            new DescribeDBInstancesRequest
            {
                DBInstanceIdentifier = dbInstanceIdentifier
            });
        // Get the entire list using the paginator.
        await foreach (var instances in instancesPaginator.DBInstances)
        {
            results.Add(instances);
        }
        return results;
    }



    /// <summary>
    /// Create an RDS DB instance with a particular set of properties. Use the
action DescribeDBInstancesAsync
    /// to determine when the DB instance is ready to use.
    /// </summary>
```

```
   /// <param name="dbName">Name for the DB instance.</param>
   /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
   /// <param name="parameterGroupName">DB parameter group to associate with the
instance.</param>
   /// <param name="dbEngine">The engine for the DB instance.</param>
   /// <param name="dbEngineVersion">Version for the DB instance.</param>
   /// <param name="instanceClass">Class for the DB instance.</param>
   /// <param name="allocatedStorage">The amount of storage in gibibytes (GiB)
to allocate to the DB instance.</param>
   /// <param name="adminName">Admin user name.</param>
   /// <param name="adminPassword">Admin user password.</param>
   /// <returns>DB instance object.</returns>
   public async Task<DBInstance> CreateDBInstance(string dbName, string
dbInstanceIdentifier,
       string parameterGroupName, string dbEngine, string dbEngineVersion,
       string instanceClass, int allocatedStorage, string adminName, string
adminPassword)
   {
       var response = await _amazonRDS.CreateDBInstanceAsync(
           new CreateDBInstanceRequest()
           {
               DBName = dbName,
               DBInstanceIdentifier = dbInstanceIdentifier,
               DBParameterGroupName = parameterGroupName,
               Engine = dbEngine,
               EngineVersion = dbEngineVersion,
               DBInstanceClass = instanceClass,
               AllocatedStorage = allocatedStorage,
               MasterUsername = adminName,
               MasterUserPassword = adminPassword
           });

       return response.DBInstance;
   }



   /// <summary>
   /// Delete a particular DB instance.
   /// </summary>
   /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
   /// <returns>DB instance object.</returns>
   public async Task<DBInstance> DeleteDBInstance(string dbInstanceIdentifier)
   {
```

```
        var response = await _amazonRDS.DeleteDBInstanceAsync(
            new DeleteDBInstanceRequest()
            {
                DBInstanceIdentifier = dbInstanceIdentifier,
                SkipFinalSnapshot = true,
                DeleteAutomatedBackups = true
            });

        return response.DBInstance;
    }
```

Méthodes d'encapsulation utilisées par le scénario pour les groupes de paramètres de base de données.

```
/// <summary>
/// Wrapper methods to use Amazon Relational Database Service (Amazon RDS) with
 parameter groups.
/// </summary>
public partial class RDSWrapper
{

    /// <summary>
    /// Get descriptions of DB parameter groups.
    /// </summary>
    /// <param name="name">Optional name of the DB parameter group to describe.</
param>
    /// <returns>The list of DB parameter group descriptions.</returns>
    public async Task<List<DBParameterGroup>> DescribeDBParameterGroups(string
 name = null)
    {
        var response = await _amazonRDS.DescribeDBParameterGroupsAsync(
            new DescribeDBParameterGroupsRequest()
            {
                DBParameterGroupName = name
            });
        return response.DBParameterGroups;
    }
```

```
    /// <summary>
    /// Create a new DB parameter group. Use the action
 DescribeDBParameterGroupsAsync
    /// to determine when the DB parameter group is ready to use.
    /// </summary>
    /// <param name="name">Name of the DB parameter group.</param>
    /// <param name="family">Family of the DB parameter group.</param>
    /// <param name="description">Description of the DB parameter group.</param>
    /// <returns>The new DB parameter group.</returns>
    public async Task<DBParameterGroup> CreateDBParameterGroup(
        string name, string family, string description)
    {
        var response = await _amazonRDS.CreateDBParameterGroupAsync(
            new CreateDBParameterGroupRequest()
            {
                DBParameterGroupName = name,
                DBParameterGroupFamily = family,
                Description = description
            });
        return response.DBParameterGroup;
    }



    /// <summary>
    /// Update a DB parameter group. Use the action
 DescribeDBParameterGroupsAsync
    /// to determine when the DB parameter group is ready to use.
    /// </summary>
    /// <param name="name">Name of the DB parameter group.</param>
    /// <param name="parameters">List of parameters. Maximum of 20 per request.</
param>
    /// <returns>The updated DB parameter group name.</returns>
    public async Task<string> ModifyDBParameterGroup(
        string name, List<Parameter> parameters)
    {
        var response = await _amazonRDS.ModifyDBParameterGroupAsync(
            new ModifyDBParameterGroupRequest()
            {
                DBParameterGroupName = name,
                Parameters = parameters,
            });
        return response.DBParameterGroupName;
    }
```

```
    /// <summary>
    /// Delete a DB parameter group. The group cannot be a default DB parameter
group
    /// or be associated with any DB instances.
    /// </summary>
    /// <param name="name">Name of the DB parameter group.</param>
    /// <returns>True if successful.</returns>
    public async Task<bool> DeleteDBParameterGroup(string name)
    {
        var response = await _amazonRDS.DeleteDBParameterGroupAsync(
            new DeleteDBParameterGroupRequest()
            {
                DBParameterGroupName = name,
            });
        return response.HttpStatusCode == HttpStatusCode.OK;
    }



    /// <summary>
    /// Get a list of DB parameters from a specific parameter group.
    /// </summary>
    /// <param name="dbParameterGroupName">Name of a specific DB parameter
group.</param>
    /// <param name="source">Optional source for selecting parameters.</param>
    /// <returns>List of parameter values.</returns>
    public async Task<List<Parameter>> DescribeDBParameters(string
dbParameterGroupName, string source = null)
    {
        var results = new List<Parameter>();
        var paginateParameters = _amazonRDS.Paginators.DescribeDBParameters(
            new DescribeDBParametersRequest()
            {
                DBParameterGroupName = dbParameterGroupName,
                Source = source
            });
        // Get the entire list using the paginator.
        await foreach (var parameters in paginateParameters.Parameters)
        {
            results.Add(parameters);
        }
```

```
        return results;
    }
```

Méthodes d'encapsulation utilisées par le scénario pour les actions d'instantané de base de données.

```
/// <summary>
/// Wrapper methods to use Amazon Relational Database Service (Amazon RDS) with
 snapshots.
/// </summary>
public partial class RDSWrapper
{

    /// <summary>
    /// Create a snapshot of a DB instance.
    /// </summary>
    /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
    /// <param name="snapshotIdentifier">Identifier for the snapshot.</param>
    /// <returns>DB snapshot object.</returns>
    public async Task<DBSnapshot> CreateDBSnapshot(string dbInstanceIdentifier,
 string snapshotIdentifier)
    {
        var response = await _amazonRDS.CreateDBSnapshotAsync(
            new CreateDBSnapshotRequest()
            {
                DBSnapshotIdentifier = snapshotIdentifier,
                DBInstanceIdentifier = dbInstanceIdentifier
            });

        return response.DBSnapshot;
    }



    /// <summary>
    /// Return a list of DB snapshots for a particular DB instance.
    /// </summary>
    /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
    /// <returns>List of DB snapshots.</returns>
```

```
   public async Task<List<DBSnapshot>> DescribeDBSnapshots(string
dbInstanceIdentifier)
   {
       var results = new List<DBSnapshot>();
       var snapshotsPaginator = _amazonRDS.Paginators.DescribeDBSnapshots(
           new DescribeDBSnapshotsRequest()
           {
               DBInstanceIdentifier = dbInstanceIdentifier
           });

       // Get the entire list using the paginator.
       await foreach (var snapshots in snapshotsPaginator.DBSnapshots)
       {
           results.Add(snapshots);
       }
       return results;
   }
```

- Pour plus d'informations sur l'API consultez les rubriques suivantes dans la référence de l'API AWS SDK pour .NET .
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

//! Routine which creates an Amazon RDS instance and demonstrates several
 operations
//! on that instance.
/*!
 \sa gettingStartedWithDBInstances()
 \param clientConfiguration: AWS client configuration.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::gettingStartedWithDBInstances(
        const Aws::Client::ClientConfiguration &clientConfig) {
    Aws::RDS::RDSClient client(clientConfig);

    printAsterisksLine();
    std::cout << "Welcome to the Amazon Relational Database Service (Amazon RDS)"
              << std::endl;
    std::cout << "get started with DB instances demo." << std::endl;
    printAsterisksLine();

    std::cout << "Checking for an existing DB parameter group named '" <<
              PARAMETER_GROUP_NAME << "'." << std::endl;
    Aws::String dbParameterGroupFamily("Undefined");
    bool parameterGroupFound = true;
    {
        // 1. Check if the DB parameter group already exists.
        Aws::RDS::Model::DescribeDBParameterGroupsRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);

        Aws::RDS::Model::DescribeDBParameterGroupsOutcome outcome =
```

```
                client.DescribeDBParameterGroups(request);

        if (outcome.IsSuccess()) {
            std::cout << "DB parameter group named '" <<
                      PARAMETER_GROUP_NAME << "' already exists." << std::endl;
            dbParameterGroupFamily = outcome.GetResult().GetDBParameterGroups()
[0].GetDBParameterGroupFamily();
        }
        else if (outcome.GetError().GetErrorType() ==
                 Aws::RDS::RDSErrors::D_B_PARAMETER_GROUP_NOT_FOUND_FAULT) {
            std::cout << "DB parameter group named '" <<
                      PARAMETER_GROUP_NAME << "' does not exist." << std::endl;
            parameterGroupFound = false;
        }
        else {
            std::cerr << "Error with RDS::DescribeDBParameterGroups. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
    }

    if (!parameterGroupFound) {
        Aws::Vector<Aws::RDS::Model::DBEngineVersion> engineVersions;

        // 2. Get available engine versions for the specified engine.
        if (!getDBEngineVersions(DB_ENGINE, NO_PARAMETER_GROUP_FAMILY,
                                 engineVersions, client)) {
            return false;
        }

        std::cout << "Getting available database engine versions for " <<
 DB_ENGINE
                  << "."
                  << std::endl;
        std::vector<Aws::String> families;
        for (const Aws::RDS::Model::DBEngineVersion &version: engineVersions) {
            Aws::String family = version.GetDBParameterGroupFamily();
            if (std::find(families.begin(), families.end(), family) ==
                families.end()) {
                families.push_back(family);
                std::cout << "  " << families.size() << ": " << family <<
 std::endl;
            }
```

```
        }

        int choice = askQuestionForIntRange("Which family do you want to use? ",
1,
                                            static_cast<int>(families.size()));
        dbParameterGroupFamily = families[choice - 1];
    }
    if (!parameterGroupFound) {
        // 3.  Create a DB parameter group.
        Aws::RDS::Model::CreateDBParameterGroupRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        request.SetDBParameterGroupFamily(dbParameterGroupFamily);
        request.SetDescription("Example parameter group.");

        Aws::RDS::Model::CreateDBParameterGroupOutcome outcome =
                client.CreateDBParameterGroup(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB parameter group was successfully created."
                      << std::endl;
        }
        else {
            std::cerr << "Error with RDS::CreateDBParameterGroup. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
    }

    printAsterisksLine();
    std::cout << "Let's set some parameter values in your parameter group."
              << std::endl;

    Aws::String marker;
    Aws::Vector<Aws::RDS::Model::Parameter> autoIncrementParameters;
    // 4.  Get the parameters in the DB parameter group.
    if (!getDBParameters(PARAMETER_GROUP_NAME, AUTO_INCREMENT_PREFIX, NO_SOURCE,
                         autoIncrementParameters,
                         client)) {
        cleanUpResources(PARAMETER_GROUP_NAME, "", client);
        return false;
    }

    Aws::Vector<Aws::RDS::Model::Parameter> updateParameters;
```

```
    for (Aws::RDS::Model::Parameter &autoIncParameter: autoIncrementParameters) {
        if (autoIncParameter.GetIsModifiable() &&
            (autoIncParameter.GetDataType() == "integer")) {
            std::cout << "The " << autoIncParameter.GetParameterName()
                      << " is described as: " <<
                      autoIncParameter.GetDescription() << "." << std::endl;
            if (autoIncParameter.ParameterValueHasBeenSet()) {
                std::cout << "The current value is "
                          << autoIncParameter.GetParameterValue()
                          << "." << std::endl;
            }
            std::vector<int> splitValues = splitToInts(
                    autoIncParameter.GetAllowedValues(), '-');
            if (splitValues.size() == 2) {
                int newValue = askQuestionForIntRange(
                        Aws::String("Enter a new value in the range ") +
                        autoIncParameter.GetAllowedValues() + ": ",
                        splitValues[0], splitValues[1]);
                autoIncParameter.SetParameterValue(std::to_string(newValue));
                updateParameters.push_back(autoIncParameter);

            }
            else {
                std::cerr << "Error parsing " <<
autoIncParameter.GetAllowedValues()
                          << std::endl;
            }
        }
    }

    {
        // 5.  Modify the auto increment parameters in the group.
        Aws::RDS::Model::ModifyDBParameterGroupRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        request.SetParameters(updateParameters);

        Aws::RDS::Model::ModifyDBParameterGroupOutcome outcome =
                client.ModifyDBParameterGroup(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB parameter group was successfully modified."
                      << std::endl;
        }
```

```
        else {
            std::cerr << "Error with RDS::ModifyDBParameterGroup. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
        }
    }

    std::cout
            << "You can get a list of parameters you've set by specifying a
source of 'user'."
            << std::endl;

    Aws::Vector<Aws::RDS::Model::Parameter> userParameters;
    // 6.  Display the modified parameters in the group.
    if (!getDBParameters(PARAMETER_GROUP_NAME, NO_NAME_PREFIX, "user",
userParameters,
                         client)) {
        cleanUpResources(PARAMETER_GROUP_NAME, "", client);
        return false;
    }

    for (const auto &userParameter: userParameters) {
        std::cout << "  " << userParameter.GetParameterName() << ", " <<
                  userParameter.GetDescription() << ", parameter value - "
                  << userParameter.GetParameterValue() << std::endl;
    }

    printAsterisksLine();
    std::cout << "Checking for an existing DB instance." << std::endl;

    Aws::RDS::Model::DBInstance dbInstance;
    // 7.  Check if the DB instance already exists.
    if (!describeDBInstance(DB_INSTANCE_IDENTIFIER, dbInstance, client)) {
        cleanUpResources(PARAMETER_GROUP_NAME, "", client);
        return false;
    }

    if (dbInstance.DbInstancePortHasBeenSet()) {
        std::cout << "The DB instance already exists." << std::endl;
    }
    else {
        std::cout << "Let's create a DB instance." << std::endl;
        const Aws::String administratorName = askQuestion(
                "Enter an administrator username for the database: ");
```

```
        const Aws::String administratorPassword = askQuestion(
                "Enter a password for the administrator (at least 8 characters):
");
        Aws::Vector<Aws::RDS::Model::DBEngineVersion> engineVersions;

        // 8.  Get a list of available engine versions.
        if (!getDBEngineVersions(DB_ENGINE, dbParameterGroupFamily,
engineVersions,
                                 client)) {
            cleanUpResources(PARAMETER_GROUP_NAME, "", client);
            return false;
        }

        std::cout << "The available engines for your parameter group are:" <<
std::endl;

        int index = 1;
        for (const Aws::RDS::Model::DBEngineVersion &engineVersion:
engineVersions) {
            std::cout << "  " << index << ": " <<
engineVersion.GetEngineVersion()
                      << std::endl;
            ++index;
        }
        int choice = askQuestionForIntRange("Which engine do you want to use? ",
1,

static_cast<int>(engineVersions.size()));
        const Aws::RDS::Model::DBEngineVersion engineVersion =
engineVersions[choice -
                                                                              1];

        Aws::String dbInstanceClass;
        // 9.  Get a list of micro instance classes.
        if (!chooseMicroDBInstanceClass(engineVersion.GetEngine(),
                                        engineVersion.GetEngineVersion(),
                                        dbInstanceClass,
                                        client)) {
            cleanUpResources(PARAMETER_GROUP_NAME, "", client);
            return false;
        }

        std::cout << "Creating a DB instance named '" << DB_INSTANCE_IDENTIFIER
                  << "' and database '" << DB_NAME << "'.\n"
```

```
                << "The DB instance is configured to use your custom parameter
group '"
                << PARAMETER_GROUP_NAME << "',\n"
                << "selected engine version " <<
engineVersion.GetEngineVersion()
                << ",\n"
                << "selected DB instance class '" << dbInstanceClass << "',"
                << " and " << DB_ALLOCATED_STORAGE << " GiB of " <<
DB_STORAGE_TYPE
                << " storage.\nThis typically takes several minutes." <<
std::endl;

       Aws::RDS::Model::CreateDBInstanceRequest request;
       request.SetDBName(DB_NAME);
       request.SetDBInstanceIdentifier(DB_INSTANCE_IDENTIFIER);
       request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
       request.SetEngine(engineVersion.GetEngine());
       request.SetEngineVersion(engineVersion.GetEngineVersion());
       request.SetDBInstanceClass(dbInstanceClass);
       request.SetStorageType(DB_STORAGE_TYPE);
       request.SetAllocatedStorage(DB_ALLOCATED_STORAGE);
       request.SetMasterUsername(administratorName);
       request.SetMasterUserPassword(administratorPassword);

       Aws::RDS::Model::CreateDBInstanceOutcome outcome =
               client.CreateDBInstance(request);

       if (outcome.IsSuccess()) {
           std::cout << "The DB instance creation has started."
                     << std::endl;
       }
       else {
           std::cerr << "Error with RDS::CreateDBInstance. "
                     << outcome.GetError().GetMessage()
                     << std::endl;
           cleanUpResources(PARAMETER_GROUP_NAME, "", client);
           return false;
       }
   }

   std::cout << "Waiting for the DB instance to become available." << std::endl;

   int counter = 0;
   // 11. Wait for the DB instance to become available.
```

```
    do {
        std::this_thread::sleep_for(std::chrono::seconds(1));
        ++counter;
        if (counter > 900) {
            std::cerr << "Wait for instance to become available timed out ofter "
                      << counter
                      << " seconds." << std::endl;
            cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
client);
            return false;
        }

        dbInstance = Aws::RDS::Model::DBInstance();
        if (!describeDBInstance(DB_INSTANCE_IDENTIFIER, dbInstance, client)) {
            cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
client);
            return false;
        }

        if ((counter % 20) == 0) {
            std::cout << "Current DB instance status is '"
                      << dbInstance.GetDBInstanceStatus()
                      << "' after " << counter << " seconds." << std::endl;
        }
    } while (dbInstance.GetDBInstanceStatus() != "available");

    if (dbInstance.GetDBInstanceStatus() == "available") {
        std::cout << "The DB instance has been created." << std::endl;
    }

    printAsterisksLine();

    // 12. Display the connection string that can be used to connect a 'mysql'
shell to the database.
    displayConnection(dbInstance);

    printAsterisksLine();

    if (askYesNoQuestion(
            "Do you want to create a snapshot of your DB instance (y/n)? ")) {
        Aws::String snapshotID(DB_INSTANCE_IDENTIFIER + "-" +
                               Aws::String(Aws::Utils::UUID::RandomUUID()));
        {
```

```
            std::cout << "Creating a snapshot named " << snapshotID << "." <<
std::endl;
            std::cout << "This typically takes a few minutes." << std::endl;

            // 13. Create a snapshot of the DB instance.
            Aws::RDS::Model::CreateDBSnapshotRequest request;
            request.SetDBInstanceIdentifier(DB_INSTANCE_IDENTIFIER);
            request.SetDBSnapshotIdentifier(snapshotID);

            Aws::RDS::Model::CreateDBSnapshotOutcome outcome =
                    client.CreateDBSnapshot(request);

            if (outcome.IsSuccess()) {
                std::cout << "Snapshot creation has started."
                          << std::endl;
            }
            else {
                std::cerr << "Error with RDS::CreateDBSnapshot. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
client);
                return false;
            }
        }

        std::cout << "Waiting for snapshot to become available." << std::endl;

        Aws::RDS::Model::DBSnapshot snapshot;
        counter = 0;
        do {
            std::this_thread::sleep_for(std::chrono::seconds(1));
            ++counter;
            if (counter > 600) {
                std::cerr << "Wait for snapshot to be available timed out ofter "
                          << counter
                          << " seconds." << std::endl;
                cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
client);
                return false;
            }

            // 14. Wait for the snapshot to become available.
            Aws::RDS::Model::DescribeDBSnapshotsRequest request;
```

```
            request.SetDBSnapshotIdentifier(snapshotID);

            Aws::RDS::Model::DescribeDBSnapshotsOutcome outcome =
                    client.DescribeDBSnapshots(request);

            if (outcome.IsSuccess()) {
                snapshot = outcome.GetResult().GetDBSnapshots()[0];
            }
            else {
                std::cerr << "Error with RDS::DescribeDBSnapshots. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
 client);
                return false;
            }

            if ((counter % 20) == 0) {
                std::cout << "Current snapshot status is '"
                          << snapshot.GetStatus()
                          << "' after " << counter << " seconds." << std::endl;
            }
        } while (snapshot.GetStatus() != "available");

        if (snapshot.GetStatus() != "available") {
            std::cout << "A snapshot has been created." << std::endl;
        }
    }

    printAsterisksLine();

    bool result = true;
    if (askYesNoQuestion(
            "Do you want to delete the DB instance and parameter group (y/n)? "))
 {
        result = cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
 client);
    }

    return result;
}


//! Routine which gets DB parameters using the 'DescribeDBParameters' api.
```

```
/*!
 \sa getDBParameters()
 \param parameterGroupName: The name of the parameter group.
 \param namePrefix: Prefix string to filter results by parameter name.
 \param source: A source such as 'user', ignored if empty.
 \param parametersResult: Vector of 'Parameter' objects returned by the routine.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::getDBParameters(const Aws::String &parameterGroupName,
                                  const Aws::String &namePrefix,
                                  const Aws::String &source,
                                  Aws::Vector<Aws::RDS::Model::Parameter>
 &parametersResult,
                                  const Aws::RDS::RDSClient &client) {
    Aws::String marker;
    do {
        Aws::RDS::Model::DescribeDBParametersRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        if (!marker.empty()) {
            request.SetMarker(marker);
        }
        if (!source.empty()) {
            request.SetSource(source);
        }

        Aws::RDS::Model::DescribeDBParametersOutcome outcome =
                client.DescribeDBParameters(request);

        if (outcome.IsSuccess()) {
            const Aws::Vector<Aws::RDS::Model::Parameter> &parameters =
                    outcome.GetResult().GetParameters();
            for (const Aws::RDS::Model::Parameter &parameter: parameters) {
                if (!namePrefix.empty()) {
                    if (parameter.GetParameterName().find(namePrefix) == 0) {
                        parametersResult.push_back(parameter);
                    }
                }
                else {
                    parametersResult.push_back(parameter);
                }
            }

            marker = outcome.GetResult().GetMarker();
```

```
        }
        else {
            std::cerr << "Error with RDS::DescribeDBParameters. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
    } while (!marker.empty());

    return true;
}


//! Routine which gets available DB engine versions for an engine name and
//! an optional parameter group family.
/*!
 \sa getDBEngineVersions()
 \param engineName: A DB engine name.
 \param parameterGroupFamily: A parameter group family name, ignored if empty.
 \param engineVersionsResult: Vector of 'DBEngineVersion' objects returned by the
 routine.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::getDBEngineVersions(const Aws::String &engineName,
                                      const Aws::String &parameterGroupFamily,

 Aws::Vector<Aws::RDS::Model::DBEngineVersion> &engineVersionsResult,
                                      const Aws::RDS::RDSClient &client) {
    Aws::RDS::Model::DescribeDBEngineVersionsRequest request;
    request.SetEngine(engineName);
    if (!parameterGroupFamily.empty()) {
        request.SetDBParameterGroupFamily(parameterGroupFamily);
    }

    engineVersionsResult.clear();
    Aws::String marker; // Used for pagination.

    do {
        if (!marker.empty()) {
            request.SetMarker(marker);
        }
```

```
        Aws::RDS::Model::DescribeDBEngineVersionsOutcome outcome =
                client.DescribeDBEngineVersions(request);

        if (outcome.IsSuccess()) {
            auto &engineVersions = outcome.GetResult().GetDBEngineVersions();
            engineVersionsResult.insert(engineVersionsResult.end(),
 engineVersions.begin(),
                                        engineVersions.end());
            marker = outcome.GetResult().GetMarker();
        }
        else {
            std::cerr << "Error with RDS::DescribeDBEngineVersionsRequest. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }

    } while (!marker.empty());


    return true;
}


//! Routine which gets a DB instance description.
/*!
 \sa describeDBInstance()
 \param dbInstanceIdentifier: A DB instance identifier.
 \param instanceResult: The 'DBInstance' object containing the description.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::describeDBInstance(const Aws::String &dbInstanceIdentifier,
                                     Aws::RDS::Model::DBInstance &instanceResult,
                                     const Aws::RDS::RDSClient &client) {
    Aws::RDS::Model::DescribeDBInstancesRequest request;
    request.SetDBInstanceIdentifier(dbInstanceIdentifier);

    Aws::RDS::Model::DescribeDBInstancesOutcome outcome =
            client.DescribeDBInstances(request);

    bool result = true;
    if (outcome.IsSuccess()) {
        instanceResult = outcome.GetResult().GetDBInstances()[0];
```

```
    }
    else if (outcome.GetError().GetErrorType() !=
             Aws::RDS::RDSErrors::D_B_INSTANCE_NOT_FOUND_FAULT) {
        result = false;
        std::cerr << "Error with RDS::DescribeDBInstances. "
                  << outcome.GetError().GetMessage()
                  << std::endl;
    }
        // This example does not log an error if the DB instance does not exist.
        // Instead, instanceResult is set to empty.
    else {
        instanceResult = Aws::RDS::Model::DBInstance();
    }

    return result;
}


//! Routine which gets available 'micro' DB instance classes, displays the list
//! to the user, and returns the user selection.
/*!
 \sa chooseMicroDBInstanceClass()
 \param engineName: The DB engine name.
 \param engineVersion: The DB engine version.
 \param dbInstanceClass: String for DB instance class chosen by the user.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::chooseMicroDBInstanceClass(const Aws::String &engine,
                                             const Aws::String &engineVersion,
                                             Aws::String &dbInstanceClass,
                                             const Aws::RDS::RDSClient &client) {
    std::vector<Aws::String> instanceClasses;
    Aws::String marker;
    do {
        Aws::RDS::Model::DescribeOrderableDBInstanceOptionsRequest request;
        request.SetEngine(engine);
        request.SetEngineVersion(engineVersion);
        if (!marker.empty()) {
            request.SetMarker(marker);
        }

        Aws::RDS::Model::DescribeOrderableDBInstanceOptionsOutcome outcome =
                client.DescribeOrderableDBInstanceOptions(request);
```

```
        if (outcome.IsSuccess()) {
            const Aws::Vector<Aws::RDS::Model::OrderableDBInstanceOption>
 &options =
                    outcome.GetResult().GetOrderableDBInstanceOptions();
            for (const Aws::RDS::Model::OrderableDBInstanceOption &option:
 options) {
                const Aws::String &instanceClass = option.GetDBInstanceClass();
                if (instanceClass.find("micro") != std::string::npos) {
                    if (std::find(instanceClasses.begin(), instanceClasses.end(),
                                  instanceClass) ==
                        instanceClasses.end()) {
                        instanceClasses.push_back(instanceClass);
                    }
                }
            }
            marker = outcome.GetResult().GetMarker();
        }
        else {
            std::cerr << "Error with RDS::DescribeOrderableDBInstanceOptions. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
    } while (!marker.empty());

    std::cout << "The available micro DB instance classes for your database
 engine are:"
              << std::endl;
    for (int i = 0; i < instanceClasses.size(); ++i) {
        std::cout << "   " << i + 1 << ": " << instanceClasses[i] << std::endl;
    }

    int choice = askQuestionForIntRange(
            "Which micro DB instance class do you want to use? ",
            1, static_cast<int>(instanceClasses.size()));
    dbInstanceClass = instanceClasses[choice - 1];
    return true;
}

//! Routine which deletes resources created by the scenario.
/*!
\sa cleanUpResources()
\param parameterGroupName: A parameter group name, this may be empty.
```

```
\param dbInstanceIdentifier: A DB instance identifier, this may be empty.
\param client: 'RDSClient' instance.
\return bool: Successful completion.
*/
bool AwsDoc::RDS::cleanUpResources(const Aws::String &parameterGroupName,
                                   const Aws::String &dbInstanceIdentifier,
                                   const Aws::RDS::RDSClient &client) {
    bool result = true;
    if (!dbInstanceIdentifier.empty()) {
        {
            // 15. Delete the DB instance.
            Aws::RDS::Model::DeleteDBInstanceRequest request;
            request.SetDBInstanceIdentifier(dbInstanceIdentifier);
            request.SetSkipFinalSnapshot(true);
            request.SetDeleteAutomatedBackups(true);

            Aws::RDS::Model::DeleteDBInstanceOutcome outcome =
                    client.DeleteDBInstance(request);

            if (outcome.IsSuccess()) {
                std::cout << "DB instance deletion has started."
                          << std::endl;
            }
            else {
                std::cerr << "Error with RDS::DeleteDBInstance. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                result = false;
            }
        }

        std::cout
                << "Waiting for DB instance to delete before deleting the
 parameter group."
                << std::endl;
        std::cout << "This may take a while." << std::endl;

        int counter = 0;
        Aws::RDS::Model::DBInstance dbInstance;
        do {
            std::this_thread::sleep_for(std::chrono::seconds(1));
            ++counter;
            if (counter > 800) {
```

```
                std::cerr << "Wait for instance to delete timed out ofter " <<
 counter
                          << " seconds." << std::endl;
                return false;
            }

            dbInstance = Aws::RDS::Model::DBInstance();
            // 16. Wait for the DB instance to be deleted.
            if (!describeDBInstance(dbInstanceIdentifier, dbInstance, client)) {
                return false;
            }

            if (dbInstance.DBInstanceIdentifierHasBeenSet() && (counter % 20) ==
 0) {
                std::cout << "Current DB instance status is '"
                          << dbInstance.GetDBInstanceStatus()
                          << "' after " << counter << " seconds." << std::endl;
            }
        } while (dbInstance.DBInstanceIdentifierHasBeenSet());
    }

    if (!parameterGroupName.empty()) {
        // 17. Delete the parameter group.
        Aws::RDS::Model::DeleteDBParameterGroupRequest request;
        request.SetDBParameterGroupName(parameterGroupName);

        Aws::RDS::Model::DeleteDBParameterGroupOutcome outcome =
                client.DeleteDBParameterGroup(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB parameter group was successfully deleted."
                      << std::endl;
        }
        else {
            std::cerr << "Error with RDS::DeleteDBParameterGroup. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            result = false;
        }
    }

    return result;
}
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans la référence de l'API AWS SDK pour C++ .
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Exécutez un scénario interactif à une invite de commande.

```
import (
 "context"
 "fmt"
 "log"
 "sort"
 "strconv"
```

```
 "strings"
 "time"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
 "github.com/awsdocs/aws-doc-sdk-examples/gov2/demotools"
 "github.com/awsdocs/aws-doc-sdk-examples/gov2/rds/actions"
 "github.com/google/uuid"
)

// GetStartedInstances is an interactive example that shows you how to use the
 AWS SDK for Go
// with Amazon Relation Database Service (Amazon RDS) to do the following:
//
//  1. Create a custom DB parameter group and set parameter values.
//  2. Create a DB instance that is configured to use the parameter group. The DB
 instance
//     also contains a database.
//  3. Take a snapshot of the DB instance.
//  4. Delete the DB instance and parameter group.
type GetStartedInstances struct {
 sdkConfig  aws.Config
 instances  actions.DbInstances
 questioner demotools.IQuestioner
 helper     IScenarioHelper
 isTestRun  bool
}

// NewGetStartedInstances constructs a GetStartedInstances instance from a
 configuration.
// It uses the specified config to get an Amazon RDS
// client and create wrappers for the actions used in the scenario.
func NewGetStartedInstances(sdkConfig aws.Config, questioner
 demotools.IQuestioner,
 helper IScenarioHelper) GetStartedInstances {
 rdsClient := rds.NewFromConfig(sdkConfig)
 return GetStartedInstances{
  sdkConfig:  sdkConfig,
  instances:  actions.DbInstances{RdsClient: rdsClient},
  questioner: questioner,
  helper:     helper,
 }
}
```

```
// Run runs the interactive scenario.
func (scenario GetStartedInstances) Run(ctx context.Context, dbEngine string,
 parameterGroupName string,
 instanceName string, dbName string) {
 defer func() {
  if r := recover(); r != nil {
   log.Println("Something went wrong with the demo.")
  }
 }()

 log.Println(strings.Repeat("-", 88))
 log.Println("Welcome to the Amazon Relational Database Service (Amazon RDS) DB
 Instance demo.")
 log.Println(strings.Repeat("-", 88))

 parameterGroup := scenario.CreateParameterGroup(ctx, dbEngine,
 parameterGroupName)
 scenario.SetUserParameters(ctx, parameterGroupName)
 instance := scenario.CreateInstance(ctx, instanceName, dbEngine, dbName,
 parameterGroup)
 scenario.DisplayConnection(instance)
 scenario.CreateSnapshot(ctx, instance)
 scenario.Cleanup(ctx, instance, parameterGroup)

 log.Println(strings.Repeat("-", 88))
 log.Println("Thanks for watching!")
 log.Println(strings.Repeat("-", 88))
}

// CreateParameterGroup shows how to get available engine versions for a
 specified
// database engine and create a DB parameter group that is compatible with a
// selected engine family.
func (scenario GetStartedInstances) CreateParameterGroup(ctx context.Context,
 dbEngine string,
 parameterGroupName string) *types.DBParameterGroup {

 log.Printf("Checking for an existing DB parameter group named %v.\n",
  parameterGroupName)
 parameterGroup, err := scenario.instances.GetParameterGroup(ctx,
 parameterGroupName)
 if err != nil {
  panic(err)
```

```
 }
 if parameterGroup == nil {
  log.Printf("Getting available database engine versions for %v.\n", dbEngine)
  engineVersions, err := scenario.instances.GetEngineVersions(ctx, dbEngine, "")
  if err != nil {
   panic(err)
  }

  familySet := map[string]struct{}{}
  for _, family := range engineVersions {
   familySet[*family.DBParameterGroupFamily] = struct{}{}
  }
  var families []string
  for family := range familySet {
   families = append(families, family)
  }
  sort.Strings(families)
  familyIndex := scenario.questioner.AskChoice("Which family do you want to use?
\n", families)
  log.Println("Creating a DB parameter group.")
  _, err = scenario.instances.CreateParameterGroup(
   ctx, parameterGroupName, families[familyIndex], "Example parameter group.")
  if err != nil {
   panic(err)
  }
  parameterGroup, err = scenario.instances.GetParameterGroup(ctx,
 parameterGroupName)
  if err != nil {
   panic(err)
  }
 }
 log.Printf("Parameter group %v:\n", *parameterGroup.DBParameterGroupFamily)
 log.Printf("\tName: %v\n", *parameterGroup.DBParameterGroupName)
 log.Printf("\tARN: %v\n", *parameterGroup.DBParameterGroupArn)
 log.Printf("\tFamily: %v\n", *parameterGroup.DBParameterGroupFamily)
 log.Printf("\tDescription: %v\n", *parameterGroup.Description)
 log.Println(strings.Repeat("-", 88))
 return parameterGroup
}

// SetUserParameters shows how to get the parameters contained in a custom
 parameter
// group and update some of the parameter values in the group.
```

```
func (scenario GetStartedInstances) SetUserParameters(ctx context.Context,
 parameterGroupName string) {
 log.Println("Let's set some parameter values in your parameter group.")
 dbParameters, err := scenario.instances.GetParameters(ctx, parameterGroupName,
 "")
 if err != nil {
  panic(err)
 }
 var updateParams []types.Parameter
 for _, dbParam := range dbParameters {
  if strings.HasPrefix(*dbParam.ParameterName, "auto_increment") &&
   *dbParam.IsModifiable && *dbParam.DataType == "integer" {
   log.Printf("The %v parameter is described as:\n\t%v",
    *dbParam.ParameterName, *dbParam.Description)
   rangeSplit := strings.Split(*dbParam.AllowedValues, "-")
   lower, _ := strconv.Atoi(rangeSplit[0])
   upper, _ := strconv.Atoi(rangeSplit[1])
   newValue := scenario.questioner.AskInt(
    fmt.Sprintf("Enter a value between %v and %v:", lower, upper),
    demotools.InIntRange{Lower: lower, Upper: upper})
   dbParam.ParameterValue = aws.String(strconv.Itoa(newValue))
   updateParams = append(updateParams, dbParam)
  }
 }
 err = scenario.instances.UpdateParameters(ctx, parameterGroupName, updateParams)
 if err != nil {
  panic(err)
 }
 log.Println("To get a list of parameters that you set previously, specify a
 source of 'user'.")
 userParameters, err := scenario.instances.GetParameters(ctx, parameterGroupName,
 "user")
 if err != nil {
  panic(err)
 }
 log.Println("Here are the parameters you set:")
 for _, param := range userParameters {
  log.Printf("\t%v: %v\n", *param.ParameterName, *param.ParameterValue)
 }
 log.Println(strings.Repeat("-", 88))
}

// CreateInstance shows how to create a DB instance that contains a database of a
```

```
// specified type. The database is also configured to use a custom DB parameter
 group.
func (scenario GetStartedInstances) CreateInstance(ctx context.Context,
 instanceName string, dbEngine string,
 dbName string, parameterGroup *types.DBParameterGroup) *types.DBInstance {

 log.Println("Checking for an existing DB instance.")
 instance, err := scenario.instances.GetInstance(ctx, instanceName)
 if err != nil {
  panic(err)
 }
 if instance == nil {
  adminUsername := scenario.questioner.Ask(
   "Enter an administrator username for the database: ", demotools.NotEmpty{})
  adminPassword := scenario.questioner.AskPassword(
   "Enter a password for the administrator (at least 8 characters): ", 7)
  engineVersions, err := scenario.instances.GetEngineVersions(ctx, dbEngine,
   *parameterGroup.DBParameterGroupFamily)
  if err != nil {
   panic(err)
  }
  var engineChoices []string
  for _, engine := range engineVersions {
   engineChoices = append(engineChoices, *engine.EngineVersion)
  }
  engineIndex := scenario.questioner.AskChoice(
   "The available engines for your parameter group are:\n", engineChoices)
  engineSelection := engineVersions[engineIndex]
  instOpts, err := scenario.instances.GetOrderableInstances(ctx,
 *engineSelection.Engine,
   *engineSelection.EngineVersion)
  if err != nil {
   panic(err)
  }
  optSet := map[string]struct{}{}
  for _, opt := range instOpts {
   if strings.Contains(*opt.DBInstanceClass, "micro") {
    optSet[*opt.DBInstanceClass] = struct{}{}
   }
  }
  var optChoices []string
  for opt := range optSet {
   optChoices = append(optChoices, opt)
  }
```

```
  sort.Strings(optChoices)
  optIndex := scenario.questioner.AskChoice(
   "The available micro DB instance classes for your database engine are:\n",
 optChoices)
  storageType := "standard"
  allocatedStorage := int32(5)
  log.Printf("Creating a DB instance named %v and database %v.\n"+
   "The DB instance is configured to use your custom parameter group %v,\n"+
   "selected engine %v,\n"+
   "selected DB instance class %v,"+
   "and %v GiB of %v storage.\n"+
   "This typically takes several minutes.",
   instanceName, dbName, *parameterGroup.DBParameterGroupName,
 *engineSelection.EngineVersion,
   optChoices[optIndex], allocatedStorage, storageType)
  instance, err = scenario.instances.CreateInstance(
   ctx, instanceName, dbName, *engineSelection.Engine,
 *engineSelection.EngineVersion,
   *parameterGroup.DBParameterGroupName, optChoices[optIndex], storageType,
   allocatedStorage, adminUsername, adminPassword)
  if err != nil {
   panic(err)
  }
  for *instance.DBInstanceStatus != "available" {
   scenario.helper.Pause(30)
   instance, err = scenario.instances.GetInstance(ctx, instanceName)
   if err != nil {
    panic(err)
   }
  }
  log.Println("Instance created and available.")
 }
 log.Println("Instance data:")
 log.Printf("\tDBInstanceIdentifier: %v\n", *instance.DBInstanceIdentifier)
 log.Printf("\tARN: %v\n", *instance.DBInstanceArn)
 log.Printf("\tStatus: %v\n", *instance.DBInstanceStatus)
 log.Printf("\tEngine: %v\n", *instance.Engine)
 log.Printf("\tEngine version: %v\n", *instance.EngineVersion)
 log.Println(strings.Repeat("-", 88))
 return instance
}

// DisplayConnection displays connection information about a DB instance and tips
// on how to connect to it.
```

```
func (scenario GetStartedInstances) DisplayConnection(instance *types.DBInstance)
 {
 log.Println(
  "You can now connect to your database by using your favorite MySQL client.\n" +
   "One way to connect is by using the 'mysql' shell on an Amazon EC2 instance\n"
 +
   "that is running in the same VPC as your DB instance. Pass the endpoint,\n" +
   "port, and administrator username to 'mysql'. Then, enter your password\n" +
   "when prompted:")
 log.Printf("\n\tmysql -h %v -P %v -u %v -p\n",
  *instance.Endpoint.Address, instance.Endpoint.Port, *instance.MasterUsername)
 log.Println("For more information, see the User Guide for RDS:\n" +
  "\thttps://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
CHAP_GettingStarted.CreatingConnecting.MySQL.html#CHAP_GettingStarted.Connecting.MySQL")
 log.Println(strings.Repeat("-", 88))
}

// CreateSnapshot shows how to create a DB instance snapshot and wait until it's
 available.
func (scenario GetStartedInstances) CreateSnapshot(ctx context.Context, instance
 *types.DBInstance) {
 if scenario.questioner.AskBool(
  "Do you want to create a snapshot of your DB instance (y/n)? ", "y") {
  snapshotId := fmt.Sprintf("%v-%v", *instance.DBInstanceIdentifier,
 scenario.helper.UniqueId())
  log.Printf("Creating a snapshot named %v. This typically takes a few minutes.
\n", snapshotId)
  snapshot, err := scenario.instances.CreateSnapshot(ctx,
 *instance.DBInstanceIdentifier, snapshotId)
  if err != nil {
   panic(err)
  }
  for *snapshot.Status != "available" {
   scenario.helper.Pause(30)
   snapshot, err = scenario.instances.GetSnapshot(ctx, snapshotId)
   if err != nil {
    panic(err)
   }
  }
  log.Println("Snapshot data:")
  log.Printf("\tDBSnapshotIdentifier: %v\n", *snapshot.DBSnapshotIdentifier)
  log.Printf("\tARN: %v\n", *snapshot.DBSnapshotArn)
  log.Printf("\tStatus: %v\n", *snapshot.Status)
  log.Printf("\tEngine: %v\n", *snapshot.Engine)
```

```
  log.Printf("\tEngine version: %v\n", *snapshot.EngineVersion)
  log.Printf("\tDBInstanceIdentifier: %v\n", *snapshot.DBInstanceIdentifier)
  log.Printf("\tSnapshotCreateTime: %v\n", *snapshot.SnapshotCreateTime)
  log.Println(strings.Repeat("-", 88))
 }
}

// Cleanup shows how to clean up a DB instance and DB parameter group.
// Before the DB parameter group can be deleted, all associated DB instances must
 first be deleted.
func (scenario GetStartedInstances) Cleanup(
 ctx context.Context, instance *types.DBInstance, parameterGroup
 *types.DBParameterGroup) {

 if scenario.questioner.AskBool(
  "\nDo you want to delete the database instance and parameter group (y/n)? ",
 "y") {
  log.Printf("Deleting database instance %v.\n", *instance.DBInstanceIdentifier)
  err := scenario.instances.DeleteInstance(ctx, *instance.DBInstanceIdentifier)
  if err != nil {
   panic(err)
  }
  log.Println(
   "Waiting for the DB instance to delete. This typically takes several
 minutes.")
  for instance != nil {
   scenario.helper.Pause(30)
   instance, err = scenario.instances.GetInstance(ctx,
 *instance.DBInstanceIdentifier)
   if err != nil {
    panic(err)
   }
  }
  log.Printf("Deleting parameter group %v.",
 *parameterGroup.DBParameterGroupName)
  err = scenario.instances.DeleteParameterGroup(ctx,
 *parameterGroup.DBParameterGroupName)
  if err != nil {
   panic(err)
  }
 }
}

// IScenarioHelper abstracts the function from a scenario so that it
```

```
// can be mocked for unit testing.
type IScenarioHelper interface {
 Pause(secs int)
 UniqueId() string
}
type ScenarioHelper struct{}

// Pause waits for the specified number of seconds.
func (helper ScenarioHelper) Pause(secs int) {
 time.Sleep(time.Duration(secs) * time.Second)
}

// UniqueId returns a new UUID.
func (helper ScenarioHelper) UniqueId() string {
 return uuid.New().String()
}
```

Définissez des fonctions appelées par le scénario pour gérer des actions Amazon RDS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}


// GetParameterGroup gets a DB parameter group by name.
func (instances *DbInstances) GetParameterGroup(ctx context.Context,
 parameterGroupName string) (
 *types.DBParameterGroup, error) {
 output, err := instances.RdsClient.DescribeDBParameterGroups(
  ctx, &rds.DescribeDBParameterGroupsInput{
```

```
   DBParameterGroupName: aws.String(parameterGroupName),
  })
 if err != nil {
  var notFoundError *types.DBParameterGroupNotFoundFault
  if errors.As(err, &notFoundError) {
   log.Printf("Parameter group %v does not exist.\n", parameterGroupName)
   err = nil
  } else {
   log.Printf("Error getting parameter group %v: %v\n", parameterGroupName, err)
  }
  return nil, err
 } else {
  return &output.DBParameterGroups[0], err
 }
}



// CreateParameterGroup creates a DB parameter group that is based on the
 specified
// parameter group family.
func (instances *DbInstances) CreateParameterGroup(
 ctx context.Context, parameterGroupName string, parameterGroupFamily string,
 description string) (
 *types.DBParameterGroup, error) {

 output, err := instances.RdsClient.CreateDBParameterGroup(ctx,
  &rds.CreateDBParameterGroupInput{
   DBParameterGroupName:   aws.String(parameterGroupName),
   DBParameterGroupFamily: aws.String(parameterGroupFamily),
   Description:            aws.String(description),
  })
 if err != nil {
  log.Printf("Couldn't create parameter group %v: %v\n", parameterGroupName, err)
  return nil, err
 } else {
  return output.DBParameterGroup, err
 }
}



// DeleteParameterGroup deletes the named DB parameter group.
```

```
func (instances *DbInstances) DeleteParameterGroup(ctx context.Context,
 parameterGroupName string) error {
 _, err := instances.RdsClient.DeleteDBParameterGroup(ctx,
  &rds.DeleteDBParameterGroupInput{
   DBParameterGroupName: aws.String(parameterGroupName),
  })
 if err != nil {
  log.Printf("Couldn't delete parameter group %v: %v\n", parameterGroupName, err)
  return err
 } else {
  return nil
 }
}



// GetParameters gets the parameters that are contained in a DB parameter group.
func (instances *DbInstances) GetParameters(ctx context.Context,
 parameterGroupName string, source string) (
 []types.Parameter, error) {

 var output *rds.DescribeDBParametersOutput
 var params []types.Parameter
 var err error
 parameterPaginator := rds.NewDescribeDBParametersPaginator(instances.RdsClient,
  &rds.DescribeDBParametersInput{
   DBParameterGroupName: aws.String(parameterGroupName),
   Source:               aws.String(source),
  })
 for parameterPaginator.HasMorePages() {
  output, err = parameterPaginator.NextPage(ctx)
  if err != nil {
   log.Printf("Couldn't get parameters for %v: %v\n", parameterGroupName, err)
   break
  } else {
   params = append(params, output.Parameters...)
  }
 }
 return params, err
}



// UpdateParameters updates parameters in a named DB parameter group.
```

```
func (instances *DbInstances) UpdateParameters(ctx context.Context,
 parameterGroupName string, params []types.Parameter) error {
 _, err := instances.RdsClient.ModifyDBParameterGroup(ctx,
  &rds.ModifyDBParameterGroupInput{
   DBParameterGroupName: aws.String(parameterGroupName),
   Parameters:           params,
  })
 if err != nil {
  log.Printf("Couldn't update parameters in %v: %v\n", parameterGroupName, err)
  return err
 } else {
  return nil
 }
}



// CreateSnapshot creates a snapshot of a DB instance.
func (instances *DbInstances) CreateSnapshot(ctx context.Context, instanceName
 string, snapshotName string) (
 *types.DBSnapshot, error) {
 output, err := instances.RdsClient.CreateDBSnapshot(ctx,
 &rds.CreateDBSnapshotInput{
  DBInstanceIdentifier: aws.String(instanceName),
  DBSnapshotIdentifier: aws.String(snapshotName),
 })
 if err != nil {
  log.Printf("Couldn't create snapshot %v: %v\n", snapshotName, err)
  return nil, err
 } else {
  return output.DBSnapshot, nil
 }
}



// GetSnapshot gets a DB instance snapshot.
func (instances *DbInstances) GetSnapshot(ctx context.Context, snapshotName
 string) (*types.DBSnapshot, error) {
 output, err := instances.RdsClient.DescribeDBSnapshots(ctx,
  &rds.DescribeDBSnapshotsInput{
   DBSnapshotIdentifier: aws.String(snapshotName),
  })
 if err != nil {
```

```
  log.Printf("Couldn't get snapshot %v: %v\n", snapshotName, err)
  return nil, err
 } else {
  return &output.DBSnapshots[0], nil
 }
}



// CreateInstance creates a DB instance.
func (instances *DbInstances) CreateInstance(ctx context.Context, instanceName
 string, dbName string,
 dbEngine string, dbEngineVersion string, parameterGroupName string,
 dbInstanceClass string,
 storageType string, allocatedStorage int32, adminName string, adminPassword
 string) (
 *types.DBInstance, error) {
 output, err := instances.RdsClient.CreateDBInstance(ctx,
 &rds.CreateDBInstanceInput{
  DBInstanceIdentifier: aws.String(instanceName),
  DBName:               aws.String(dbName),
  DBParameterGroupName: aws.String(parameterGroupName),
  Engine:               aws.String(dbEngine),
  EngineVersion:        aws.String(dbEngineVersion),
  DBInstanceClass:      aws.String(dbInstanceClass),
  StorageType:          aws.String(storageType),
  AllocatedStorage:     aws.Int32(allocatedStorage),
  MasterUsername:       aws.String(adminName),
  MasterUserPassword:   aws.String(adminPassword),
 })
 if err != nil {
  log.Printf("Couldn't create instance %v: %v\n", instanceName, err)
  return nil, err
 } else {
  return output.DBInstance, nil
 }
}



// GetInstance gets data about a DB instance.
func (instances *DbInstances) GetInstance(ctx context.Context, instanceName
 string) (
 *types.DBInstance, error) {
```

```
 output, err := instances.RdsClient.DescribeDBInstances(ctx,
  &rds.DescribeDBInstancesInput{
   DBInstanceIdentifier: aws.String(instanceName),
  })
 if err != nil {
  var notFoundError *types.DBInstanceNotFoundFault
  if errors.As(err, &notFoundError) {
   log.Printf("DB instance %v does not exist.\n", instanceName)
   err = nil
  } else {
   log.Printf("Couldn't get instance %v: %v\n", instanceName, err)
  }
  return nil, err
 } else {
  return &output.DBInstances[0], nil
 }
}



// DeleteInstance deletes a DB instance.
func (instances *DbInstances) DeleteInstance(ctx context.Context, instanceName
 string) error {
 _, err := instances.RdsClient.DeleteDBInstance(ctx, &rds.DeleteDBInstanceInput{
  DBInstanceIdentifier:   aws.String(instanceName),
  SkipFinalSnapshot:      aws.Bool(true),
  DeleteAutomatedBackups: aws.Bool(true),
 })
 if err != nil {
  log.Printf("Couldn't delete instance %v: %v\n", instanceName, err)
  return err
 } else {
  return nil
 }
}



// GetEngineVersions gets database engine versions that are available for the
 specified engine
// and parameter group family.
func (instances *DbInstances) GetEngineVersions(ctx context.Context, engine
 string, parameterGroupFamily string) (
 []types.DBEngineVersion, error) {
```

```
 output, err := instances.RdsClient.DescribeDBEngineVersions(ctx,
  &rds.DescribeDBEngineVersionsInput{
   Engine:                 aws.String(engine),
   DBParameterGroupFamily: aws.String(parameterGroupFamily),
  })
 if err != nil {
  log.Printf("Couldn't get engine versions for %v: %v\n", engine, err)
  return nil, err
 } else {
  return output.DBEngineVersions, nil
 }
}



// GetOrderableInstances uses a paginator to get DB instance options that can be
 used to create DB instances that are
// compatible with a set of specifications.
func (instances *DbInstances) GetOrderableInstances(ctx context.Context, engine
 string, engineVersion string) (
 []types.OrderableDBInstanceOption, error) {

 var output *rds.DescribeOrderableDBInstanceOptionsOutput
 var instanceOptions []types.OrderableDBInstanceOption
 var err error
 orderablePaginator :=
 rds.NewDescribeOrderableDBInstanceOptionsPaginator(instances.RdsClient,
  &rds.DescribeOrderableDBInstanceOptionsInput{
   Engine:        aws.String(engine),
   EngineVersion: aws.String(engineVersion),
  })
 for orderablePaginator.HasMorePages() {
  output, err = orderablePaginator.NextPage(ctx)
  if err != nil {
   log.Printf("Couldn't get orderable DB instance options: %v\n", err)
   break
  } else {
   instanceOptions = append(instanceOptions,
 output.OrderableDBInstanceOptions...)
  }
 }
 return instanceOptions, err
}
```

- Pour plus d'informations sur l'API consultez les rubriques suivantes dans la référence de l'API AWS SDK pour Go .
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Exécutez plusieurs opérations.

```
import com.google.gson.Gson;
import
 software.amazon.awssdk.auth.credentials.EnvironmentVariableCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.CreateDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.CreateDbInstanceResponse;
```

```
import software.amazon.awssdk.services.rds.model.CreateDbParameterGroupResponse;
import software.amazon.awssdk.services.rds.model.CreateDbSnapshotRequest;
import software.amazon.awssdk.services.rds.model.CreateDbSnapshotResponse;
import software.amazon.awssdk.services.rds.model.DBEngineVersion;
import software.amazon.awssdk.services.rds.model.DBInstance;
import software.amazon.awssdk.services.rds.model.DBParameterGroup;
import software.amazon.awssdk.services.rds.model.DBSnapshot;
import software.amazon.awssdk.services.rds.model.DeleteDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.DeleteDbInstanceResponse;
import software.amazon.awssdk.services.rds.model.DescribeDbEngineVersionsRequest;
import
 software.amazon.awssdk.services.rds.model.DescribeDbEngineVersionsResponse;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesRequest;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesResponse;
import
 software.amazon.awssdk.services.rds.model.DescribeDbParameterGroupsResponse;
import software.amazon.awssdk.services.rds.model.DescribeDbParametersResponse;
import software.amazon.awssdk.services.rds.model.DescribeDbSnapshotsRequest;
import software.amazon.awssdk.services.rds.model.DescribeDbSnapshotsResponse;
import
 software.amazon.awssdk.services.rds.model.DescribeOrderableDbInstanceOptionsResponse;
import software.amazon.awssdk.services.rds.model.ModifyDbParameterGroupResponse;
import software.amazon.awssdk.services.rds.model.OrderableDBInstanceOption;
import software.amazon.awssdk.services.rds.model.Parameter;
import software.amazon.awssdk.services.rds.model.RdsException;
import software.amazon.awssdk.services.rds.model.CreateDbParameterGroupRequest;
import
 software.amazon.awssdk.services.rds.model.DescribeDbParameterGroupsRequest;
import software.amazon.awssdk.services.rds.model.DescribeDbParametersRequest;
import software.amazon.awssdk.services.rds.model.ModifyDbParameterGroupRequest;
import
 software.amazon.awssdk.services.rds.model.DescribeOrderableDbInstanceOptionsRequest;
import software.amazon.awssdk.services.rds.model.DeleteDbParameterGroupRequest;
import software.amazon.awssdk.services.secretsmanager.SecretsManagerClient;
import
 software.amazon.awssdk.services.secretsmanager.model.GetSecretValueRequest;
import
 software.amazon.awssdk.services.secretsmanager.model.GetSecretValueResponse;
import java.util.ArrayList;
import java.util.List;

/**
 * Before running this Java (v2) code example, set up your development
 * environment, including your credentials.
```

```
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * This example requires an AWS Secrets Manager secret that contains the
 * database credentials. If you do not create a
 * secret, this example will not work. For details, see:
 *
 * https://docs.aws.amazon.com/secretsmanager/latest/userguide/integrating_how-
services-use-secrets_RS.html
 *
 * This Java example performs these tasks:
 *
 * 1. Returns a list of the available DB engines.
 * 2. Selects an engine family and create a custom DB parameter group.
 * 3. Gets the parameter groups.
 * 4. Gets parameters in the group.
 * 5. Modifies the auto_increment_offset parameter.
 * 6. Gets and displays the updated parameters.
 * 7. Gets a list of allowed engine versions.
 * 8. Gets a list of micro instance classes available for the selected engine.
 * 9. Creates an RDS database instance that contains a MySql database and uses
 * the parameter group.
 * 10. Waits for the DB instance to be ready and prints out the connection
 * endpoint value.
 * 11. Creates a snapshot of the DB instance.
 * 12. Waits for an RDS DB snapshot to be ready.
 * 13. Deletes the RDS DB instance.
 * 14. Deletes the parameter group.
 */
public class RDSScenario {
    public static long sleepTime = 20;
    public static final String DASHES = new String(new char[80]).replace("\0",
 "-");

    public static void main(String[] args) throws InterruptedException {
        final String usage = """

                Usage:
                    <dbGroupName> <dbParameterGroupFamily> <dbInstanceIdentifier>
 <dbName> <dbSnapshotIdentifier> <secretName>
```

```
                Where:
                    dbGroupName - The database group name.\s
                    dbParameterGroupFamily - The database parameter group name
(for example, mysql8.0).
                    dbInstanceIdentifier - The database instance identifier\s
                    dbName - The database name.\s
                    dbSnapshotIdentifier - The snapshot identifier.\s
                    secretName - The name of the AWS Secrets Manager secret that
contains the database credentials"
                """;

        if (args.length != 6) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbGroupName = args[0];
        String dbParameterGroupFamily = args[1];
        String dbInstanceIdentifier = args[2];
        String dbName = args[3];
        String dbSnapshotIdentifier = args[4];
        String secretName = args[5];

        Gson gson = new Gson();
        User user = gson.fromJson(String.valueOf(getSecretValues(secretName)),
User.class);
        String masterUsername = user.getUsername();
        String masterUserPassword = user.getPassword();

        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();
        System.out.println(DASHES);
        System.out.println("Welcome to the Amazon RDS example scenario.");
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("1. Return a list of the available DB engines");
        describeDBEngines(rdsClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("2. Create a custom parameter group");
```

```
        createDBParameterGroup(rdsClient, dbGroupName, dbParameterGroupFamily);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("3. Get the parameter group");
        describeDbParameterGroups(rdsClient, dbGroupName);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("4. Get the parameters in the group");
        describeDbParameters(rdsClient, dbGroupName, 0);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("5. Modify the auto_increment_offset parameter");
        modifyDBParas(rdsClient, dbGroupName);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("6. Display the updated value");
        describeDbParameters(rdsClient, dbGroupName, -1);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("7. Get a list of allowed engine versions");
        getAllowedEngines(rdsClient, dbParameterGroupFamily);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("8. Get a list of micro instance classes available for
the selected engine");
        getMicroInstances(rdsClient);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println(
                "9. Create an RDS database instance that contains a MySql
database and uses the parameter group");
        String dbARN = createDatabaseInstance(rdsClient, dbGroupName,
dbInstanceIdentifier, dbName, masterUsername,
                masterUserPassword);
        System.out.println("The ARN of the new database is " + dbARN);
        System.out.println(DASHES);
```

```
        System.out.println(DASHES);
        System.out.println("10. Wait for DB instance to be ready");
        waitForInstanceReady(rdsClient, dbInstanceIdentifier);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("11. Create a snapshot of the DB instance");
        createSnapshot(rdsClient, dbInstanceIdentifier, dbSnapshotIdentifier);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("12. Wait for DB snapshot to be ready");
        waitForSnapshotReady(rdsClient, dbInstanceIdentifier,
dbSnapshotIdentifier);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("13. Delete the DB instance");
        deleteDatabaseInstance(rdsClient, dbInstanceIdentifier);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("14. Delete the parameter group");
        deleteParaGroup(rdsClient, dbGroupName, dbARN);
        System.out.println(DASHES);

        System.out.println(DASHES);
        System.out.println("The Scenario has successfully completed.");
        System.out.println(DASHES);

        rdsClient.close();
    }

    private static SecretsManagerClient getSecretClient() {
        Region region = Region.US_WEST_2;
        return SecretsManagerClient.builder()
            .region(region)
            .build();
    }

    public static String getSecretValues(String secretName) {
        SecretsManagerClient secretClient = getSecretClient();
        GetSecretValueRequest valueRequest = GetSecretValueRequest.builder()
                .secretId(secretName)
```

```
                .build();

        GetSecretValueResponse valueResponse =
secretClient.getSecretValue(valueRequest);
        return valueResponse.secretString();
    }

    // Delete the parameter group after database has been deleted.
    // An exception is thrown if you attempt to delete the para group while
database
    // exists.
    public static void deleteParaGroup(RdsClient rdsClient, String dbGroupName,
String dbARN)
            throws InterruptedException {
        try {
            boolean isDataDel = false;
            boolean didFind;
            String instanceARN;

            // Make sure that the database has been deleted.
            while (!isDataDel) {
                DescribeDbInstancesResponse response =
rdsClient.describeDBInstances();
                List<DBInstance> instanceList = response.dbInstances();
                int listSize = instanceList.size();
                didFind = false;
                int index = 1;
                for (DBInstance instance : instanceList) {
                    instanceARN = instance.dbInstanceArn();
                    if (instanceARN.compareTo(dbARN) == 0) {
                        System.out.println(dbARN + " still exists");
                        didFind = true;
                    }
                    if ((index == listSize) && (!didFind)) {
                        // Went through the entire list and did not find the
database ARN.
                        isDataDel = true;
                    }
                    Thread.sleep(sleepTime * 1000);
                    index++;
                }
            }

            // Delete the para group.
```

```
            DeleteDbParameterGroupRequest parameterGroupRequest =
DeleteDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .build();

            rdsClient.deleteDBParameterGroup(parameterGroupRequest);
            System.out.println(dbGroupName + " was deleted.");

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Delete the DB instance.
    public static void deleteDatabaseInstance(RdsClient rdsClient, String
dbInstanceIdentifier) {
        try {
            DeleteDbInstanceRequest deleteDbInstanceRequest =
DeleteDbInstanceRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .deleteAutomatedBackups(true)
                    .skipFinalSnapshot(true)
                    .build();

            DeleteDbInstanceResponse response =
rdsClient.deleteDBInstance(deleteDbInstanceRequest);
            System.out.print("The status of the database is " +
response.dbInstance().dbInstanceStatus());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Waits until the snapshot instance is available.
    public static void waitForSnapshotReady(RdsClient rdsClient, String
dbInstanceIdentifier,
            String dbSnapshotIdentifier) {
        try {
            boolean snapshotReady = false;
            String snapshotReadyStr;
            System.out.println("Waiting for the snapshot to become available.");
```

```
            DescribeDbSnapshotsRequest snapshotsRequest =
DescribeDbSnapshotsRequest.builder()
                    .dbSnapshotIdentifier(dbSnapshotIdentifier)
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .build();

            while (!snapshotReady) {
                DescribeDbSnapshotsResponse response =
rdsClient.describeDBSnapshots(snapshotsRequest);
                List<DBSnapshot> snapshotList = response.dbSnapshots();
                for (DBSnapshot snapshot : snapshotList) {
                    snapshotReadyStr = snapshot.status();
                    if (snapshotReadyStr.contains("available")) {
                        snapshotReady = true;
                    } else {
                        System.out.print(".");
                        Thread.sleep(sleepTime * 1000);
                    }
                }
            }

            System.out.println("The Snapshot is available!");
        } catch (RdsException | InterruptedException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Create an Amazon RDS snapshot.
    public static void createSnapshot(RdsClient rdsClient, String
dbInstanceIdentifier, String dbSnapshotIdentifier) {
        try {
            CreateDbSnapshotRequest snapshotRequest =
CreateDbSnapshotRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .dbSnapshotIdentifier(dbSnapshotIdentifier)
                    .build();

            CreateDbSnapshotResponse response =
rdsClient.createDBSnapshot(snapshotRequest);
            System.out.println("The Snapshot id is " +
response.dbSnapshot().dbiResourceId());
```

```
        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Waits until the database instance is available.
    public static void waitForInstanceReady(RdsClient rdsClient, String
dbInstanceIdentifier) {
        boolean instanceReady = false;
        String instanceReadyStr;
        System.out.println("Waiting for instance to become available.");
        try {
            DescribeDbInstancesRequest instanceRequest =
DescribeDbInstancesRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .build();

            String endpoint = "";
            while (!instanceReady) {
                DescribeDbInstancesResponse response =
rdsClient.describeDBInstances(instanceRequest);
                List<DBInstance> instanceList = response.dbInstances();
                for (DBInstance instance : instanceList) {
                    instanceReadyStr = instance.dbInstanceStatus();
                    if (instanceReadyStr.contains("available")) {
                        endpoint = instance.endpoint().address();
                        instanceReady = true;
                    } else {
                        System.out.print(".");
                        Thread.sleep(sleepTime * 1000);
                    }
                }
            }
            System.out.println("Database instance is available! The connection
endpoint is " + endpoint);

        } catch (RdsException | InterruptedException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }

    // Create a database instance and return the ARN of the database.
```

```
   public static String createDatabaseInstance(RdsClient rdsClient,
           String dbGroupName,
           String dbInstanceIdentifier,
           String dbName,
           String userName,
           String userPassword) {

       try {
           CreateDbInstanceRequest instanceRequest =
CreateDbInstanceRequest.builder()
               .dbInstanceIdentifier(dbInstanceIdentifier)
               .allocatedStorage(100)
               .dbName(dbName)
               .engine("mysql")
               .dbInstanceClass("db.t3.medium") // Updated to a supported class
               .engineVersion("8.0.32")         // Updated to a supported
version
               .storageType("gp2")             // Changed to General Purpose SSD
(gp2)
               .masterUsername(userName)
               .masterUserPassword(userPassword)
               .build();

           CreateDbInstanceResponse response =
rdsClient.createDBInstance(instanceRequest);
           System.out.print("The status is " +
response.dbInstance().dbInstanceStatus());
           return response.dbInstance().dbInstanceArn();

       } catch (RdsException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }

       return "";
   }

   // Get a list of micro instances.
   public static void getMicroInstances(RdsClient rdsClient) {
       try {
           DescribeOrderableDbInstanceOptionsRequest dbInstanceOptionsRequest =
DescribeOrderableDbInstanceOptionsRequest
                   .builder()
                   .engine("mysql")
```

```
                    .build();

            DescribeOrderableDbInstanceOptionsResponse response = rdsClient

.describeOrderableDBInstanceOptions(dbInstanceOptionsRequest);
            List<OrderableDBInstanceOption> orderableDBInstances =
response.orderableDBInstanceOptions();
            for (OrderableDBInstanceOption dbInstanceOption :
orderableDBInstances) {
                System.out.println("The engine version is " +
dbInstanceOption.engineVersion());
                System.out.println("The engine description is " +
dbInstanceOption.engine());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Get a list of allowed engine versions.
    public static void getAllowedEngines(RdsClient rdsClient, String
dbParameterGroupFamily) {
        try {
            DescribeDbEngineVersionsRequest versionsRequest =
DescribeDbEngineVersionsRequest.builder()
                    .dbParameterGroupFamily(dbParameterGroupFamily)
                    .engine("mysql")
                    .build();

            DescribeDbEngineVersionsResponse response =
rdsClient.describeDBEngineVersions(versionsRequest);
            List<DBEngineVersion> dbEngines = response.dbEngineVersions();
            for (DBEngineVersion dbEngine : dbEngines) {
                System.out.println("The engine version is " +
dbEngine.engineVersion());
                System.out.println("The engine description is " +
dbEngine.dbEngineDescription());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
```

```
        }
    }

    // Modify auto_increment_offset and auto_increment_increment parameters.
    public static void modifyDBParas(RdsClient rdsClient, String dbGroupName) {
        try {
            Parameter parameter1 = Parameter.builder()
                    .parameterName("auto_increment_offset")
                    .applyMethod("immediate")
                    .parameterValue("5")
                    .build();

            List<Parameter> paraList = new ArrayList<>();
            paraList.add(parameter1);
            ModifyDbParameterGroupRequest groupRequest =
ModifyDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .parameters(paraList)
                    .build();

            ModifyDbParameterGroupResponse response =
rdsClient.modifyDBParameterGroup(groupRequest);
            System.out.println("The parameter group " +
response.dbParameterGroupName() + " was successfully modified");

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Retrieve parameters in the group.
    public static void describeDbParameters(RdsClient rdsClient, String
dbGroupName, int flag) {
        try {
            DescribeDbParametersRequest dbParameterGroupsRequest;
            if (flag == 0) {
                dbParameterGroupsRequest = DescribeDbParametersRequest.builder()
                        .dbParameterGroupName(dbGroupName)
                        .build();
            } else {
                dbParameterGroupsRequest = DescribeDbParametersRequest.builder()
                        .dbParameterGroupName(dbGroupName)
                        .source("user")
```

```
                        .build();
            }

            DescribeDbParametersResponse response =
rdsClient.describeDBParameters(dbParameterGroupsRequest);
            List<Parameter> dbParameters = response.parameters();
            String paraName;
            for (Parameter para : dbParameters) {
                // Only print out information about either auto_increment_offset
or
                // auto_increment_increment.
                paraName = para.parameterName();
                if ((paraName.compareTo("auto_increment_offset") == 0)
                        || (paraName.compareTo("auto_increment_increment ") ==
0)) {
                    System.out.println("*** The parameter name is  " + paraName);
                    System.out.println("*** The parameter value is  " +
para.parameterValue());
                    System.out.println("*** The parameter data type is " +
para.dataType());
                    System.out.println("*** The parameter description is " +
para.description());
                    System.out.println("*** The parameter allowed values  is " +
para.allowedValues());
                }
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static void describeDbParameterGroups(RdsClient rdsClient, String
dbGroupName) {
        try {
            DescribeDbParameterGroupsRequest groupsRequest =
DescribeDbParameterGroupsRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .maxRecords(20)
                    .build();

            DescribeDbParameterGroupsResponse response =
rdsClient.describeDBParameterGroups(groupsRequest);
```

```
            List<DBParameterGroup> groups = response.dbParameterGroups();
            for (DBParameterGroup group : groups) {
                System.out.println("The group name is " +
group.dbParameterGroupName());
                System.out.println("The group description is " +
group.description());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static void createDBParameterGroup(RdsClient rdsClient, String
dbGroupName, String dbParameterGroupFamily) {
        try {
            CreateDbParameterGroupRequest groupRequest =
CreateDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .dbParameterGroupFamily(dbParameterGroupFamily)
                    .description("Created by using the AWS SDK for Java")
                    .build();

            CreateDbParameterGroupResponse response =
rdsClient.createDBParameterGroup(groupRequest);
            System.out.println("The group name is " +
response.dbParameterGroup().dbParameterGroupName());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    public static void describeDBEngines(RdsClient rdsClient) {
        try {
            DescribeDbEngineVersionsRequest engineVersionsRequest =
DescribeDbEngineVersionsRequest.builder()
                    .defaultOnly(true)
                    .engine("mysql")
                    .maxRecords(20)
                    .build();
```

```
            DescribeDbEngineVersionsResponse response =
 rdsClient.describeDBEngineVersions(engineVersionsRequest);
            List<DBEngineVersion> engines = response.dbEngineVersions();

            // Get all DBEngineVersion objects.
            for (DBEngineVersion engineOb : engines) {
                System.out.println("The name of the DB parameter group family for
 the database engine is "
                        + engineOb.dbParameterGroupFamily());
                System.out.println("The name of the database engine " +
 engineOb.engine());
                System.out.println("The version number of the database engine " +
 engineOb.engineVersion());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus d'informations sur l'API consultez les rubriques suivantes dans la référence de l'API AWS SDK for Java 2.x .
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
/**
Before running this code example, set up your development environment, including
 your credentials.

For more information, see the following documentation topic:

https://docs.aws.amazon.com/sdk-for-kotlin/latest/developer-guide/setup.html

This example requires an AWS Secrets Manager secret that contains the database
 credentials. If you do not create a
secret, this example will not work. For more details, see:

https://docs.aws.amazon.com/secretsmanager/latest/userguide/integrating_how-
services-use-secrets_RS.html

This example performs the following tasks:

1. Returns a list of the available DB engines by invoking the
 DescribeDbEngineVersions method.
2. Selects an engine family and create a custom DB parameter group by invoking
 the createDBParameterGroup method.
3. Gets the parameter groups by invoking the DescribeDbParameterGroups method.
4. Gets parameters in the group by invoking the DescribeDbParameters method.
5. Modifies both the auto_increment_offset and auto_increment_increment
 parameters by invoking the modifyDbParameterGroup method.
6. Gets and displays the updated parameters.
7. Gets a list of allowed engine versions by invoking the
 describeDbEngineVersions method.
8. Gets a list of micro instance classes available for the selected engine.
9. Creates an Amazon Relational Database Service (Amazon RDS) database instance
 that contains a MySQL database and uses the parameter group.
```

```
10. Waits for DB instance to be ready and prints out the connection endpoint
 value.
11. Creates a snapshot of the DB instance.
12. Waits for the DB snapshot to be ready.
13. Deletes the DB instance.
14. Deletes the parameter group.
 */

var sleepTime: Long = 20

suspend fun main(args: Array<String>) {
    val usage = """
        Usage:
            <dbGroupName> <dbParameterGroupFamily> <dbInstanceIdentifier>
 <dbName> <dbSnapshotIdentifier><secretName>

        Where:
            dbGroupName - The database group name.
            dbParameterGroupFamily - The database parameter group name.
            dbInstanceIdentifier - The database instance identifier.
            dbName -  The database name.
            dbSnapshotIdentifier - The snapshot identifier.
            secretName - The name of the AWS Secrets Manager secret that contains
 the database credentials.
    """

    if (args.size != 6) {
        println(usage)
        exitProcess(1)
    }

    val dbGroupName = args[0]
    val dbParameterGroupFamily = args[1]
    val dbInstanceIdentifier = args[2]
    val dbName = args[3]
    val dbSnapshotIdentifier = args[4]
    val secretName = args[5]

    val gson = Gson()
    val user = gson.fromJson(getSecretValues(secretName).toString(),
 User::class.java)
    val username = user.username
    val userPassword = user.password
```

```
    println("1. Return a list of the available DB engines")
    describeDBEngines()

    println("2. Create a custom parameter group")
    createDBParameterGroup(dbGroupName, dbParameterGroupFamily)

    println("3. Get the parameter groups")
    describeDbParameterGroups(dbGroupName)

    println("4. Get the parameters in the group")
    describeDbParameters(dbGroupName, 0)

    println("5. Modify the auto_increment_offset parameter")
    modifyDBParas(dbGroupName)

    println("6. Display the updated value")
    describeDbParameters(dbGroupName, -1)

    println("7. Get a list of allowed engine versions")
    getAllowedEngines(dbParameterGroupFamily)

    println("8. Get a list of micro instance classes available for the selected
engine")
    getMicroInstances()

    println("9. Create an RDS database instance that contains a MySql database
and uses the parameter group")
    val dbARN = createDatabaseInstance(dbGroupName, dbInstanceIdentifier, dbName,
username, userPassword)
    println("The ARN of the new database is $dbARN")

    println("10. Wait for DB instance to be ready")
    waitForDbInstanceReady(dbInstanceIdentifier)

    println("11. Create a snapshot of the DB instance")
    createDbSnapshot(dbInstanceIdentifier, dbSnapshotIdentifier)

    println("12. Wait for DB snapshot to be ready")
    waitForSnapshotReady(dbInstanceIdentifier, dbSnapshotIdentifier)

    println("13. Delete the DB instance")
    deleteDbInstance(dbInstanceIdentifier)

    println("14. Delete the parameter group")
```

```
    if (dbARN != null) {
        deleteParaGroup(dbGroupName, dbARN)
    }

    println("The Scenario has successfully completed.")
}

suspend fun deleteParaGroup(
    dbGroupName: String,
    dbARN: String,
) {
    var isDataDel = false
    var didFind: Boolean
    var instanceARN: String

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        // Make sure that the database has been deleted.
        while (!isDataDel) {
            val response = rdsClient.describeDbInstances()
            val instanceList = response.dbInstances
            val listSize = instanceList?.size
            isDataDel = false // Reset this value.
            didFind = false // Reset this value.
            var index = 1
            if (instanceList != null) {
                for (instance in instanceList) {
                    instanceARN = instance.dbInstanceArn.toString()
                    if (instanceARN.compareTo(dbARN) == 0) {
                        println("$dbARN still exists")
                        didFind = true
                    }
                    if (index == listSize && !didFind) {
                        // Went through the entire list and did not find the
 database name.
                        isDataDel = true
                    }
                    index++
                }
            }
        }

        // Delete the para group.
        val parameterGroupRequest =
            DeleteDbParameterGroupRequest {
```

```
                dbParameterGroupName = dbGroupName
            }
        rdsClient.deleteDbParameterGroup(parameterGroupRequest)
        println("$dbGroupName was deleted.")
    }
}

suspend fun deleteDbInstance(dbInstanceIdentifierVal: String) {
    val deleteDbInstanceRequest =
        DeleteDbInstanceRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            deleteAutomatedBackups = true
            skipFinalSnapshot = true
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.deleteDbInstance(deleteDbInstanceRequest)
        print("The status of the database is
 ${response.dbInstance?.dbInstanceStatus}")
    }
}

// Waits until the snapshot instance is available.
suspend fun waitForSnapshotReady(
    dbInstanceIdentifierVal: String?,
    dbSnapshotIdentifierVal: String?,
) {
    var snapshotReady = false
    var snapshotReadyStr: String
    println("Waiting for the snapshot to become available.")

    val snapshotsRequest =
        DescribeDbSnapshotsRequest {
            dbSnapshotIdentifier = dbSnapshotIdentifierVal
            dbInstanceIdentifier = dbInstanceIdentifierVal
        }

    while (!snapshotReady) {
        RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
            val response = rdsClient.describeDbSnapshots(snapshotsRequest)
            val snapshotList: List<DbSnapshot>? = response.dbSnapshots
            if (snapshotList != null) {
                for (snapshot in snapshotList) {
                    snapshotReadyStr = snapshot.status.toString()
```

```
                    if (snapshotReadyStr.contains("available")) {
                        snapshotReady = true
                    } else {
                        print(".")
                        delay(sleepTime * 1000)
                    }
                }
            }
        }
    }
    println("The Snapshot is available!")
}

// Create an Amazon RDS snapshot.
suspend fun createDbSnapshot(
    dbInstanceIdentifierVal: String?,
    dbSnapshotIdentifierVal: String?,
) {
    val snapshotRequest =
        CreateDbSnapshotRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            dbSnapshotIdentifier = dbSnapshotIdentifierVal
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.createDbSnapshot(snapshotRequest)
        print("The Snapshot id is ${response.dbSnapshot?.dbiResourceId}")
    }
}

// Waits until the database instance is available.
suspend fun waitForDbInstanceReady(dbInstanceIdentifierVal: String?) {
    var instanceReady = false
    var instanceReadyStr: String
    println("Waiting for instance to become available.")

    val instanceRequest =
        DescribeDbInstancesRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
        }
    var endpoint = ""
    while (!instanceReady) {
        RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
            val response = rdsClient.describeDbInstances(instanceRequest)
```

```
            val instanceList = response.dbInstances
            if (instanceList != null) {
                for (instance in instanceList) {
                    instanceReadyStr = instance.dbInstanceStatus.toString()
                    if (instanceReadyStr.contains("available")) {
                        endpoint = instance.endpoint?.address.toString()
                        instanceReady = true
                    } else {
                        print(".")
                        delay(sleepTime * 1000)
                    }
                }
            }
        }
    }
    println("Database instance is available! The connection endpoint is
 $endpoint")
}

// Create a database instance and return the ARN of the database.
suspend fun createDatabaseInstance(
    dbGroupNameVal: String?,
    dbInstanceIdentifierVal: String?,
    dbNameVal: String?,
    masterUsernameVal: String?,
    masterUserPasswordVal: String?,
): String? {
    val instanceRequest =
        CreateDbInstanceRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            allocatedStorage = 100
            dbName = dbNameVal
            dbParameterGroupName = dbGroupNameVal
            engine = "mysql"
            dbInstanceClass = "db.t3.micro"
            engineVersion = "8.0.35"
            storageType = "gp2"
            masterUsername = masterUsernameVal
            masterUserPassword = masterUserPasswordVal
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.createDbInstance(instanceRequest)
        print("The status is ${response.dbInstance?.dbInstanceStatus}")
```

```
        return response.dbInstance?.dbInstanceArn
    }
}

// Get a list of micro instances.
suspend fun getMicroInstances() {
    val dbInstanceOptionsRequest =
        DescribeOrderableDbInstanceOptionsRequest {
            engine = "mysql"
        }
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response =
 rdsClient.describeOrderableDbInstanceOptions(dbInstanceOptionsRequest)
        val orderableDBInstances = response.orderableDbInstanceOptions
        if (orderableDBInstances != null) {
            for (dbInstanceOption in orderableDBInstances) {
                println("The engine version is
 ${dbInstanceOption.engineVersion}")
                println("The engine description is ${dbInstanceOption.engine}")
            }
        }
    }
}

// Get a list of allowed engine versions.
suspend fun getAllowedEngines(dbParameterGroupFamilyVal: String?) {
    val versionsRequest =
        DescribeDbEngineVersionsRequest {
            dbParameterGroupFamily = dbParameterGroupFamilyVal
            engine = "mysql"
        }
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.describeDbEngineVersions(versionsRequest)
        val dbEngines: List<DbEngineVersion>? = response.dbEngineVersions
        if (dbEngines != null) {
            for (dbEngine in dbEngines) {
                println("The engine version is ${dbEngine.engineVersion}")
                println("The engine description is
 ${dbEngine.dbEngineDescription}")
            }
        }
    }
}
```

```
// Modify the auto_increment_offset parameter.
suspend fun modifyDBParas(dbGroupName: String) {
    val parameter1 =
        Parameter {
            parameterName = "auto_increment_offset"
            applyMethod = ApplyMethod.Immediate
            parameterValue = "5"
        }

    val paraList: ArrayList<Parameter> = ArrayList()
    paraList.add(parameter1)
    val groupRequest =
        ModifyDbParameterGroupRequest {
            dbParameterGroupName = dbGroupName
            parameters = paraList
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.modifyDbParameterGroup(groupRequest)
        println("The parameter group ${response.dbParameterGroupName} was
 successfully modified")
    }
}

// Retrieve parameters in the group.
suspend fun describeDbParameters(
    dbGroupName: String?,
    flag: Int,
) {
    val dbParameterGroupsRequest: DescribeDbParametersRequest
    dbParameterGroupsRequest =
        if (flag == 0) {
            DescribeDbParametersRequest {
                dbParameterGroupName = dbGroupName
            }
        } else {
            DescribeDbParametersRequest {
                dbParameterGroupName = dbGroupName
                source = "user"
            }
        }
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.describeDbParameters(dbParameterGroupsRequest)
        val dbParameters: List<Parameter>? = response.parameters
```

```
        var paraName: String
        if (dbParameters != null) {
            for (para in dbParameters) {
                // Only print out information about either auto_increment_offset
 or auto_increment_increment.
                paraName = para.parameterName.toString()
                if (paraName.compareTo("auto_increment_offset") == 0 ||
 paraName.compareTo("auto_increment_increment ") == 0) {
                    println("*** The parameter name is  $paraName")
                    System.out.println("*** The parameter value is
 ${para.parameterValue}")
                    System.out.println("*** The parameter data type is
 ${para.dataType}")
                    System.out.println("*** The parameter description is
 ${para.description}")
                    System.out.println("*** The parameter allowed values  is
 ${para.allowedValues}")
                }
            }
        }
    }
}

suspend fun describeDbParameterGroups(dbGroupName: String?) {
    val groupsRequest =
        DescribeDbParameterGroupsRequest {
            dbParameterGroupName = dbGroupName
            maxRecords = 20
        }
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.describeDbParameterGroups(groupsRequest)
        val groups = response.dbParameterGroups
        if (groups != null) {
            for (group in groups) {
                println("The group name is ${group.dbParameterGroupName}")
                println("The group description is ${group.description}")
            }
        }
    }
}

// Create a parameter group.
suspend fun createDBParameterGroup(
    dbGroupName: String?,
```

```
    dbParameterGroupFamilyVal: String?,
) {
    val groupRequest =
        CreateDbParameterGroupRequest {
            dbParameterGroupName = dbGroupName
            dbParameterGroupFamily = dbParameterGroupFamilyVal
            description = "Created by using the AWS SDK for Kotlin"
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.createDbParameterGroup(groupRequest)
        println("The group name is
 ${response.dbParameterGroup?.dbParameterGroupName}")
    }
}

// Returns a list of the available DB engines.
suspend fun describeDBEngines() {
    val engineVersionsRequest =
        DescribeDbEngineVersionsRequest {
            defaultOnly = true
            engine = "mysql"
            maxRecords = 20
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.describeDbEngineVersions(engineVersionsRequest)
        val engines: List<DbEngineVersion>? = response.dbEngineVersions

        // Get all DbEngineVersion objects.
        if (engines != null) {
            for (engineOb in engines) {
                println("The name of the DB parameter group family for the
 database engine is ${engineOb.dbParameterGroupFamily}.")
                println("The name of the database engine ${engineOb.engine}.")
                println("The version number of the database engine
 ${engineOb.engineVersion}")
            }
        }
    }
}

suspend fun getSecretValues(secretName: String?): String? {
    val valueRequest =
```

```
        GetSecretValueRequest {
            secretId = secretName
        }

    SecretsManagerClient.fromEnvironment { region = "us-west-2" }.use
 { secretsClient ->
        val valueResponse = secretsClient.getSecretValue(valueRequest)
        return valueResponse.secretString
    }
}
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans AWS SDK for Kotlin API reference.
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Exécutez un scénario interactif à une invite de commande.

```
class RdsInstanceScenario:
    """Runs a scenario that shows how to get started using Amazon RDS DB
 instances."""

    def __init__(self, instance_wrapper):
        """
        :param instance_wrapper: An object that wraps Amazon RDS DB instance
 actions.
        """
        self.instance_wrapper = instance_wrapper

    def create_parameter_group(self, parameter_group_name, db_engine):
        """
        Shows how to get available engine versions for a specified database
 engine and
        create a DB parameter group that is compatible with a selected engine
 family.

        :param parameter_group_name: The name given to the newly created
 parameter group.
        :param db_engine: The database engine to use as a basis.
        :return: The newly created parameter group.
        """
        print(
            f"Checking for an existing DB instance parameter group named
 {parameter_group_name}."
        )
        parameter_group = self.instance_wrapper.get_parameter_group(
            parameter_group_name
        )
        if parameter_group is None:
            print(f"Getting available database engine versions for {db_engine}.")
            engine_versions =
 self.instance_wrapper.get_engine_versions(db_engine)
            families = list({ver["DBParameterGroupFamily"] for ver in
 engine_versions})
            family_index = q.choose("Which family do you want to use? ",
 families)
            print(f"Creating a parameter group.")
            self.instance_wrapper.create_parameter_group(
                parameter_group_name, families[family_index], "Example parameter
 group."
```

```
            )
            parameter_group = self.instance_wrapper.get_parameter_group(
                parameter_group_name
            )
        print(f"Parameter group {parameter_group['DBParameterGroupName']}:")
        pp(parameter_group)
        print("-" * 88)
        return parameter_group

    def update_parameters(self, parameter_group_name):
        """
        Shows how to get the parameters contained in a custom parameter group and
        update some of the parameter values in the group.

        :param parameter_group_name: The name of the parameter group to query and
modify.
        """
        print("Let's set some parameter values in your parameter group.")
        auto_inc_parameters = self.instance_wrapper.get_parameters(
            parameter_group_name, name_prefix="auto_increment"
        )
        update_params = []
        for auto_inc in auto_inc_parameters:
            if auto_inc["IsModifiable"] and auto_inc["DataType"] == "integer":
                print(f"The {auto_inc['ParameterName']} parameter is described
as:")
                print(f"\t{auto_inc['Description']}")
                param_range = auto_inc["AllowedValues"].split("-")
                auto_inc["ParameterValue"] = str(
                    q.ask(
                        f"Enter a value between {param_range[0]} and
{param_range[1]}: ",
                        q.is_int,
                        q.in_range(int(param_range[0]), int(param_range[1])),
                    )
                )
                update_params.append(auto_inc)
        self.instance_wrapper.update_parameters(parameter_group_name,
update_params)
        print(
            "You can get a list of parameters you've set by specifying a source
of 'user'."
        )
        user_parameters = self.instance_wrapper.get_parameters(
```

```
            parameter_group_name, source="user"
        )
        pp(user_parameters)
        print("-" * 88)

    def create_instance(self, instance_name, db_name, db_engine,
parameter_group):
        """
        Shows how to create a DB instance that contains a database of a specified
        type and is configured to use a custom DB parameter group.

        :param instance_name: The name given to the newly created DB instance.
        :param db_name: The name given to the created database.
        :param db_engine: The engine of the created database.
        :param parameter_group: The parameter group that is associated with the
DB instance.
        :return: The newly created DB instance.
        """
        print("Checking for an existing DB instance.")
        db_inst = self.instance_wrapper.get_db_instance(instance_name)
        if db_inst is None:
            print("Let's create a DB instance.")
            admin_username = q.ask(
                "Enter an administrator user name for the database: ",
q.non_empty
            )
            admin_password = q.ask(
                "Enter a password for the administrator (at least 8 characters):
",
                q.non_empty,
            )
            engine_versions = self.instance_wrapper.get_engine_versions(
                db_engine, parameter_group["DBParameterGroupFamily"]
            )
            engine_choices = [ver["EngineVersion"] for ver in engine_versions]
            print("The available engines for your parameter group are:")
            engine_index = q.choose("Which engine do you want to use? ",
engine_choices)
            engine_selection = engine_versions[engine_index]
            print(
                "The available micro DB instance classes for your database engine
are:"
            )
            inst_opts = self.instance_wrapper.get_orderable_instances(
```

```
                engine_selection["Engine"], engine_selection["EngineVersion"]
            )
            inst_choices = list(
                {
                    opt["DBInstanceClass"]
                    for opt in inst_opts
                    if "micro" in opt["DBInstanceClass"]
                }
            )
            inst_index = q.choose(
                "Which micro DB instance class do you want to use? ",
inst_choices
            )
            group_name = parameter_group["DBParameterGroupName"]
            storage_type = "standard"
            allocated_storage = 5
            print(
                f"Creating a DB instance named {instance_name} and database
{db_name}.\n"
                f"The DB instance is configured to use your custom parameter
group {group_name},\n"
                f"selected engine {engine_selection['EngineVersion']},\n"
                f"selected DB instance class {inst_choices[inst_index]},"
                f"and {allocated_storage} GiB of {storage_type} storage.\n"
                f"This typically takes several minutes."
            )
            db_inst = self.instance_wrapper.create_db_instance(
                db_name,
                instance_name,
                group_name,
                engine_selection["Engine"],
                engine_selection["EngineVersion"],
                inst_choices[inst_index],
                storage_type,
                allocated_storage,
                admin_username,
                admin_password,
            )
            while db_inst.get("DBInstanceStatus") != "available":
                wait(10)
                db_inst = self.instance_wrapper.get_db_instance(instance_name)
        print("Instance data:")
        pp(db_inst)
        print("-" * 88)
```

```
        return db_inst

    @staticmethod
    def display_connection(db_inst):
        """
        Displays connection information about a DB instance and tips on how to
        connect to it.

        :param db_inst: The DB instance to display.
        """
        print(
            "You can now connect to your database using your favorite MySql
 client.\n"
            "One way to connect is by using the 'mysql' shell on an Amazon EC2
 instance\n"
            "that is running in the same VPC as your DB instance. Pass the
 endpoint,\n"
            "port, and administrator user name to 'mysql' and enter your password
\n"
            "when prompted:\n"
        )
        print(
            f"\n\tmysql -h {db_inst['Endpoint']['Address']} -P
 {db_inst['Endpoint']['Port']} "
            f"-u {db_inst['MasterUsername']} -p\n"
        )
        print(
            "For more information, see the User Guide for Amazon RDS:\n"
            "\thttps://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/
CHAP_GettingStarted.CreatingConnecting.MySQL.html#CHAP_GettingStarted.Connecting.MySQL"
        )
        print("-" * 88)

    def create_snapshot(self, instance_name):
        """
        Shows how to create a DB instance snapshot and wait until it's available.

        :param instance_name: The name of a DB instance to snapshot.
        """
        if q.ask(
            "Do you want to create a snapshot of your DB instance (y/n)? ",
 q.is_yesno
        ):
            snapshot_id = f"{instance_name}-{uuid.uuid4()}"
```

```
            print(
                f"Creating a snapshot named {snapshot_id}. This typically takes a
few minutes."
            )
            snapshot = self.instance_wrapper.create_snapshot(snapshot_id,
instance_name)
            while snapshot.get("Status") != "available":
                wait(10)
                snapshot = self.instance_wrapper.get_snapshot(snapshot_id)
            pp(snapshot)
            print("-" * 88)

    def cleanup(self, db_inst, parameter_group_name):
        """
        Shows how to clean up a DB instance and parameter group.
        Before the parameter group can be deleted, all associated DB instances
must first
        be deleted.

        :param db_inst: The DB instance to delete.
        :param parameter_group_name: The DB parameter group to delete.
        """
        if q.ask(
            "\nDo you want to delete the DB instance and parameter group (y/n)?
",
            q.is_yesno,
        ):
            print(f"Deleting DB instance {db_inst['DBInstanceIdentifier']}.")

self.instance_wrapper.delete_db_instance(db_inst["DBInstanceIdentifier"])
            print(
                "Waiting for the DB instance to delete. This typically takes
several minutes."
            )
            while db_inst is not None:
                wait(10)
                db_inst = self.instance_wrapper.get_db_instance(
                    db_inst["DBInstanceIdentifier"]
                )
            print(f"Deleting parameter group {parameter_group_name}.")
            self.instance_wrapper.delete_parameter_group(parameter_group_name)

    def run_scenario(self, db_engine, parameter_group_name, instance_name,
db_name):
```

```
        logging.basicConfig(level=logging.INFO, format="%(levelname)s:
 %(message)s")

        print("-" * 88)
        print(
            "Welcome to the Amazon Relational Database Service (Amazon RDS)\n"
            "get started with DB instances demo."
        )
        print("-" * 88)

        parameter_group = self.create_parameter_group(parameter_group_name,
 db_engine)
        self.update_parameters(parameter_group_name)
        db_inst = self.create_instance(
            instance_name, db_name, db_engine, parameter_group
        )
        self.display_connection(db_inst)
        self.create_snapshot(instance_name)
        self.cleanup(db_inst, parameter_group_name)

        print("\nThanks for watching!")
        print("-" * 88)


if __name__ == "__main__":
    try:
        scenario = RdsInstanceScenario(InstanceWrapper.from_client())
        scenario.run_scenario(
            "mysql",
            "doc-example-parameter-group",
            "doc-example-instance",
            "docexampledb",
        )
    except Exception:
        logging.exception("Something went wrong with the demo.")
```

Définissez des fonctions appelées par le scénario pour gérer des actions Amazon RDS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
```

```
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_parameter_group(self, parameter_group_name):
        """
        Gets a DB parameter group.

        :param parameter_group_name: The name of the parameter group to retrieve.
        :return: The parameter group.
        """
        try:
            response = self.rds_client.describe_db_parameter_groups(
                DBParameterGroupName=parameter_group_name
            )
            parameter_group = response["DBParameterGroups"][0]
        except ClientError as err:
            if err.response["Error"]["Code"] == "DBParameterGroupNotFound":
                logger.info("Parameter group %s does not exist.",
parameter_group_name)
            else:
                logger.error(
                    "Couldn't get parameter group %s. Here's why: %s: %s",
                    parameter_group_name,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return parameter_group


    def create_parameter_group(
        self, parameter_group_name, parameter_group_family, description
```

```
    ):
        """
        Creates a DB parameter group that is based on the specified parameter
group
        family.

        :param parameter_group_name: The name of the newly created parameter
group.
        :param parameter_group_family: The family that is used as the basis of
the new
                                       parameter group.
        :param description: A description given to the parameter group.
        :return: Data about the newly created parameter group.
        """
        try:
            response = self.rds_client.create_db_parameter_group(
                DBParameterGroupName=parameter_group_name,
                DBParameterGroupFamily=parameter_group_family,
                Description=description,
            )
        except ClientError as err:
            logger.error(
                "Couldn't create parameter group %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return response


    def delete_parameter_group(self, parameter_group_name):
        """
        Deletes a DB parameter group.

        :param parameter_group_name: The name of the parameter group to delete.
        :return: Data about the parameter group.
        """
        try:
            self.rds_client.delete_db_parameter_group(
                DBParameterGroupName=parameter_group_name
            )
        except ClientError as err:
```

```
            logger.error(
                "Couldn't delete parameter group %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise


    def get_parameters(self, parameter_group_name, name_prefix="", source=None):
        """
        Gets the parameters that are contained in a DB parameter group.

        :param parameter_group_name: The name of the parameter group to query.
        :param name_prefix: When specified, the retrieved list of parameters is
filtered
                            to contain only parameters that start with this
prefix.
        :param source: When specified, only parameters from this source are
retrieved.
                       For example, a source of 'user' retrieves only parameters
that
                       were set by a user.
        :return: The list of requested parameters.
        """
        try:
            kwargs = {"DBParameterGroupName": parameter_group_name}
            if source is not None:
                kwargs["Source"] = source
            parameters = []
            paginator = self.rds_client.get_paginator("describe_db_parameters")
            for page in paginator.paginate(**kwargs):
                parameters += [
                    p
                    for p in page["Parameters"]
                    if p["ParameterName"].startswith(name_prefix)
                ]
        except ClientError as err:
            logger.error(
                "Couldn't get parameters for %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
```

```
            raise
        else:
            return parameters


    def update_parameters(self, parameter_group_name, update_parameters):
        """
        Updates parameters in a custom DB parameter group.

        :param parameter_group_name: The name of the parameter group to update.
        :param update_parameters: The parameters to update in the group.
        :return: Data about the modified parameter group.
        """
        try:
            response = self.rds_client.modify_db_parameter_group(
                DBParameterGroupName=parameter_group_name,
Parameters=update_parameters
            )
        except ClientError as err:
            logger.error(
                "Couldn't update parameters in %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return response


    def create_snapshot(self, snapshot_id, instance_id):
        """
        Creates a snapshot of a DB instance.

        :param snapshot_id: The ID to give the created snapshot.
        :param instance_id: The ID of the DB instance to snapshot.
        :return: Data about the newly created snapshot.
        """
        try:
            response = self.rds_client.create_db_snapshot(
                DBSnapshotIdentifier=snapshot_id,
DBInstanceIdentifier=instance_id
            )
            snapshot = response["DBSnapshot"]
```

```
        except ClientError as err:
            logger.error(
                "Couldn't create snapshot of %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return snapshot


    def get_snapshot(self, snapshot_id):
        """
        Gets a DB instance snapshot.

        :param snapshot_id: The ID of the snapshot to retrieve.
        :return: The retrieved snapshot.
        """
        try:
            response = self.rds_client.describe_db_snapshots(
                DBSnapshotIdentifier=snapshot_id
            )
            snapshot = response["DBSnapshots"][0]
        except ClientError as err:
            logger.error(
                "Couldn't get snapshot %s. Here's why: %s: %s",
                snapshot_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return snapshot


    def get_engine_versions(self, engine, parameter_group_family=None):
        """
        Gets database engine versions that are available for the specified engine
        and parameter group family.

        :param engine: The database engine to look up.
        :param parameter_group_family: When specified, restricts the returned
list of
```

```
                                        engine versions to those that are
compatible with
                                        this parameter group family.
        :return: The list of database engine versions.
        """
        try:
            kwargs = {"Engine": engine}
            if parameter_group_family is not None:
                kwargs["DBParameterGroupFamily"] = parameter_group_family
            response = self.rds_client.describe_db_engine_versions(**kwargs)
            versions = response["DBEngineVersions"]
        except ClientError as err:
            logger.error(
                "Couldn't get engine versions for %s. Here's why: %s: %s",
                engine,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return versions


    def get_orderable_instances(self, db_engine, db_engine_version):
        """
        Gets DB instance options that can be used to create DB instances that are
        compatible with a set of specifications.

        :param db_engine: The database engine that must be supported by the DB
instance.
        :param db_engine_version: The engine version that must be supported by
the DB instance.
        :return: The list of DB instance options that can be used to create a
compatible DB instance.
        """
        try:
            inst_opts = []
            paginator = self.rds_client.get_paginator(
                "describe_orderable_db_instance_options"
            )
            for page in paginator.paginate(
                Engine=db_engine, EngineVersion=db_engine_version
            ):
                inst_opts += page["OrderableDBInstanceOptions"]
```

```
        except ClientError as err:
            logger.error(
                "Couldn't get orderable DB instances. Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return inst_opts


    def get_db_instance(self, instance_id):
        """
        Gets data about a DB instance.

        :param instance_id: The ID of the DB instance to retrieve.
        :return: The retrieved DB instance.
        """
        try:
            response = self.rds_client.describe_db_instances(
                DBInstanceIdentifier=instance_id
            )
            db_inst = response["DBInstances"][0]
        except ClientError as err:
            if err.response["Error"]["Code"] == "DBInstanceNotFound":
                logger.info("Instance %s does not exist.", instance_id)
            else:
                logger.error(
                    "Couldn't get DB instance %s. Here's why: %s: %s",
                    instance_id,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return db_inst


    def create_db_instance(
        self,
        db_name,
        instance_id,
        parameter_group_name,
        db_engine,
```

```
        db_engine_version,
        instance_class,
        storage_type,
        allocated_storage,
        admin_name,
        admin_password,
    ):
        """
        Creates a DB instance.

        :param db_name: The name of the database that is created in the DB
instance.
        :param instance_id: The ID to give the newly created DB instance.
        :param parameter_group_name: A parameter group to associate with the DB
instance.
        :param db_engine: The database engine of a database to create in the DB
instance.
        :param db_engine_version: The engine version for the created database.
        :param instance_class: The DB instance class for the newly created DB
instance.
        :param storage_type: The storage type of the DB instance.
        :param allocated_storage: The amount of storage allocated on the DB
instance, in GiBs.
        :param admin_name: The name of the admin user for the created database.
        :param admin_password: The admin password for the created database.
        :return: Data about the newly created DB instance.
        """
        try:
            response = self.rds_client.create_db_instance(
                DBName=db_name,
                DBInstanceIdentifier=instance_id,
                DBParameterGroupName=parameter_group_name,
                Engine=db_engine,
                EngineVersion=db_engine_version,
                DBInstanceClass=instance_class,
                StorageType=storage_type,
                AllocatedStorage=allocated_storage,
                MasterUsername=admin_name,
                MasterUserPassword=admin_password,
            )
            db_inst = response["DBInstance"]
        except ClientError as err:
            logger.error(
                "Couldn't create DB instance %s. Here's why: %s: %s",
```

```
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return db_inst


    def delete_db_instance(self, instance_id):
        """
        Deletes a DB instance.

        :param instance_id: The ID of the DB instance to delete.
        :return: Data about the deleted DB instance.
        """
        try:
            response = self.rds_client.delete_db_instance(
                DBInstanceIdentifier=instance_id,
                SkipFinalSnapshot=True,
                DeleteAutomatedBackups=True,
            )
            db_inst = response["DBInstance"]
        except ClientError as err:
            logger.error(
                "Couldn't delete DB instance %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return db_inst
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans AWS SDK for Python (Boto3) API Reference.
  - CréerDBInstance
  - Créer un DBParameter groupe

- CréerDBSnapshot
- SuppressionDBInstance
- Supprimer le DBParameter groupe
- Décrire DBEngine les versions
- Décrivez DBInstances
- Décrire DBParameter les groupes
- Décrivez DBParameters
- Décrivez DBSnapshots
- DescribeOrderableDBInstanceOptions
- Modifier le DBParameter groupe

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Fichier `Package.swift`.

```
// swift-tools-version: 5.9
//
// The swift-tools-version declares the minimum version of Swift required to
// build this package.

import PackageDescription

let package = Package(
    name: "rds-scenario",
    // Let Xcode know the minimum Apple platforms supported.
    platforms: [
        .macOS(.v13),
        .iOS(.v15)
    ],
    dependencies: [
```

```
        // Dependencies declare other packages that this package depends on.
        .package(
            url: "https://github.com/awslabs/aws-sdk-swift",
            from: "1.4.0"),
        .package(
            url: "https://github.com/apple/swift-argument-parser.git",
            branch: "main"
        )
    ],
    targets: [
        // Targets are the basic building blocks of a package, defining a module
 or a test suite.
        // Targets can depend on other targets in this package and products
        // from dependencies.
        .executableTarget(
            name: "rds-scenario",
            dependencies: [
                .product(name: "AWSRDS", package: "aws-sdk-swift"),
                .product(name: "ArgumentParser", package: "swift-argument-
parser")
            ],
            path: "Sources")

    ]
)
```

Le fichier de code Swift,`entry.swift`.

```
// An example that shows how to use the AWS SDK for Swift to perform a variety
// of operations using Amazon Relational Database Service (RDS).
//

import ArgumentParser
import Foundation
import AWSRDS

struct ExampleCommand: ParsableCommand {
    @Option(help: "The AWS Region to run AWS API calls in.")
    var awsRegion = "us-east-1"
    @Option(help: "The username to use for the database administrator.")
    var dbUsername = "admin"
    @Option(help: "The password to use for the database administrator.")
```

```
    var dbPassword: String

    static var configuration = CommandConfiguration(
        commandName: "rds-scenario",
        abstract: """
        Performs various operations to demonstrate the use of Amazon RDS
 Instances
        using the AWS SDK for Swift.
        """,
        discussion: """
        """
    )

    /// Called by ``main()`` to run the bulk of the example.
    func runAsync() async throws {
        let example = try await Example(region: awsRegion, username: dbUsername,
 password: dbPassword)

        await example.run()
    }
}

class Example {
    let rdsClient: RDSClient

    // Storage for AWS RDS properties

    let dbUsername: String
    let dbPassword: String
    var dbInstanceIdentifier: String
    var dbSnapshotIdentifier: String
    var dbParameterGroupName: String
    var dbParameterGroup: RDSClientTypes.DBParameterGroup?
    var selectedEngineVersion: String?

    init(region: String, username: String, password: String) async throws{
        let rdsConfig = try await RDSClient.RDSClientConfiguration(region:
 region)
        rdsClient = RDSClient(config: rdsConfig)

        dbUsername = username
        dbPassword = password
        dbParameterGroupName = ""
        dbInstanceIdentifier = ""
```

```
        dbSnapshotIdentifier = ""
    }

    /// The example's main body.
    func run() async {
        var parameterGroupFamilies: Set<String> = []

        //========================================================================
        // 1. Get available database engine families for MySQL.
        //========================================================================

        let engineVersions = await getDBEngineVersions(engineName: "mysql")

        for version in engineVersions {
            if version.dbParameterGroupFamily != nil {
                parameterGroupFamilies.insert(version.dbParameterGroupFamily!)
            }
        }

        if engineVersions.count > 0 {
            selectedEngineVersion = engineVersions.last!.engineVersion
        } else {
            print("*** Unable to find a valid database engine version. Canceling
operations.")
            await cleanUp()
            return
        }

        print("Found \(parameterGroupFamilies.count) parameter group families:")
        for family in parameterGroupFamilies {
            print("    \(family)")
        }

        //========================================================================
        // 2. Select an engine family and create a custom DB parameter group.
        //    We select a family by sorting the set of family names, then
        //    choosing the last one.
        //========================================================================

        let sortedFamilies = parameterGroupFamilies.sorted()

        guard let selectedFamily = sortedFamilies.last else {
            print("*** Unable to find a database engine family. Canceling
operations.")
```

```
            await cleanUp()
            return
        }

        print("Selected database engine family \(selectedFamily)")

        dbParameterGroupName = tempName(prefix: "rds-example")
        print("Creating a database parameter group named \(dbParameterGroupName)
using \(selectedFamily)")
        dbParameterGroup = await createDBParameterGroup(groupName:
dbParameterGroupName,
                                                        familyName:
selectedFamily)

        //===========================================================================
        // 3. Get the parameter group's details.
        //===========================================================================

        print("Getting the database parameter group list...")
        let dbParameterGroupList = await describeDBParameterGroups(groupName:
dbParameterGroupName)
        guard let dbParameterGroupList else {
            await cleanUp()
            return
        }

        print("Found \(dbParameterGroupList.count) parameter groups...")
        for group in dbParameterGroupList {
            print("    \(group.dbParameterGroupName ?? "<unknown>")")
        }
        print()

        //===========================================================================
        // 4. Get a list of the parameter group's parameters. This list is
        //    likely to be long, so use pagination. Find the
        //    auto_increment_offset and auto_increment_increment parameters.
        //===========================================================================

        let parameters = await describeDBParameters(groupName:
dbParameterGroupName)

        //===========================================================================
        // 5. Parse and display each parameter's name, description, and
        //    allowed values.
```

```
        //===========================================================================

        for parameter in parameters {
            let name = parameter.parameterName
            guard let name else {
                print("*** Unable to get parameter name!")
                continue
            }

            if name == "auto_increment_offset" || name ==
"auto_increment_increment" {
                print("Parameter \(name):")
                print("          Value: \(parameter.parameterValue ??
"<undefined>")")
                print("      Data type: \(parameter.dataType ?? "<unknown>")")
                print("    Description: \(parameter.description ?? "")")
                print(" Allowed values: \(parameter.allowedValues ??
"<unspecified")")
                print(String(repeating: "=", count: 78))
            }
        }

        //===========================================================================
        // 6. Modify both the auto_increment_offset and
        //    auto_increment_increment parameters in one call in the custom
        //    parameter group. Set their parameterValue fields to a new
        //    permitted value.
        //===========================================================================

        print("Setting auto_increment_offset and auto_increment_increment both to
5...")
        await modifyDBParameters(groupName: dbParameterGroupName)

        //===========================================================================
        // 7. Get and display the updated parameters, specifying a source of
        //    "user" to get only the modified parameters.
        //===========================================================================

        let updatedParameters = await describeDBParameters(groupName:
dbParameterGroupName, source: "user")

        for parameter in updatedParameters {
            let name = parameter.parameterName
            guard let name else {
```

```
                print("*** Unable to get parameter name!")
                continue
            }

            print("Parameter \(name):")
            print("          Value: \(parameter.parameterValue ??
"<undefined>")")
            print("      Data type: \(parameter.dataType ?? "<unknown>")")
            print("    Description: \(parameter.description ?? "")")
            print(" Allowed values: \(parameter.allowedValues ??
"<unspecified")")
            print(String(repeating: "=", count: 78))
        }

        //========================================================================
        // 8. Get a list of allowed engine versions using
        //    DescribeRDSEngineVersions.
        //========================================================================

        await listAllowedEngines(family: selectedFamily)

        //========================================================================
        // 9. Get a list of micro instance classes available for the selected
        //    engine and engine version.
        //========================================================================

        let dbInstanceClass = await chooseMicroInstance(engine: "mysql",
engineVersion: selectedEngineVersion)
        guard let dbInstanceClass else {
            print("Did not get a valid instance class. Canceling operations.")
            await cleanUp()
            return
        }

        //========================================================================
        // 10. Create an RDS database that contains a MySQL database and uses
        //     the parameter group we created.
        //========================================================================

        print("Creating the database instance...")

        guard let instanceClass = dbInstanceClass.dbInstanceClass else {
            print("Instance class name is unknown. Canceling operations.")
            await cleanUp()
```

```
            return
        }

        dbInstanceIdentifier = tempName(prefix: "sample-identifier")
        let dbInstanceArn = await createDBInstance(
            name: "SampleDatabase\(Int.random(in: 1000000..<1000000000))",
            instanceIdentifier: dbInstanceIdentifier,
            parameterGroupName: dbParameterGroupName,
            engine: "mysql",
            engineVersion: selectedEngineVersion!,
            instanceClass: instanceClass,
            username: dbUsername,
            password: dbPassword
        )

        if dbInstanceArn == nil {
            await cleanUp()
            return
        }

        //===========================================================================
        // 11. Wait for the database instance to be ready by calling
        //     DescribeDBInstances repeatedly until it reports
        //     dbInstanceStatus as "available". This can take upwards of 10
        //     minutes, let the user know that.
        //===========================================================================

        guard let endpoint = await waitUntilDBInstanceReady(instanceIdentifier:
dbInstanceIdentifier) else {
            print("\nDid not get a valid endpoint from AWS RDS.")
            await cleanUp()
            return
        }

        guard let endpointAddress = endpoint.address else {
            print("\nNo endpoint address returned.")
            await cleanUp()
            return
        }
        guard let endpointPort = endpoint.port else {
            print("\nNo endpoint port returned.")
            await cleanUp()
            return
        }
```

```
        //=========================================================================
        // 12. Display connection information for the database instance.
        //=========================================================================

        print("\nTo connect to the new database instance using 'mysql' from the
shell:")
        print("    mysql -h \(endpointAddress) -P \(endpointPort) -u
\(self.dbUsername)")

        //=========================================================================
        // 13. Create a snapshot of the database instance.
        //=========================================================================

        dbSnapshotIdentifier = tempName(prefix: "sample-snapshot")
        await createDBSnapshot(instanceIdentifier: dbInstanceIdentifier,
snapshotIdentifier: dbSnapshotIdentifier)

        //=========================================================================
        // 14. Wait for the snapshot to be ready.
        //=========================================================================

        await waitUntilDBSnapshotReady(instanceIdentifier: dbInstanceIdentifier,
snapshotIdentifier: dbSnapshotIdentifier)

        // That's it! Clean up and exit!

        print("Example complete! Cleaning up...")
        await cleanUp()
    }

    /// Clean up by discarding and closing down all allocated EC2 items.
    func cleanUp() async {
        print("Deleting the database instance \(dbInstanceIdentifier)...")
        await deleteDBInstance(instanceIdentifier: dbInstanceIdentifier)
        await waitUntilDBInstanceDeleted(instanceIdentifier:
dbInstanceIdentifier)

        print("Deleting the database parameter group \(dbParameterGroupName)...")
        await deleteDBParameterGroup(groupName: dbParameterGroupName)
    }

    /// Get all the database engine versions available for the specified
    /// database engine.
```

```
    ///
    /// - Parameter engineName: The name of the database engine to query.
    ///
    /// - Returns: An array of `RDSClientTypes.DBEngineVersion` structures,
    ///   each describing one supported version of the specified database.
    func getDBEngineVersions(engineName: String) async ->
[RDSClientTypes.DBEngineVersion] {
        do {
            let output = try await rdsClient.describeDBEngineVersions(
                input: DescribeDBEngineVersionsInput(
                    engine: engineName
                )
            )

            return output.dbEngineVersions ?? []
        } catch {
            return []
        }
    }

    /// Create a new database parameter group with the specified name.
    ///
    /// - Parameters:
    ///   - groupName: The name of the new parameter group.
    ///   - familyName: The name of the parameter group family.
    /// - Returns:
    func createDBParameterGroup(groupName: String, familyName: String) async ->
RDSClientTypes.DBParameterGroup? {
        do {
            let output = try await rdsClient.createDBParameterGroup(
                input: CreateDBParameterGroupInput(
                    dbParameterGroupFamily: familyName,
                    dbParameterGroupName: groupName,
                    description: "Created using the AWS SDK for Swift"
                )
            )
            return output.dbParameterGroup
        } catch {
            print("*** Error creating the parameter group:
\(error.localizedDescription)")
            return nil
        }
    }
```

```
    /// Get descriptions of the database parameter groups matching the given
    /// name.
    ///
    /// - Parameter groupName: The name of the parameter group to describe.
    ///
    /// - Returns: An array of [RDSClientTypes.DBParameterGroup] objects
    ///   describing the parameter group.
    func describeDBParameterGroups(groupName: String) async ->
[RDSClientTypes.DBParameterGroup]? {
        do {
            let output = try await rdsClient.describeDBParameterGroups(
                input: DescribeDBParameterGroupsInput(
                    dbParameterGroupName: groupName
                )
            )
            return output.dbParameterGroups
        } catch {
            print("*** Error getting the database parameter group's details:
\(error.localizedDescription)")
            return nil
        }
    }

    /// Returns the detailed parameter list for the specified database
    /// parameter group.
    ///
    /// - Parameters:
    ///   - groupName: The name of the parameter group to return parameters for.
    ///   - source: The types of parameters to return (`user`, `system`, or
    ///     `engine-default`).
    ///
    /// - Returns: An array of `RdSClientTypes.Parameter` objects, each
    ///   describing one of the group's parameters.
    func describeDBParameters(groupName: String, source: String? = nil) async ->
[RDSClientTypes.Parameter] {
        var parameterList: [RDSClientTypes.Parameter] = []

        do {
            let pages = rdsClient.describeDBParametersPaginated(
                input: DescribeDBParametersInput(
                    dbParameterGroupName: groupName,
                    source: source
                )
            )
```

```
            for try await page in pages {
                guard let parameters = page.parameters else {
                    return []
                }

                parameterList += parameters
            }
        } catch {
            print("*** Error getting database parameters:
\(error.localizedDescription)")
            return []
        }

        return parameterList
    }

    /// Demonstrates modifying two of the specified database parameter group's
    /// parameters.
    ///
    /// - Parameter groupName: The name of the parameter group to change
    ///   parameters for.
    func modifyDBParameters(groupName: String) async {
        let parameter1 = RDSClientTypes.Parameter(
            applyMethod: RDSClientTypes.ApplyMethod.immediate,
            parameterName: "auto_increment_offset",
            parameterValue: "5"
        )
        let parameter2 = RDSClientTypes.Parameter(
            applyMethod: RDSClientTypes.ApplyMethod.immediate,
            parameterName: "auto_increment_increment",
            parameterValue: "5"
        )

        let parameterList = [parameter1, parameter2]

        do {
            _ = try await rdsClient.modifyDBParameterGroup(
                input: ModifyDBParameterGroupInput(
                    dbParameterGroupName: groupName,
                    parameters: parameterList
                )
            )
```

```
            print("Successfully modified the parameter group \(groupName).")
        } catch {
            print("*** Error modifying the parameter group \(groupName):
\(error.localizedDescription)")
        }
    }

    /// Output a list of the database engine versions supported by the
    /// specified family.
    ///
    /// - Parameter family: The family for which to list allowed database
    ///   engines.
    func listAllowedEngines(family: String?) async {
        do {
            let output = try await rdsClient.describeDBEngineVersions(
                input: DescribeDBEngineVersionsInput(
                    dbParameterGroupFamily: family,
                    engine: "mysql"
                )
            )

            guard let engineVersions = output.dbEngineVersions else {
                print("No engine versions returned.")
                return
            }

            print("Found \(engineVersions.count) database engine versions:")
            for version in engineVersions {
                print("    \(version.engineVersion ?? "<unknown>"):
\(version.dbEngineDescription ?? "")")
            }
        } catch {
            print("*** Error getting database engine version list:
\(error.localizedDescription)")
            return
        }
    }

    /// Print a list of available database instances with "micro" in the class
    /// name, then return one of them to be used by other code.
    ///
    /// - Parameters:
    ///   - engine: The database engine for which to list database instance
    ///     classes.
```

```
    ///   - engineVersion: The database version for which to list instances.
    ///
    /// - Returns: An `RDSClientTypes.OrderableDBInstanceOption` describing
    ///   the selected instance type.
    func chooseMicroInstance(engine: String = "mysql", engineVersion: String? =
nil) async -> RDSClientTypes.OrderableDBInstanceOption? {
        do {
            let pages = rdsClient.describeOrderableDBInstanceOptionsPaginated(
                input: DescribeOrderableDBInstanceOptionsInput(
                    engine: engine,
                    engineVersion: engineVersion
                )
            )

            var optionsList: [RDSClientTypes.OrderableDBInstanceOption] = []

            for try await page in pages {
                guard let orderableDBInstanceOptions =
page.orderableDBInstanceOptions else {
                    continue
                }

                for dbInstanceOption in orderableDBInstanceOptions {
                    guard let className = dbInstanceOption.dbInstanceClass else {
                        continue
                    }
                    if className.contains("micro") {
                        optionsList.append(dbInstanceOption)
                    }
                }
            }

            print("Found \(optionsList.count) database instances of 'micro' class
types:")
            for dbInstanceOption in optionsList {
                print("    \(dbInstanceOption.engine ??
"<unknown>") \(dbInstanceOption.engineVersion ?? "<unknown>")
(\(dbInstanceOption.dbInstanceClass ?? "<unknown class>"))")
            }

            return optionsList[0]
        } catch {
            print("*** Error getting a list of orderable instance options:
\(error.localizedDescription)")
```

```
            return nil
        }
    }

    /// Create a new database instance.
    ///
    /// - Parameters:
    ///   - name: The name of the database to create.
    ///   - instanceIdentifier: The identifier to give the new database
    ///     instance.
    ///   - parameterGroupName: The name of the parameter group to associate
    ///     with the new database instance.
    ///   - engine: The database engine to use.
    ///   - engineVersion: The version of the database given by `engine` to
    ///     use.
    ///   - instanceClass: The memory and compute capacity of the database
    ///     instance, such as `db.m5.large``.
    ///   - username: The admin user's username to establish for the new
    ///     instance.
    ///   - password: The password to use for the specified user's access.
    ///
    /// - Returns: A string indicating the ARN of the newly created database
    ///   instance, or nil if the instance couldn't be created.
    func createDBInstance(name: String, instanceIdentifier: String,
parameterGroupName: String,
                          engine: String, engineVersion: String, instanceClass:
String,
                          username: String, password: String) async -> String? {
        do {
            let output = try await rdsClient.createDBInstance(
                input: CreateDBInstanceInput(
                    allocatedStorage: 100,
                    dbInstanceClass: instanceClass,
                    dbInstanceIdentifier: instanceIdentifier,
                    dbName: name,
                    dbParameterGroupName: parameterGroupName,
                    engine: engine,
                    engineVersion: engineVersion,
                    masterUserPassword: password,
                    masterUsername: username,
                    storageType: "gp2"
                )
            )
```

```
            guard let dbInstance = output.dbInstance else {
                print("*** Unable to get the database instance.")
                return nil
            }

            return dbInstance.dbInstanceArn
        } catch {
            print("*** An error occurred while creating the database instance:
\(error.localizedDescription)")
            return nil
        }
    }

    /// Wait until the specified database is available to use.
    ///
    /// - Parameter instanceIdentifier: The database instance identifier of the
    ///   database to wait for.
    func waitUntilDBInstanceReady(instanceIdentifier: String) async ->
RDSClientTypes.Endpoint? {
        do {
            putString("Waiting for the database instance to be ready to use. This
may take 10 minutes or more...")
            while true {
                let output = try await rdsClient.describeDBInstances(
                    input: DescribeDBInstancesInput(
                        dbInstanceIdentifier: instanceIdentifier
                    )
                )

                guard let instanceList = output.dbInstances else {
                    continue
                }

                for instance in instanceList {
                    let status = instance.dbInstanceStatus

                    guard let status else {
                        print("\nUnable to determine the status.")
                        continue
                    }

                    if status.contains("available") {
                        return instance.endpoint
                    } else {
```

```
                        putString(".")
                        do {
                            try await Task.sleep(for: .seconds(15))
                        } catch {
                            print("*** Error pausing the task!")
                        }
                    }
                }
            }
        } catch {
            print("*** Unable to wait until the database is ready:
\(error.localizedDescription)")
            return nil
        }
    }

    /// Create a snapshot of the specified name.
    ///
    /// - Parameters:
    ///   - instanceIdentifier: The identifier of the database instance to
    ///     snapshot.
    ///   - snapshotIdentifier: A unique identifier to give the newly-created
    ///     snapshot.
    func createDBSnapshot(instanceIdentifier: String, snapshotIdentifier: String)
async {
        do {
            let output = try await rdsClient.createDBSnapshot(
                input: CreateDBSnapshotInput(
                    dbInstanceIdentifier: instanceIdentifier,
                    dbSnapshotIdentifier: snapshotIdentifier
                )
            )

            guard let snapshot = output.dbSnapshot else {
                print("No snapshot returned.")
                return
            }

            print("The snapshot has been created with ID
\(snapshot.dbiResourceId ?? "<unknown>")")
        } catch {
            print("*** Unable to create the database snapshot named
\(snapshotIdentifier): \(error.localizedDescription)")
        }
```

```
    }

    /// Wait until the specified database snapshot is available to use.
    ///
    /// - Parameters:
    ///   - instanceIdentifier: The identifier of the database for which the
    ///     snapshot was taken.
    ///   - snapshotIdentifier: The identifier of the snapshot to wait for.
    func waitUntilDBSnapshotReady(instanceIdentifier: String, snapshotIdentifier:
String) async {
        var snapshotReady = false

        putString("Waiting for the snapshot to be ready...")

        do {
            while !snapshotReady {
                let output = try await rdsClient.describeDBSnapshots(
                    input: DescribeDBSnapshotsInput(
                        dbInstanceIdentifier: instanceIdentifier,
                        dbSnapshotIdentifier: snapshotIdentifier
                    )
                )

                guard let snapshotList = output.dbSnapshots else {
                    return
                }

                for snapshot in snapshotList {
                    guard let snapshotReadyStr = snapshot.status else {
                        return
                    }

                    if snapshotReadyStr.contains("available") {
                        snapshotReady = true
                        print()
                    } else {
                        putString(".")
                        do {
                            try await Task.sleep(for: .seconds(15))
                        } catch {
                            print("\n*** Error pausing the task!")
                        }
                    }
                }
```

```
            }
        } catch {
            print("\n*** Unable to wait for the database snapshot to be ready:
\(error.localizedDescription)")
        }
    }

    /// Delete the specified database instance.
    ///
    /// - Parameter instanceIdentifier: The identifier of the database
    ///   instance to delete.
    func deleteDBInstance(instanceIdentifier: String) async {
        do {
            _ = try await rdsClient.deleteDBInstance(
                input: DeleteDBInstanceInput(
                    dbInstanceIdentifier: instanceIdentifier,
                    deleteAutomatedBackups: true,
                    skipFinalSnapshot: true
                )
            )
        } catch {
            print("*** Error deleting the database instance
\(instanceIdentifier): \(error.localizedDescription)")
        }
    }

    /// Wait until the specified database instance has been deleted.
    ///
    /// - Parameter instanceIdentifier: The identifier of the database
    ///   instance to wait for.
    func waitUntilDBInstanceDeleted(instanceIdentifier: String) async {
        putString("Waiting for the database instance to be deleted. This may take
a few minutes...")
        do {
            var isDatabaseDeleted = false
            var foundInstance = false

            while !isDatabaseDeleted {
                let output = try await rdsClient.describeDBInstances(input:
DescribeDBInstancesInput())
                guard let instanceList = output.dbInstances else {
                    return
                }
```

```
                foundInstance = false

                for instance in instanceList {
                    guard let foundInstanceIdentifier =
instance.dbInstanceIdentifier else {
                        continue
                    }

                    if instanceIdentifier == foundInstanceIdentifier {
                        foundInstance = true
                        break
                    } else {
                        putString(".")
                        do {
                            try await Task.sleep(for: .seconds(15))
                        } catch {
                            print("\n*** Error pausing the task!")
                        }
                    }
                }
                if !foundInstance {
                    isDatabaseDeleted = true
                    print()
                }
            }
        } catch {
            print("\n*** Error waiting for the database instance to be deleted:
\(error.localizedDescription)")
        }
    }

    /// Delete the specified database parameter group.
    ///
    /// - Parameter groupName: The name of the parameter group to delete.
    func deleteDBParameterGroup(groupName: String) async {
        do {
            _ = try await rdsClient.deleteDBParameterGroup(
                input: DeleteDBParameterGroupInput(
                    dbParameterGroupName: groupName
                )
            )
        } catch {
            print("*** Error deleting the database parameter group \(groupName):
\(error.localizedDescription)")
```

```
        }
    }

    /// Generate and return a unique file name that begins with the specified
    /// string.
    ///
    /// - Parameters:
    ///   - prefix: Text to use at the beginning of the returned name.
    ///
    /// - Returns: A string containing a unique filename that begins with the
    ///   specified `prefix`.
    ///
    /// The returned name uses a random number between 1 million and 1 billion to
    /// provide reasonable certainty of uniqueness for the purposes of this
    /// example.
    func tempName(prefix: String) -> String {
        return "\(prefix)-\(Int.random(in: 1000000..<1000000000))"
    }

    /// Print a string to standard output without a trailing newline, and
    /// without buffering.
    ///
    /// - Parameter str: The string to output.
    func putString(_ str: String = "") {
        if str.length >= 1 {
            let data = str.data(using: .utf8)
            guard let data else {
                return
            }
            FileHandle.standardOutput.write(data)
        }
    }
}

/// The program's asynchronous entry point.
@main
struct Main {
    static func main() async {
        let args = Array(CommandLine.arguments.dropFirst())

        do {
            let command = try ExampleCommand.parse(args)
            try await command.runAsync()
        } catch {
```

```
            ExampleCommand.exit(withError: error)
        }
    }
}
```

- Pour plus d'informations sur l'API, consultez les rubriques suivantes dans la référence de l'API du kit SDK AWS pour Swift.
  - CréerDBInstance
  - Créer un DBParameter groupe
  - CréerDBSnapshot
  - SuppressionDBInstance
  - Supprimer le DBParameter groupe
  - Décrire DBEngine les versions
  - Décrivez DBInstances
  - Décrire DBParameter les groupes
  - Décrivez DBParameters
  - Décrivez DBSnapshots
  - DescribeOrderableDBInstanceOptions
  - Modifier le DBParameter groupe

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Actions pour Amazon RDS utilisant AWS SDKs

Les exemples de code suivants montrent comment effectuer des actions Amazon RDS individuelles avec AWS SDKs. Chaque exemple inclut un lien vers GitHub, où vous pouvez trouver des instructions pour configurer et exécuter le code.

Ces extraits appellent l'API Amazon RDS et sont des extraits de code de programmes plus volumineux qui doivent être exécutés en contexte. Vous pouvez voir les actions dans leur contexte dans Scénarios d'utilisation d'Amazon RDS AWS SDKs .

Les exemples suivants incluent uniquement les actions les plus couramment utilisées. Pour une liste complète, consultez le manuel Amazon Relational Database Service API Reference.

Exemples

- Utilisation CreateDBInstance avec un AWS SDK ou une CLI
- Utilisation CreateDBParameterGroup avec un AWS SDK ou une CLI
- Utilisation CreateDBSnapshot avec un AWS SDK ou une CLI
- Utilisation DeleteDBInstance avec un AWS SDK ou une CLI
- Utilisation DeleteDBParameterGroup avec un AWS SDK ou une CLI
- Utilisation DescribeAccountAttributes avec un AWS SDK ou une CLI
- Utilisation DescribeDBEngineVersions avec un AWS SDK ou une CLI
- Utilisation DescribeDBInstances avec un AWS SDK ou une CLI
- Utilisation DescribeDBParameterGroups avec un AWS SDK ou une CLI
- Utilisation DescribeDBParameters avec un AWS SDK ou une CLI
- Utilisation DescribeDBSnapshots avec un AWS SDK ou une CLI
- Utilisation DescribeOrderableDBInstanceOptions avec un AWS SDK ou une CLI
- Utilisation GenerateRDSAuthToken avec un AWS SDK
- Utilisation ModifyDBInstance avec un AWS SDK ou une CLI
- Utilisation ModifyDBParameterGroup avec un AWS SDK ou une CLI
- Utilisation RebootDBInstance avec un AWS SDK ou une CLI

Utilisation **`CreateDBInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `CreateDBInstance`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Create an RDS DB instance with a particular set of properties. Use the
action DescribeDBInstancesAsync
   /// to determine when the DB instance is ready to use.
   /// </summary>
   /// <param name="dbName">Name for the DB instance.</param>
   /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
   /// <param name="parameterGroupName">DB parameter group to associate with the
instance.</param>
   /// <param name="dbEngine">The engine for the DB instance.</param>
   /// <param name="dbEngineVersion">Version for the DB instance.</param>
   /// <param name="instanceClass">Class for the DB instance.</param>
   /// <param name="allocatedStorage">The amount of storage in gibibytes (GiB)
to allocate to the DB instance.</param>
   /// <param name="adminName">Admin user name.</param>
   /// <param name="adminPassword">Admin user password.</param>
   /// <returns>DB instance object.</returns>
   public async Task<DBInstance> CreateDBInstance(string dbName, string
dbInstanceIdentifier,
       string parameterGroupName, string dbEngine, string dbEngineVersion,
       string instanceClass, int allocatedStorage, string adminName, string
adminPassword)
   {
       var response = await _amazonRDS.CreateDBInstanceAsync(
           new CreateDBInstanceRequest()
           {
               DBName = dbName,
               DBInstanceIdentifier = dbInstanceIdentifier,
               DBParameterGroupName = parameterGroupName,
               Engine = dbEngine,
               EngineVersion = dbEngineVersion,
```

```
                DBInstanceClass = instanceClass,
                AllocatedStorage = allocatedStorage,
                MasterUsername = adminName,
                MasterUserPassword = adminPassword
            });

        return response.DBInstance;
    }
```

- Pour plus de détails sur l'API, consultez Create DBInstance in AWS SDK pour .NET API Reference.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);

        Aws::RDS::Model::CreateDBInstanceRequest request;
        request.SetDBName(DB_NAME);
        request.SetDBInstanceIdentifier(DB_INSTANCE_IDENTIFIER);
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        request.SetEngine(engineVersion.GetEngine());
        request.SetEngineVersion(engineVersion.GetEngineVersion());
        request.SetDBInstanceClass(dbInstanceClass);
        request.SetStorageType(DB_STORAGE_TYPE);
        request.SetAllocatedStorage(DB_ALLOCATED_STORAGE);
        request.SetMasterUsername(administratorName);
        request.SetMasterUserPassword(administratorPassword);
```

```
        Aws::RDS::Model::CreateDBInstanceOutcome outcome =
                client.CreateDBInstance(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB instance creation has started."
                      << std::endl;
        }
        else {
            std::cerr << "Error with RDS::CreateDBInstance. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            cleanUpResources(PARAMETER_GROUP_NAME, "", client);
            return false;
        }
```

- Pour plus de détails sur l'API, consultez Create DBInstance in AWS SDK pour C++ API Reference.

CLI

AWS CLI

Pour créer une instance de base de données

L'`create-db-instance`exemple suivant utilise les options requises pour lancer une nouvelle instance de base de données.

```
aws rds create-db-instance \
    --db-instance-identifier test-mysql-instance \
    --db-instance-class db.t3.micro \
    --engine mysql \
    --master-username admin \
    --master-user-password secret99 \
    --allocated-storage 20
```

Sortie :

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-mysql-instance",
```

```
        "DBInstanceClass": "db.t3.micro",
        "Engine": "mysql",
        "DBInstanceStatus": "creating",
        "MasterUsername": "admin",
        "AllocatedStorage": 20,
        "PreferredBackupWindow": "12:55-13:25",
        "BackupRetentionPeriod": 1,
        "DBSecurityGroups": [],
        "VpcSecurityGroups": [
            {
                "VpcSecurityGroupId": "sg-12345abc",
                "Status": "active"
            }
        ],
        "DBParameterGroups": [
            {
                "DBParameterGroupName": "default.mysql5.7",
                "ParameterApplyStatus": "in-sync"
            }
        ],
        "DBSubnetGroup": {
            "DBSubnetGroupName": "default",
            "DBSubnetGroupDescription": "default",
            "VpcId": "vpc-2ff2ff2f",
            "SubnetGroupStatus": "Complete",
            "Subnets": [
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2c"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2d"
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2a"
```

```
                    },
                    "SubnetStatus": "Active"
                },
                {
                    "SubnetIdentifier": "subnet-########",
                    "SubnetAvailabilityZone": {
                        "Name": "us-west-2b"
                    },
                    "SubnetStatus": "Active"
                }
            ]
        },
        "PreferredMaintenanceWindow": "sun:08:07-sun:08:37",
        "PendingModifiedValues": {
            "MasterUserPassword": "****"
        },
        "MultiAZ": false,
        "EngineVersion": "5.7.22",
        "AutoMinorVersionUpgrade": true,
        "ReadReplicaDBInstanceIdentifiers": [],
        "LicenseModel": "general-public-license",
        "OptionGroupMemberships": [
            {
                "OptionGroupName": "default:mysql-5-7",
                "Status": "in-sync"
            }
        ],
        "PubliclyAccessible": true,
        "StorageType": "gp2",
        "DbInstancePort": 0,
        "StorageEncrypted": false,
        "DbiResourceId": "db-5555EXAMPLE44444444EXAMPLE",
        "CACertificateIdentifier": "rds-ca-2019",
        "DomainMemberships": [],
        "CopyTagsToSnapshot": false,
        "MonitoringInterval": 0,
        "DBInstanceArn": "arn:aws:rds:us-west-2:123456789012:db:test-mysql-
instance",
        "IAMDatabaseAuthenticationEnabled": false,
        "PerformanceInsightsEnabled": false,
        "DeletionProtection": false,
        "AssociatedRoles": []
    }
```

```
}
```

Pour plus d'informations, consultez la section Création d'une instance de base de données Amazon RDS dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Create DBInstance in AWS CLI Command Reference.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// CreateInstance creates a DB instance.
func (instances *DbInstances) CreateInstance(ctx context.Context, instanceName
 string, dbName string,
 dbEngine string, dbEngineVersion string, parameterGroupName string,
 dbInstanceClass string,
 storageType string, allocatedStorage int32, adminName string, adminPassword
 string) (
 *types.DBInstance, error) {
```

```
 output, err := instances.RdsClient.CreateDBInstance(ctx,
 &rds.CreateDBInstanceInput{
  DBInstanceIdentifier: aws.String(instanceName),
  DBName:               aws.String(dbName),
  DBParameterGroupName: aws.String(parameterGroupName),
  Engine:               aws.String(dbEngine),
  EngineVersion:        aws.String(dbEngineVersion),
  DBInstanceClass:      aws.String(dbInstanceClass),
  StorageType:          aws.String(storageType),
  AllocatedStorage:     aws.Int32(allocatedStorage),
  MasterUsername:       aws.String(adminName),
  MasterUserPassword:   aws.String(adminPassword),
 })
 if err != nil {
  log.Printf("Couldn't create instance %v: %v\n", instanceName, err)
  return nil, err
 } else {
  return output.DBInstance, nil
 }
}
```

- Pour plus de détails sur l'API, consultez Create DBInstance in AWS SDK pour Go API Reference.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import com.google.gson.Gson;
import
 software.amazon.awssdk.auth.credentials.EnvironmentVariableCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
```

```
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesRequest;
import software.amazon.awssdk.services.rds.model.CreateDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.CreateDbInstanceResponse;
import software.amazon.awssdk.services.rds.model.RdsException;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesResponse;
import software.amazon.awssdk.services.rds.model.DBInstance;
import software.amazon.awssdk.services.secretsmanager.SecretsManagerClient;
import
 software.amazon.awssdk.services.secretsmanager.model.GetSecretValueRequest;
import
 software.amazon.awssdk.services.secretsmanager.model.GetSecretValueResponse;

import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 *
 * This example requires an AWS Secrets Manager secret that contains the
 * database credentials. If you do not create a
 * secret, this example will not work. For more details, see:
 *
 * https://docs.aws.amazon.com/secretsmanager/latest/userguide/integrating_how-
services-use-secrets_RS.html
 *
 *
 */

public class CreateDBInstance {
    public static long sleepTime = 20;

    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <dbInstanceIdentifier> <dbName> <secretName>

                Where:
                    dbInstanceIdentifier - The database instance identifier.\s
```

```
                    dbName - The database name.\s
                    secretName - The name of the AWS Secrets Manager secret that
contains the database credentials."
                """;

        if (args.length != 3) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbInstanceIdentifier = args[0];
        String dbName = args[1];
        String secretName = args[2];
        Gson gson = new Gson();
        User user = gson.fromJson(String.valueOf(getSecretValues(secretName)),
User.class);
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        createDatabaseInstance(rdsClient, dbInstanceIdentifier, dbName,
user.getUsername(), user.getPassword());
        waitForInstanceReady(rdsClient, dbInstanceIdentifier);
        rdsClient.close();
    }

    private static SecretsManagerClient getSecretClient() {
        Region region = Region.US_WEST_2;
        return SecretsManagerClient.builder()
                .region(region)

.credentialsProvider(EnvironmentVariableCredentialsProvider.create())
                .build();
    }

    private static String getSecretValues(String secretName) {
        SecretsManagerClient secretClient = getSecretClient();
        GetSecretValueRequest valueRequest = GetSecretValueRequest.builder()
                .secretId(secretName)
                .build();

        GetSecretValueResponse valueResponse =
secretClient.getSecretValue(valueRequest);
```

```
        return valueResponse.secretString();
    }

    public static void createDatabaseInstance(RdsClient rdsClient,
            String dbInstanceIdentifier,
            String dbName,
            String userName,
            String userPassword) {

        try {
            CreateDbInstanceRequest instanceRequest =
CreateDbInstanceRequest.builder()
                .dbInstanceIdentifier(dbInstanceIdentifier)
                .allocatedStorage(100)
                .dbName(dbName)
                .engine("mysql")
                .dbInstanceClass("db.t3.medium") // Updated to a supported class
                .engineVersion("8.0.32")         // Updated to a supported
version
                .storageType("gp2")             // Changed to General Purpose SSD
(gp2)
                .masterUsername(userName)
                .masterUserPassword(userPassword)
                .build();

            CreateDbInstanceResponse response =
rdsClient.createDBInstance(instanceRequest);
            System.out.print("The status is " +
response.dbInstance().dbInstanceStatus());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }

    // Waits until the database instance is available.
    public static void waitForInstanceReady(RdsClient rdsClient, String
dbInstanceIdentifier) {
        boolean instanceReady = false;
        String instanceReadyStr;
        System.out.println("Waiting for instance to become available.");
        try {
```

```
            DescribeDbInstancesRequest instanceRequest =
DescribeDbInstancesRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .build();

            // Loop until the cluster is ready.
            while (!instanceReady) {
                DescribeDbInstancesResponse response =
rdsClient.describeDBInstances(instanceRequest);
                List<DBInstance> instanceList = response.dbInstances();
                for (DBInstance instance : instanceList) {
                    instanceReadyStr = instance.dbInstanceStatus();
                    if (instanceReadyStr.contains("available"))
                        instanceReady = true;
                    else {
                        System.out.print(".");
                        Thread.sleep(sleepTime * 1000);
                    }
                }
            }
            System.out.println("Database instance is available!");

        } catch (RdsException | InterruptedException e) {
            System.err.println(e.getMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, consultez Create DBInstance in AWS SDK for Java 2.x API Reference.

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
suspend fun createDatabaseInstance(
    dbInstanceIdentifierVal: String?,
    dbNamedbVal: String?,
    masterUsernameVal: String?,
    masterUserPasswordVal: String?,
) {
    val instanceRequest =
        CreateDbInstanceRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            allocatedStorage = 100
            dbName = dbNamedbVal
            engine = "mysql"
            dbInstanceClass = "db.t3.micro" // Use a supported instance class
            engineVersion = "8.0.39" // Use a supported engine version
            storageType = "gp2"
            masterUsername = masterUsernameVal
            masterUserPassword = masterUserPasswordVal
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.createDbInstance(instanceRequest)
        print("The status is ${response.dbInstance?.dbInstanceStatus}")
    }
}

// Waits until the database instance is available.
suspend fun waitForInstanceReady(dbInstanceIdentifierVal: String?) {
    val sleepTime: Long = 20
    var instanceReady = false
    var instanceReadyStr: String
    println("Waiting for instance to become available.")

    val instanceRequest =
        DescribeDbInstancesRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        while (!instanceReady) {
            val response = rdsClient.describeDbInstances(instanceRequest)
            val instanceList = response.dbInstances
            if (instanceList != null) {
                for (instance in instanceList) {
```

```
                    instanceReadyStr = instance.dbInstanceStatus.toString()
                    if (instanceReadyStr.contains("available")) {
                        instanceReady = true
                    } else {
                        println("...$instanceReadyStr")
                        delay(sleepTime * 1000)
                    }
                }
            }
        }
        println("Database instance is available!")
    }
}
```

- Pour plus de détails sur l'API, voir Create DBInstance in AWS SDK for Kotlin API reference.

PHP

Kit SDK pour PHP

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require __DIR__ . '/vendor/autoload.php';

use Aws\Exception\AwsException;



$rdsClient = new Aws\Rds\RdsClient([
    'region' => 'us-east-2'
]);

$dbIdentifier = '<<{{db-identifier}}>>';
$dbClass = 'db.t2.micro';
$storage = 5;
```

```
$engine = 'MySQL';
$username = 'MyUser';
$password = 'MyPassword';

try {
    $result = $rdsClient->createDBInstance([
        'DBInstanceIdentifier' => $dbIdentifier,
        'DBInstanceClass' => $dbClass,
        'AllocatedStorage' => $storage,
        'Engine' => $engine,
        'MasterUsername' => $username,
        'MasterUserPassword' => $password,
    ]);
    var_dump($result);
} catch (AwsException $e) {
    echo $e->getMessage();
    echo "\n";
}
```

- Pour plus de détails sur l'API, consultez Create DBInstance in AWS SDK pour PHP API Reference.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
```

```
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def create_db_instance(
        self,
        db_name,
        instance_id,
        parameter_group_name,
        db_engine,
        db_engine_version,
        instance_class,
        storage_type,
        allocated_storage,
        admin_name,
        admin_password,
    ):
        """
        Creates a DB instance.

        :param db_name: The name of the database that is created in the DB
instance.
        :param instance_id: The ID to give the newly created DB instance.
        :param parameter_group_name: A parameter group to associate with the DB
instance.
        :param db_engine: The database engine of a database to create in the DB
instance.
        :param db_engine_version: The engine version for the created database.
        :param instance_class: The DB instance class for the newly created DB
instance.
        :param storage_type: The storage type of the DB instance.
        :param allocated_storage: The amount of storage allocated on the DB
instance, in GiBs.
        :param admin_name: The name of the admin user for the created database.
        :param admin_password: The admin password for the created database.
        :return: Data about the newly created DB instance.
        """
```

```
        try:
            response = self.rds_client.create_db_instance(
                DBName=db_name,
                DBInstanceIdentifier=instance_id,
                DBParameterGroupName=parameter_group_name,
                Engine=db_engine,
                EngineVersion=db_engine_version,
                DBInstanceClass=instance_class,
                StorageType=storage_type,
                AllocatedStorage=allocated_storage,
                MasterUsername=admin_name,
                MasterUserPassword=admin_password,
            )
            db_inst = response["DBInstance"]
        except ClientError as err:
            logger.error(
                "Couldn't create DB instance %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return db_inst
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Create DBInstance in AWS SDK for Python (Boto3).

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS
```

```
   /// Create a new database instance.
   ///
   /// - Parameters:
   ///   - name: The name of the database to create.
   ///   - instanceIdentifier: The identifier to give the new database
   ///     instance.
   ///   - parameterGroupName: The name of the parameter group to associate
   ///     with the new database instance.
   ///   - engine: The database engine to use.
   ///   - engineVersion: The version of the database given by `engine` to
   ///     use.
   ///   - instanceClass: The memory and compute capacity of the database
   ///     instance, such as `db.m5.large``.
   ///   - username: The admin user's username to establish for the new
   ///     instance.
   ///   - password: The password to use for the specified user's access.
   ///
   /// - Returns: A string indicating the ARN of the newly created database
   ///   instance, or nil if the instance couldn't be created.
   func createDBInstance(name: String, instanceIdentifier: String,
parameterGroupName: String,
                         engine: String, engineVersion: String, instanceClass:
String,
                         username: String, password: String) async -> String? {
       do {
           let output = try await rdsClient.createDBInstance(
               input: CreateDBInstanceInput(
                   allocatedStorage: 100,
                   dbInstanceClass: instanceClass,
                   dbInstanceIdentifier: instanceIdentifier,
                   dbName: name,
                   dbParameterGroupName: parameterGroupName,
                   engine: engine,
                   engineVersion: engineVersion,
                   masterUserPassword: password,
                   masterUsername: username,
                   storageType: "gp2"
               )
           )

           guard let dbInstance = output.dbInstance else {
               print("*** Unable to get the database instance.")
               return nil
```

```
            }

            return dbInstance.dbInstanceArn
        } catch {
            print("*** An error occurred while creating the database instance:
\(error.localizedDescription)")
            return nil
        }
    }
```

- Pour plus de détails sur l'API, voir Create DBInstance in AWS SDK for Swift API reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateDBParameterGroup`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `CreateDBParameterGroup`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
```

```
    /// Create a new DB parameter group. Use the action
DescribeDBParameterGroupsAsync
    /// to determine when the DB parameter group is ready to use.
    /// </summary>
    /// <param name="name">Name of the DB parameter group.</param>
    /// <param name="family">Family of the DB parameter group.</param>
    /// <param name="description">Description of the DB parameter group.</param>
    /// <returns>The new DB parameter group.</returns>
    public async Task<DBParameterGroup> CreateDBParameterGroup(
        string name, string family, string description)
    {
        var response = await _amazonRDS.CreateDBParameterGroupAsync(
            new CreateDBParameterGroupRequest()
            {
                DBParameterGroupName = name,
                DBParameterGroupFamily = family,
                Description = description
            });
        return response.DBParameterGroup;
    }
```

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);
```

```
        Aws::RDS::Model::CreateDBParameterGroupRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        request.SetDBParameterGroupFamily(dbParameterGroupFamily);
        request.SetDescription("Example parameter group.");

        Aws::RDS::Model::CreateDBParameterGroupOutcome outcome =
                client.CreateDBParameterGroup(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB parameter group was successfully created."
                      << std::endl;
        }
        else {
            std::cerr << "Error with RDS::CreateDBParameterGroup. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
```

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour créer un groupe de paramètres de base de données

L'`create-db-parameter-group`exemple suivant crée un groupe de paramètres de base de données.

```
aws rds create-db-parameter-group \
    --db-parameter-group-name mydbparametergroup \
    --db-parameter-group-family MySQL5.6 \
    --description "My new parameter group"
```

Sortie :

```
{
```

```
    "DBParameterGroup": {
        "DBParameterGroupName": "mydbparametergroup",
        "DBParameterGroupFamily": "mysql5.6",
        "Description": "My new parameter group",
        "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:mydbparametergroup"
    }
}
```

Pour plus d'informations, consultez la section Création d'un groupe de paramètres de base de données dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}
```

```
// CreateParameterGroup creates a DB parameter group that is based on the
 specified
// parameter group family.
func (instances *DbInstances) CreateParameterGroup(
 ctx context.Context, parameterGroupName string, parameterGroupFamily string,
 description string) (
 *types.DBParameterGroup, error) {

 output, err := instances.RdsClient.CreateDBParameterGroup(ctx,
  &rds.CreateDBParameterGroupInput{
   DBParameterGroupName:   aws.String(parameterGroupName),
   DBParameterGroupFamily: aws.String(parameterGroupFamily),
   Description:            aws.String(description),
  })
 if err != nil {
  log.Printf("Couldn't create parameter group %v: %v\n", parameterGroupName, err)
  return nil, err
 } else {
  return output.DBParameterGroup, err
 }
}
```

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   public static void createDBParameterGroup(RdsClient rdsClient, String
dbGroupName, String dbParameterGroupFamily) {
       try {
```

```
            CreateDbParameterGroupRequest groupRequest =
 CreateDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .dbParameterGroupFamily(dbParameterGroupFamily)
                    .description("Created by using the AWS SDK for Java")
                    .build();

            CreateDbParameterGroupResponse response =
 rdsClient.createDBParameterGroup(groupRequest);
            System.out.println("The group name is " +
 response.dbParameterGroup().dbParameterGroupName());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client
```

```
    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def create_parameter_group(
        self, parameter_group_name, parameter_group_family, description
    ):
        """
        Creates a DB parameter group that is based on the specified parameter
group
        family.

        :param parameter_group_name: The name of the newly created parameter
group.
        :param parameter_group_family: The family that is used as the basis of
the new
                                       parameter group.
        :param description: A description given to the parameter group.
        :return: Data about the newly created parameter group.
        """
        try:
            response = self.rds_client.create_db_parameter_group(
                DBParameterGroupName=parameter_group_name,
                DBParameterGroupFamily=parameter_group_family,
                Description=description,
            )
        except ClientError as err:
            logger.error(
                "Couldn't create parameter group %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return response
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Create DBParameter Group in AWS SDK for Python (Boto3).

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Create a new database parameter group with the specified name.
    ///
    /// - Parameters:
    ///   - groupName: The name of the new parameter group.
    ///   - familyName: The name of the parameter group family.
    /// - Returns:
    func createDBParameterGroup(groupName: String, familyName: String) async ->
 RDSClientTypes.DBParameterGroup? {
        do {
            let output = try await rdsClient.createDBParameterGroup(
                input: CreateDBParameterGroupInput(
                    dbParameterGroupFamily: familyName,
                    dbParameterGroupName: groupName,
                    description: "Created using the AWS SDK for Swift"
                )
            )
            return output.dbParameterGroup
        } catch {
            print("*** Error creating the parameter group:
 \(error.localizedDescription)")
            return nil
        }
    }
```

- Pour plus de détails sur l'API, voir Créer un DBParameter groupe dans le AWS SDK pour une référence sur l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`CreateDBSnapshot`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `CreateDBSnapshot`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Create a snapshot of a DB instance.
   /// </summary>
   /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
   /// <param name="snapshotIdentifier">Identifier for the snapshot.</param>
   /// <returns>DB snapshot object.</returns>
   public async Task<DBSnapshot> CreateDBSnapshot(string dbInstanceIdentifier,
string snapshotIdentifier)
   {
       var response = await _amazonRDS.CreateDBSnapshotAsync(
           new CreateDBSnapshotRequest()
           {
```

```
                DBSnapshotIdentifier = snapshotIdentifier,
                DBInstanceIdentifier = dbInstanceIdentifier
            });

        return response.DBSnapshot;
    }
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK pour .NET API Reference.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);

            Aws::RDS::Model::CreateDBSnapshotRequest request;
            request.SetDBInstanceIdentifier(DB_INSTANCE_IDENTIFIER);
            request.SetDBSnapshotIdentifier(snapshotID);

            Aws::RDS::Model::CreateDBSnapshotOutcome outcome =
                    client.CreateDBSnapshot(request);

            if (outcome.IsSuccess()) {
                std::cout << "Snapshot creation has started."
                          << std::endl;
            }
            else {
                std::cerr << "Error with RDS::CreateDBSnapshot. "
```

```
                        << outcome.GetError().GetMessage()
                        << std::endl;
                cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
client);
                return false;
            }
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK pour C++ API Reference.

CLI

AWS CLI

Pour créer un instantané de base de données

L'`create-db-snapshot`exemple suivant crée un instantané de base de données.

```
aws rds create-db-snapshot \
    --db-instance-identifier database-mysql \
    --db-snapshot-identifier mydbsnapshot
```

Sortie :

```
{
    "DBSnapshot": {
        "DBSnapshotIdentifier": "mydbsnapshot",
        "DBInstanceIdentifier": "database-mysql",
        "Engine": "mysql",
        "AllocatedStorage": 100,
        "Status": "creating",
        "Port": 3306,
        "AvailabilityZone": "us-east-1b",
        "VpcId": "vpc-6594f31c",
        "InstanceCreateTime": "2019-04-30T15:45:53.663Z",
        "MasterUsername": "admin",
        "EngineVersion": "5.6.40",
        "LicenseModel": "general-public-license",
        "SnapshotType": "manual",
        "Iops": 1000,
        "OptionGroupName": "default:mysql-5-6",
```

```
        "PercentProgress": 0,
        "StorageType": "io1",
        "Encrypted": true,
        "KmsKeyId": "arn:aws:kms:us-east-1:123456789012:key/
AKIAIOSFODNN7EXAMPLE",
        "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshot",
        "IAMDatabaseAuthenticationEnabled": false,
        "ProcessorFeatures": [],
        "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
    }
}
```

Pour plus d'informations, consultez la section Création d'un instantané de base de données dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Create DBSnapshot in AWS CLI Command Reference.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}
```

```
// CreateSnapshot creates a snapshot of a DB instance.
func (instances *DbInstances) CreateSnapshot(ctx context.Context, instanceName
 string, snapshotName string) (
 *types.DBSnapshot, error) {
 output, err := instances.RdsClient.CreateDBSnapshot(ctx,
 &rds.CreateDBSnapshotInput{
  DBInstanceIdentifier: aws.String(instanceName),
  DBSnapshotIdentifier: aws.String(snapshotName),
 })
 if err != nil {
  log.Printf("Couldn't create snapshot %v: %v\n", snapshotName, err)
  return nil, err
 } else {
  return output.DBSnapshot, nil
 }
}
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK pour Go API Reference.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   // Create an Amazon RDS snapshot.
   public static void createSnapshot(RdsClient rdsClient, String
dbInstanceIdentifier, String dbSnapshotIdentifier) {
       try {
           CreateDbSnapshotRequest snapshotRequest =
CreateDbSnapshotRequest.builder()
```

```
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .dbSnapshotIdentifier(dbSnapshotIdentifier)
                    .build();

            CreateDbSnapshotResponse response =
rdsClient.createDBSnapshot(snapshotRequest);
            System.out.println("The Snapshot id is " +
response.dbSnapshot().dbiResourceId());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK for Java 2.x API Reference.

PHP

Kit SDK pour PHP

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require __DIR__ . '/vendor/autoload.php';

use Aws\Exception\AwsException;



$rdsClient = new Aws\Rds\RdsClient([
    'region' => 'us-east-2'
]);

$dbIdentifier = '<<{{db-identifier}}>>';
```

```
$snapshotName = '<<{{backup_2018_12_25}}>>';

try {
    $result = $rdsClient->createDBSnapshot([
        'DBInstanceIdentifier' => $dbIdentifier,
        'DBSnapshotIdentifier' => $snapshotName,
    ]);
    var_dump($result);
} catch (AwsException $e) {
    echo $e->getMessage();
    echo "\n";
}
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK pour PHP API Reference.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
```

```
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def create_snapshot(self, snapshot_id, instance_id):
        """
        Creates a snapshot of a DB instance.

        :param snapshot_id: The ID to give the created snapshot.
        :param instance_id: The ID of the DB instance to snapshot.
        :return: Data about the newly created snapshot.
        """
        try:
            response = self.rds_client.create_db_snapshot(
                DBSnapshotIdentifier=snapshot_id,
DBInstanceIdentifier=instance_id
            )
            snapshot = response["DBSnapshot"]
        except ClientError as err:
            logger.error(
                "Couldn't create snapshot of %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return snapshot
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Create DBSnapshot in AWS SDK for Python (Boto3).

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds' # v2: require 'aws-sdk'

# Create a snapshot for an Amazon Relational Database Service (Amazon RDS)
# DB instance.
#
# @param rds_resource [Aws::RDS::Resource] The resource containing SDK logic.
# @param db_instance_name [String] The name of the Amazon RDS DB instance.
# @return [Aws::RDS::DBSnapshot, nil] The snapshot created, or nil if error.
def create_snapshot(rds_resource, db_instance_name)
  id = "snapshot-#{rand(10**6)}"
  db_instance = rds_resource.db_instance(db_instance_name)
  db_instance.create_snapshot({
                                db_snapshot_identifier: id
                              })
rescue Aws::Errors::ServiceError => e
  puts "Couldn't create DB instance snapshot #{id}:\n #{e.message}"
end
```

- Pour plus de détails sur l'API, consultez Create DBSnapshot in AWS SDK pour Ruby API Reference.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Create a snapshot of the specified name.
    ///
    /// - Parameters:
    ///   - instanceIdentifier: The identifier of the database instance to
    ///     snapshot.
    ///   - snapshotIdentifier: A unique identifier to give the newly-created
    ///     snapshot.
    func createDBSnapshot(instanceIdentifier: String, snapshotIdentifier: String)
 async {
        do {
            let output = try await rdsClient.createDBSnapshot(
                input: CreateDBSnapshotInput(
                    dbInstanceIdentifier: instanceIdentifier,
                    dbSnapshotIdentifier: snapshotIdentifier
                )
            )

            guard let snapshot = output.dbSnapshot else {
                print("No snapshot returned.")
                return
            }

            print("The snapshot has been created with ID
\(snapshot.dbiResourceId ?? "<unknown>")")
        } catch {
            print("*** Unable to create the database snapshot named
\(snapshotIdentifier): \(error.localizedDescription)")
        }
    }
```

- Pour plus de détails sur l'API, voir Create DBSnapshot in AWS SDK for Swift API reference.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteDBInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DeleteDBInstance`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
/// <summary>
/// Delete a particular DB instance.
/// </summary>
/// <param name="dbInstanceIdentifier">DB instance identifier.</param>
/// <returns>DB instance object.</returns>
public async Task<DBInstance> DeleteDBInstance(string dbInstanceIdentifier)
{
    var response = await _amazonRDS.DeleteDBInstanceAsync(
        new DeleteDBInstanceRequest()
        {
            DBInstanceIdentifier = dbInstanceIdentifier,
            SkipFinalSnapshot = true,
            DeleteAutomatedBackups = true
        });

    return response.DBInstance;
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);

            Aws::RDS::Model::DeleteDBInstanceRequest request;
            request.SetDBInstanceIdentifier(dbInstanceIdentifier);
            request.SetSkipFinalSnapshot(true);
            request.SetDeleteAutomatedBackups(true);

            Aws::RDS::Model::DeleteDBInstanceOutcome outcome =
                    client.DeleteDBInstance(request);

            if (outcome.IsSuccess()) {
                std::cout << "DB instance deletion has started."
                          << std::endl;
            }
            else {
                std::cerr << "Error with RDS::DeleteDBInstance. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                result = false;
            }
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour supprimer une instance de base de données

L'`delete-db-instance`exemple suivant supprime l'instance de base de données spécifiée après avoir créé un instantané de base de données final nommé`test-instance-final-snap`.

```
aws rds delete-db-instance \
    --db-instance-identifier test-instance \
    --final-db-snapshot-identifier test-instance-final-snap
```

Sortie :

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-instance",
        "DBInstanceStatus": "deleting",
        ...some output truncated...
    }
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS CLI commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
```

```
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// DeleteInstance deletes a DB instance.
func (instances *DbInstances) DeleteInstance(ctx context.Context, instanceName
 string) error {
 _, err := instances.RdsClient.DeleteDBInstance(ctx, &rds.DeleteDBInstanceInput{
  DBInstanceIdentifier:   aws.String(instanceName),
  SkipFinalSnapshot:      aws.Bool(true),
  DeleteAutomatedBackups: aws.Bool(true),
 })
 if err != nil {
  log.Printf("Couldn't delete instance %v: %v\n", instanceName, err)
  return err
 } else {
  return nil
 }
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.DeleteDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.DeleteDbInstanceResponse;
import software.amazon.awssdk.services.rds.model.RdsException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DeleteDBInstance {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <dbInstanceIdentifier>\s

                Where:
                    dbInstanceIdentifier - The database instance identifier\s
                """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbInstanceIdentifier = args[0];
```

```
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        deleteDatabaseInstance(rdsClient, dbInstanceIdentifier);
        rdsClient.close();
    }

    public static void deleteDatabaseInstance(RdsClient rdsClient, String
 dbInstanceIdentifier) {
        try {
            DeleteDbInstanceRequest deleteDbInstanceRequest =
 DeleteDbInstanceRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .deleteAutomatedBackups(true)
                    .skipFinalSnapshot(true)
                    .build();

            DeleteDbInstanceResponse response =
 rdsClient.deleteDBInstance(deleteDbInstanceRequest);
            System.out.print("The status of the database is " +
 response.dbInstance().dbInstanceStatus());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS SDK for Java 2.x API.

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
suspend fun deleteDatabaseInstance(dbInstanceIdentifierVal: String?) {
    val deleteDbInstanceRequest =
        DeleteDbInstanceRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            deleteAutomatedBackups = true
            skipFinalSnapshot = true
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.deleteDbInstance(deleteDbInstanceRequest)
        print("The status of the database is
 ${response.dbInstance?.dbInstanceStatus}")
    }
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le AWS SDK pour la référence de l'API Kotlin.

PHP

Kit SDK pour PHP

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require __DIR__ . '/vendor/autoload.php';

use Aws\Exception\AwsException;


//Create an RDSClient
$rdsClient = new Aws\Rds\RdsClient([
    'region' => 'us-east-1'
]);

$dbIdentifier = '<<{{db-identifier}}>>';

try {
    $result = $rdsClient->deleteDBInstance([
        'DBInstanceIdentifier' => $dbIdentifier,
    ]);
    var_dump($result);
} catch (AwsException $e) {
    echo $e->getMessage();
    echo "\n";
}
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le manuel de référence des AWS SDK pour PHP API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
```

```
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def delete_db_instance(self, instance_id):
        """
        Deletes a DB instance.

        :param instance_id: The ID of the DB instance to delete.
        :return: Data about the deleted DB instance.
        """
        try:
            response = self.rds_client.delete_db_instance(
                DBInstanceIdentifier=instance_id,
                SkipFinalSnapshot=True,
                DeleteAutomatedBackups=True,
            )
            db_inst = response["DBInstance"]
        except ClientError as err:
            logger.error(
                "Couldn't delete DB instance %s. Here's why: %s: %s",
                instance_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return db_inst
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Delete DBInstance in AWS SDK for Python (Boto3).

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Delete the specified database instance.
    ///
    /// - Parameter instanceIdentifier: The identifier of the database
    ///   instance to delete.
    func deleteDBInstance(instanceIdentifier: String) async {
        do {
            _ = try await rdsClient.deleteDBInstance(
                input: DeleteDBInstanceInput(
                    dbInstanceIdentifier: instanceIdentifier,
                    deleteAutomatedBackups: true,
                    skipFinalSnapshot: true
                )
            )
        } catch {
            print("*** Error deleting the database instance
 \(instanceIdentifier): \(error.localizedDescription)")
        }
    }
```

- Pour plus de détails sur l'API, voir Supprimer DBInstance dans le AWS SDK pour une référence à l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DeleteDBParameterGroup`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DeleteDBParameterGroup`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Delete a DB parameter group. The group cannot be a default DB parameter
group
   /// or be associated with any DB instances.
   /// </summary>
   /// <param name="name">Name of the DB parameter group.</param>
   /// <returns>True if successful.</returns>
   public async Task<bool> DeleteDBParameterGroup(string name)
   {
       var response = await _amazonRDS.DeleteDBParameterGroupAsync(
           new DeleteDBParameterGroupRequest()
           {
               DBParameterGroupName = name,
           });
       return response.HttpStatusCode == HttpStatusCode.OK;
   }
```

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    Aws::Client::ClientConfiguration clientConfig;
    // Optional: Set to the AWS Region (overrides config file).
    // clientConfig.region = "us-east-1";

Aws::RDS::RDSClient client(clientConfig);

    Aws::RDS::Model::DeleteDBParameterGroupRequest request;
    request.SetDBParameterGroupName(parameterGroupName);

    Aws::RDS::Model::DeleteDBParameterGroupOutcome outcome =
            client.DeleteDBParameterGroup(request);

    if (outcome.IsSuccess()) {
        std::cout << "The DB parameter group was successfully deleted."
                  << std::endl;
    }
    else {
        std::cerr << "Error with RDS::DeleteDBParameterGroup. "
                  << outcome.GetError().GetMessage()
                  << std::endl;
        result = false;
    }
```

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour supprimer un groupe de paramètres de base de données

L'commandexemple suivant supprime un groupe de paramètres de base de données.

```
aws rds delete-db-parameter-group \
    --db-parameter-group-name mydbparametergroup
```

Cette commande ne produit aucun résultat.

Pour de plus amples informations, veuillez consulter Utilisation des groupes de paramètres de base de données dans le Guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
```

```
}



// DeleteParameterGroup deletes the named DB parameter group.
func (instances *DbInstances) DeleteParameterGroup(ctx context.Context,
 parameterGroupName string) error {
 _, err := instances.RdsClient.DeleteDBParameterGroup(ctx,
  &rds.DeleteDBParameterGroupInput{
   DBParameterGroupName: aws.String(parameterGroupName),
  })
 if err != nil {
  log.Printf("Couldn't delete parameter group %v: %v\n", parameterGroupName, err)
  return err
 } else {
  return nil
 }
}
```

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   // Delete the parameter group after database has been deleted.
   // An exception is thrown if you attempt to delete the para group while
database
   // exists.
   public static void deleteParaGroup(RdsClient rdsClient, String dbGroupName,
String dbARN)
           throws InterruptedException {
```

```
        try {
            boolean isDataDel = false;
            boolean didFind;
            String instanceARN;

            // Make sure that the database has been deleted.
            while (!isDataDel) {
                DescribeDbInstancesResponse response =
rdsClient.describeDBInstances();
                List<DBInstance> instanceList = response.dbInstances();
                int listSize = instanceList.size();
                didFind = false;
                int index = 1;
                for (DBInstance instance : instanceList) {
                    instanceARN = instance.dbInstanceArn();
                    if (instanceARN.compareTo(dbARN) == 0) {
                        System.out.println(dbARN + " still exists");
                        didFind = true;
                    }
                    if ((index == listSize) && (!didFind)) {
                        // Went through the entire list and did not find the
database ARN.
                        isDataDel = true;
                    }
                    Thread.sleep(sleepTime * 1000);
                    index++;
                }
            }

            // Delete the para group.
            DeleteDbParameterGroupRequest parameterGroupRequest =
DeleteDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .build();

            rdsClient.deleteDBParameterGroup(parameterGroupRequest);
            System.out.println(dbGroupName + " was deleted.");

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def delete_parameter_group(self, parameter_group_name):
        """
        Deletes a DB parameter group.

        :param parameter_group_name: The name of the parameter group to delete.
        :return: Data about the parameter group.
        """
        try:
```

```
            self.rds_client.delete_db_parameter_group(
                DBParameterGroupName=parameter_group_name
            )
        except ClientError as err:
            logger.error(
                "Couldn't delete parameter group %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Delete DBParameter Group in AWS SDK for Python (Boto3).

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Delete the specified database parameter group.
    ///
    /// - Parameter groupName: The name of the parameter group to delete.
    func deleteDBParameterGroup(groupName: String) async {
        do {
            _ = try await rdsClient.deleteDBParameterGroup(
                input: DeleteDBParameterGroupInput(
                    dbParameterGroupName: groupName
                )
            )
        } catch {
```

```
            print("*** Error deleting the database parameter group \(groupName):
 \(error.localizedDescription)")
        }
    }
```

- Pour plus de détails sur l'API, voir Supprimer le DBParameter groupe dans le AWS SDK pour une référence à l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeAccountAttributes`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeAccountAttributes`.

CLI

AWS CLI

Pour décrire les attributs du compte

L'`describe-account-attributes`exemple suivant récupère les attributs du AWS compte courant.

```
aws rds describe-account-attributes
```

Sortie :

```
{
    "AccountQuotas": [
        {
            "Max": 40,
            "Used": 4,
            "AccountQuotaName": "DBInstances"
        },
        {
            "Max": 40,
            "Used": 0,
            "AccountQuotaName": "ReservedDBInstances"
        },
```

```
        {
            "Max": 100000,
            "Used": 40,
            "AccountQuotaName": "AllocatedStorage"
        },
        {
            "Max": 25,
            "Used": 0,
            "AccountQuotaName": "DBSecurityGroups"
        },
        {
            "Max": 20,
            "Used": 0,
            "AccountQuotaName": "AuthorizationsPerDBSecurityGroup"
        },
        {
            "Max": 50,
            "Used": 1,
            "AccountQuotaName": "DBParameterGroups"
        },
        {
            "Max": 100,
            "Used": 3,
            "AccountQuotaName": "ManualSnapshots"
        },
        {
            "Max": 20,
            "Used": 0,
            "AccountQuotaName": "EventSubscriptions"
        },
        {
            "Max": 50,
            "Used": 1,
            "AccountQuotaName": "DBSubnetGroups"
        },
        {
            "Max": 20,
            "Used": 1,
            "AccountQuotaName": "OptionGroups"
        },
        {
            "Max": 20,
            "Used": 6,
            "AccountQuotaName": "SubnetsPerDBSubnetGroup"
```

```
        },
        {
            "Max": 5,
            "Used": 0,
            "AccountQuotaName": "ReadReplicasPerMaster"
        },
        {
            "Max": 40,
            "Used": 1,
            "AccountQuotaName": "DBClusters"
        },
        {
            "Max": 50,
            "Used": 0,
            "AccountQuotaName": "DBClusterParameterGroups"
        },
        {
            "Max": 5,
            "Used": 0,
            "AccountQuotaName": "DBClusterRoles"
        }
    ]
}
```

- Pour plus de détails sur l'API, reportez-vous DescribeAccountAttributesà la section Référence des AWS CLI commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.AccountQuota;
import software.amazon.awssdk.services.rds.model.RdsException;
```

```
import
 software.amazon.awssdk.services.rds.model.DescribeAccountAttributesResponse;
import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DescribeAccountAttributes {
    public static void main(String[] args) {
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        getAccountAttributes(rdsClient);
        rdsClient.close();
    }

    public static void getAccountAttributes(RdsClient rdsClient) {
        try {
            DescribeAccountAttributesResponse response =
 rdsClient.describeAccountAttributes();
            List<AccountQuota> quotasList = response.accountQuotas();
            for (AccountQuota quotas : quotasList) {
                System.out.println("Name is: " + quotas.accountQuotaName());
                System.out.println("Max value is " + quotas.max());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir DescribeAccountAttributesla section Référence des AWS SDK for Java 2.x API.

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
suspend fun getAccountAttributes() {
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response =
 rdsClient.describeAccountAttributes(DescribeAccountAttributesRequest {})
        response.accountQuotas?.forEach { quotas ->
            val response = response.accountQuotas
            println("Name is: ${quotas.accountQuotaName}")
            println("Max value is ${quotas.max}")
        }
    }
}
```

- Pour plus de détails sur l'API, consultez DescribeAccountAttributesla section AWS SDK pour la référence de l'API Kotlin.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDBEngineVersions`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeDBEngineVersions`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
    /// Get a list of DB engine versions for a particular DB engine.
    /// </summary>
    /// <param name="engine">Name of the engine.</param>
    /// <param name="dbParameterGroupFamily">Optional parameter group family
name.</param>
    /// <returns>List of DBEngineVersions.</returns>
    public async Task<List<DBEngineVersion>> DescribeDBEngineVersions(string
engine,
        string dbParameterGroupFamily = null)
    {
        var response = await _amazonRDS.DescribeDBEngineVersionsAsync(
            new DescribeDBEngineVersionsRequest()
            {
                Engine = engine,
                DBParameterGroupFamily = dbParameterGroupFamily
            });
        return response.DBEngineVersions;
    }
```

- Pour plus de détails sur l'API, consultez la section Description DBEngine des versions dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);


//! Routine which gets available DB engine versions for an engine name and
//! an optional parameter group family.
/*!
 \sa getDBEngineVersions()
 \param engineName: A DB engine name.
 \param parameterGroupFamily: A parameter group family name, ignored if empty.
 \param engineVersionsResult: Vector of 'DBEngineVersion' objects returned by the
 routine.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::getDBEngineVersions(const Aws::String &engineName,
                                      const Aws::String &parameterGroupFamily,

 Aws::Vector<Aws::RDS::Model::DBEngineVersion> &engineVersionsResult,
                                      const Aws::RDS::RDSClient &client) {
    Aws::RDS::Model::DescribeDBEngineVersionsRequest request;
    request.SetEngine(engineName);
    if (!parameterGroupFamily.empty()) {
        request.SetDBParameterGroupFamily(parameterGroupFamily);
    }

    engineVersionsResult.clear();
    Aws::String marker; // Used for pagination.
```

```
    do {
        if (!marker.empty()) {
            request.SetMarker(marker);
        }


        Aws::RDS::Model::DescribeDBEngineVersionsOutcome outcome =
                client.DescribeDBEngineVersions(request);

        if (outcome.IsSuccess()) {
            auto &engineVersions = outcome.GetResult().GetDBEngineVersions();
            engineVersionsResult.insert(engineVersionsResult.end(),
 engineVersions.begin(),
                                        engineVersions.end());
            marker = outcome.GetResult().GetMarker();
        }
        else {
            std::cerr << "Error with RDS::DescribeDBEngineVersionsRequest. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }

    } while (!marker.empty());


    return true;
}
```

- Pour plus de détails sur l'API, consultez la section Description DBEngine des versions dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour décrire les versions du moteur de base de données pour le moteur de base de données MySQL

L'`describe-db-engine-versions`exemple suivant affiche des détails sur chacune des versions du moteur de base de données pour le moteur de base de données spécifié.

```
aws rds describe-db-engine-versions \
    --engine mysql
```

Sortie :

```
{
    "DBEngineVersions": [
        {
            "Engine": "mysql",
            "EngineVersion": "5.5.46",
            "DBParameterGroupFamily": "mysql5.5",
            "DBEngineDescription": "MySQL Community Edition",
            "DBEngineVersionDescription": "MySQL 5.5.46",
            "ValidUpgradeTarget": [
                {
                    "Engine": "mysql",
                    "EngineVersion": "5.5.53",
                    "Description": "MySQL 5.5.53",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                {
                    "Engine": "mysql",
                    "EngineVersion": "5.5.54",
                    "Description": "MySQL 5.5.54",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                {
                    "Engine": "mysql",
                    "EngineVersion": "5.5.57",
                    "Description": "MySQL 5.5.57",
                    "AutoUpgrade": false,
                    "IsMajorVersionUpgrade": false
                },
                ...some output truncated...
            ]
        }
```

Pour plus d'informations, consultez Qu'est-ce qu'Amazon Relational Database Service (Amazon RDS) ? dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Décrire DBEngine les versions dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// GetEngineVersions gets database engine versions that are available for the
 specified engine
// and parameter group family.
func (instances *DbInstances) GetEngineVersions(ctx context.Context, engine
 string, parameterGroupFamily string) (
 []types.DBEngineVersion, error) {
 output, err := instances.RdsClient.DescribeDBEngineVersions(ctx,
  &rds.DescribeDBEngineVersionsInput{
   Engine:                 aws.String(engine),
   DBParameterGroupFamily: aws.String(parameterGroupFamily),
  })
```

```
 if err != nil {
  log.Printf("Couldn't get engine versions for %v: %v\n", engine, err)
  return nil, err
 } else {
  return output.DBEngineVersions, nil
 }
}
```

- Pour plus de détails sur l'API, consultez la section Description DBEngine des versions dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    public static void describeDBEngines(RdsClient rdsClient) {
        try {
            DescribeDbEngineVersionsRequest engineVersionsRequest =
DescribeDbEngineVersionsRequest.builder()
                    .defaultOnly(true)
                    .engine("mysql")
                    .maxRecords(20)
                    .build();

            DescribeDbEngineVersionsResponse response =
rdsClient.describeDBEngineVersions(engineVersionsRequest);
            List<DBEngineVersion> engines = response.dbEngineVersions();

            // Get all DBEngineVersion objects.
            for (DBEngineVersion engineOb : engines) {
                System.out.println("The name of the DB parameter group family for
the database engine is "
                        + engineOb.dbParameterGroupFamily());
```

```
                System.out.println("The name of the database engine " +
engineOb.engine());
                System.out.println("The version number of the database engine " +
engineOb.engineVersion());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, consultez la section Description DBEngine des versions dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
```

```
        return cls(rds_client)


    def get_engine_versions(self, engine, parameter_group_family=None):
        """
        Gets database engine versions that are available for the specified engine
        and parameter group family.

        :param engine: The database engine to look up.
        :param parameter_group_family: When specified, restricts the returned
list of
                                       engine versions to those that are
compatible with
                                       this parameter group family.
        :return: The list of database engine versions.
        """
        try:
            kwargs = {"Engine": engine}
            if parameter_group_family is not None:
                kwargs["DBParameterGroupFamily"] = parameter_group_family
            response = self.rds_client.describe_db_engine_versions(**kwargs)
            versions = response["DBEngineVersions"]
        except ClientError as err:
            logger.error(
                "Couldn't get engine versions for %s. Here's why: %s: %s",
                engine,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return versions
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Describe DBEngine Versions in AWS SDK for Python (Boto3).

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Get all the database engine versions available for the specified
    /// database engine.
    ///
    /// - Parameter engineName: The name of the database engine to query.
    ///
    /// - Returns: An array of `RDSClientTypes.DBEngineVersion` structures,
    ///   each describing one supported version of the specified database.
    func getDBEngineVersions(engineName: String) async ->
 [RDSClientTypes.DBEngineVersion] {
        do {
            let output = try await rdsClient.describeDBEngineVersions(
                input: DescribeDBEngineVersionsInput(
                    engine: engineName
                )
            )

            return output.dbEngineVersions ?? []
        } catch {
            return []
        }
    }
```

- Pour plus de détails sur l'API, voir Décrire DBEngine les versions dans le AWS SDK pour une référence à l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDBInstances`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeDBInstances`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Returns a list of DB instances.
   /// </summary>
   /// <param name="dbInstanceIdentifier">Optional name of a specific DB
instance.</param>
   /// <returns>List of DB instances.</returns>
   public async Task<List<DBInstance>> DescribeDBInstances(string
dbInstanceIdentifier = null)
   {
       var results = new List<DBInstance>();
       var instancesPaginator = _amazonRDS.Paginators.DescribeDBInstances(
           new DescribeDBInstancesRequest
           {
               DBInstanceIdentifier = dbInstanceIdentifier
           });
       // Get the entire list using the paginator.
       await foreach (var instances in instancesPaginator.DBInstances)
       {
           results.Add(instances);
       }
```

```
        return results;
    }
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);


//! Routine which gets a DB instance description.
/*!
 \sa describeDBInstance()
 \param dbInstanceIdentifier: A DB instance identifier.
 \param instanceResult: The 'DBInstance' object containing the description.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::describeDBInstance(const Aws::String &dbInstanceIdentifier,
                                     Aws::RDS::Model::DBInstance &instanceResult,
                                     const Aws::RDS::RDSClient &client) {
    Aws::RDS::Model::DescribeDBInstancesRequest request;
    request.SetDBInstanceIdentifier(dbInstanceIdentifier);

    Aws::RDS::Model::DescribeDBInstancesOutcome outcome =
            client.DescribeDBInstances(request);
```

```
    bool result = true;
    if (outcome.IsSuccess()) {
        instanceResult = outcome.GetResult().GetDBInstances()[0];
    }
    else if (outcome.GetError().GetErrorType() !=
             Aws::RDS::RDSErrors::D_B_INSTANCE_NOT_FOUND_FAULT) {
        result = false;
        std::cerr << "Error with RDS::DescribeDBInstances. "
                  << outcome.GetError().GetMessage()
                  << std::endl;
    }
        // This example does not log an error if the DB instance does not exist.
        // Instead, instanceResult is set to empty.
    else {
        instanceResult = Aws::RDS::Model::DBInstance();
    }

    return result;
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour décrire une instance de base de données

L'`describe-db-instances`exemple suivant récupère les détails de l'instance de base de données spécifiée.

```
aws rds describe-db-instances \
    --db-instance-identifier mydbinstancecf
```

Sortie :

```
{
    "DBInstances": [
        {
```

```
            "DBInstanceIdentifier": "mydbinstancecf",
            "DBInstanceClass": "db.t3.small",
            "Engine": "mysql",
            "DBInstanceStatus": "available",
            "MasterUsername": "masterawsuser",
            "Endpoint": {
                "Address": "mydbinstancecf.abcexample.us-
east-1.rds.amazonaws.com",
                "Port": 3306,
                "HostedZoneId": "Z2R2ITUGPM61AM"
            },
            ...some output truncated...
        }
    ]
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS CLI commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
```

```
 RdsClient *rds.Client
}



// GetInstance gets data about a DB instance.
func (instances *DbInstances) GetInstance(ctx context.Context, instanceName
 string) (
 *types.DBInstance, error) {
 output, err := instances.RdsClient.DescribeDBInstances(ctx,
  &rds.DescribeDBInstancesInput{
   DBInstanceIdentifier: aws.String(instanceName),
  })
 if err != nil {
  var notFoundError *types.DBInstanceNotFoundFault
  if errors.As(err, &notFoundError) {
   log.Printf("DB instance %v does not exist.\n", instanceName)
   err = nil
  } else {
   log.Printf("Couldn't get instance %v: %v\n", instanceName, err)
  }
  return nil, err
 } else {
  return &output.DBInstances[0], nil
 }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.DescribeDbInstancesResponse;
import software.amazon.awssdk.services.rds.model.DBInstance;
import software.amazon.awssdk.services.rds.model.RdsException;
import java.util.List;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class DescribeDBInstances {

    public static void main(String[] args) {
        Region region = Region.US_EAST_1;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        describeInstances(rdsClient);
        rdsClient.close();
    }

    public static void describeInstances(RdsClient rdsClient) {
        try {
            DescribeDbInstancesResponse response =
 rdsClient.describeDBInstances();
            List<DBInstance> instanceList = response.dbInstances();
            for (DBInstance instance : instanceList) {
                System.out.println("Instance ARN is: " +
 instance.dbInstanceArn());
                System.out.println("The Engine is " + instance.engine());
                System.out.println("Connection endpoint is" +
 instance.endpoint().address());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
```

```
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK for Java 2.x API.

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
suspend fun describeInstances() {
    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val response = rdsClient.describeDbInstances(DescribeDbInstancesRequest
 {})
        response.dbInstances?.forEach { instance ->
            println("Instance Identifier is ${instance.dbInstanceIdentifier}")
            println("The Engine is ${instance.engine}")
            println("Connection endpoint is ${instance.endpoint?.address}")
        }
    }
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le AWS SDK pour la référence de l'API Kotlin.

PHP

Kit SDK pour PHP

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require __DIR__ . '/vendor/autoload.php';

use Aws\Exception\AwsException;


//Create an RDSClient
$rdsClient = new Aws\Rds\RdsClient([
    'region' => 'us-east-2'
]);

try {
    $result = $rdsClient->describeDBInstances();
    foreach ($result['DBInstances'] as $instance) {
        print('<p>DB Identifier: ' . $instance['DBInstanceIdentifier']);
        print('<br />Endpoint: ' . $instance['Endpoint']["Address"]
            . ':' . $instance['Endpoint']["Port"]);
        print('<br />Current Status: ' . $instance["DBInstanceStatus"]);
        print('</p>');
    }
    print(" Raw Result ");
    var_dump($result);
} catch (AwsException $e) {
    echo $e->getMessage();
    echo "\n";
}
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour PHP API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_db_instance(self, instance_id):
        """
        Gets data about a DB instance.

        :param instance_id: The ID of the DB instance to retrieve.
        :return: The retrieved DB instance.
        """
        try:
            response = self.rds_client.describe_db_instances(
                DBInstanceIdentifier=instance_id
            )
            db_inst = response["DBInstances"][0]
        except ClientError as err:
            if err.response["Error"]["Code"] == "DBInstanceNotFound":
```

```
                logger.info("Instance %s does not exist.", instance_id)
            else:
                logger.error(
                    "Couldn't get DB instance %s. Here's why: %s: %s",
                    instance_id,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return db_inst
```

- Pour plus de détails sur l'API, voir AWS Describe DBInstances in SDK for Python (Boto3) API Reference.

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds' # v2: require 'aws-sdk'

# List all Amazon Relational Database Service (Amazon RDS) DB instances.
#
# @param rds_resource [Aws::RDS::Resource] An SDK for Ruby Amazon RDS resource.
# @return [Array, nil] List of all DB instances, or nil if error.
def list_instances(rds_resource)
  db_instances = []
  rds_resource.db_instances.each do |i|
    db_instances.append({
                          "name": i.id,
                          "status": i.db_instance_status
                        })
  end
```

```
  db_instances
rescue Aws::Errors::ServiceError => e
  puts "Couldn't list instances:\n#{e.message}"
end
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le manuel de référence des AWS SDK pour Ruby API.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Wait until the specified database is available to use.
    ///
    /// - Parameter instanceIdentifier: The database instance identifier of the
    ///   database to wait for.
    func waitUntilDBInstanceReady(instanceIdentifier: String) async ->
 RDSClientTypes.Endpoint? {
        do {
            putString("Waiting for the database instance to be ready to use. This
 may take 10 minutes or more...")
            while true {
                let output = try await rdsClient.describeDBInstances(
                    input: DescribeDBInstancesInput(
                        dbInstanceIdentifier: instanceIdentifier
                    )
                )

                guard let instanceList = output.dbInstances else {
                    continue
                }
```

```
                for instance in instanceList {
                    let status = instance.dbInstanceStatus

                    guard let status else {
                        print("\nUnable to determine the status.")
                        continue
                    }

                    if status.contains("available") {
                        return instance.endpoint
                    } else {
                        putString(".")
                        do {
                            try await Task.sleep(for: .seconds(15))
                        } catch {
                            print("*** Error pausing the task!")
                        }
                    }
                }
            }
        } catch {
            print("*** Unable to wait until the database is ready:
\(error.localizedDescription)")
            return nil
        }
    }
```

- Pour plus de détails sur l'API, voir Description DBInstances dans le AWS SDK pour la référence de l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDBParameterGroups`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeDBParameterGroups`.

Les exemples d’actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l’exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
    /// Get descriptions of DB parameter groups.
    /// </summary>
    /// <param name="name">Optional name of the DB parameter group to describe.</
param>
    /// <returns>The list of DB parameter group descriptions.</returns>
    public async Task<List<DBParameterGroup>> DescribeDBParameterGroups(string
 name = null)
    {
        var response = await _amazonRDS.DescribeDBParameterGroupsAsync(
            new DescribeDBParameterGroupsRequest()
            {
                DBParameterGroupName = name
            });
        return response.DBParameterGroups;
    }
```

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans le AWS SDK pour .NET manuel de référence des API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);

        Aws::RDS::Model::DescribeDBParameterGroupsRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);

        Aws::RDS::Model::DescribeDBParameterGroupsOutcome outcome =
                client.DescribeDBParameterGroups(request);

        if (outcome.IsSuccess()) {
            std::cout << "DB parameter group named '" <<
                      PARAMETER_GROUP_NAME << "' already exists." << std::endl;
            dbParameterGroupFamily = outcome.GetResult().GetDBParameterGroups()
[0].GetDBParameterGroupFamily();
        }

        else {
            std::cerr << "Error with RDS::DescribeDBParameterGroups. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
```

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans le AWS SDK pour C++ manuel de référence des API.

CLI

AWS CLI

Pour décrire votre groupe de paramètres de base de données

L'`describe-db-parameter-groups`exemple suivant permet de récupérer des informations sur vos groupes de paramètres de base de données.

```
aws rds describe-db-parameter-groups
```

Sortie :

```
{
    "DBParameterGroups": [
        {
            "DBParameterGroupName": "default.aurora-mysql5.7",
            "DBParameterGroupFamily": "aurora-mysql5.7",
            "Description": "Default parameter group for aurora-mysql5.7",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-mysql5.7"
        },
        {
            "DBParameterGroupName": "default.aurora-postgresql9.6",
            "DBParameterGroupFamily": "aurora-postgresql9.6",
            "Description": "Default parameter group for aurora-postgresql9.6",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora-postgresql9.6"
        },
        {
            "DBParameterGroupName": "default.aurora5.6",
            "DBParameterGroupFamily": "aurora5.6",
            "Description": "Default parameter group for aurora5.6",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.aurora5.6"
        },
        {
            "DBParameterGroupName": "default.mariadb10.1",
            "DBParameterGroupFamily": "mariadb10.1",
            "Description": "Default parameter group for mariadb10.1",
            "DBParameterGroupArn": "arn:aws:rds:us-
east-1:123456789012:pg:default.mariadb10.1"
        },
```

```
        ...some output truncated...
    ]
}
```

Pour de plus amples informations, veuillez consulter Utilisation des groupes de paramètres de base de données dans le Guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// GetParameterGroup gets a DB parameter group by name.
func (instances *DbInstances) GetParameterGroup(ctx context.Context,
 parameterGroupName string) (
 *types.DBParameterGroup, error) {
 output, err := instances.RdsClient.DescribeDBParameterGroups(
```

```
  ctx, &rds.DescribeDBParameterGroupsInput{
   DBParameterGroupName: aws.String(parameterGroupName),
  })
 if err != nil {
  var notFoundError *types.DBParameterGroupNotFoundFault
  if errors.As(err, &notFoundError) {
   log.Printf("Parameter group %v does not exist.\n", parameterGroupName)
   err = nil
  } else {
   log.Printf("Error getting parameter group %v: %v\n", parameterGroupName, err)
  }
  return nil, err
 } else {
  return &output.DBParameterGroups[0], err
 }
}
```

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans le AWS SDK pour Go manuel de référence des API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   public static void describeDbParameterGroups(RdsClient rdsClient, String
dbGroupName) {
       try {
           DescribeDbParameterGroupsRequest groupsRequest =
DescribeDbParameterGroupsRequest.builder()
                   .dbParameterGroupName(dbGroupName)
                   .maxRecords(20)
                   .build();
```

```
            DescribeDbParameterGroupsResponse response =
 rdsClient.describeDBParameterGroups(groupsRequest);
            List<DBParameterGroup> groups = response.dbParameterGroups();
            for (DBParameterGroup group : groups) {
                System.out.println("The group name is " +
 group.dbParameterGroupName());
                System.out.println("The group description is " +
 group.description());
            }

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans le AWS SDK for Java 2.x manuel de référence des API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
```

```
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_parameter_group(self, parameter_group_name):
        """
        Gets a DB parameter group.

        :param parameter_group_name: The name of the parameter group to retrieve.
        :return: The parameter group.
        """
        try:
            response = self.rds_client.describe_db_parameter_groups(
                DBParameterGroupName=parameter_group_name
            )
            parameter_group = response["DBParameterGroups"][0]
        except ClientError as err:
            if err.response["Error"]["Code"] == "DBParameterGroupNotFound":
                logger.info("Parameter group %s does not exist.",
parameter_group_name)
            else:
                logger.error(
                    "Couldn't get parameter group %s. Here's why: %s: %s",
                    parameter_group_name,
                    err.response["Error"]["Code"],
                    err.response["Error"]["Message"],
                )
                raise
        else:
            return parameter_group
```

- Pour plus de détails sur l'API, consultez le manuel de référence de l'API Describe DBParameter Groups in AWS SDK for Python (Boto3).

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds' # v2: require 'aws-sdk'

# List all Amazon Relational Database Service (Amazon RDS) parameter groups.
#
# @param rds_resource [Aws::RDS::Resource] An SDK for Ruby Amazon RDS resource.
# @return [Array, nil] List of all parameter groups, or nil if error.
def list_parameter_groups(rds_resource)
  parameter_groups = []
  rds_resource.db_parameter_groups.each do |p|
    parameter_groups.append({
                              "name": p.db_parameter_group_name,
                              "description": p.description
                            })
  end
  parameter_groups
rescue Aws::Errors::ServiceError => e
  puts "Couldn't list parameter groups:\n #{e.message}"
end
```

- Pour plus de détails sur l'API, consultez la section Décrire DBParameter les groupes dans le AWS SDK pour Ruby manuel de référence des API.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Get descriptions of the database parameter groups matching the given
    /// name.
    ///
    /// - Parameter groupName: The name of the parameter group to describe.
    ///
    /// - Returns: An array of [RDSClientTypes.DBParameterGroup] objects
    ///   describing the parameter group.
    func describeDBParameterGroups(groupName: String) async ->
 [RDSClientTypes.DBParameterGroup]? {
        do {
            let output = try await rdsClient.describeDBParameterGroups(
                input: DescribeDBParameterGroupsInput(
                    dbParameterGroupName: groupName
                )
            )
            return output.dbParameterGroups
        } catch {
            print("*** Error getting the database parameter group's details:
 \(error.localizedDescription)")
            return nil
        }
    }
```

- Pour plus de détails sur l'API, consultez la section Description DBParameter des groupes dans le AWS SDK pour une référence sur l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDBParameters`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeDBParameters`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
   /// <summary>
   /// Get a list of DB parameters from a specific parameter group.
   /// </summary>
   /// <param name="dbParameterGroupName">Name of a specific DB parameter
group.</param>
   /// <param name="source">Optional source for selecting parameters.</param>
   /// <returns>List of parameter values.</returns>
   public async Task<List<Parameter>> DescribeDBParameters(string
dbParameterGroupName, string source = null)
   {
       var results = new List<Parameter>();
       var paginateParameters = _amazonRDS.Paginators.DescribeDBParameters(
           new DescribeDBParametersRequest()
           {
               DBParameterGroupName = dbParameterGroupName,
               Source = source
```

```
            });
        // Get the entire list using the paginator.
        await foreach (var parameters in paginateParameters.Parameters)
        {
            results.Add(parameters);
        }
        return results;
    }
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);


//! Routine which gets DB parameters using the 'DescribeDBParameters' api.
/*!
 \sa getDBParameters()
 \param parameterGroupName: The name of the parameter group.
 \param namePrefix: Prefix string to filter results by parameter name.
 \param source: A source such as 'user', ignored if empty.
 \param parametersResult: Vector of 'Parameter' objects returned by the routine.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
```

```
bool AwsDoc::RDS::getDBParameters(const Aws::String &parameterGroupName,
                                  const Aws::String &namePrefix,
                                  const Aws::String &source,
                                  Aws::Vector<Aws::RDS::Model::Parameter>
 &parametersResult,
                                  const Aws::RDS::RDSClient &client) {
   Aws::String marker;
   do {
       Aws::RDS::Model::DescribeDBParametersRequest request;
       request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
       if (!marker.empty()) {
           request.SetMarker(marker);
       }
       if (!source.empty()) {
           request.SetSource(source);
       }

       Aws::RDS::Model::DescribeDBParametersOutcome outcome =
               client.DescribeDBParameters(request);

       if (outcome.IsSuccess()) {
           const Aws::Vector<Aws::RDS::Model::Parameter> &parameters =
                   outcome.GetResult().GetParameters();
           for (const Aws::RDS::Model::Parameter &parameter: parameters) {
               if (!namePrefix.empty()) {
                   if (parameter.GetParameterName().find(namePrefix) == 0) {
                       parametersResult.push_back(parameter);
                   }
               }
               else {
                   parametersResult.push_back(parameter);
               }
           }

           marker = outcome.GetResult().GetMarker();
       }
       else {
           std::cerr << "Error with RDS::DescribeDBParameters. "
                     << outcome.GetError().GetMessage()
                     << std::endl;
           return false;
       }
   } while (!marker.empty());
```

```
    return true;
}
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour décrire les paramètres d'un groupe de paramètres de base de données

L'`describe-db-parameters`exemple suivant récupère les détails du groupe de paramètres de base de données spécifié.

```
aws rds describe-db-parameters \
    --db-parameter-group-name mydbpg
```

Sortie :

```
{
    "Parameters": [
        {
            "ParameterName": "allow-suspicious-udfs",
            "Description": "Controls whether user-defined functions that have
 only an xxx symbol for the main function can be loaded",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot"
        },
        {
            "ParameterName": "auto_generate_certs",
            "Description": "Controls whether the server autogenerates SSL key and
 certificate files in the data directory, if they do not already exist.",
            "Source": "engine-default",
            "ApplyType": "static",
            "DataType": "boolean",
```

```
            "AllowedValues": "0,1",
            "IsModifiable": false,
            "ApplyMethod": "pending-reboot"
        },
        ...some output truncated...
    ]
}
```

Pour de plus amples informations, veuillez consulter Utilisation des groupes de paramètres de base de données dans le Guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS CLI commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}
```

```
// GetParameters gets the parameters that are contained in a DB parameter group.
func (instances *DbInstances) GetParameters(ctx context.Context,
 parameterGroupName string, source string) (
 []types.Parameter, error) {

 var output *rds.DescribeDBParametersOutput
 var params []types.Parameter
 var err error
 parameterPaginator := rds.NewDescribeDBParametersPaginator(instances.RdsClient,
  &rds.DescribeDBParametersInput{
   DBParameterGroupName: aws.String(parameterGroupName),
   Source:               aws.String(source),
  })
 for parameterPaginator.HasMorePages() {
  output, err = parameterPaginator.NextPage(ctx)
  if err != nil {
   log.Printf("Couldn't get parameters for %v: %v\n", parameterGroupName, err)
   break
  } else {
   params = append(params, output.Parameters...)
  }
 }
 return params, err
}
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    // Retrieve parameters in the group.
```

```
   public static void describeDbParameters(RdsClient rdsClient, String
dbGroupName, int flag) {
       try {
           DescribeDbParametersRequest dbParameterGroupsRequest;
           if (flag == 0) {
               dbParameterGroupsRequest = DescribeDbParametersRequest.builder()
                       .dbParameterGroupName(dbGroupName)
                       .build();
           } else {
               dbParameterGroupsRequest = DescribeDbParametersRequest.builder()
                       .dbParameterGroupName(dbGroupName)
                       .source("user")
                       .build();
           }

           DescribeDbParametersResponse response =
rdsClient.describeDBParameters(dbParameterGroupsRequest);
           List<Parameter> dbParameters = response.parameters();
           String paraName;
           for (Parameter para : dbParameters) {
               // Only print out information about either auto_increment_offset
or
               // auto_increment_increment.
               paraName = para.parameterName();
               if ((paraName.compareTo("auto_increment_offset") == 0)
                       || (paraName.compareTo("auto_increment_increment ") ==
0)) {
                   System.out.println("*** The parameter name is  " + paraName);
                   System.out.println("*** The parameter value is  " +
para.parameterValue());
                   System.out.println("*** The parameter data type is " +
para.dataType());
                   System.out.println("*** The parameter description is " +
para.description());
                   System.out.println("*** The parameter allowed values  is " +
para.allowedValues());
               }
           }

       } catch (RdsException e) {
           System.out.println(e.getLocalizedMessage());
           System.exit(1);
       }
   }
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_parameters(self, parameter_group_name, name_prefix="", source=None):
        """
        Gets the parameters that are contained in a DB parameter group.

        :param parameter_group_name: The name of the parameter group to query.
        :param name_prefix: When specified, the retrieved list of parameters is
 filtered
```

```
                            to contain only parameters that start with this
prefix.
        :param source: When specified, only parameters from this source are
retrieved.
                       For example, a source of 'user' retrieves only parameters
that
                       were set by a user.
        :return: The list of requested parameters.
        """
        try:
            kwargs = {"DBParameterGroupName": parameter_group_name}
            if source is not None:
                kwargs["Source"] = source
            parameters = []
            paginator = self.rds_client.get_paginator("describe_db_parameters")
            for page in paginator.paginate(**kwargs):
                parameters += [
                    p
                    for p in page["Parameters"]
                    if p["ParameterName"].startswith(name_prefix)
                ]
        except ClientError as err:
            logger.error(
                "Couldn't get parameters for %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return parameters
```

- Pour plus de détails sur l'API, voir AWS Describe DBParameters in SDK for Python (Boto3) API Reference.

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds' # v2: require 'aws-sdk'

# List all Amazon Relational Database Service (Amazon RDS) parameter groups.
#
# @param rds_resource [Aws::RDS::Resource] An SDK for Ruby Amazon RDS resource.
# @return [Array, nil] List of all parameter groups, or nil if error.
def list_parameter_groups(rds_resource)
  parameter_groups = []
  rds_resource.db_parameter_groups.each do |p|
    parameter_groups.append({
                              "name": p.db_parameter_group_name,
                              "description": p.description
                            })
  end
  parameter_groups
rescue Aws::Errors::ServiceError => e
  puts "Couldn't list parameter groups:\n #{e.message}"
end
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le manuel de référence des AWS SDK pour Ruby API.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Returns the detailed parameter list for the specified database
    /// parameter group.
    ///
    /// - Parameters:
    ///   - groupName: The name of the parameter group to return parameters for.
    ///   - source: The types of parameters to return (`user`, `system`, or
    ///     `engine-default`).
    ///
    /// - Returns: An array of `RdSClientTypes.Parameter` objects, each
    ///   describing one of the group's parameters.
    func describeDBParameters(groupName: String, source: String? = nil) async ->
 [RDSClientTypes.Parameter] {
        var parameterList: [RDSClientTypes.Parameter] = []

        do {
            let pages = rdsClient.describeDBParametersPaginated(
                input: DescribeDBParametersInput(
                    dbParameterGroupName: groupName,
                    source: source
                )
            )

            for try await page in pages {
                guard let parameters = page.parameters else {
                    return []
                }

                parameterList += parameters
            }
        } catch {
```

```
            print("*** Error getting database parameters:
\(error.localizedDescription)")
            return []
        }

        return parameterList
    }
```

- Pour plus de détails sur l'API, voir Description DBParameters dans le AWS SDK pour la référence de l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeDBSnapshots`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeDBSnapshots`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
/// <summary>
/// Return a list of DB snapshots for a particular DB instance.
/// </summary>
```

```
    /// <param name="dbInstanceIdentifier">DB instance identifier.</param>
    /// <returns>List of DB snapshots.</returns>
    public async Task<List<DBSnapshot>> DescribeDBSnapshots(string
dbInstanceIdentifier)
    {
        var results = new List<DBSnapshot>();
        var snapshotsPaginator = _amazonRDS.Paginators.DescribeDBSnapshots(
            new DescribeDBSnapshotsRequest()
            {
                DBInstanceIdentifier = dbInstanceIdentifier
            });

        // Get the entire list using the paginator.
        await foreach (var snapshots in snapshotsPaginator.DBSnapshots)
        {
            results.Add(snapshots);
        }
        return results;
    }
```

- Pour plus de détails sur l'API, voir Description DBSnapshots dans le manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);
```

```
            Aws::RDS::Model::DescribeDBSnapshotsRequest request;
            request.SetDBSnapshotIdentifier(snapshotID);

            Aws::RDS::Model::DescribeDBSnapshotsOutcome outcome =
                    client.DescribeDBSnapshots(request);

            if (outcome.IsSuccess()) {
                snapshot = outcome.GetResult().GetDBSnapshots()[0];
            }
            else {
                std::cerr << "Error with RDS::DescribeDBSnapshots. "
                          << outcome.GetError().GetMessage()
                          << std::endl;
                cleanUpResources(PARAMETER_GROUP_NAME, DB_INSTANCE_IDENTIFIER,
 client);
                return false;
            }
```

- Pour plus de détails sur l'API, voir Description DBSnapshots dans le manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Exemple 1 : pour décrire un instantané de base de données pour une instance de base de données

L'`describe-db-snapshots`exemple suivant récupère les détails d'un instantané de base de données pour une instance de base de données.

```
aws rds describe-db-snapshots \
    --db-snapshot-identifier mydbsnapshot
```

Sortie :

```
{
    "DBSnapshots": [
        {
            "DBSnapshotIdentifier": "mydbsnapshot",
            "DBInstanceIdentifier": "mysqldb",
```

```
            "SnapshotCreateTime": "2018-02-08T22:28:08.598Z",
            "Engine": "mysql",
            "AllocatedStorage": 20,
            "Status": "available",
            "Port": 3306,
            "AvailabilityZone": "us-east-1f",
            "VpcId": "vpc-6594f31c",
            "InstanceCreateTime": "2018-02-08T22:24:55.973Z",
            "MasterUsername": "mysqladmin",
            "EngineVersion": "5.6.37",
            "LicenseModel": "general-public-license",
            "SnapshotType": "manual",
            "OptionGroupName": "default:mysql-5-6",
            "PercentProgress": 100,
            "StorageType": "gp2",
            "Encrypted": false,
            "DBSnapshotArn": "arn:aws:rds:us-
east-1:123456789012:snapshot:mydbsnapshot",
            "IAMDatabaseAuthenticationEnabled": false,
            "ProcessorFeatures": [],
            "DbiResourceId": "db-AKIAIOSFODNN7EXAMPLE"
        }
    ]
}
```

Pour plus d'informations, consultez la section Création d'un instantané de base de données dans le guide de l'utilisateur Amazon RDS.

Exemple 2 : pour trouver le nombre de clichés pris manuellement

L'`describe-db-snapshots`exemple suivant utilise l'`length`opérateur dans l'`--query`option pour renvoyer le nombre de clichés manuels qui ont été pris dans une AWS région donnée.

```
aws rds describe-db-snapshots \
    --snapshot-type manual \
    --query "length(*[].{DBSnapshots:SnapshotType})" \
    --region eu-central-1
```

Sortie :

```
35
```

Pour plus d'informations, consultez la section Création d'un instantané de base de données dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Description DBSnapshots dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// GetSnapshot gets a DB instance snapshot.
func (instances *DbInstances) GetSnapshot(ctx context.Context, snapshotName
 string) (*types.DBSnapshot, error) {
 output, err := instances.RdsClient.DescribeDBSnapshots(ctx,
  &rds.DescribeDBSnapshotsInput{
   DBSnapshotIdentifier: aws.String(snapshotName),
  })
 if err != nil {
  log.Printf("Couldn't get snapshot %v: %v\n", snapshotName, err)
```

```
  return nil, err
 } else {
  return &output.DBSnapshots[0], nil
 }
}
```

- Pour plus de détails sur l'API, voir Description DBSnapshots dans le manuel de référence des AWS SDK pour Go API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_snapshot(self, snapshot_id):
        """
```

```
        Gets a DB instance snapshot.

        :param snapshot_id: The ID of the snapshot to retrieve.
        :return: The retrieved snapshot.
        """
        try:
            response = self.rds_client.describe_db_snapshots(
                DBSnapshotIdentifier=snapshot_id
            )
            snapshot = response["DBSnapshots"][0]
        except ClientError as err:
            logger.error(
                "Couldn't get snapshot %s. Here's why: %s: %s",
                snapshot_id,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return snapshot
```

- Pour plus de détails sur l'API, voir AWS Describe DBSnapshots in SDK for Python (Boto3) API Reference.

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
require 'aws-sdk-rds' # v2: require 'aws-sdk'

# List all Amazon Relational Database Service (Amazon RDS) DB instance
# snapshots.
#
```

```
# @param rds_resource [Aws::RDS::Resource] An SDK for Ruby Amazon RDS resource.
# @return instance_snapshots [Array, nil] All instance snapshots, or nil if
 error.
def list_instance_snapshots(rds_resource)
  instance_snapshots = []
  rds_resource.db_snapshots.each do |s|
    instance_snapshots.append({
                                "id": s.snapshot_id,
                                "status": s.status
                              })
  end
  instance_snapshots
rescue Aws::Errors::ServiceError => e
  puts "Couldn't list instance snapshots:\n #{e.message}"
end
```

- Pour plus de détails sur l'API, voir Description DBSnapshots dans le manuel de référence des AWS SDK pour Ruby API.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Wait until the specified database snapshot is available to use.
    ///
    /// - Parameters:
    ///   - instanceIdentifier: The identifier of the database for which the
    ///     snapshot was taken.
    ///   - snapshotIdentifier: The identifier of the snapshot to wait for.
    func waitUntilDBSnapshotReady(instanceIdentifier: String, snapshotIdentifier:
 String) async {
```

```
        var snapshotReady = false

        putString("Waiting for the snapshot to be ready...")

        do {
            while !snapshotReady {
                let output = try await rdsClient.describeDBSnapshots(
                    input: DescribeDBSnapshotsInput(
                        dbInstanceIdentifier: instanceIdentifier,
                        dbSnapshotIdentifier: snapshotIdentifier
                    )
                )

                guard let snapshotList = output.dbSnapshots else {
                    return
                }

                for snapshot in snapshotList {
                    guard let snapshotReadyStr = snapshot.status else {
                        return
                    }

                    if snapshotReadyStr.contains("available") {
                        snapshotReady = true
                        print()
                    } else {
                        putString(".")
                        do {
                            try await Task.sleep(for: .seconds(15))
                        } catch {
                            print("\n*** Error pausing the task!")
                        }
                    }
                }
            }
        } catch {
            print("\n*** Unable to wait for the database snapshot to be ready:
\(error.localizedDescription)")
        }
    }
```

- Pour plus de détails sur l'API, voir Description DBSnapshots dans le AWS SDK pour la référence de l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`DescribeOrderableDBInstanceOptions`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `DescribeOrderableDBInstanceOptions`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
    /// Get a list of orderable DB instance options for a specific
    /// engine and engine version.
    /// </summary>
    /// <param name="engine">Name of the engine.</param>
    /// <param name="engineVersion">Version of the engine.</param>
    /// <returns>List of OrderableDBInstanceOptions.</returns>
    public async Task<List<OrderableDBInstanceOption>>
DescribeOrderableDBInstanceOptions(string engine, string engineVersion)
```

```
    {
        // Use a paginator to get a list of DB instance options.
        var results = new List<OrderableDBInstanceOption>();
        var paginateInstanceOptions =
_amazonRDS.Paginators.DescribeOrderableDBInstanceOptions(
            new DescribeOrderableDBInstanceOptionsRequest()
            {
                Engine = engine,
                EngineVersion = engineVersion,
            });
        // Get the entire list using the paginator.
        await foreach (var instanceOptions in
paginateInstanceOptions.OrderableDBInstanceOptions)
        {
            results.Add(instanceOptions);
        }
        return results;
    }
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du manuel de référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);
```

```
//! Routine which gets available 'micro' DB instance classes, displays the list
//! to the user, and returns the user selection.
/*!
 \sa chooseMicroDBInstanceClass()
 \param engineName: The DB engine name.
 \param engineVersion: The DB engine version.
 \param dbInstanceClass: String for DB instance class chosen by the user.
 \param client: 'RDSClient' instance.
 \return bool: Successful completion.
 */
bool AwsDoc::RDS::chooseMicroDBInstanceClass(const Aws::String &engine,
                                             const Aws::String &engineVersion,
                                             Aws::String &dbInstanceClass,
                                             const Aws::RDS::RDSClient &client) {
    std::vector<Aws::String> instanceClasses;
    Aws::String marker;
    do {
        Aws::RDS::Model::DescribeOrderableDBInstanceOptionsRequest request;
        request.SetEngine(engine);
        request.SetEngineVersion(engineVersion);
        if (!marker.empty()) {
            request.SetMarker(marker);
        }

        Aws::RDS::Model::DescribeOrderableDBInstanceOptionsOutcome outcome =
                client.DescribeOrderableDBInstanceOptions(request);

        if (outcome.IsSuccess()) {
            const Aws::Vector<Aws::RDS::Model::OrderableDBInstanceOption>
 &options =
                    outcome.GetResult().GetOrderableDBInstanceOptions();
            for (const Aws::RDS::Model::OrderableDBInstanceOption &option:
 options) {
                const Aws::String &instanceClass = option.GetDBInstanceClass();
                if (instanceClass.find("micro") != std::string::npos) {
                    if (std::find(instanceClasses.begin(), instanceClasses.end(),
                                  instanceClass) ==
                        instanceClasses.end()) {
                        instanceClasses.push_back(instanceClass);
                    }
                }
            }
            marker = outcome.GetResult().GetMarker();
        }
```

```
        else {
            std::cerr << "Error with RDS::DescribeOrderableDBInstanceOptions. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
            return false;
        }
    } while (!marker.empty());

    std::cout << "The available micro DB instance classes for your database
 engine are:"
              << std::endl;
    for (int i = 0; i < instanceClasses.size(); ++i) {
        std::cout << "   " << i + 1 << ": " << instanceClasses[i] << std::endl;
    }

    int choice = askQuestionForIntRange(
            "Which micro DB instance class do you want to use? ",
            1, static_cast<int>(instanceClasses.size()));
    dbInstanceClass = instanceClasses[choice - 1];
    return true;
}
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du manuel de référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour décrire les options d'instance de base de données pouvant être commandées

L'`describe-orderable-db-instance-options`exemple suivant récupère des détails sur les options commandables pour une instance de base de données exécutant le moteur de base de données MySQL.

```
aws rds describe-orderable-db-instance-options \
    --engine mysql
```

Sortie :

```
{
```

```
    "OrderableDBInstanceOptions": [
        {
            "MinStorageSize": 5,
            "ReadReplicaCapable": true,
            "MaxStorageSize": 6144,
            "AvailabilityZones": [
                {
                    "Name": "us-east-1a"
                },
                {
                    "Name": "us-east-1b"
                },
                {
                    "Name": "us-east-1c"
                },
                {
                    "Name": "us-east-1d"
                }
            ],
            "SupportsIops": false,
            "AvailableProcessorFeatures": [],
            "MultiAZCapable": true,
            "DBInstanceClass": "db.m1.large",
            "Vpc": true,
            "StorageType": "gp2",
            "LicenseModel": "general-public-license",
            "EngineVersion": "5.5.46",
            "SupportsStorageEncryption": false,
            "SupportsEnhancedMonitoring": true,
            "Engine": "mysql",
            "SupportsIAMDatabaseAuthentication": false,
            "SupportsPerformanceInsights": false
        }
    ]
    ...some output truncated...
}
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du manuel de référence des AWS CLI commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// GetOrderableInstances uses a paginator to get DB instance options that can be
 used to create DB instances that are
// compatible with a set of specifications.
func (instances *DbInstances) GetOrderableInstances(ctx context.Context, engine
 string, engineVersion string) (
 []types.OrderableDBInstanceOption, error) {

 var output *rds.DescribeOrderableDBInstanceOptionsOutput
 var instanceOptions []types.OrderableDBInstanceOption
 var err error
 orderablePaginator :=
 rds.NewDescribeOrderableDBInstanceOptionsPaginator(instances.RdsClient,
  &rds.DescribeOrderableDBInstanceOptionsInput{
   Engine:        aws.String(engine),
   EngineVersion: aws.String(engineVersion),
```

```
  })
 for orderablePaginator.HasMorePages() {
  output, err = orderablePaginator.NextPage(ctx)
  if err != nil {
   log.Printf("Couldn't get orderable DB instance options: %v\n", err)
   break
  } else {
   instanceOptions = append(instanceOptions,
 output.OrderableDBInstanceOptions...)
  }
 }
 return instanceOptions, err
}
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du manuel de référence des AWS SDK pour Go API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
```

```
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def get_orderable_instances(self, db_engine, db_engine_version):
        """
        Gets DB instance options that can be used to create DB instances that are
        compatible with a set of specifications.

        :param db_engine: The database engine that must be supported by the DB
instance.
        :param db_engine_version: The engine version that must be supported by
the DB instance.
        :return: The list of DB instance options that can be used to create a
compatible DB instance.
        """
        try:
            inst_opts = []
            paginator = self.rds_client.get_paginator(
                "describe_orderable_db_instance_options"
            )
            for page in paginator.paginate(
                Engine=db_engine, EngineVersion=db_engine_version
            ):
                inst_opts += page["OrderableDBInstanceOptions"]
        except ClientError as err:
            logger.error(
                "Couldn't get orderable DB instances. Here's why: %s: %s",
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return inst_opts
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du AWS SDK pour Python (Boto3) API Reference.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Print a list of available database instances with "micro" in the class
    /// name, then return one of them to be used by other code.
    ///
    /// - Parameters:
    ///   - engine: The database engine for which to list database instance
    ///     classes.
    ///   - engineVersion: The database version for which to list instances.
    ///
    /// - Returns: An `RDSClientTypes.OrderableDBInstanceOption` describing
    ///   the selected instance type.
    func chooseMicroInstance(engine: String = "mysql", engineVersion: String? =
 nil) async -> RDSClientTypes.OrderableDBInstanceOption? {
        do {
            let pages = rdsClient.describeOrderableDBInstanceOptionsPaginated(
                input: DescribeOrderableDBInstanceOptionsInput(
                    engine: engine,
                    engineVersion: engineVersion
                )
            )

            var optionsList: [RDSClientTypes.OrderableDBInstanceOption] = []

            for try await page in pages {
                guard let orderableDBInstanceOptions =
 page.orderableDBInstanceOptions else {
                    continue
                }

                for dbInstanceOption in orderableDBInstanceOptions {
                    guard let className = dbInstanceOption.dbInstanceClass else {
```

```
                    continue
                }
                if className.contains("micro") {
                    optionsList.append(dbInstanceOption)
                }
            }
        }

        print("Found \(optionsList.count) database instances of 'micro' class
types:")
        for dbInstanceOption in optionsList {
            print("    \(dbInstanceOption.engine ??
"<unknown>") \(dbInstanceOption.engineVersion ?? "<unknown>")
(\(dbInstanceOption.dbInstanceClass ?? "<unknown class>"))")
        }

        return optionsList[0]
    } catch {
        print("*** Error getting a list of orderable instance options:
\(error.localizedDescription)")
        return nil
    }
}
```

- Pour plus de détails sur l'API, consultez la section DescribeOrderableDBInstanceOptions du AWS SDK pour la référence de l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`GenerateRDSAuthToken`** avec un AWS SDK

L'exemple de code suivant montre comment utiliser`GenerateRDSAuthToken`.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

Utilisez la RdsUtilitiesclasse pour générer un jeton d'authentification.

```
public class GenerateRDSAuthToken {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <dbInstanceIdentifier> <masterUsername>

                Where:
                    dbInstanceIdentifier - The database instance identifier.\s
                    masterUsername - The master user name.\s
                """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbInstanceIdentifier = args[0];
        String masterUsername = args[1];
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        String token = getAuthToken(rdsClient, dbInstanceIdentifier,
 masterUsername);
        System.out.println("The token response is " + token);
    }

    public static String getAuthToken(RdsClient rdsClient, String
 dbInstanceIdentifier, String masterUsername) {
```

```
        RdsUtilities utilities = rdsClient.utilities();
        try {
            GenerateAuthenticationTokenRequest tokenRequest =
 GenerateAuthenticationTokenRequest.builder()
                    .credentialsProvider(ProfileCredentialsProvider.create())
                    .username(masterUsername)
                    .port(3306)
                    .hostname(dbInstanceIdentifier)
                    .build();

            return utilities.generateAuthenticationToken(tokenRequest);

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
        return "";
    }
}
```

- Pour plus de détails sur l'API, voir Générer un RDSAuth jeton dans la référence des AWS SDK for Java 2.x API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ModifyDBInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `ModifyDBInstance`.

CLI

AWS CLI

Exemple 1 : pour modifier une instance de base de données

L'`modify-db-instance`exemple suivant associe un groupe d'options et un groupe de paramètres à une instance de base de données Microsoft SQL Server compatible. Le `--`

`apply-immediately` paramètre entraîne l'association immédiate des groupes d'options et de paramètres, au lieu d'attendre la fenêtre de maintenance suivante.

```
aws rds modify-db-instance \
    --db-instance-identifier database-2 \
    --option-group-name test-se-2017 \
    --db-parameter-group-name test-sqlserver-se-2017 \
    --apply-immediately
```

Sortie :

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "database-2",
        "DBInstanceClass": "db.r4.large",
        "Engine": "sqlserver-se",
        "DBInstanceStatus": "available",

        ...output omitted...

        "DBParameterGroups": [
            {
                "DBParameterGroupName": "test-sqlserver-se-2017",
                "ParameterApplyStatus": "applying"
            }
        ],
        "AvailabilityZone": "us-west-2d",

        ...output omitted...

        "MultiAZ": true,
        "EngineVersion": "14.00.3281.6.v1",
        "AutoMinorVersionUpgrade": false,
        "ReadReplicaDBInstanceIdentifiers": [],
        "LicenseModel": "license-included",
        "OptionGroupMemberships": [
            {
                "OptionGroupName": "test-se-2017",
                "Status": "pending-apply"
            }
        ],
        "CharacterSetName": "SQL_Latin1_General_CP1_CI_AS",
        "SecondaryAvailabilityZone": "us-west-2c",
```

```
        "PubliclyAccessible": true,
        "StorageType": "gp2",

        ...output omitted...

        "DeletionProtection": false,
        "AssociatedRoles": [],
        "MaxAllocatedStorage": 1000
    }
}
```

Pour plus d'informations, consultez la section Modification d'une instance de base de données Amazon RDS dans le guide de l'utilisateur Amazon RDS.

Exemple 2 : pour associer un groupe de sécurité VPC à une instance de base de données

L'`modify-db-instance`exemple suivant associe un groupe de sécurité VPC spécifique et supprime les groupes de sécurité de base de données d'une instance de base de données :

```
aws rds modify-db-instance \
    --db-instance-identifier dbName \
    --vpc-security-group-ids sg-ID
```

Sortie :

```
{
"DBInstance": {
    "DBInstanceIdentifier": "dbName",
    "DBInstanceClass": "db.t3.micro",
    "Engine": "mysql",
    "DBInstanceStatus": "available",
    "MasterUsername": "admin",
    "Endpoint": {
        "Address": "dbName.abcdefghijk.us-west-2.rds.amazonaws.com",
        "Port": 3306,
        "HostedZoneId": "ABCDEFGHIJK1234"
    },
    "AllocatedStorage": 20,
    "InstanceCreateTime": "2024-02-15T00:37:58.793000+00:00",
    "PreferredBackupWindow": "11:57-12:27",
    "BackupRetentionPeriod": 7,
    "DBSecurityGroups": [],
    "VpcSecurityGroups": [
```

```
        {
            "VpcSecurityGroupId": "sg-ID",
            "Status": "active"
        }
    ],
    ... output omitted ...
    "MultiAZ": false,
    "EngineVersion": "8.0.35",
    "AutoMinorVersionUpgrade": true,
    "ReadReplicaDBInstanceIdentifiers": [],
    "LicenseModel": "general-public-license",

    ... output ommited ...
    }
}
```

Pour plus d'informations, consultez la section Contrôle de l'accès avec les groupes de sécurité dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Modifier DBInstance dans AWS CLI la référence des commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.ModifyDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.ModifyDbInstanceResponse;
import software.amazon.awssdk.services.rds.model.RdsException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
```

```
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class ModifyDBInstance {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <dbInstanceIdentifier> <dbSnapshotIdentifier>\s
                Where:
                    dbInstanceIdentifier - The database instance identifier.\s
                    masterUserPassword - The updated password that corresponds to
 the master user name.\s
                """;

        if (args.length != 2) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbInstanceIdentifier = args[0];
        String masterUserPassword = args[1];
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        updateIntance(rdsClient, dbInstanceIdentifier, masterUserPassword);
        rdsClient.close();
    }

    public static void updateIntance(RdsClient rdsClient, String
 dbInstanceIdentifier, String masterUserPassword) {
        try {
            // For a demo - modify the DB instance by modifying the master
 password.
            ModifyDbInstanceRequest modifyDbInstanceRequest =
 ModifyDbInstanceRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .publiclyAccessible(true)
                    .masterUserPassword(masterUserPassword)
                    .build();
```

```
            ModifyDbInstanceResponse instanceResponse =
 rdsClient.modifyDBInstance(modifyDbInstanceRequest);
            System.out.print("The ARN of the modified database is: " +
 instanceResponse.dbInstance().dbInstanceArn());

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Modifier DBInstance dans la référence des AWS SDK for Java 2.x API.

Kotlin

SDK pour Kotlin

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
suspend fun updateIntance(
    dbInstanceIdentifierVal: String?,
    masterUserPasswordVal: String?,
) {
    val request =
        ModifyDbInstanceRequest {
            dbInstanceIdentifier = dbInstanceIdentifierVal
            publiclyAccessible = true
            masterUserPassword = masterUserPasswordVal
        }

    RdsClient.fromEnvironment { region = "us-west-2" }.use { rdsClient ->
        val instanceResponse = rdsClient.modifyDbInstance(request)
```

```
        println("The ARN of the modified database is
 ${instanceResponse.dbInstance?.dbInstanceArn}")
    }
}
```

- Pour plus de détails sur l'API, voir Modifier DBInstance dans le AWS SDK pour la référence de l'API Kotlin.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`ModifyDBParameterGroup`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `ModifyDBParameterGroup`.

Les exemples d'actions sont des extraits de code de programmes de plus grande envergure et doivent être exécutés en contexte. Vous pouvez voir cette action en contexte dans l'exemple de code suivant :

- Principes de base

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    /// <summary>
    /// Update a DB parameter group. Use the action
DescribeDBParameterGroupsAsync
    /// to determine when the DB parameter group is ready to use.
    /// </summary>
    /// <param name="name">Name of the DB parameter group.</param>
```

```
    /// <param name="parameters">List of parameters. Maximum of 20 per request.</
param>
    /// <returns>The updated DB parameter group name.</returns>
    public async Task<string> ModifyDBParameterGroup(
        string name, List<Parameter> parameters)
    {
        var response = await _amazonRDS.ModifyDBParameterGroupAsync(
            new ModifyDBParameterGroupRequest()
            {
                DBParameterGroupName = name,
                Parameters = parameters,
            });
        return response.DBParameterGroupName;
    }
```

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans la référence des AWS SDK pour .NET API.

C++

SDK pour C++

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
        Aws::Client::ClientConfiguration clientConfig;
        // Optional: Set to the AWS Region (overrides config file).
        // clientConfig.region = "us-east-1";

    Aws::RDS::RDSClient client(clientConfig);

        Aws::RDS::Model::ModifyDBParameterGroupRequest request;
        request.SetDBParameterGroupName(PARAMETER_GROUP_NAME);
        request.SetParameters(updateParameters);

        Aws::RDS::Model::ModifyDBParameterGroupOutcome outcome =
```

```
                client.ModifyDBParameterGroup(request);

        if (outcome.IsSuccess()) {
            std::cout << "The DB parameter group was successfully modified."
                      << std::endl;
        }
        else {
            std::cerr << "Error with RDS::ModifyDBParameterGroup. "
                      << outcome.GetError().GetMessage()
                      << std::endl;
        }
```

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans la référence des AWS SDK pour C++ API.

CLI

AWS CLI

Pour modifier un groupe de paramètres de base de données

L'`modify-db-parameter-group`exemple suivant modifie la valeur du `clr enabled` paramètre dans un groupe de paramètres de base de données. Le `--apply-immediately` paramètre entraîne la modification immédiate du groupe de paramètres de base de données, au lieu d'attendre la fenêtre de maintenance suivante.

```
aws rds modify-db-parameter-group \
    --db-parameter-group-name test-sqlserver-se-2017 \
    --parameters "ParameterName='clr
 enabled',ParameterValue=1,ApplyMethod=immediate"
```

Sortie :

```
{
    "DBParameterGroupName": "test-sqlserver-se-2017"
}
```

Pour plus d'informations, consultez la section Modification des paramètres d'un groupe de paramètres de base de données dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans AWS CLI la référence des commandes.

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import (
 "context"
 "errors"
 "log"

 "github.com/aws/aws-sdk-go-v2/aws"
 "github.com/aws/aws-sdk-go-v2/service/rds"
 "github.com/aws/aws-sdk-go-v2/service/rds/types"
)

type DbInstances struct {
 RdsClient *rds.Client
}



// UpdateParameters updates parameters in a named DB parameter group.
func (instances *DbInstances) UpdateParameters(ctx context.Context,
 parameterGroupName string, params []types.Parameter) error {
 _, err := instances.RdsClient.ModifyDBParameterGroup(ctx,
  &rds.ModifyDBParameterGroupInput{
   DBParameterGroupName: aws.String(parameterGroupName),
   Parameters:           params,
  })
 if err != nil {
  log.Printf("Couldn't update parameters in %v: %v\n", parameterGroupName, err)
  return err
```

```
 } else {
  return nil
 }
}
```

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans la référence des AWS SDK pour Go API.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
    // Modify auto_increment_offset and auto_increment_increment parameters.
    public static void modifyDBParas(RdsClient rdsClient, String dbGroupName) {
        try {
            Parameter parameter1 = Parameter.builder()
                    .parameterName("auto_increment_offset")
                    .applyMethod("immediate")
                    .parameterValue("5")
                    .build();

            List<Parameter> paraList = new ArrayList<>();
            paraList.add(parameter1);
            ModifyDbParameterGroupRequest groupRequest =
ModifyDbParameterGroupRequest.builder()
                    .dbParameterGroupName(dbGroupName)
                    .parameters(paraList)
                    .build();

            ModifyDbParameterGroupResponse response =
rdsClient.modifyDBParameterGroup(groupRequest);
            System.out.println("The parameter group " +
response.dbParameterGroupName() + " was successfully modified");
```

```
        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
```

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans la référence des AWS SDK for Java 2.x API.

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
class InstanceWrapper:
    """Encapsulates Amazon RDS DB instance actions."""

    def __init__(self, rds_client):
        """
        :param rds_client: A Boto3 Amazon RDS client.
        """
        self.rds_client = rds_client

    @classmethod
    def from_client(cls):
        """
        Instantiates this class from a Boto3 client.
        """
        rds_client = boto3.client("rds")
        return cls(rds_client)


    def update_parameters(self, parameter_group_name, update_parameters):
        """
```

```
        Updates parameters in a custom DB parameter group.

        :param parameter_group_name: The name of the parameter group to update.
        :param update_parameters: The parameters to update in the group.
        :return: Data about the modified parameter group.
        """
        try:
            response = self.rds_client.modify_db_parameter_group(
                DBParameterGroupName=parameter_group_name,
Parameters=update_parameters
            )
        except ClientError as err:
            logger.error(
                "Couldn't update parameters in %s. Here's why: %s: %s",
                parameter_group_name,
                err.response["Error"]["Code"],
                err.response["Error"]["Message"],
            )
            raise
        else:
            return response
```

- Pour plus de détails sur l'API, voir DBParameterModify Group in AWS SDK for Python (Boto3) API Reference.

Swift

Kit SDK pour Swift

> **Note**
>
> Il y en a plus sur GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import AWSRDS

    /// Demonstrates modifying two of the specified database parameter group's
    /// parameters.
```

```
    ///
    /// - Parameter groupName: The name of the parameter group to change
    ///   parameters for.
    func modifyDBParameters(groupName: String) async {
        let parameter1 = RDSClientTypes.Parameter(
            applyMethod: RDSClientTypes.ApplyMethod.immediate,
            parameterName: "auto_increment_offset",
            parameterValue: "5"
        )
        let parameter2 = RDSClientTypes.Parameter(
            applyMethod: RDSClientTypes.ApplyMethod.immediate,
            parameterName: "auto_increment_increment",
            parameterValue: "5"
        )

        let parameterList = [parameter1, parameter2]

        do {
            _ = try await rdsClient.modifyDBParameterGroup(
                input: ModifyDBParameterGroupInput(
                    dbParameterGroupName: groupName,
                    parameters: parameterList
                )
            )

            print("Successfully modified the parameter group \(groupName).")
        } catch {
            print("*** Error modifying the parameter group \(groupName):
\(error.localizedDescription)")
        }
    }
```

- Pour plus de détails sur l'API, voir Modifier DBParameter le groupe dans le AWS SDK pour une référence à l'API Swift.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Utilisation **`RebootDBInstance`** avec un AWS SDK ou une CLI

Les exemples de code suivants illustrent comment utiliser `RebootDBInstance`.

CLI

AWS CLI

Pour redémarrer une instance de base de données

L'`reboot-db-instance`exemple suivant lance un redémarrage de l'instance de base de données spécifiée.

```
aws rds reboot-db-instance \
    --db-instance-identifier test-mysql-instance
```

Sortie :

```
{
    "DBInstance": {
        "DBInstanceIdentifier": "test-mysql-instance",
        "DBInstanceClass": "db.t3.micro",
        "Engine": "mysql",
        "DBInstanceStatus": "rebooting",
        "MasterUsername": "admin",
        "Endpoint": {
            "Address": "test-mysql-instance.############.us-
west-2.rds.amazonaws.com",
            "Port": 3306,
            "HostedZoneId": "Z1PVIF0EXAMPLE"
        },

    ... output omitted...

    }
}
```

Pour plus d'informations, consultez la section Redémarrage d'une instance de base de données dans le guide de l'utilisateur Amazon RDS.

- Pour plus de détails sur l'API, voir Reboot DBInstance dans AWS CLI la référence des commandes.

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples de code AWS.

```
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rds.RdsClient;
import software.amazon.awssdk.services.rds.model.RebootDbInstanceRequest;
import software.amazon.awssdk.services.rds.model.RebootDbInstanceResponse;
import software.amazon.awssdk.services.rds.model.RdsException;

/**
 * Before running this Java V2 code example, set up your development
 * environment, including your credentials.
 *
 * For more information, see the following documentation topic:
 *
 * https://docs.aws.amazon.com/sdk-for-java/latest/developer-guide/get-
started.html
 */
public class RebootDBInstance {
    public static void main(String[] args) {
        final String usage = """

                Usage:
                    <dbInstanceIdentifier>\s

                Where:
                    dbInstanceIdentifier - The database instance identifier\s
                """;

        if (args.length != 1) {
            System.out.println(usage);
            System.exit(1);
        }

        String dbInstanceIdentifier = args[0];
```

```
        Region region = Region.US_WEST_2;
        RdsClient rdsClient = RdsClient.builder()
                .region(region)
                .build();

        rebootInstance(rdsClient, dbInstanceIdentifier);
        rdsClient.close();
    }

    public static void rebootInstance(RdsClient rdsClient, String
 dbInstanceIdentifier) {
        try {
            RebootDbInstanceRequest rebootDbInstanceRequest =
 RebootDbInstanceRequest.builder()
                    .dbInstanceIdentifier(dbInstanceIdentifier)
                    .build();

            RebootDbInstanceResponse instanceResponse =
 rdsClient.rebootDBInstance(rebootDbInstanceRequest);
            System.out.print("The database " +
 instanceResponse.dbInstance().dbInstanceArn() + " was rebooted");

        } catch (RdsException e) {
            System.out.println(e.getLocalizedMessage());
            System.exit(1);
        }
    }
}
```

- Pour plus de détails sur l'API, voir Reboot DBInstance dans le manuel de référence des AWS SDK for Java 2.x API.

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Scénarios d'utilisation d'Amazon RDS AWS SDKs

Les exemples de code suivants vous montrent comment implémenter des scénarios courants dans Amazon RDS avec AWS SDKs. Ces scénarios vous montrent comment accomplir des tâches

spécifiques en appelant plusieurs fonctions au sein d'Amazon RDS ou en les combinant à d'autres Services AWS. Chaque exemple inclut un lien vers le code source complet, où vous trouverez des instructions sur la configuration et l'exécution du code.

Les scénarios ciblent un niveau d'expérience intermédiaire pour vous aider à comprendre les actions de service dans leur contexte.

Exemples

- Créer un outil de suivi des éléments de travail sans serveur Aurora

Créer un outil de suivi des éléments de travail sans serveur Aurora

Les exemples de code suivants montrent comment créer une application web qui suit des éléments de travail dans une base de données Amazon Aurora sans serveur et envoie des rapports par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES).

.NET

SDK pour .NET

Montre comment utiliser le AWS SDK pour .NET pour créer une application Web qui suit les éléments de travail dans une base de données Amazon Aurora et envoie des rapports par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES). Cet exemple utilise un front-end créé avec React.js pour interagir avec un backend RESTful .NET.

- Intégrez une application Web React à AWS des services.
- Listez, ajoutez et mettez à jour des éléments dans une table Aurora.
- Envoyez un rapport par e-mail sur les éléments de travail filtrés à l'aide d'Amazon SES.
- Déployez et gérez des exemples de ressources à l'aide du AWS CloudFormation script inclus.

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS

- Services de données Amazon RDS
- Amazon SES

C++

SDK pour C++

Montre comment créer une application web qui suit et génère des rapports sur les éléments de travail stockés dans une base de données Amazon Aurora sans serveur.

Pour obtenir le code source complet et les instructions sur la façon de configurer une API REST C++ qui interroge les données Amazon Aurora Serverless et à utiliser par une application React, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS
- Services de données Amazon RDS
- Amazon SES

Java

SDK pour Java 2.x

Montre comment créer une application web qui suit et génère des rapports sur les éléments de travail stockés dans une base de données Amazon RDS.

Pour obtenir le code source complet et les instructions sur la façon de configurer une API Spring REST qui interroge les données Amazon Aurora Serverless et pour une utilisation par une application React, consultez l'exemple complet sur GitHub.

Pour obtenir le code source complet et les instructions sur la façon de configurer et d'exécuter un exemple utilisant l'API JDBC, consultez l'exemple complet sur. GitHub

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS

- Services de données Amazon RDS
- Amazon SES

JavaScript

SDK pour JavaScript (v3)

Montre comment utiliser le AWS SDK pour JavaScript (v3) pour créer une application Web qui suit les éléments de travail dans une base de données Amazon Aurora et envoie des rapports par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES). Cet exemple utilise un front end créé avec React.js pour interagir avec un backend Express Node.js.

- Intégrez une application Web React.js à Services AWS.
- Lister, ajouter et mettre à jour des éléments dans une table Aurora.
- Envoyez un rapport par e-mail sur les éléments de travail filtrés en utilisant Amazon SES.
- Déployez et gérez des exemples de ressources à l'aide du AWS CloudFormation script inclus.

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS
- Services de données Amazon RDS
- Amazon SES

Kotlin

SDK pour Kotlin

Montre comment créer une application web qui suit et génère des rapports sur les éléments de travail stockés dans une base de données Amazon RDS.

Pour obtenir le code source complet et les instructions sur la façon de configurer une API Spring REST qui interroge les données Amazon Aurora Serverless et pour une utilisation par une application React, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS
- Services de données Amazon RDS
- Amazon SES

PHP

Kit SDK pour PHP

Montre comment utiliser le AWS SDK pour PHP pour créer une application Web qui suit les éléments de travail dans une base de données Amazon RDS et envoie des rapports par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES). Cet exemple utilise un front-end créé avec React.js pour interagir avec un backend RESTful PHP.

- Intégrez une application Web React.js à AWS des services.
- Répertoriez, ajoutez, mettez à jour et supprimez des éléments dans une table Amazon RDS.
- Envoyez un rapport par e-mail sur les éléments de travail filtrés à l'aide d'Amazon SES.
- Déployez et gérez des exemples de ressources à l'aide du AWS CloudFormation script inclus.

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS
- Services de données Amazon RDS
- Amazon SES

Python

SDK pour Python (Boto3)

Montre comment utiliser le AWS SDK pour Python (Boto3) pour créer un service REST qui suit les éléments de travail dans une base de données Amazon Aurora Serverless et envoie des rapports par e-mail à l'aide d'Amazon Simple Email Service (Amazon SES). Cet exemple

utilise la structure web Flask pour gérer le routage HTTP et s'intègre à une page web React pour présenter une application web entièrement fonctionnelle.

- Créez un service Flask REST qui s'intègre à Services AWS.
- Lisez, écrivez et mettez à jour les éléments de travail stockés dans une base de données Aurora sans serveur.
- Créez un AWS Secrets Manager secret contenant les informations d'identification de la base de données et utilisez-le pour authentifier les appels à la base de données.
- Utilisez Amazon SES pour envoyer des rapports par e-mail sur les éléments de travail.

Pour obtenir le code source complet et les instructions de configuration et d'exécution, consultez l'exemple complet sur GitHub.

Les services utilisés dans cet exemple

- Aurora
- Amazon RDS
- Services de données Amazon RDS
- Amazon SES

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Exemples de solutions sans serveur pour Amazon RDS

Les exemples de code suivants montrent comment utiliser Amazon RDS avec AWS SDKs.

Exemples

- Connexion à une base de données Amazon RDS dans une fonction Lambda

Connexion à une base de données Amazon RDS dans une fonction Lambda

Les exemples de code suivants illustrent comment implémenter une fonction Lambda qui se connecte à une base de données RDS. La fonction effectue une simple requête de base de données et renvoie le résultat.

.NET

SDK pour .NET

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de .NET.

```
using System.Data;
using System.Text.Json;
using Amazon.Lambda.APIGatewayEvents;
using Amazon.Lambda.Core;
using MySql.Data.MySqlClient;

// Assembly attribute to enable the Lambda function's JSON input to be converted
 into a .NET class.
[assembly:
 LambdaSerializer(typeof(Amazon.Lambda.Serialization.SystemTextJson.DefaultLambdaJsonSeri

namespace aws_rds;

public class InputModel
{
    public string key1 { get; set; }
    public string key2 { get; set; }
}

public class Function
{
    /// <summary>
    // Handles the Lambda function execution for connecting to RDS using IAM
 authentication.
    /// </summary>
    /// <param name="input">The input event data passed to the Lambda function</
param>
    /// <param name="context">The Lambda execution context that provides runtime
 information</param>
    /// <returns>A response object containing the execution result</returns>
```

```
   public async Task<APIGatewayProxyResponse>
FunctionHandler(APIGatewayProxyRequest request, ILambdaContext context)
   {
       // Sample Input: {"body": "{\"key1\":\"20\", \"key2\":\"25\"}"}
       var input = JsonSerializer.Deserialize<InputModel>(request.Body);

       /// Obtain authentication token
       var authToken = RDSAuthTokenGenerator.GenerateAuthToken(
           Environment.GetEnvironmentVariable("RDS_ENDPOINT"),
           Convert.ToInt32(Environment.GetEnvironmentVariable("RDS_PORT")),
           Environment.GetEnvironmentVariable("RDS_USERNAME")
       );

       /// Build the Connection String with the Token
       string connectionString =
$"Server={Environment.GetEnvironmentVariable("RDS_ENDPOINT")};" +

$"Port={Environment.GetEnvironmentVariable("RDS_PORT")};" +

$"Uid={Environment.GetEnvironmentVariable("RDS_USERNAME")};" +
                                  $"Pwd={authToken};";


       try
       {
           await using var connection = new MySqlConnection(connectionString);
           await connection.OpenAsync();

           const string sql = "SELECT @param1 + @param2 AS Sum";

           await using var command = new MySqlCommand(sql, connection);
           command.Parameters.AddWithValue("@param1", int.Parse(input.key1 ??
"0"));
           command.Parameters.AddWithValue("@param2", int.Parse(input.key2 ??
"0"));

           await using var reader = await command.ExecuteReaderAsync();
           if (await reader.ReadAsync())
           {
               int result = reader.GetInt32("Sum");

               //Sample Response: {"statusCode":200,"body":"{\"message\":\"The
sum is: 45\"}","isBase64Encoded":false}
               return new APIGatewayProxyResponse
```

```
                {
                    StatusCode = 200,
                    Body = JsonSerializer.Serialize(new { message = $"The sum is:
 {result}" })
                };
            }

        }
        catch (Exception ex)
        {
            Console.WriteLine($"Error: {ex.Message}");
        }

        return new APIGatewayProxyResponse
        {
            StatusCode = 500,
            Body = JsonSerializer.Serialize(new { error = "Internal server
 error" })
        };
    }
}
```

Go

Kit SDK for Go V2

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de Go.

```
/*
Golang v2 code here.
*/

package main

import (
```

```
 "context"
 "database/sql"
 "encoding/json"
 "fmt"
 "os"

 "github.com/aws/aws-lambda-go/lambda"
 "github.com/aws/aws-sdk-go-v2/config"
 "github.com/aws/aws-sdk-go-v2/feature/rds/auth"
 _ "github.com/go-sql-driver/mysql"
)

type MyEvent struct {
 Name string `json:"name"`
}

func HandleRequest(event *MyEvent) (map[string]interface{}, error) {

 var dbName string = os.Getenv("DatabaseName")
 var dbUser string = os.Getenv("DatabaseUser")
 var dbHost string = os.Getenv("DBHost") // Add hostname without https
 var dbPort int = os.Getenv("Port")      // Add port number
 var dbEndpoint string = fmt.Sprintf("%s:%d", dbHost, dbPort)
 var region string = os.Getenv("AWS_REGION")

 cfg, err := config.LoadDefaultConfig(context.TODO())
 if err != nil {
  panic("configuration error: " + err.Error())
 }

 authenticationToken, err := auth.BuildAuthToken(
  context.TODO(), dbEndpoint, region, dbUser, cfg.Credentials)
 if err != nil {
  panic("failed to create authentication token: " + err.Error())
 }

 dsn := fmt.Sprintf("%s:%s@tcp(%s)/%s?tls=true&allowCleartextPasswords=true",
  dbUser, authenticationToken, dbEndpoint, dbName,
 )

 db, err := sql.Open("mysql", dsn)
 if err != nil {
  panic(err)
 }
```

```
 defer db.Close()

 var sum int
 err = db.QueryRow("SELECT ?+? AS sum", 3, 2).Scan(&sum)
 if err != nil {
  panic(err)
 }
 s := fmt.Sprint(sum)
 message := fmt.Sprintf("The selected sum is: %s", s)

 messageBytes, err := json.Marshal(message)
 if err != nil {
  return nil, err
 }

 messageString := string(messageBytes)
 return map[string]interface{}{
  "statusCode": 200,
  "headers":    map[string]string{"Content-Type": "application/json"},
  "body":       messageString,
 }, nil
}

func main() {
 lambda.Start(HandleRequest)
}
```

Java

SDK pour Java 2.x

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de Java.

```
import com.amazonaws.services.lambda.runtime.Context;
```

```
import com.amazonaws.services.lambda.runtime.RequestHandler;
import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyRequestEvent;
import com.amazonaws.services.lambda.runtime.events.APIGatewayProxyResponseEvent;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.rdsdata.RdsDataClient;
import software.amazon.awssdk.services.rdsdata.model.ExecuteStatementRequest;
import software.amazon.awssdk.services.rdsdata.model.ExecuteStatementResponse;
import software.amazon.awssdk.services.rdsdata.model.Field;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

public class RdsLambdaHandler implements
 RequestHandler<APIGatewayProxyRequestEvent, APIGatewayProxyResponseEvent> {

    @Override
    public APIGatewayProxyResponseEvent handleRequest(APIGatewayProxyRequestEvent
 event, Context context) {
        APIGatewayProxyResponseEvent response = new
 APIGatewayProxyResponseEvent();

        try {
            // Obtain auth token
            String token = createAuthToken();

            // Define connection configuration
            String connectionString = String.format("jdbc:mysql://%s:%s/%s?
useSSL=true&requireSSL=true",
                    System.getenv("ProxyHostName"),
                    System.getenv("Port"),
                    System.getenv("DBName"));

            // Establish a connection to the database
            try (Connection connection =
 DriverManager.getConnection(connectionString, System.getenv("DBUserName"),
 token);
                 PreparedStatement statement =
 connection.prepareStatement("SELECT ? + ? AS sum")) {

                statement.setInt(1, 3);
                statement.setInt(2, 2);
```

```
                try (ResultSet resultSet = statement.executeQuery()) {
                    if (resultSet.next()) {
                        int sum = resultSet.getInt("sum");
                        response.setStatusCode(200);
                        response.setBody("The selected sum is: " + sum);
                    }
                }
            }

        } catch (Exception e) {
            response.setStatusCode(500);
            response.setBody("Error: " + e.getMessage());
        }

        return response;
    }

    private String createAuthToken() {
        // Create RDS Data Service client
        RdsDataClient rdsDataClient = RdsDataClient.builder()
                .region(Region.of(System.getenv("AWS_REGION")))
                .credentialsProvider(DefaultCredentialsProvider.create())
                .build();

        // Define authentication request
        ExecuteStatementRequest request = ExecuteStatementRequest.builder()
                .resourceArn(System.getenv("ProxyHostName"))
                .secretArn(System.getenv("DBUserName"))
                .database(System.getenv("DBName"))
                .sql("SELECT 'RDS IAM Authentication'")
                .build();

        // Execute request and obtain authentication token
        ExecuteStatementResponse response =
 rdsDataClient.executeStatement(request);
        Field tokenField = response.records().get(0).get(0);

        return tokenField.stringValue();
    }
}
```

JavaScript

SDK pour JavaScript (v3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de. JavaScript

```
// Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
// SPDX-License-Identifier: Apache-2.0
/*
Node.js code here.
*/
// ES6+ example
import { Signer } from "@aws-sdk/rds-signer";
import mysql from 'mysql2/promise';

async function createAuthToken() {
  // Define connection authentication parameters
  const dbinfo = {

    hostname: process.env.ProxyHostName,
    port: process.env.Port,
    username: process.env.DBUserName,
    region: process.env.AWS_REGION,

  }

  // Create RDS Signer object
  const signer = new Signer(dbinfo);

  // Request authorization token from RDS, specifying the username
  const token = await signer.getAuthToken();
  return token;
}

async function dbOps() {
```

```
  // Obtain auth token
  const token = await createAuthToken();
  // Define connection configuration
  let connectionConfig = {
    host: process.env.ProxyHostName,
    user: process.env.DBUserName,
    password: token,
    database: process.env.DBName,
    ssl: 'Amazon RDS'
  }
  // Create the connection to the DB
  const conn = await mysql.createConnection(connectionConfig);
  // Obtain the result of the query
  const [res,] = await conn.execute('select ?+? as sum', [3, 2]);
  return res;

}

export const handler = async (event) => {
  // Execute database flow
  const result = await dbOps();
  // Return result
  return {
    statusCode: 200,
    body: JSON.stringify("The selected sum is: " + result[0].sum)
  }
};
```

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de. TypeScript

```
import { Signer } from "@aws-sdk/rds-signer";
import mysql from 'mysql2/promise';

// RDS settings
// Using '!' (non-null assertion operator) to tell the TypeScript compiler that
 the DB settings are not null or undefined,
const proxy_host_name = process.env.PROXY_HOST_NAME!
const port = parseInt(process.env.PORT!)
const db_name = process.env.DB_NAME!
```

```
const db_user_name = process.env.DB_USER_NAME!
const aws_region = process.env.AWS_REGION!


async function createAuthToken(): Promise<string> {

    // Create RDS Signer object
    const signer = new Signer({
        hostname: proxy_host_name,
        port: port,
        region: aws_region,
        username: db_user_name
    });

    // Request authorization token from RDS, specifying the username
    const token = await signer.getAuthToken();
    return token;
}

async function dbOps(): Promise<mysql.QueryResult | undefined> {
    try {
        // Obtain auth token
        const token = await createAuthToken();
        const conn = await mysql.createConnection({
            host: proxy_host_name,
            user: db_user_name,
            password: token,
            database: db_name,
            ssl: 'Amazon RDS' // Ensure you have the CA bundle for SSL connection
        });
        const [rows, fields] = await conn.execute('SELECT ? + ? AS sum', [3, 2]);
        console.log('result:', rows);
        return rows;
    }
    catch (err) {
        console.log(err);
    }
}

export const lambdaHandler = async (event: any): Promise<{ statusCode: number;
 body: string }> => {
    // Execute database flow
    const result = await dbOps();
```

```
    // Return error is result is undefined
    if (result == undefined)
        return {
            statusCode: 500,
            body: JSON.stringify(`Error with connection to DB host`)
        }

    // Return result
    return {
        statusCode: 200,
        body: JSON.stringify(`The selected sum is: ${result[0].sum}`)
    };
};
```

PHP

Kit SDK pour PHP

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de PHP.

```
<?php
# Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
# SPDX-License-Identifier: Apache-2.0

# using bref/bref and bref/logger for simplicity

use Bref\Context\Context;
use Bref\Event\Handler as StdHandler;
use Bref\Logger\StderrLogger;
use Aws\Rds\AuthTokenGenerator;
use Aws\Credentials\CredentialProvider;

require __DIR__ . '/vendor/autoload.php';

class Handler implements StdHandler
{
```

```
    private StderrLogger $logger;
    public function __construct(StderrLogger $logger)
    {
        $this->logger = $logger;
    }


    private function getAuthToken(): string {
        // Define connection authentication parameters
        $dbConnection = [
            'hostname' => getenv('DB_HOSTNAME'),
            'port' => getenv('DB_PORT'),
            'username' => getenv('DB_USERNAME'),
            'region' => getenv('AWS_REGION'),
        ];

        // Create RDS AuthTokenGenerator object
        $generator = new
AuthTokenGenerator(CredentialProvider::defaultProvider());

        // Request authorization token from RDS, specifying the username
        return $generator->createToken(
            $dbConnection['hostname'] . ':' . $dbConnection['port'],
            $dbConnection['region'],
            $dbConnection['username']
        );
    }

    private function getQueryResults() {
        // Obtain auth token
        $token = $this->getAuthToken();

        // Define connection configuration
        $connectionConfig = [
            'host' => getenv('DB_HOSTNAME'),
            'user' => getenv('DB_USERNAME'),
            'password' => $token,
            'database' => getenv('DB_NAME'),
        ];

        // Create the connection to the DB
        $conn = new PDO(

"mysql:host={$connectionConfig['host']};dbname={$connectionConfig['database']}",
```

```
            $connectionConfig['user'],
            $connectionConfig['password'],
            [
                PDO::MYSQL_ATTR_SSL_CA => '/path/to/rds-ca-2019-root.pem',
                PDO::MYSQL_ATTR_SSL_VERIFY_SERVER_CERT => true,
            ]
        );

        // Obtain the result of the query
        $stmt = $conn->prepare('SELECT ?+? AS sum');
        $stmt->execute([3, 2]);

        return $stmt->fetch(PDO::FETCH_ASSOC);
    }

    /**
     * @param mixed $event
     * @param Context $context
     * @return array
     */
    public function handle(mixed $event, Context $context): array
    {
        $this->logger->info("Processing query");

        // Execute database flow
        $result = $this->getQueryResults();

        return [
            'sum' => $result['sum']
        ];
    }
}

$logger = new StderrLogger();
return new Handler($logger);
```

Python

SDK pour Python (Boto3)

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de Python.

```
import json
import os
import boto3
import pymysql

# RDS settings
proxy_host_name = os.environ['PROXY_HOST_NAME']
port = int(os.environ['PORT'])
db_name = os.environ['DB_NAME']
db_user_name = os.environ['DB_USER_NAME']
aws_region = os.environ['AWS_REGION']


# Fetch RDS Auth Token
def get_auth_token():
    client = boto3.client('rds')
    token = client.generate_db_auth_token(
        DBHostname=proxy_host_name,
        Port=port
        DBUsername=db_user_name
        Region=aws_region
    )
    return token

def lambda_handler(event, context):
    token = get_auth_token()
    try:
        connection = pymysql.connect(
            host=proxy_host_name,
```

```
            user=db_user_name,
            password=token,
            db=db_name,
            port=port,
            ssl={'ca': 'Amazon RDS'}  # Ensure you have the CA bundle for SSL
 connection
        )

        with connection.cursor() as cursor:
            cursor.execute('SELECT %s + %s AS sum', (3, 2))
            result = cursor.fetchone()

        return result

    except Exception as e:
        return (f"Error: {str(e)}")  # Return an error message if an exception
 occurs
```

Ruby

Kit SDK pour Ruby

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de Ruby.

```
# Ruby code here.

require 'aws-sdk-rds'
require 'json'
require 'mysql2'

def lambda_handler(event:, context:)
  endpoint = ENV['DBEndpoint'] # Add the endpoint without https"
  port = ENV['Port']           # 3306
  user = ENV['DBUser']
  region = ENV['DBRegion']     # 'us-east-1'
```

```
  db_name = ENV['DBName']

  credentials = Aws::Credentials.new(
    ENV['AWS_ACCESS_KEY_ID'],
    ENV['AWS_SECRET_ACCESS_KEY'],
    ENV['AWS_SESSION_TOKEN']
  )
  rds_client = Aws::RDS::AuthTokenGenerator.new(
    region: region,
    credentials: credentials
  )

  token = rds_client.auth_token(
    endpoint: endpoint+ ':' + port,
    user_name: user,
    region: region
  )

  begin
    conn = Mysql2::Client.new(
      host: endpoint,
      username: user,
      password: token,
      port: port,
      database: db_name,
      sslca: '/var/task/global-bundle.pem',
      sslverify: true,
      enable_cleartext_plugin: true
    )
    a = 3
    b = 2
    result = conn.query("SELECT #{a} + #{b} AS sum").first['sum']
    puts result
    conn.close
    {
      statusCode: 200,
      body: result.to_json
    }
  rescue => e
    puts "Database connection failed due to #{e}"
  end
end
```

Rust

SDK pour Rust

> **Note**
>
> Il y en a plus à ce sujet GitHub. Trouvez l'exemple complet et découvrez comment le configurer et l'exécuter dans le référentiel d'exemples sans serveur.

Connexion à une base de données Amazon RDS dans une fonction Lambda à l'aide de Rust.

```
use aws_config::BehaviorVersion;
use aws_credential_types::provider::ProvideCredentials;
use aws_sigv4::{
    http_request::{sign, SignableBody, SignableRequest, SigningSettings},
    sign::v4,
};
use lambda_runtime::{run, service_fn, Error, LambdaEvent};
use serde_json::{json, Value};
use sqlx::postgres::PgConnectOptions;
use std::env;
use std::time::{Duration, SystemTime};

const RDS_CERTS: &[u8] = include_bytes!("global-bundle.pem");

async fn generate_rds_iam_token(
    db_hostname: &str,
    port: u16,
    db_username: &str,
) -> Result<String, Error> {
    let config = aws_config::load_defaults(BehaviorVersion::v2024_03_28()).await;

    let credentials = config
        .credentials_provider()
        .expect("no credentials provider found")
        .provide_credentials()
        .await
        .expect("unable to load credentials");
    let identity = credentials.into();
    let region = config.region().unwrap().to_string();

    let mut signing_settings = SigningSettings::default();
```

```
    signing_settings.expires_in = Some(Duration::from_secs(900));
    signing_settings.signature_location =
 aws_sigv4::http_request::SignatureLocation::QueryParams;

    let signing_params = v4::SigningParams::builder()
        .identity(&identity)
        .region(&region)
        .name("rds-db")
        .time(SystemTime::now())
        .settings(signing_settings)
        .build()?;

    let url = format!(
        "https://{db_hostname}:{port}/?Action=connect&DBUser={db_user}",
        db_hostname = db_hostname,
        port = port,
        db_user = db_username
    );

    let signable_request =
        SignableRequest::new("GET", &url, std::iter::empty(),
 SignableBody::Bytes(&[]))
            .expect("signable request");

    let (signing_instructions, _signature) =
        sign(signable_request, &signing_params.into())?.into_parts();

    let mut url = url::Url::parse(&url).unwrap();
    for (name, value) in signing_instructions.params() {
        url.query_pairs_mut().append_pair(name, &value);
    }

    let response = url.to_string().split_off("https://".len());

    Ok(response)
}

#[tokio::main]
async fn main() -> Result<(), Error> {
    run(service_fn(handler)).await
}

async fn handler(_event: LambdaEvent<Value>) -> Result<Value, Error> {
    let db_host = env::var("DB_HOSTNAME").expect("DB_HOSTNAME must be set");
```

```
    let db_port = env::var("DB_PORT")
        .expect("DB_PORT must be set")
        .parse::<u16>()
        .expect("PORT must be a valid number");
    let db_name = env::var("DB_NAME").expect("DB_NAME must be set");
    let db_user_name = env::var("DB_USERNAME").expect("DB_USERNAME must be set");

    let token = generate_rds_iam_token(&db_host, db_port, &db_user_name).await?;

    let opts = PgConnectOptions::new()
        .host(&db_host)
        .port(db_port)
        .username(&db_user_name)
        .password(&token)
        .database(&db_name)
        .ssl_root_cert_from_pem(RDS_CERTS.to_vec())
        .ssl_mode(sqlx::postgres::PgSslMode::Require);

    let pool = sqlx::postgres::PgPoolOptions::new()
        .connect_with(opts)
        .await?;

    let result: i32 = sqlx::query_scalar("SELECT $1 + $2")
        .bind(3)
        .bind(2)
        .fetch_one(&pool)
        .await?;

    println!("Result: {:?}", result);

    Ok(json!({
        "statusCode": 200,
        "content-type": "text/plain",
        "body": format!("The selected sum is: {result}")
    }))
}
```

Pour obtenir la liste complète des guides de développement du AWS SDK et des exemples de code, consultezUtilisation de ce service avec un AWS SDK. Cette rubrique comprend également des informations sur le démarrage et sur les versions précédentes de SDK.

Sécurité dans Amazon RDS

La sécurité du cloud AWS est la priorité absolue. En tant que AWS client, vous bénéficiez d'un centre de données et d'une architecture réseau conçus pour répondre aux exigences des entreprises les plus sensibles en matière de sécurité.

La sécurité est une responsabilité partagée entre vous AWS et vous. Le modèle de responsabilité partagée décrit cette notion par les termes sécurité du cloud et sécurité dans le cloud :

- Sécurité du cloud : AWS est chargée de protéger l'infrastructure qui exécute les AWS services dans le AWS cloud. AWS vous fournit également des services que vous pouvez utiliser en toute sécurité. Des auditeurs tiers testent et vérifient régulièrement l'efficacité de notre sécurité dans le cadre des programmes de conformitéAWS. Pour en savoir plus sur les programmes de conformité qui s'appliquent à Amazon RDS , consultez la section AWS Services concernés par programme de conformité.
- Sécurité dans le cloud — Votre responsabilité est déterminée par le AWS service que vous utilisez. Vous êtes également responsable d'autres facteurs, y compris la sensibilité de vos données, les exigences de votre organisation, et la législation et la réglementation applicables.

Cette documentation vous aide à comprendre comment appliquer le modèle de responsabilité partagée lors de l'utilisation d'Amazon RDS (). Les rubriques suivantes expliquent comment configurer Amazon RDS (Amazon) pour répondre à vos objectifs de sécurité et de conformité. Vous apprendrez également à utiliser d'autres AWS services qui vous aident à surveiller et à sécuriser vos ressources Amazon RDS .

Vous pouvez gérer l'accès à vos ressources Amazon RDS et à vos bases de données sur un d'instances de base de données. La méthode que vous utilisez pour gérer l'accès dépend du type de tâche que l'utilisateur doit effectuer avec Amazon RDS :

- Exécutez votre d'instances de base de données dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC pour un contrôle d'accès réseau optimal. Pour plus d'informations sur la création d'un d'instances de base de données dans un VPC, consultez. Amazon VPC et Amazon RDS
- Utilisez des politiques AWS Identity and Access Management (IAM) pour attribuer des autorisations qui déterminent qui est autorisé à gérer les ressources Amazon RDS . Par exemple, vous pouvez utiliser IAM pour déterminer qui est autorisé à créer, décrire, modifier et supprimer des d'instances de base de données, à étiqueter des ressources ou à modifier des groupes de sécurité.

- Utilisez des groupes de sécurité pour contrôler les adresses IP ou les EC2 instances Amazon qui peuvent se connecter à vos bases de données sur un d'instances de base de données. Lorsque vous créez un d'instances de base de données pour la première fois, son pare-feu empêche tout accès à la base de données, sauf par le biais de règles spécifiées par un groupe de sécurité associé.
- Utilisez des connexions SSL (Secure Socket Layer) ou TLS (Transport Layer Security) avec des instances de base de données exécutant les moteurs de base de données DB2, MySQL, MariaDB, PostgreSQL, Oracle ou Microsoft SQL Server. Pour plus d'informations sur l'utilisation de SSL/TLS avec un d'instances de base de données, consultez.
- Utilisez le chiffrement Amazon RDS pour sécuriser vos instances de base de données, vos de données et vos instantanés au repos. Amazon RDS Pour plus d'informations, consultez Chiffrement des ressources Amazon RDS.
- Utilisez un chiffrement réseau et un chiffrement TDE (Transparent Data Encryption) avec les instances de base de données Oracle. Pour plus d'informations, consultez Oracle NNE (Native Network Encryption) et Oracle Transparent Data Encryption
- Utilisez les fonctionnalités de sécurité de votre moteur de base de données pour contrôler qui peut se connecter aux bases de données d'un d'instances de base de données. Ces fonctions agissent comme si la base de données se trouvait sur votre réseau local.

> **Note**
>
> Vous devez uniquement configurer la sécurité de vos cas d'utilisation. Vous n'avez pas à configurer l'accès de sécurité pour les processus gérés par Amazon RDS. Cela inclut, par exemple, la création de sauvegardes et la réplication de données entre une instance de base de données principale et un réplica en lecture, et d'autres processus.

Pour plus d'informations sur la gestion de l'accès aux ressources Amazon RDS et à vos bases de données sur un d'instances de base de données, consultez les rubriques suivantes.

Rubriques

- Authentification de base de données avec Amazon RDS
- Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager
- Protection des données dans Amazon RDS
- Gestion des identités et des accès pour Amazon RDS

- Journalisation et surveillance dans Amazon RDS
- Validation de conformité pour Amazon RDS
- Résilience dans Amazon RDS
- Sécurité de l'infrastructure dans Amazon RDS
- API Amazon RDS et points de terminaison d'un VPC d'interface (AWS PrivateLink)
- Bonnes pratiques de sécurité pour Amazon RDS
- Contrôle d'accès par groupe de sécurité
- Privilèges du compte utilisateur principal
- Utilisation des rôles liés à un service pour Amazon RDS
- Amazon VPC et Amazon RDS

Authentification de base de données avec Amazon RDS

Amazon RDS prend en charge plusieurs méthodes pour authentifier les utilisateurs de bases de données.

L'authentification par mot de passe, Kerberos et IAM utilisent différentes méthodes d'authentification auprès de la base de données. Par conséquent, un utilisateur spécifique peut se connecter à une base de données en utilisant une seule méthode d'authentification.

Pour PostgreSQL, utilisez un seul des paramètres de rôle suivants pour un utilisateur d'une base de données spécifique :

- Pour utiliser l'authentification de base de données IAM, affectez le rôle `rds_iam` à l'utilisateur.
- Pour utiliser l'authentification Kerberos, affectez le rôle `rds_ad` à l'utilisateur.
- Pour utiliser l'authentification par mot de passe, n'affectez pas les rôles `rds_iam` ou `rds_ad` à l'utilisateur.

N'affectez pas à la fois les rôles `rds_iam` et `rds_ad` à un utilisateur d'une base de données PostgreSQL, directement ou indirectement par l'intermédiaire d'un accès accordé imbriqué. Si le rôle `rds_iam` est ajouté à l'utilisateur principal, l'authentification IAM a priorité sur l'authentification par mot de passe, de sorte que l'utilisateur principal doit se connecter en tant qu'utilisateur IAM.

> **Important**
>
> Nous vous recommandons vivement de ne pas avoir recours au rôle d'utilisateur principal directement dans vos applications. Au lieu de cela, respectez la bonne pratique qui consiste à avoir recours à un utilisateur de base de données doté des privilèges minimum requis pour votre application.

Rubriques

- Authentification par mot de passe
- Authentification de base de données IAM
- Authentification Kerberos

Authentification par mot de passe

Avec l'authentification par mot de passe, votre base de données se charge de toute l'administration des comptes utilisateurs. Vous créez des utilisateurs avec des instructions SQL telles que `CREATE USER`, avec la clause appropriée requise par le moteur de base de données pour spécifier des mots de passe. Par exemple, dans MySQL, l'instruction est `CREATE USER` *`name`* `IDENTIFIED BY` *`password`*, tandis que dans PostgreSQL, l'instruction est. `CREATE USER` *`name`* `WITH PASSWORD` *`password`*

Avec l'authentification par mot de passe, votre base de données contrôle et authentifie les comptes d'utilisateurs. Si un moteur de base de données dispose de fonctionnalités de gestion de mot de passe solides, il peut améliorer la sécurité. L'authentification de base de données peut être plus facile à administrer en utilisant l'authentification par mot de passe lorsque vous avez de petites communautés d'utilisateurs. Étant donné que des mots de passe en texte clair sont générés dans ce cas, leur intégration AWS Secrets Manager peut améliorer la sécurité.

Pour plus d'informations sur l'utilisation de Secrets Manager avec Amazon RDS , consultez les sections Création d'un secret de base et Rotation de secrets pour les bases de données Amazon RDS prises en charge dans le Guide de l'AWS Secrets Manager utilisateur. Pour plus d'informations sur la récupération par programme de vos secrets dans vos applications personnalisées, consultez Récupération de la valeur de secret dans le Guide de l'utilisateur AWS Secrets Manager .

Authentification de base de données IAM

Vous pouvez vous authentifier auprès de votre d'instances de base de données à l'aide de l'authentification de base de données AWS Identity and Access Management (IAM). Avec cette méthode d'authentification, vous n'avez pas besoin d'utiliser de mot de passe lorsque vous vous connectez à un d'instances de base de données. En revanche, un jeton d'authentification est nécessaire.

Pour plus d'informations sur l'authentification de base de données IAM, notamment sur la disponibilité de moteurs de base de données spécifiques, consultezAuthentification de base de données IAM pour MariaDB, MySQL et PostgreSQL.

Authentification Kerberos

RDS prend en charge l'authentification externe des utilisateurs de bases de données à l'aide de Kerberos et de Microsoft Active Directory. Kerberos est un protocole d'authentification réseau qui utilise les tickets et la cryptographie de clé symétrique pour vous éviter d'acheminer vos mots de passe via le réseau. Intégré dans Active Directory, Kerberos est conçu pour authentifier les utilisateurs sur les ressources réseau, par exemple les bases de données.

Le support d'Amazon RDS pour Kerberos et Active Directory offre les avantages de l'authentification unique et de l'authentification centralisée des utilisateurs de bases de données. Vous pouvez conserver vos informations d'identification utilisateur dans Active Directory. Active Directory fournit un emplacement centralisé pour le stockage et la gestion des informations d'identification pour plusieurs d'instances de base de données.

Pour utiliser les informations d'identification de votre Active Directory autogéré, vous devez configurer une relation de confiance avec Microsoft Active Directory AWS Directory Service auquel le de base de données est joint.

RDS pour PostgreSQL et RDS pour MySQL Aurora PostgreSQL et Aurora MySQL prennent en charge les avec une authentification à l'échelle de la forêt ou une authentification sélective.

Dans certains scénarios, vous pouvez configurer l'authentification Kerberos via une relation de confiance externe. Cela nécessite que votre Active Directory autogéré dispose de paramètres supplémentaires. Cela inclut, mais sans s'y limiter, l'ordre de recherche des forêts Kerberos.

Les instances de base de données Microsoft SQL Server et PostgreSQL prennent en charge les relations de confiance unidirectionnelles et bidirectionnelles en matière de forêt. Les instances

de base de données Oracle prennent en charge les relations de confiance externes et forestières unidirectionnelles et bidirectionnelles. Pour plus d'informations, veuillez consulter Quand créer une relation d'approbation dans le Guide d'administration AWS Directory Service .

Pour plus d'informations sur l'authentification Kerberos avec un moteur de base de données spécifique, veuillez consulter les sections suivantes :

- Utilisation d'Active Directory AWS géré avec RDS pour SQL Server
- Utilisation de Kerberos l'authentification pour Amazon RDS for MySQL
- Configuration de l'authentification Kerberos pour Amazon RDS for Oracle
- Utilisation de l'authentification Kerberos avec Amazon RDS for PostgreSQL
- Utilisation de Kerberos l'authentification pour Amazon RDS pour Db2 .

> **Note**
>
> Actuellement, l'authentification Kerberos n'est pas prise en charge pour les instances de base de données MariaDB.

Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager

Amazon RDS s'intègre à Secrets Manager pour gérer les mots de passe d'utilisateur principal de vos clusters de bases de données multi-AZ et de vos instances de bases de données.

Rubriques

- Limites de l'intégration de Secrets Manager avec Amazon RDS
- Présentation de la gestion des mots de passe des utilisateurs principaux avec AWS Secrets Manager
- Avantages de la gestion des mots de passe d'utilisateur principal avec Secrets Manager
- Autorisations requises pour l'intégration de Secrets Manager
- Application de la gestion du mot de passe de l'utilisateur principal par RDS dans AWS Secrets Manager
- Gestion du mot de passe d'utilisateur principal pour une instance de base de données avec Secrets Manager
- Gestion du mot de passe utilisateur principal pour une base de données cliente RDS pour Oracle avec Secrets Manager
- Gestion du mot de passe d'utilisateur principal pour un cluster de bases de données multi-AZ avec Secrets Manager
- Rotation du secret de mot de passe d'utilisateur principal pour une instance de base de données
- Rotation du secret de mot de passe d'utilisateur principal pour un cluster de bases de données multi-AZ
- Affichage des détails concernant un secret pour une instance de base de données
- Affichage des détails concernant un secret pour un cluster de bases de données multi-AZ
- Afficher les détails d'un secret pour une base de données mutualisée
- Disponibilité des régions et des versions

Limites de l'intégration de Secrets Manager avec Amazon RDS

La gestion des mots de passe d'utilisateur principal à l'aide de Secrets Manager n'est pas prise en charge pour les fonctionnalités suivantes :

- Création d'une réplique en lecture lorsque la base de données source ou le cluster de base de données gère les informations d'identification avec Secrets Manager. Cela s'applique à tous les moteurs de base de données, à l'exception de RDS pour SQL Server.
- Déploiements Amazon RDS Blue/Green
- Amazon RDS Custom
- Basculement Oracle Data Guard

Présentation de la gestion des mots de passe des utilisateurs principaux avec AWS Secrets Manager

Vous pouvez AWS Secrets Manager ainsi remplacer les informations d'identification codées en dur dans votre code, y compris les mots de passe de base de données, par un appel d'API à Secrets Manager pour récupérer le secret par programmation. Pour plus d'informations sur Secrets Manager, consultez le Guide de l'utilisateur AWS Secrets Manager.

Lorsque vous stockez des secrets de base de données dans Secrets Manager, des frais Compte AWS vous sont facturés. Pour plus d'informations sur la tarification, consultez Tarification AWS Secrets Manager.

Vous pouvez spécifier que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager pour une instance de base de données Amazon RDS ou un cluster de bases de données multi-AZ quand vous effectuez l'une des opérations suivantes :

- Créez une instance de base de données.
- Création d'un cluster de base de données multi-AZ
- Création d'une base de données client dans un RDS pour Oracle CDB
- Modification d'une instance de base de données
- Modifier un cluster de base de données multi-AZ
- Modifier une base de données client (RDS pour Oracle uniquement)
- Restaurer une instance de base de données depuis Amazon S3
- Restaurer une instance de base de données à partir d'un instantané ou à un moment donné (RDS pour Oracle uniquement)

Quand vous spécifiez que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager, RDS génère le mot de passe et le stocke dans Secrets Manager. Vous pouvez interagir

directement avec le secret pour récupérer les informations d'identification de l'utilisateur principal. Vous pouvez également spécifier une clé gérée par le client pour chiffrer le secret, ou utiliser la clé KMS fournie par Secrets Manager.

RDS gère les paramètres du secret et effectue la rotation du secret tous les sept jours, par défaut. Vous pouvez modifier certains paramètres, tels que la planification de la rotation. Si vous supprimez une instance de base de données qui gère un secret dans Secrets Manager, le secret et les métadonnées associées sont également supprimés.

Pour vous connecter à un cluster de bases de données multi-AZ ou à une instance de base de données avec les informations d'identification contenues dans un secret, vous pouvez récupérer le secret à partir de Secrets Manager. Pour plus d'informations, voir Extraire des secrets depuis une base de données SQL AWS Secrets Manager et Se connecter à une base de données SQL avec des informations d'identification inscrites dans un AWS Secrets Manager secret dans le Guide de AWS Secrets Manager l'utilisateur.

Avantages de la gestion des mots de passe d'utilisateur principal avec Secrets Manager

La gestion des mots de passe d'utilisateur principal RDS avec Secrets Manager présente les avantages suivants :

- RDS génère automatiquement des informations d'identification de base de données.
- RDS stocke et gère automatiquement les informations d'identification de la base de données dans AWS Secrets Manager.
- RDS effectue une rotation régulière des informations d'identification de base de données, sans exiger de modifications d'application.
- Secrets Manager sécurise les informations d'identification de base de données contre tout accès humain et tout affichage en texte brut.
- Secrets Manager permet de récupérer les informations d'identification de base de données dans des secrets pour les connexions à une base de données.
- Secrets Manager permet un contrôle précis de l'accès aux informations d'identification de base de données dans des secrets à l'aide d'IAM.
- Vous pouvez éventuellement séparer le chiffrement d'une base de données du chiffrement des informations d'identification à l'aide de clés KMS différentes.
- Vous pouvez éliminer la gestion et la rotation manuelles des informations d'identification de base de données.

- Vous pouvez facilement surveiller les informations d'identification de la base AWS CloudTrail de données avec Amazon CloudWatch.

Pour en savoir plus sur les avantages de Secrets Manager, consultez le Guide de l'utilisateur AWS Secrets Manager.

Autorisations requises pour l'intégration de Secrets Manager

Les utilisateurs doivent disposer des autorisations requises pour effectuer des opérations liées à l'intégration de Secrets Manager. Vous pouvez créer des politiques IAM qui accordent des autorisations pour effectuer des opérations API spécifiques sur les ressources spécifiées dont ils ont besoin. Vous pouvez ensuite attacher ces politiques aux jeux d'autorisations ou rôles IAM qui requièrent ces autorisations. Pour de plus amples informations, veuillez consulter Gestion des identités et des accès pour Amazon RDS.

Pour les opérations de création, de modification ou de restauration, l'utilisateur qui spécifie qu'Amazon RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager doit avoir les autorisations nécessaires pour effectuer les opérations suivantes :

- `kms:DescribeKey`
- `secretsmanager:CreateSecret`
- `secretsmanager:TagResource`

L'`kms:DescribeKey`autorisation est requise pour accéder à votre clé gérée par le client pour `MasterUserSecretKmsKeyId` et pour décrire. `aws/secretsmanager`

Pour les opérations de création, de modification ou de restauration, l'utilisateur qui spécifie la clé gérée par le client pour chiffrer le secret dans Secrets Manager doit avoir les autorisations nécessaires pour effectuer les opérations suivantes :

- `kms:Decrypt`
- `kms:GenerateDataKey`
- `kms:CreateGrant`

Pour les opérations de modification, l'utilisateur qui effectue la rotation du mot de passe d'utilisateur principal dans Secrets Manager doit être autorisé à effectuer l'opération suivante :

- `secretsmanager:RotateSecret`

Application de la gestion du mot de passe de l'utilisateur principal par RDS dans AWS Secrets Manager

Vous pouvez utiliser des clés de condition IAM pour mettre en œuvre la gestion par RDS du mot de passe d'utilisateur principal dans AWS Secrets Manager. La politique suivante n'autorise pas les utilisateurs à créer ou à restaurer des instances de base de données ou des clusters de bases de données, ni à créer ou à modifier des bases de données mutualisées, sauf si le mot de passe de l'utilisateur principal est géré par RDS dans Secrets Manager.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Deny",
            "Action": ["rds:CreateDBInstance", "rds:CreateDBCluster",
 "rds:RestoreDBInstanceFromS3", "rds:RestoreDBClusterFromS3",
                       "rds:RestoreDBInstanceFromDBSnapshot",
 "rds:RestoreDBInstanceToPointInTime", "rds:CreateTenantDatabase",
                       "rds:ModifyTenantDatabase"],
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "rds:ManageMasterUserPassword": false
                }
            }
        }
    ]
}
```

> **Note**
>
> Cette politique impose la gestion des mots de passe dès AWS Secrets Manager leur création. Toutefois, vous pouvez toujours désactiver l'intégration de Secrets Manager et définir manuellement un mot de passe principal en modifiant l'instance.

Pour éviter cela, incluez `rds:ModifyDBInstance`, `rds:ModifyDBCluster` dans le bloc action de la politique. Sachez que cela empêche l'utilisateur d'appliquer d'autres modifications aux instances existant(e)s n'ayant pas l'intégration de Secrets Manager activée.

Pour plus d'informations sur l'utilisation de clés de condition dans les politiques IAM, consultez Clés de conditions de politique pour Amazon RDS et Exemples de politiques : Utilisation des clés de condition.

Gestion du mot de passe d'utilisateur principal pour une instance de base de données avec Secrets Manager

Vous pouvez configurer la gestion RDS du mot de passe d'utilisateur principal dans Secrets Manager lorsque vous effectuez les actions suivantes :

- Création d'une instance de base de données Amazon RDS
- Modification d'une RDS instance de base de données Amazon
- Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL
- Restauration vers une instance de base de données(RDS pour Oracle uniquement)
- Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS(RDS pour Oracle uniquement)

Vous pouvez effectuer les opérations précédentes à l'aide de la console RDS, de l' AWS CLI API RDS ou de l'API RDS.

console

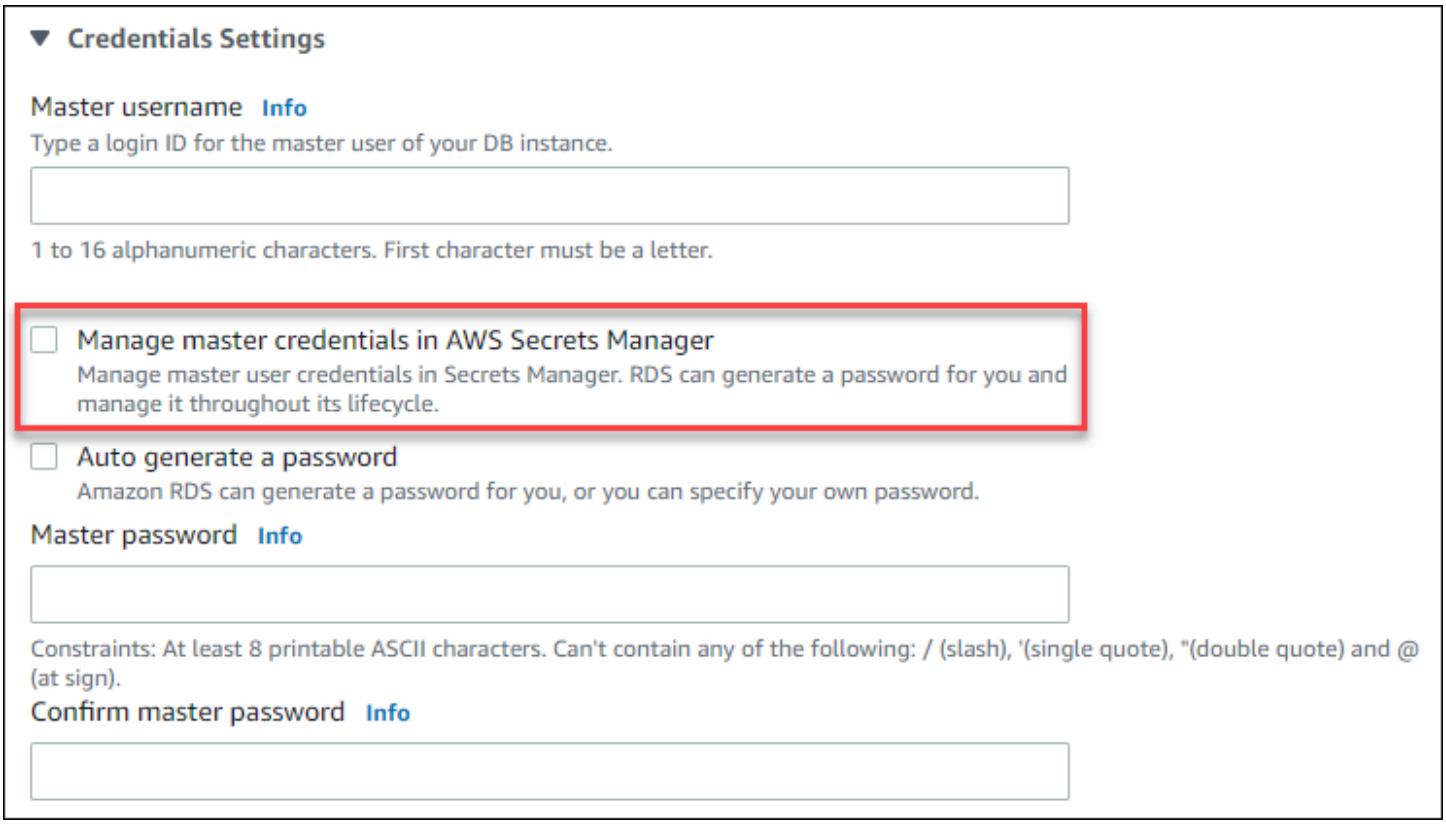
Suivez les instructions pour créer ou modifier une instance de base de données à l'aide de la console RDS :

- Création d'une instance de base de données
- Modification d'une RDS instance de base de données Amazon
- Pour importer des données à partir d'Amazon S3 vers une nouvelle instance de base de données MySQL

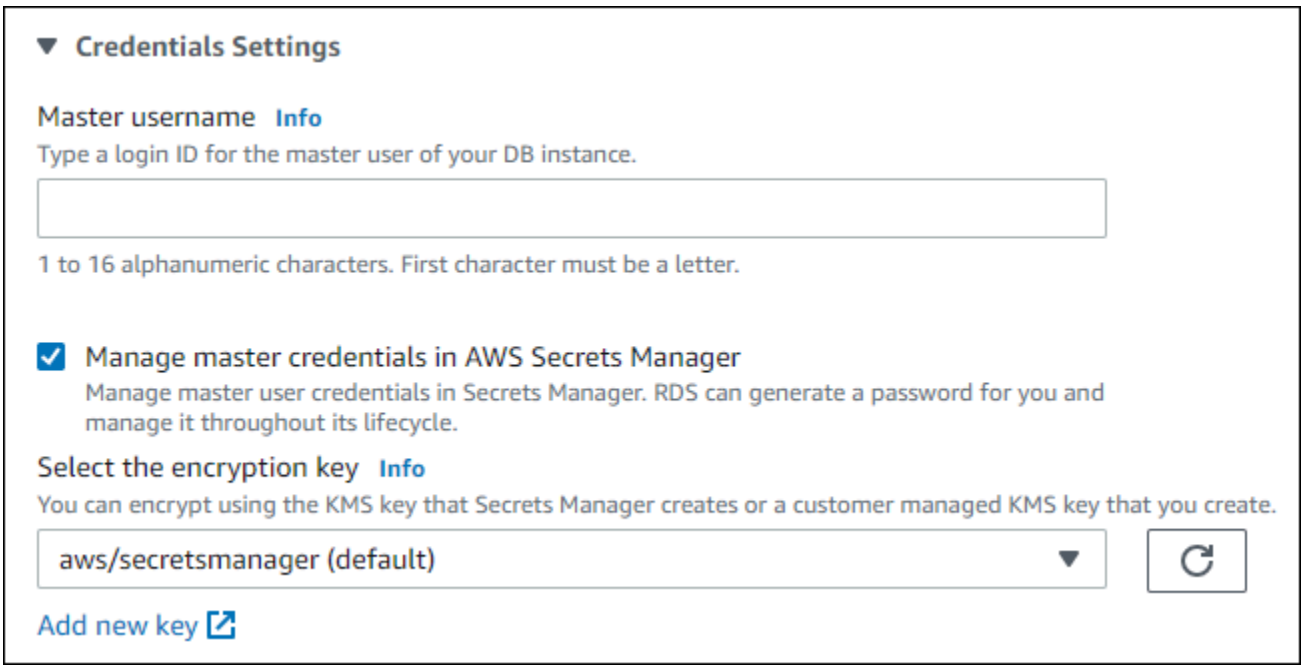
Lorsque vous utilisez la console RDS pour effectuer l'une de ces opérations, vous pouvez spécifier que le mot de passe d'utilisateur principal doit être géré par RDS dans Secrets Manager.

Lorsque vous créez ou restaurez une instance de base de données, sélectionnez Gérer les informations d'identification principales AWS Secrets Manager dans les paramètres des informations d'identification. Lorsque vous modifiez une instance de base de données, sélectionnez Gérer les informations d'identification principales AWS Secrets Manager dans les paramètres.

L'image suivante est un exemple du paramètre Manage master credentials in AWS Secrets Manager (Gérer les informations d'identification principales dans) lors de la création ou de la restauration d'une instance de base de données.



Lorsque vous sélectionnez cette option, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.



Vous pouvez choisir de chiffrer le secret à l'aide d'une clé KMS fournie par Secrets Manager ou d'une clé gérée par le client que vous créez. Une fois que RDS a géré les informations d'identification de base de données pour une instance de base de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins. Pour plus d'informations sur les paramètres disponibles lorsque vous créez une instance de base de données, consultezParamètres des instances de base de données. Pour plus d'informations sur les paramètres disponibles lorsque vous modifiez une instance de base de données, consultezParamètres des instances de base de données.

AWS CLI

Pour gérer le mot de passe de l'utilisateur principal avec RDS dans Secrets Manager, spécifiez l'`--manage-master-user-password`option dans l'une des AWS CLI commandes suivantes :

- create-db-instance
- modify-db-instance
- restore-db-instance-from-s3
- restore-db-instance-from-db-snapshot (RDS pour Oracle uniquement)
- restore-db-instance-to- point-in-time (RDS pour Oracle uniquement)

Lorsque vous spécifiez l'option `--manage-master-user-password` dans ces commandes, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez l'option `--master-user-secret-kms-key-id` pour spécifier une clé gérée par le client. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans une autre Compte AWS, spécifiez l'ARN de la clé ou l'alias ARN. Quand RDS gère les informations d'identification de base de données pour une instance de base de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins. Pour plus d'informations sur les paramètres disponibles quand vous créez une instance de base de données, consultez Paramètres des instances de base de données. Pour plus d'informations sur les paramètres disponibles quand vous modifiez une instance de base de données, consultez Paramètres des instances de base de données.

L'exemple suivant crée une instance de base de données et indique que RDS gère le mot de passe de l'utilisateur principal dans Secrets Manager. Ce secret est chiffré à l'aide de la clé KMS fournie par Secrets Manager.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --engine mysql \
    --engine-version 8.0.39 \
    --db-instance-class db.r5b.large \
    --allocated-storage 200 \
    --master-username testUser \
    --manage-master-user-password
```

Dans Windows :

```
aws rds create-db-instance ^
    --db-instance-identifier mydbinstance ^
    --engine mysql ^
```

```
    --engine-version 8.0.39 ^
    --db-instance-class db.r5b.large ^
    --allocated-storage 200 ^
    --master-username testUser ^
    --manage-master-user-password
```

API RDS

Pour spécifier que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager, affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une des opérations d'API RDS suivantes :

- CréerDBInstance
- ModifyDBInstance
- Restaurer à DBInstance partir de S3
- Restauration DBInstance FromSnapshot (RDS pour Oracle uniquement)
- Restauration DBInstance ToPointInTime (RDS pour Oracle uniquement)

Lorsque vous affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une de ces opérations, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez le paramètre `MasterUserSecretKmsKeyId` pour spécifier une clé gérée par le client. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans un autre Compte AWS, spécifiez l'ARN de la clé ou l'ARN de l'alias. Quand RDS gère les informations d'identification de base de données pour une instance de base de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Gestion du mot de passe utilisateur principal pour une base de données cliente RDS pour Oracle avec Secrets Manager

Vous pouvez configurer la gestion RDS du mot de passe d'utilisateur principal dans Secrets Manager lorsque vous effectuez les actions suivantes :

- Ajout d'une base de données locataire RDS for Oracle à votre instance de CDB
- Modification d'une base de données locataire RDS for Oracle

Vous pouvez utiliser la console RDS AWS CLI, ou l'API RDS pour effectuer les actions précédentes.

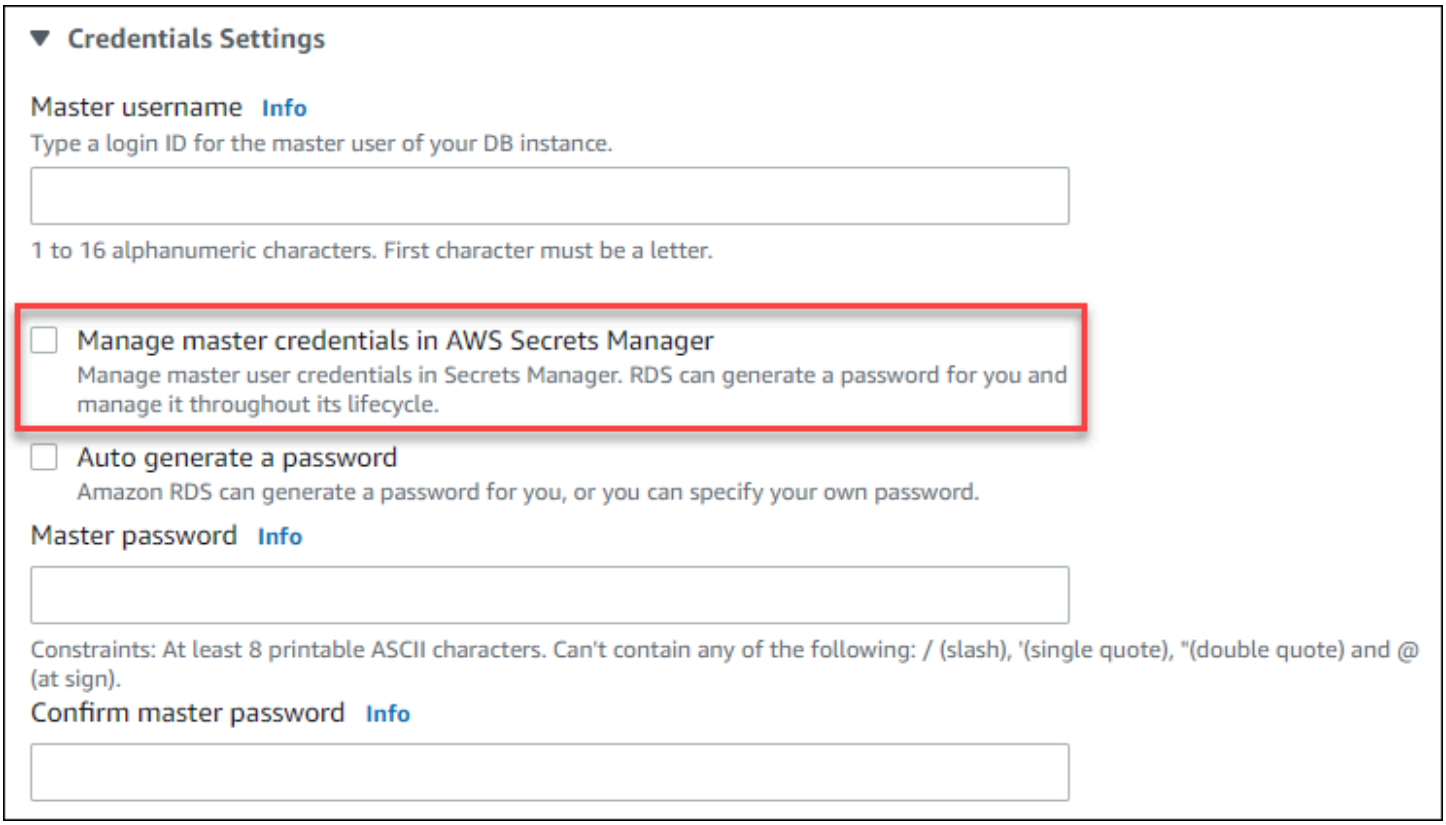
console

Suivez les instructions pour créer ou modifier une base de données client RDS pour Oracle à l'aide de la console RDS :

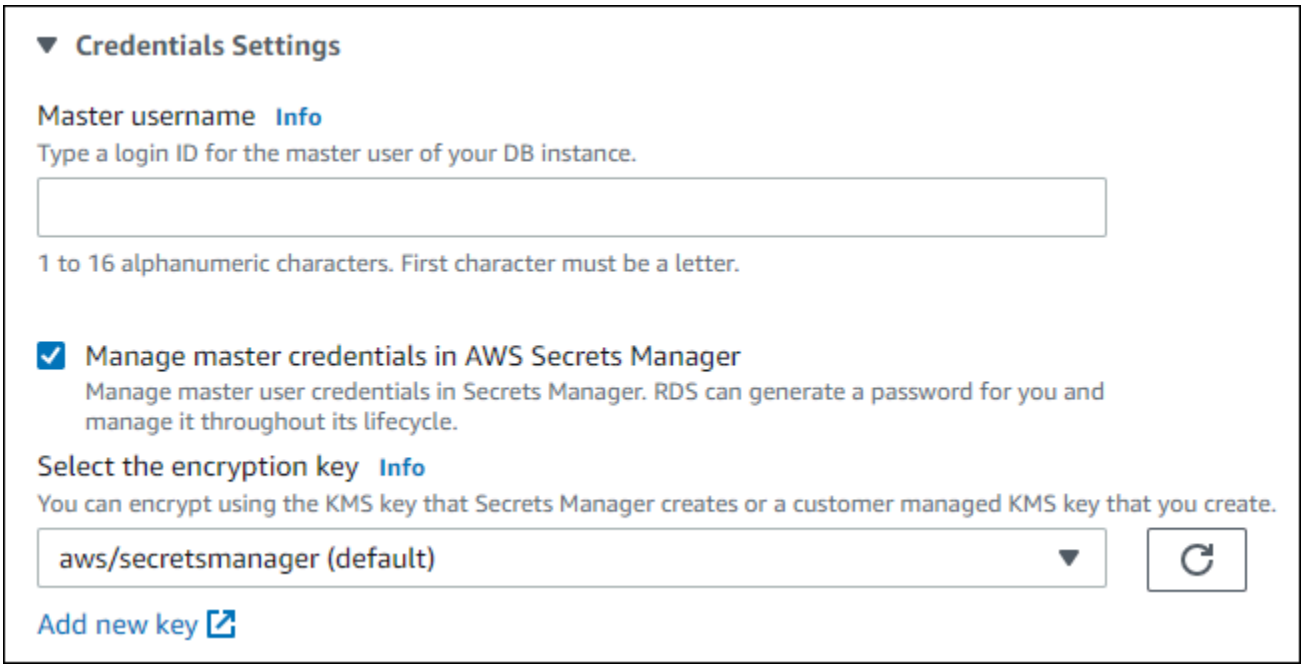
- Ajout d'une base de données locataire RDS for Oracle à votre instance de CDB
- Modification d'une base de données locataire RDS for Oracle

Lorsque vous utilisez la console RDS pour effectuer l'une des opérations précédentes, vous pouvez spécifier que RDS gère le mot de passe principal dans Secrets Manager. Lorsque vous créez une base de données de locataires, sélectionnez Gérer les informations d'identification principales AWS Secrets Manager dans les paramètres des informations d'identification. Lorsque vous modifiez une base de données mutualisée, sélectionnez Gérer les informations d'identification principales AWS Secrets Manager dans les paramètres.

L'image suivante est un exemple du AWS Secrets Manager paramètre Gérer les informations d'identification principales lors de la création d'une base de données mutualisée.



Lorsque vous sélectionnez cette option, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.



Vous pouvez choisir de chiffrer le secret à l'aide d'une clé KMS fournie par Secrets Manager ou d'une clé gérée par le client que vous créez. Une fois que RDS a géré les informations d'identification d'une base de données mutualisée, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins. Pour plus d'informations sur les paramètres disponibles lors de la création d'une base de données mutualisée, consultezParamètres des instances de base de données. Pour plus d'informations sur les paramètres disponibles lorsque vous modifiez une base de données mutualisée, consultezParamètres des instances de base de données.

AWS CLI

Pour gérer le mot de passe d'utilisateur principal avec RDS dans Secrets Manager, spécifiez l'option `--manage-master-user-password` dans l'une des commandes AWS CLI suivantes :

- create-tenant-database
- modify-tenant-database

Lorsque vous spécifiez l'`--manage-master-user-password`option dans les commandes précédentes, RDS génère le mot de passe de l'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez l'option `--master-user-secret-kms-key-id` pour spécifier une clé gérée par le client. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans une autre Compte AWS, spécifiez l'ARN de la clé ou l'alias ARN. Une fois que RDS a géré les informations d'identification d'une base de données mutualisée, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins. Pour plus d'informations sur les paramètres disponibles lors de la création d'une base de données mutualisée, consultez create-tenant-database. Pour plus d'informations sur les paramètres disponibles lorsque vous modifiez une base de données mutualisée, consultez modify-tenant-database.

L'exemple suivant crée une base de données client RDS pour Oracle et indique que RDS gère le mot de passe de l'utilisateur principal dans Secrets Manager. Ce secret est chiffré à l'aide de la clé KMS fournie par Secrets Manager.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-tenant-database --region us-east-1 \
    --db-instance-identifier my-cdb-inst \
    --tenant-db-name mypdb2 \
    --master-username mypdb2-admin \
    --character-set-name UTF-16 \
    --manage-master-user-password
```

Dans Windows :

```
aws rds create-tenant-database --region us-east-1 ^
    --db-instance-identifier my-cdb-inst ^
    --tenant-db-name mypdb2 ^
    --master-username mypdb2-admin ^
    --character-set-name UTF-16 ^
    --manage-master-user-password
```

API RDS

Pour spécifier que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager, affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une des opérations d'API RDS suivantes :

- CreateTenantDatabase
- ModifyTenantDatabase

Lorsque vous affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une de ces opérations, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez le paramètre `MasterUserSecretKmsKeyId` pour spécifier une clé gérée par le client. L'identifiant de clé KMS AWS est l'ARN de clé, l'ID de clé, l'ARN d'alias ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans une autre Compte AWS, spécifiez l'ARN de la clé ou l'alias ARN. Une fois que RDS a géré les informations d'identification d'une base de données mutualisée, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Gestion du mot de passe d'utilisateur principal pour un cluster de bases de données multi-AZ avec Secrets Manager

Vous pouvez configurer la gestion RDS du mot de passe d'utilisateur principal dans Secrets Manager lorsque vous effectuez les actions suivantes :

- Création d'un cluster de base de données multi-AZ pour Amazon RDS
- Modification d'un cluster de base de données multi-AZ pour Amazon RDS

Vous pouvez utiliser la console RDS AWS CLI, ou l'API RDS pour effectuer ces actions.

console

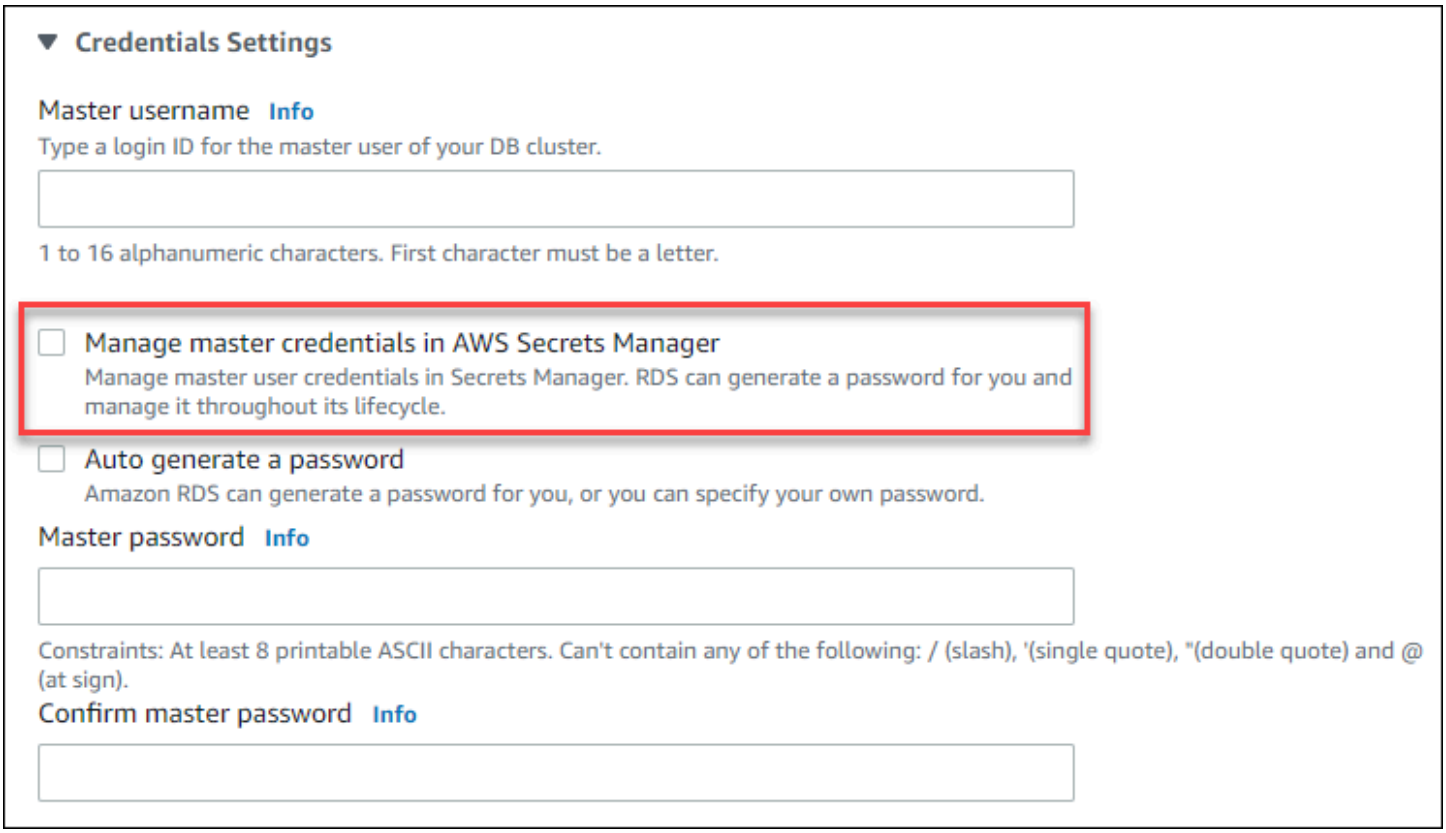
Suivez les instructions pour créer ou modifier un cluster de bases de données multi-AZ à l'aide de la console RDS :

- Création d'un cluster de base de données

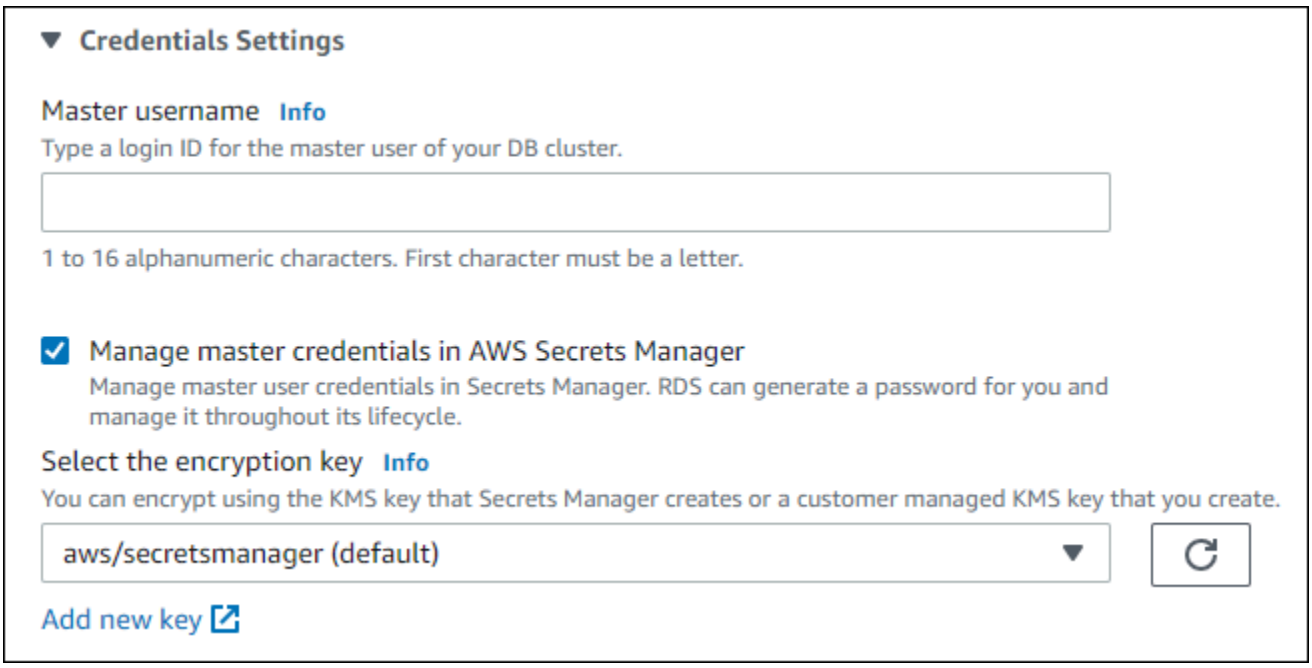
- Modification d'un cluster de base de données multi-AZ pour Amazon RDS

Lorsque vous utilisez la console RDS pour effectuer l'une de ces opérations, vous pouvez spécifier que le mot de passe d'utilisateur principal est géré par RDS dans Secrets Manager. Pour ce faire, lorsque vous créez un cluster de bases de données, sélectionnez Manage master credentials in AWS Secrets Manager (Gérer les informations d'identification principales dans) dans Credential settings (Paramètres des informations d'identification). Lorsque vous modifiez un cluster de bases de données, sélectionnez Manage master credentials in AWS Secrets Manager (Gérer les informations d'identification principales dans) dans Settings (Paramètres).

L'image suivante est un exemple du paramètre Manage master credentials in AWS Secrets Manager (Gérer les informations d'identification principales dans) lors de la création d'un cluster de bases de données.



Lorsque vous sélectionnez cette option, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.



Vous pouvez choisir de chiffrer le secret à l'aide d'une clé KMS fournie par Secrets Manager ou d'une clé gérée par le client que vous créez. Quand RDS gère les informations d'identification de base de données pour un cluster de bases de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins.

Pour plus d'informations sur les paramètres disponibles quand vous créez un cluster de bases de données multi-AZ, consultez Paramètres de création de clusters de base de données multi-AZ. Pour plus d'informations sur les paramètres disponibles quand vous modifiez un cluster de bases de données multi-AZ, consultez Paramètres de modification des clusters de base de données multi-AZ.

AWS CLI

Pour spécifier que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager, spécifiez l'option `--manage-master-user-password` dans l'une des commandes suivantes :

- create-db-cluster
- modify-db-cluster

Lorsque vous spécifiez l'option `--manage-master-user-password` dans ces commandes, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez l'option `--master-user-secret-kms-key-id` pour spécifier une clé gérée par le client. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans une autre Compte AWS, spécifiez l'ARN de la clé ou l'alias ARN. Quand RDS gère les informations d'identification de base de données pour un cluster de bases de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Vous pouvez choisir d'autres paramètres en fonction de vos besoins.

Pour plus d'informations sur les paramètres disponibles quand vous créez un cluster de bases de données multi-AZ, consultez Paramètres de création de clusters de base de données multi-AZ. Pour plus d'informations sur les paramètres disponibles quand vous modifiez un cluster de bases de données multi-AZ, consultez Paramètres de modification des clusters de base de données multi-AZ.

Cet exemple crée un cluster de bases de données multi-AZ et spécifie que RDS doit gérer le mot de passe dans Secrets Manager. Ce secret est chiffré à l'aide de la clé KMS fournie par Secrets Manager.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds create-db-cluster \
   --db-cluster-identifier mysql-multi-az-db-cluster \
   --engine mysql \
   --engine-version 8.0.39  \
   --backup-retention-period 1  \
   --allocated-storage 4000 \
   --storage-type io1 \
   --iops 10000 \
   --db-cluster-instance-class db.r6gd.xlarge \
   --master-username testUser \
   --manage-master-user-password
```

Dans Windows :

```
aws rds create-db-cluster ^
   --db-cluster-identifier mysql-multi-az-db-cluster ^
   --engine mysql ^
   --engine-version 8.0.39 ^
   --backup-retention-period 1 ^
```

```
    --allocated-storage 4000 ^
    --storage-type io1 ^
    --iops 10000 ^
    --db-cluster-instance-class db.r6gd.xlarge ^
    --master-username testUser ^
    --manage-master-user-password
```

API RDS

Pour spécifier que RDS doit gérer le mot de passe d'utilisateur principal dans Secrets Manager, affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une des opérations suivantes :

- CréerDBCluster
- ModifyDBCluster

Lorsque vous affectez au paramètre `ManageMasterUserPassword` la valeur `true` dans l'une de ces opérations, RDS génère le mot de passe d'utilisateur principal et le gère tout au long de son cycle de vie dans Secrets Manager.

Pour chiffrer le secret, vous pouvez spécifier une clé gérée par le client ou utiliser la clé KMS par défaut, fournie par Secrets Manager. Utilisez le paramètre `MasterUserSecretKmsKeyId` pour spécifier une clé gérée par le client. L'identifiant de clé AWS KMS est l'ARN de la clé, l'ID de clé, l'alias ARN ou le nom d'alias de la clé KMS. Pour utiliser une clé KMS dans un autre Compte AWS, spécifiez l'ARN de la clé ou l'ARN de l'alias. Quand RDS gère les informations d'identification de base de données pour un cluster de bases de données, vous ne pouvez pas modifier la clé KMS utilisée pour chiffrer le secret.

Rotation du secret de mot de passe d'utilisateur principal pour une instance de base de données

Quand RDS effectue la rotation d'un secret de mot de passe d'utilisateur principal, Secrets Manager génère une nouvelle version de secret pour le secret existant. La nouvelle version du secret contient le nouveau mot de passe d'utilisateur principal. Amazon RDS modifie le mot de passe d'utilisateur principal de l'instance de base de données pour qu'il corresponde au mot de passe de la nouvelle version de secret.

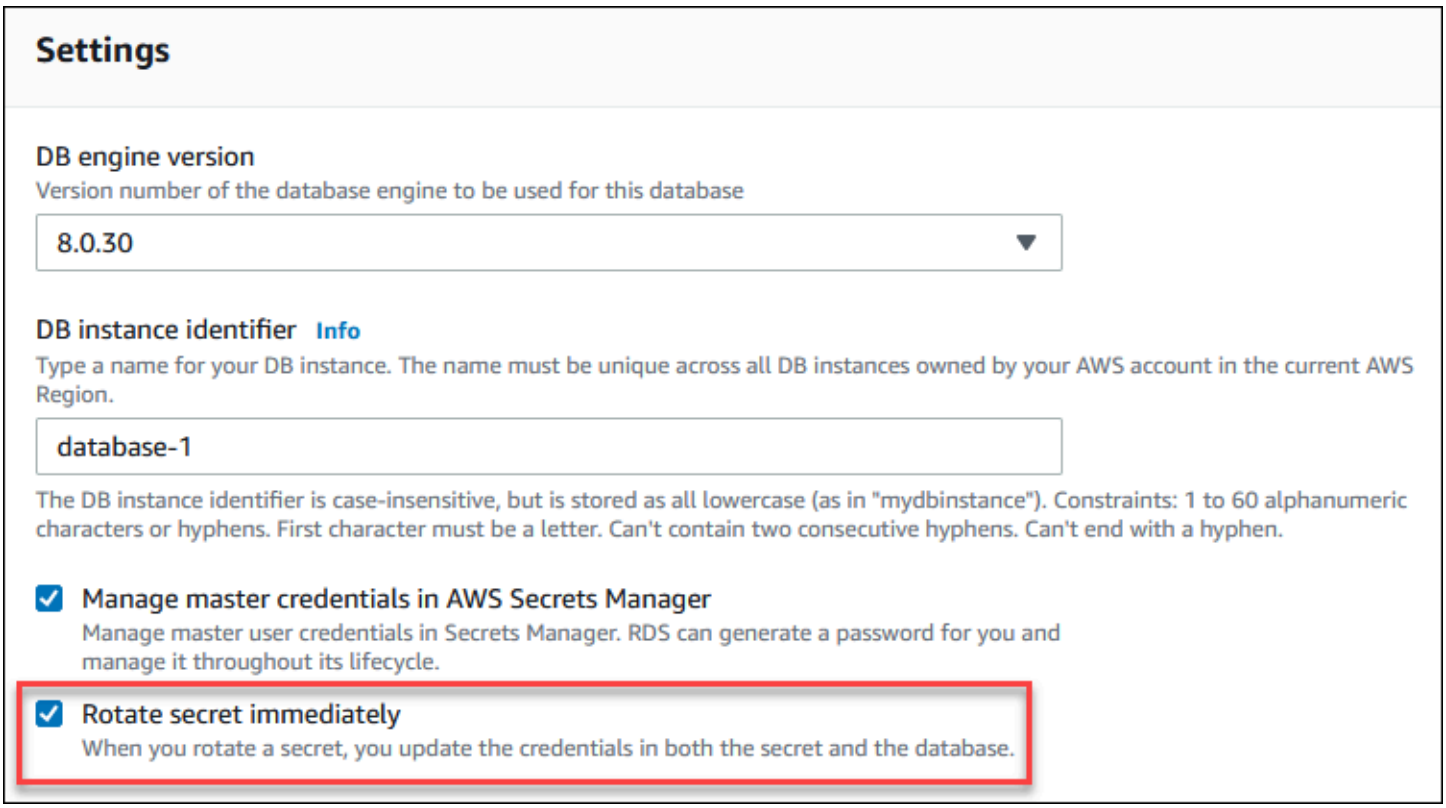
Vous pouvez effectuer immédiatement la rotation d'un secret au lieu d'attendre une rotation planifiée. Pour effectuer la rotation d'un secret de mot de passe d'utilisateur principal dans Secrets Manager,

modifiez l'instance de base de données. Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Vous pouvez modifier immédiatement le secret d'un mot de passe utilisateur principal à l'aide de la console RDS, de l' AWS CLI API RDS ou de l'API RDS. Le nouveau mot de passe comporte toujours 28 caractères, dont au moins une majuscule et une minuscule, un chiffre et un signe de ponctuation.

console

Pour effectuer la rotation d'un secret de mot de passe d'utilisateur principal à l'aide de la console RDS, modifiez l'instance de base de données et sélectionnez Rotate secret immediately (Effectuer immédiatement une rotation du secret) dans Settings (Paramètres).



Suivez les instructions pour modifier une instance de base de données à l'aide de la console RDS dans Modification d'une RDS instance de base de données Amazon. Vous devez choisir Apply immediately (Appliquer immédiatement) sur la page de confirmation.

AWS CLI

Pour faire pivoter le secret du mot de passe d'un utilisateur principal à l'aide de AWS CLI, utilisez la modify-db-instancecommande et spécifiez l'`--rotate-master-user-password`option. Vous

devez spécifier l'option `--apply-immediately` lorsque vous effectuez la rotation du mot de passe principal.

Cet exemple effectue la rotation d'un secret de mot de passe d'utilisateur principal.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --rotate-master-user-password \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --rotate-master-user-password ^
    --apply-immediately
```

API RDS

Vous pouvez faire pivoter le secret du mot de passe d'un utilisateur principal à l'aide de DBInstance l'opération Modifier et en définissant le `RotateMasterUserPassword` paramètre sur`true`. Vous devez affecter au paramètre `ApplyImmediately` la valeur `true` lorsque vous effectuez la rotation du mot de passe principal.

Rotation du secret de mot de passe d'utilisateur principal pour un cluster de bases de données multi-AZ

Quand RDS effectue la rotation d'un secret de mot de passe d'utilisateur principal, Secrets Manager génère une nouvelle version de secret pour le secret existant. La nouvelle version du secret contient le nouveau mot de passe d'utilisateur principal. Amazon RDS modifie le mot de passe d'utilisateur principal du cluster de bases de données multi-AZ pour qu'il corresponde au mot de passe de la nouvelle version de secret.

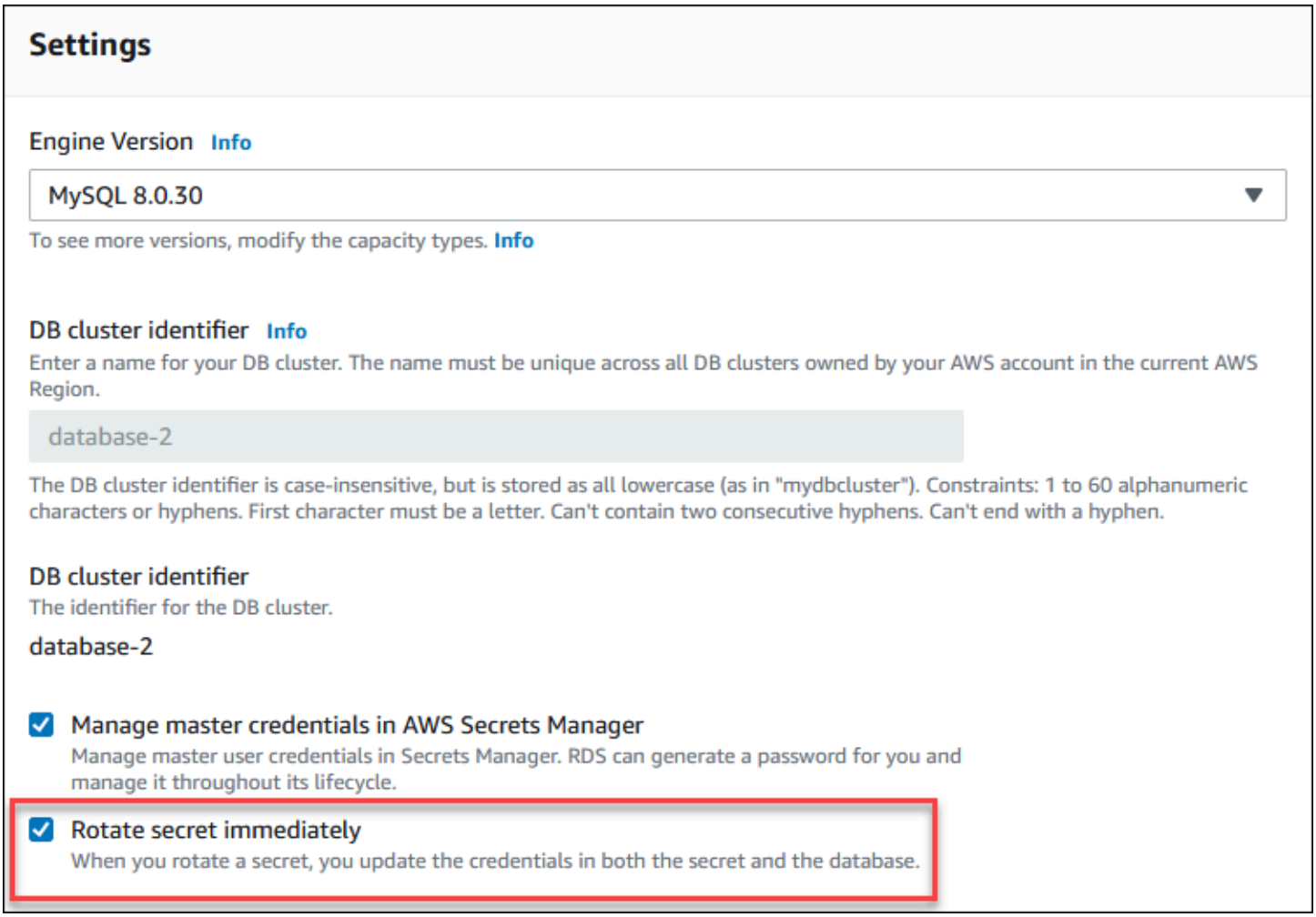
Vous pouvez effectuer immédiatement la rotation d'un secret au lieu d'attendre une rotation planifiée. Pour effectuer la rotation d'un secret de mot de passe d'utilisateur principal dans Secrets Manager,

modifiez le cluster de bases de données multi-AZ. Pour obtenir des informations sur la modification d'un cluster de bases de données multi-AZ, consultez Modification d'un cluster de base de données multi-AZ pour Amazon RDS.

Vous pouvez modifier immédiatement le secret d'un mot de passe utilisateur principal à l'aide de la console RDS, de l' AWS CLI API RDS ou de l'API RDS. Le nouveau mot de passe comporte toujours 28 caractères, dont au moins une majuscule et une minuscule, un chiffre et un signe de ponctuation.

console

Pour effectuer la rotation d'un secret de mot de passe d'utilisateur principal à l'aide de la console RDS, modifiez le cluster de bases de données multi-AZ et sélectionnez Rotate secret immediately (Effectuer immédiatement une rotation du secret) dans Settings (Paramètres).



Suivez les instructions pour modifier un cluster de bases de données multi-AZ avec la console RDS dans Modification d'un cluster de base de données multi-AZ pour Amazon RDS. Vous devez choisir Apply immediately (Appliquer immédiatement) sur la page de confirmation.

AWS CLI

Pour faire pivoter le secret du mot de passe d'un utilisateur principal à l'aide de AWS CLI, utilisez la modify-db-clustercommande et spécifiez l'`--rotate-master-user-password`option. Vous devez spécifier l'option `--apply-immediately` lorsque vous effectuez la rotation du mot de passe principal.

Cet exemple effectue la rotation d'un secret de mot de passe d'utilisateur principal.

Example

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-cluster \
    --db-cluster-identifier mydbcluster \
    --rotate-master-user-password \
    --apply-immediately
```

Dans Windows :

```
aws rds modify-db-cluster ^
    --db-cluster-identifier mydbcluster ^
    --rotate-master-user-password ^
    --apply-immediately
```

API RDS

Vous pouvez faire pivoter le secret du mot de passe d'un utilisateur principal à l'aide de DBCluster l'opération Modifier et en définissant le `RotateMasterUserPassword` paramètre sur`true`. Vous devez affecter au paramètre `ApplyImmediately` la valeur `true` lorsque vous effectuez la rotation du mot de passe principal.

Affichage des détails concernant un secret pour une instance de base de données

Vous pouvez récupérer vos secrets à l'aide de la console (https://console.aws.amazon.com/secretsmanager/) ou de la commande AWS CLI (get-secret-valueSecrets Manager).

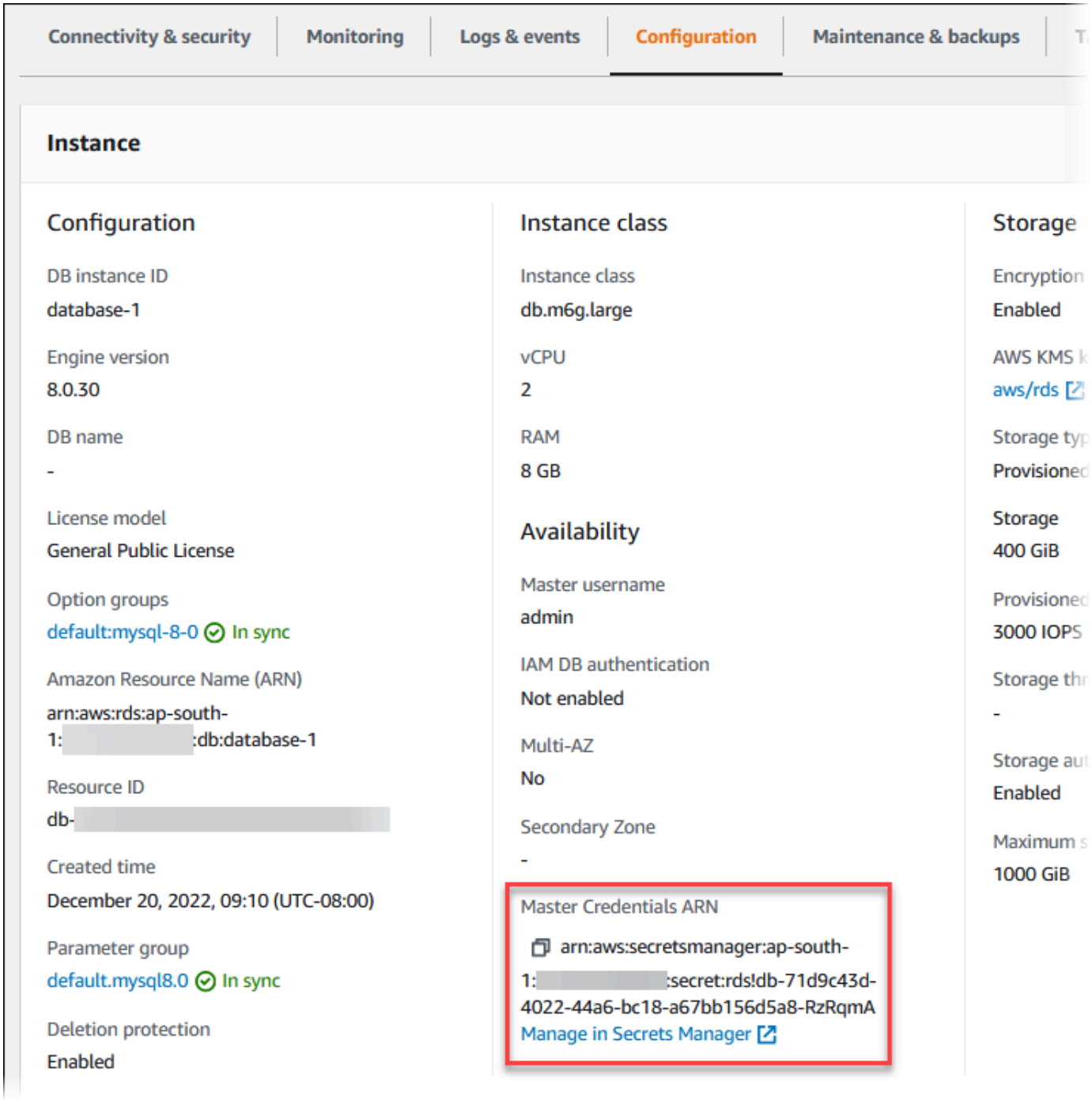
Vous pouvez trouver le Amazon Resource Name (ARN) d'un secret géré par RDS dans Secrets Manager avec la console RDS AWS CLI, ou l'API RDS.

console

Pour afficher les détails d'un secret géré par RDS dans Secrets Manager

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez le nom de l'instance de base de données pour afficher ses détails.
4. Cliquez sur l’onglet Configuration.

   Dans Master Credentials ARN (ARN des informations d'identification principales), vous pouvez consulter l'ARN du secret.



Vous pouvez suivre le lien Manage in Secrets Manager (Gérer dans Secrets Manager) pour consulter et gérer le secret dans la console Secrets Manager.

AWS CLI

Vous pouvez utiliser la commande describe-db-instancesRDS CLI pour trouver les informations suivantes sur un secret géré par RDS dans Secrets Manager :

- `SecretArn` : l'ARN du secret
- `SecretStatus` : le statut du secret

  Les valeurs de statut possibles incluent les suivantes :

  - `creating` : le secret est en cours de création.
  - `active` : le secret est disponible pour une utilisation et une rotation normales.
  - `rotating` : la rotation du secret est en cours.
  - `impaired` : le secret peut être utilisé pour accéder aux informations d'identification de base de données, mais il est impossible d'effectuer sa rotation. Un secret peut avoir ce statut si, par exemple, les autorisations sont modifiées de telle sorte que RDS ne puisse plus accéder au secret ou à la clé KMS associée à ce secret.

    Lorsqu'un secret possède ce statut, vous pouvez corriger la condition à l'origine de ce statut. Si vous corrigez la condition à l'origine du statut, celui-ci reste `impaired` jusqu'à la rotation suivante. Vous pouvez également modifier l'instance de base de données pour désactiver la gestion automatique des informations d'identification de base de données, puis modifier à nouveau l'instance de base de données pour activer la gestion automatique des informations d'identification de base de données. Pour modifier l'instance de base de données, utilisez l'`--manage-master-user-password`option de la modify-db-instancecommande.
- `KmsKeyId` : l'ARN de la clé KMS utilisée pour chiffrer le secret

Spécifiez l'option `--db-instance-identifier` permettant d'afficher la sortie pour une instance de base de données spécifique. Cet exemple montre la sortie d'un secret utilisé par une instance de base de données.

Example

```
aws rds describe-db-instances --db-instance-identifier mydbinstance
```

Voici un exemple de sortie pour un secret :

```
"MasterUserSecret": {
                "SecretArn": "arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
db-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx",
                "SecretStatus": "active",
                "KmsKeyId": "arn:aws:kms:eu-
west-1:123456789012:key/0987dcba-09fe-87dc-65ba-ab0987654321"
```

```
            }
```

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de la commande get-secret-valueSecrets Manager CLI.

Cet exemple montre les détails du secret dans l'exemple de sortie précédent.

Example

Pour LinuxmacOS, ou Unix :

```
aws secretsmanager get-secret-value \
    --secret-id 'arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
db-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx'
```

Dans Windows :

```
aws secretsmanager get-secret-value ^
    --secret-id 'arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
db-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx'
```

API RDS

Vous pouvez consulter l'ARN, le statut et la clé KMS d'un secret géré par RDS dans Secrets Manager en utilisant l'DBInstancesopération Describe et en définissant le `DBInstanceIdentifier` paramètre sur un identifiant d'instance de base de données. Les détails sur le secret sont inclus dans la sortie.

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de l'opération GetSecretValueSecrets Manager.

Affichage des détails concernant un secret pour un cluster de bases de données multi-AZ

Vous pouvez récupérer vos secrets à l'aide de la console (https://console.aws.amazon.com/secretsmanager/) ou de la commande AWS CLI (get-secret-valueSecrets Manager).

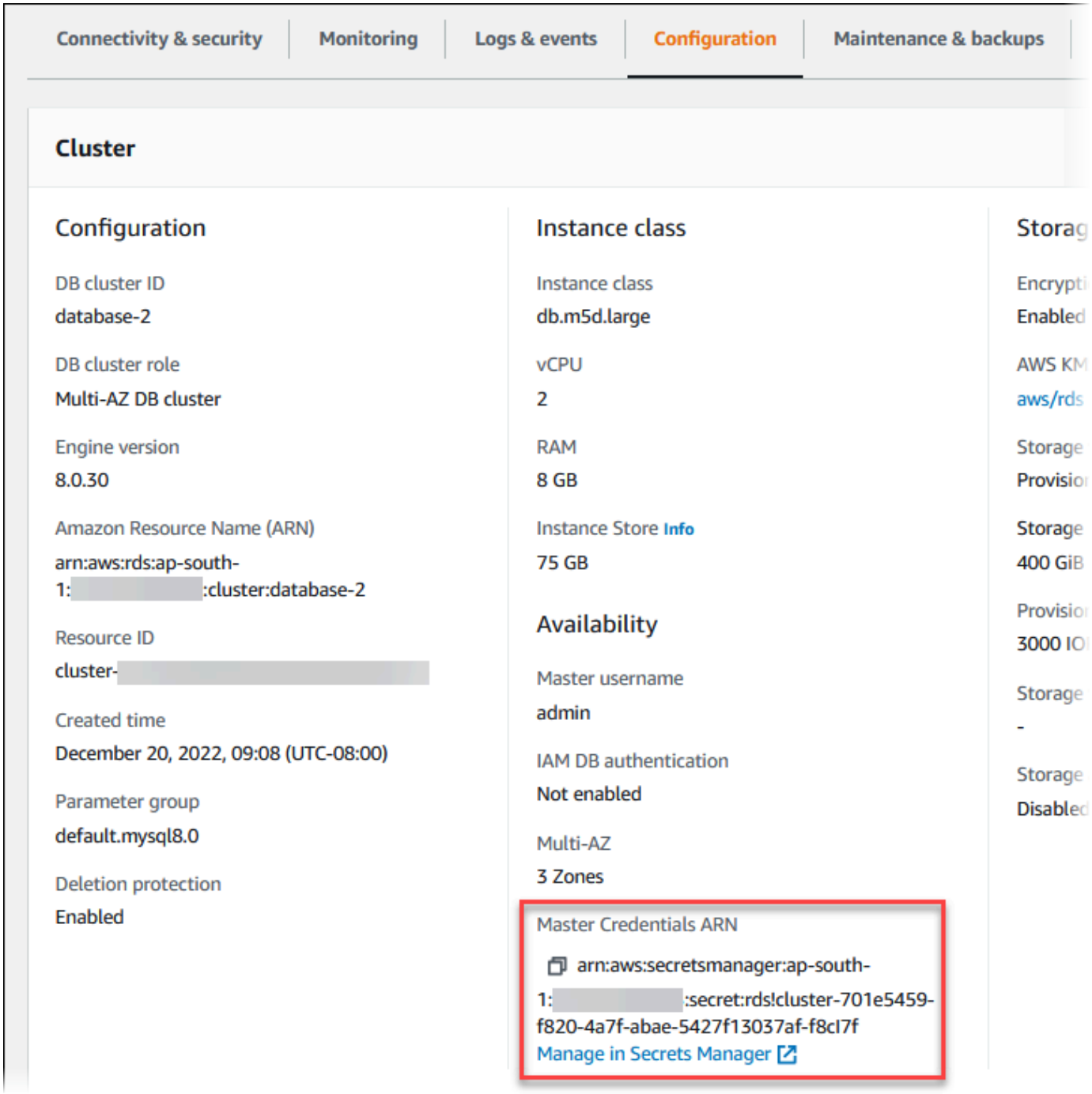
Vous pouvez trouver le Amazon Resource Name (ARN) d'un secret géré par RDS dans Secrets Manager avec la console RDS AWS CLI, ou l'API RDS.

console

Pour afficher les détails d'un secret géré par RDS dans Secrets Manager

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom du cluster de bases de données multi-AZ pour afficher ses détails.
4. Cliquez sur l’onglet Configuration.

   Dans Master Credentials ARN (ARN des informations d'identification principales), vous pouvez consulter l'ARN du secret.



Vous pouvez suivre le lien Manage in Secrets Manager (Gérer dans Secrets Manager) pour consulter et gérer le secret dans la console Secrets Manager.

AWS CLI

Vous pouvez utiliser la AWS CLI describe-db-clusterscommande RDS pour trouver les informations suivantes sur un secret géré par RDS dans Secrets Manager :

- `SecretArn` : l'ARN du secret
- `SecretStatus` : le statut du secret

  Les valeurs de statut possibles incluent les suivantes :

  - `creating` : le secret est en cours de création.
  - `active` : le secret est disponible pour une utilisation et une rotation normales.
  - `rotating` : la rotation du secret est en cours.
  - `impaired` : le secret peut être utilisé pour accéder aux informations d'identification de base de données, mais il est impossible d'effectuer sa rotation. Un secret peut avoir ce statut si, par exemple, les autorisations sont modifiées de telle sorte que RDS ne puisse plus accéder au secret ou à la clé KMS associée à ce secret.

    Lorsqu'un secret possède ce statut, vous pouvez corriger la condition à l'origine de ce statut. Si vous corrigez la condition à l'origine du statut, celui-ci reste `impaired` jusqu'à la rotation suivante. Vous pouvez également modifier le cluster de bases de données pour désactiver la gestion automatique des informations d'identification de base de données, puis modifier à nouveau le cluster de bases de données pour activer la gestion automatique des informations d'identification de base de données. Pour modifier le cluster de base de données, utilisez l'`--manage-master-user-password`option de la modify-db-clustercommande.
- `KmsKeyId` : l'ARN de la clé KMS utilisée pour chiffrer le secret

Spécifiez l'option `--db-cluster-identifier` permettant d'afficher la sortie pour un cluster de bases de données spécifique. Cet exemple montre la sortie d'un secret utilisé par un cluster de bases de données.

Example

```
aws rds describe-db-clusters --db-cluster-identifier mydbcluster
```

L'exemple suivant montre la sortie pour un secret :

```
"MasterUserSecret": {
                "SecretArn": "arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
cluster-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx",
                "SecretStatus": "active",
                "KmsKeyId": "arn:aws:kms:eu-
west-1:123456789012:key/0987dcba-09fe-87dc-65ba-ab0987654321"
```

```
        }
```

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de la commande get-secret-valueSecrets Manager CLI.

Cet exemple montre les détails du secret dans l'exemple de sortie précédent.

Example

Pour LinuxmacOS, ou Unix :

```
aws secretsmanager get-secret-value \
    --secret-id 'arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
cluster-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx'
```

Dans Windows :

```
aws secretsmanager get-secret-value ^
    --secret-id 'arn:aws:secretsmanager:eu-west-1:123456789012:secret:rds!
cluster-033d7456-2c96-450d-9d48-f5de3025e51c-xmJRDx'
```

API RDS

Vous pouvez consulter l'ARN, le statut et la clé KMS d'un secret géré par RDS dans Secrets Manager à l'aide de l'opération Describe DBClusters RDS et en définissant le `DBClusterIdentifier` paramètre sur un identifiant de cluster de base de données. Les détails sur le secret sont inclus dans la sortie.

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de l'opération GetSecretValueSecrets Manager.

Afficher les détails d'un secret pour une base de données mutualisée

Vous pouvez récupérer vos secrets à l'aide de la console (https://console.aws.amazon.com/secretsmanager/) ou de la commande AWS CLI (get-secret-valueSecrets Manager).

Vous pouvez trouver le nom de ressource Amazon (ARN) d'un secret géré par Amazon RDS dans AWS Secrets Manager la console Amazon RDS, l'API Amazon RDS ou l' AWS CLI API Amazon RDS.

console

Pour consulter les informations relatives à un secret géré par Amazon RDS dans une base AWS Secrets Manager de données mutualisée

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez le nom de l'instance de base de données qui contient la base de données mutualisée pour afficher ses détails.
4. Cliquez sur l'onglet Configuration.

   Dans la section Bases de données des locataires, recherchez la base de données des locataires et consultez son ARN d'identification principal.

   Vous pouvez suivre le lien Manage in Secrets Manager (Gérer dans Secrets Manager) pour consulter et gérer le secret dans la console Secrets Manager.

AWS CLI

Vous pouvez utiliser la AWS CLI commande describe-tenant-databasesAmazon RDS pour trouver les informations suivantes concernant un secret géré par Amazon RDS dans une base AWS Secrets Manager de données mutualisée :

- `SecretArn` : l'ARN du secret
- `SecretStatus` : le statut du secret

  Les valeurs de statut possibles incluent les suivantes :

  - `creating` : le secret est en cours de création.
  - `active` : le secret est disponible pour une utilisation et une rotation normales.
  - `rotating` : la rotation du secret est en cours.
  - `impaired` : le secret peut être utilisé pour accéder aux informations d'identification de base de données, mais il est impossible d'effectuer sa rotation. Un secret peut avoir ce statut si, par exemple, les autorisations sont modifiées afin qu'Amazon RDS ne puisse plus accéder au secret ou à la clé KMS correspondant au secret.

Lorsqu'un secret possède ce statut, vous pouvez corriger la condition à l'origine de ce statut. Si vous corrigez la condition à l'origine du statut, celui-ci reste `impaired` jusqu'à la rotation suivante. Vous pouvez également modifier la base de données client pour désactiver la gestion automatique des informations d'identification de base de données, puis modifier à nouveau la base de données client pour activer la gestion automatique des informations d'identification de base de données. Pour modifier la base de données des locataires, utilisez l'`--manage-master-user-password`option de la modify-tenant-databasecommande.

- `KmsKeyId` : l'ARN de la clé KMS utilisée pour chiffrer le secret

Spécifiez l'`--db-instance-identifier`option permettant d'afficher la sortie des bases de données mutualisées dans une instance de base de données spécifique. Vous pouvez également spécifier l'`--tenant-db-name`option permettant d'afficher la sortie d'une base de données mutualisée spécifique. Cet exemple montre le résultat d'un secret utilisé par une base de données mutualisée.

Example

```
aws rds describe-tenant-databases \
    --db-instance-identifier database-3 \
    --query "TenantDatabases[0].MasterUserSecret"
```

Voici un exemple de sortie pour un secret :

```
{
    "SecretArn": "arn:aws:secretsmanager:us-east-2:123456789012:secret:rds!db-ABC123",
    "SecretStatus": "active",
    "KmsKeyId": "arn:aws:kms:us-east-2:123456789012:key/aa11bb22-####-####-####-
fedcba123456"
}
```

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de la AWS CLI commande get-secret-valueSecrets Manager.

Cet exemple montre les détails du secret dans l'exemple de sortie précédent.

Example

Pour LinuxmacOS, ou Unix :

```
aws secretsmanager get-secret-value \
    --secret-id 'arn:aws:secretsmanager:us-east-2:123456789012:secret:rds!db-ABC123'
```

Dans Windows :

```
aws secretsmanager get-secret-value ^
    --secret-id 'arn:aws:secretsmanager:us-east-2:123456789012:secret:rds!db-ABC123'
```

API Amazon RDS

Vous pouvez consulter l'ARN, le statut et la clé KMS d'un secret géré par Amazon RDS en AWS Secrets Manager utilisant l'DescribeTenantDatabasesopération et en définissant le `DBInstanceIdentifier` paramètre sur un identifiant d'instance de base de données. Vous pouvez également définir le `TenantDBName` paramètre sur un nom de base de données client spécifique. Les détails sur le secret sont inclus dans la sortie.

Lorsque vous disposez de l'ARN secret, vous pouvez consulter les détails du secret à l'aide de l'opération GetSecretValueSecrets Manager.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour plus d'informations sur la disponibilité des versions et des régions avec l'intégration de Secrets Manager avec Amazon RDS, consultez Régions et moteurs de base de données pris en charge pour l'intégration de Secrets Manager à Amazon RDS.

Protection des données dans Amazon RDS

Le modèle de responsabilité AWS partagée s'applique à la protection des données dans Amazon Relational Database Service. Comme décrit dans ce modèle, AWS est chargé de protéger l'infrastructure mondiale qui gère tous les AWS Cloud. La gestion du contrôle de votre contenu hébergé sur cette infrastructure relève de votre responsabilité. Vous êtes également responsable des tâches de configuration et de gestion de la sécurité des Services AWS que vous utilisez. Pour plus d'informations sur la confidentialité des données, consultez Questions fréquentes (FAQ) sur la confidentialité des données. Pour en savoir plus sur la protection des données en Europe, consultez le billet de blog Modèle de responsabilité partagée AWS et RGPD (Règlement général sur la protection des données) sur le Blog de sécuritéAWS .

À des fins de protection des données, nous vous recommandons de protéger les Compte AWS informations d'identification et de configurer les utilisateurs individuels avec AWS IAM Identity Center ou AWS Identity and Access Management (IAM). Ainsi, chaque utilisateur se voit attribuer uniquement les autorisations nécessaires pour exécuter ses tâches. Nous vous recommandons également de sécuriser vos données comme indiqué ci-dessous :

- Utilisez l'authentification multifactorielle (MFA) avec chaque compte.
- Utilisez le protocole SSL/TLS pour communiquer avec les ressources. AWS Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Configurez l'API et la journalisation de l'activité des utilisateurs avec AWS CloudTrail. Pour plus d'informations sur l'utilisation des CloudTrail sentiers pour capturer AWS des activités, consultez la section Utilisation des CloudTrail sentiers dans le guide de AWS CloudTrail l'utilisateur.
- Utilisez des solutions de AWS chiffrement, ainsi que tous les contrôles de sécurité par défaut qu'ils contiennent Services AWS.
- Utilisez des services de sécurité gérés avancés tels qu'Amazon Macie, qui contribuent à la découverte et à la sécurisation des données sensibles stockées dans Amazon S3.
- Si vous avez besoin de modules cryptographiques validés par la norme FIPS 140-3 pour accéder AWS via une interface de ligne de commande ou une API, utilisez un point de terminaison FIPS. Pour plus d'informations sur les points de terminaison FIPS disponibles, consultez Norme FIPS (Federal Information Processing Standard) 140-3.

Nous vous recommandons fortement de ne jamais placer d'informations confidentielles ou sensibles, telles que les adresses e-mail de vos clients, dans des balises ou des champs de texte libre tels que le champ Nom. Cela inclut lorsque vous travaillez avec Amazon RDS ou une autre entreprise à Services AWS l'aide de la console, de l'API ou AWS SDKs. AWS CLI Toutes les données que vous entrez dans des balises ou des champs de texte de forme libre utilisés pour les noms peuvent être utilisées à des fins de facturation ou dans les journaux de diagnostic. Si vous fournissez une adresse URL à un serveur externe, nous vous recommandons fortement de ne pas inclure d'informations d'identification dans l'adresse URL permettant de valider votre demande adressée à ce serveur.

Rubriques

- Protection des données à l'aide du chiffrement
- Confidentialité du trafic inter-réseau

Protection des données à l'aide du chiffrement

Vous pouvez activer le chiffrement pour vos ressources de base de données. Vous pouvez également chiffrer les connexions aux d'instances de base de données.

Rubriques

- Chiffrement des ressources Amazon RDS
- AWS KMS key gestion
-
- Rotation de votre certificat SSL/TLS

Chiffrement des ressources Amazon RDS

Amazon RDS peut chiffrer vos Amazon RDS clusters d'instances. Les données chiffrées au repos incluent le stockage sous-jacent pour les instances de base de données, les sauvegardes automatiques, les réplicas en lecture et les instantanés.

Les instances de base de données chiffrée Amazon RDS utilisent l'algorithme de chiffrement AES-256 standard pour chiffrer vos données sur le serveur qui héberge vos instances de base de données Amazon RDS. Une fois que vos données ont été chiffrées, Amazon RDS traite l'authentification de l'accès et le déchiffrement de vos données de façon transparente, avec un impact minimal sur les performances. Vous n'avez pas besoin de modifier vos applications clientes de base de données pour utiliser le chiffrement.

> **Note**
>
> Pour les d'instances de base de données chiffrés et non chiffrés, les données en transit entre la source et les répliques lues sont chiffrées, même lors de la réplication entre régions. AWS

Rubriques

- Présentation du chiffrement des ressources Amazon RDS
- Chiffrement d'une instance de base de données
- Détermination si le chiffrement est activé pour une instance de base de données
- Disponibilité du chiffrement Amazon RDS

- Chiffrement en transit
- Limitations des instances de base de données chiffrées Amazon RDS

Présentation du chiffrement des ressources Amazon RDS

Les instances de base de données chiffrée Amazon RDS fournissent une couche supplémentaire de protection des données en sécurisant vos données contre tout accès non autorisé au stockage sous-jacent. Vous pouvez utiliser le chiffrement Amazon RDS for renforcer la protection des données de vos applications déployées dans le cloud et pour satisfaire aux exigences de conformité pour le chiffrement au repos. Pour une instance de base de données chiffrée Amazon RDS, l'ensemble des journaux, sauvegardes et instantanés sont chiffrés. Pour plus d'informations sur la disponibilité et les limites du chiffrement, consultez Disponibilité du chiffrement Amazon RDS etLimitations des instances de base de données chiffrées Amazon RDS.

Amazon RDS utilise une AWS Key Management Service clé pour chiffrer ces ressources. AWS KMS combine du matériel et des logiciels sécurisés et hautement disponibles pour fournir un système de gestion des clés adapté au cloud. Vous pouvez utiliser un Clé gérée par AWS, ou vous pouvez créer des clés gérées par le client.

Lorsque vous créez une instance de base de données chiffrée, vous pouvez choisir une clé gérée par le client ou la Clé gérée par AWS pour Amazon RDS pour chiffrer votre instance de base de données. Si vous ne spécifiez pas l'identifiant de clé pour une clé gérée par le client, Amazon RDS l'utilise Clé gérée par AWS pour votre nouvelle instance de base de données. Amazon RDS crée un Clé gérée par AWS pour Amazon RDS pour votre AWS compte. Votre AWS compte possède un identifiant Amazon RDS différent Clé gérée par AWS pour chaque AWS région.

Pour gérer les clés gérées par le client utilisées pour chiffrer et déchiffrer vos ressources Amazon RDS, vous utilisez le ().AWS Key Management ServiceAWS KMS

À l'aide de AWS KMS, vous pouvez créer des clés gérées par le client et définir les politiques qui contrôlent l'utilisation de ces clés gérées par le client. AWS KMS prend en charge CloudTrail, afin que vous puissiez auditer l'utilisation des clés KMS afin de vérifier que les clés gérées par le client sont utilisées de manière appropriée. Vous pouvez utiliser vos clés gérées par le client avec Amazon Aurora et les AWS services pris en charge tels qu'Amazon S3, Amazon EBS et Amazon Redshift. Pour obtenir la liste des services intégrés AWS KMS, consultez la section Intégration des AWS services. Quelques considérations relatives à l'utilisation des clés KMS :

- Une fois que vous avez créé une instance de base de données chiffrée, vous ne pouvez pas modifier la clé KMS utilisée par cette instance de base de données. Vous devez donc prendre

soin de déterminer vos besoins en termes de clés KMS avant de créer votre instance de base de données chiffrée.

Si vous devez modifier la clé de chiffrement de votre instance de base de données, créez un instantané manuel de votre instance et activez le chiffrement lors de la copie de l'instantané. Pour plus d'informations, consultez l'article Re:Post Knowledge.

- Si vous copiez un instantané chiffré, vous pouvez utiliser une clé KMS différente pour chiffrer l'instantané cible que celle utilisée pour chiffrer l'instantané source.
- Une réplique en lecture d'une instance cryptée Amazon RDS doit être chiffrée à l'aide de la même clé KMS que l'instance de base de données principale lorsque les deux se trouvent dans la même AWS région.
- Si l'instance de base de données principale et la réplique en lecture se trouvent dans des AWS régions différentes, vous chiffrez la réplique en lecture à l'aide de la clé KMS de cette AWS région.
- Vous ne pouvez pas partager un instantané chiffré à l'aide Clé gérée par AWS du AWS compte qui l'a partagé.
- Amazon RDS prend également en charge le chiffrement d'une instance de base de données Oracle ou SQL Server à l'aide du chiffrement TDE (Transparent Data Encryption). Le chiffrement TDE peut être utilisé avec le chiffrement RDS au repos, bien que l'utilisation simultanée de ces deux chiffrements puisse affecter quelque peu les performances de votre base de données. Vous devez gérer des clés différentes pour chaque méthode de chiffrement. Pour plus d'informations sur TDE, veuillez consulter Oracle Transparent Data Encryption ou Support pour le chiffrement transparent des données sur le SQL serveur.

> **Important**
>
> Amazon RDS perd l'accès à la clé KMS pour une instance de base de données lorsque vous désactivez la clé KMS. Si vous perdez l'accès à une clé KMS, l'instance de base de données cryptée passe à l'`inaccessible-encryption-credentials-recoverable`état 2 heures après la détection dans les instances où les sauvegardes sont activées. L'instance de base de données reste dans cet état pendant sept jours, au cours desquels elle est arrêtée. Les appels d'API effectués vers l'instance de base de données pendant cette période risquent d'échouer. Pour récupérer l'instance de base de données, activez la clé KMS et redémarrez cette instance de base de données. Activez la clé KMS depuis l'API AWS Management Console AWS CLI, ou RDS. Redémarrez l'instance de base de données à l'aide de la AWS CLI commande start-db-instanceou AWS Management Console.

L'`inaccessible-encryption-credentials-recoverable`état s'applique uniquement aux instances de base de données qui peuvent s'arrêter. Cet état récupérable ne s'applique pas aux instances qui ne peuvent pas s'arrêter, telles que les répliques en lecture et les instances avec des répliques en lecture. Pour de plus amples informations, veuillez consulter Limites liées à l'arrêt de votre instance de base de données.

Si l'instance de base de données n'est pas récupérée dans les sept jours, elle passe à `inaccessible-encryption-credentials` l'état terminal. Dans cet état, l'instance de base de données n'est plus utilisable et vous ne pouvez la restaurer qu'à partir d'une sauvegarde. Nous vous recommandons vivement de toujours activer les sauvegardes pour les instances de base de données chiffrées afin de vous prémunir contre la perte de données chiffrées dans vos bases de données.

Lors de la création d'une instance de base de données, Amazon RDS vérifie si le principal appelant a accès à la clé KMS et génère une autorisation à partir de la clé KMS qu'il utilise pendant toute la durée de vie de l'instance de base de données. La révocation de l'accès du principal appelant à la clé KMS n'affecte pas la base de données en cours d'exécution. Lorsque vous utilisez des clés KMS dans des scénarios entre comptes, tels que la copie d'un instantané sur un autre compte, la clé KMS doit être partagée avec l'autre compte. Si vous créez une instance de base de données à partir du snapshot sans spécifier de clé KMS différente, la nouvelle instance utilise la clé KMS du compte source. La révocation de l'accès à la clé après avoir créé l'instance de base de données n'affecte pas l'instance. Toutefois, la désactivation de la clé a un impact sur toutes les instances de base de données chiffrées avec cette clé. Pour éviter cela, spécifiez une autre clé lors de l'opération de copie instantanée.

Les instances de base de données dont les sauvegardes sont désactivées restent disponibles jusqu'à ce que les volumes soient détachés de l'hôte lors d'une modification d'instance ou d'une restauration. RDS déplace les instances dans un `inaccessible-encryption-credentials-recoverable` état ou un `inaccessible-encryption-credentials` état, selon le cas.

Pour plus d'informations sur les clés KMS, consultez AWS KMS keys dans le Guide du développeur AWS Key Management Service et AWS KMS key gestion.

Chiffrement d'une instance de base de données

Pour chiffrer une nouvelle instance de base de données, sélectionnez Enable encryption (Activer le chiffrement) dans la console Amazon RDS. Pour plus d'informations sur la création d'une instance de base de données, veuillez consulter Création d'une instance de base de données Amazon RDS.

Si vous utilisez la create-db-instance AWS CLI commande pour créer une instance de base de données cryptée, définissez le `--storage-encrypted` paramètre. Si vous utilisez l'opération Create DBInstance API, définissez le `StorageEncrypted` paramètre sur true.

Si vous utilisez la AWS CLI `create-db-instance` commande pour créer une instance de base de données cryptée avec une clé gérée par le client, définissez le `--kms-key-id` paramètre sur n'importe quel identifiant de clé pour la clé KMS. Si vous utilisez l'opération Amazon RDS de l'API `CreateDBInstance`, définissez le paramètre `KmsKeyId` sur n'importe quel identifiant de clé pour la clé KMS. Pour utiliser une clé gérée par le client dans un autre AWS compte, spécifiez l'ARN de la clé ou l'alias ARN.

Détermination si le chiffrement est activé pour une instance de base de données

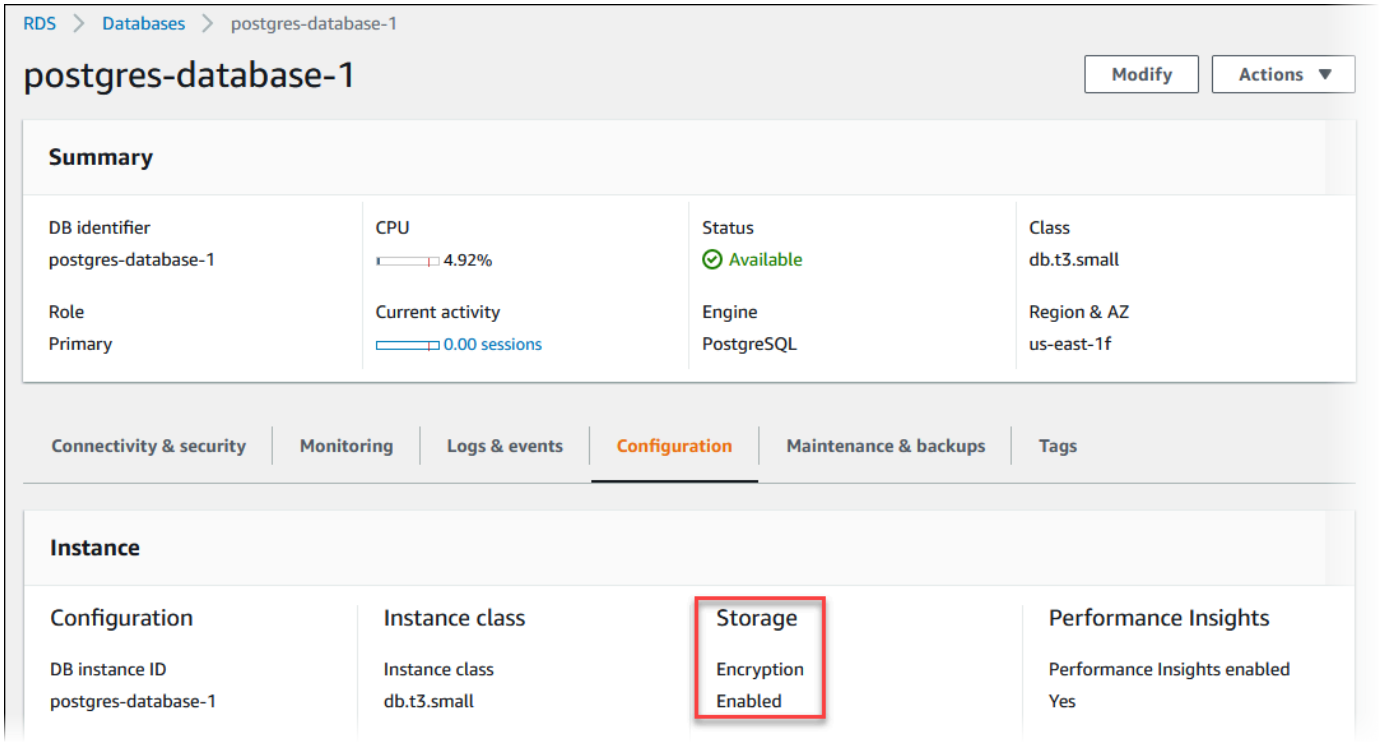
Vous pouvez utiliser l'API AWS Management Console AWS CLI, ou RDS pour déterminer si le chiffrement au repos est activé pour une instance de base de données.

console

Pour déterminer si le chiffrement au repos est activé pour une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Sélectionnez le nom de l'instance de base de données que vous souhaitez vérifier pour afficher ses détails.
4. Cliquez sur l'onglet Configuration et cochez la case Encryption (Chiffrement) sous Storage (Stockage).

   Il indique Enabled (Activé) ou Not enabled (Non activé).



AWS CLI

Pour déterminer si le chiffrement au repos est activé pour une instance de base de données à l'aide de AWS CLI, appelez la describe-db-instancescommande avec l'option suivante :

- `--db-instance-identifier` – Nom de l'instance de base de données.

L'exemple suivant utilise une requête pour renvoyer `TRUE` ou `FALSE` concernant le chiffrement au repos pour l'instance de base de données `mydb`.

Example

```
aws rds describe-db-instances --db-instance-identifier mydb --query "*[].
{StorageEncrypted:StorageEncrypted}" --output text
```

API RDS

Pour déterminer si le chiffrement au repos est activé pour une instance de base de données à l'aide de l'API Amazon RDS, appelez l'DBInstancesopération Describe avec le paramètre suivant :

- `DBInstanceIdentifier` – Nom de l'instance de base de données.

Disponibilité du chiffrement Amazon RDS

Le chiffrement Amazon RDS est actuellement disponible pour tous les moteurs de base de données et types de stockage.

Le chiffrement Amazon RDS est disponible pour la plupart des classes d'instance de base de données. Le tableau suivant répertorie les classes d'instance de base de données qui ne prennent pas en charge le chiffrement Amazon RDS :

Type d’instance	Classe d'instance
Polyvalent (M1)	db.m1.small db.m1.medium db.m1.large db.m1.xlarge
Mémoire optimisée (M2)	db.m2.xlarge db.m2.2xlarge db.m2.4xlarge
Capacité extensible (T2)	db.t2.micro

Chiffrement en transit

Chiffrement au niveau de la couche physique

Toutes les données circulant Régions AWS sur le réseau AWS mondial sont automatiquement cryptées au niveau de la couche physique avant de quitter les installations AWS sécurisées. Tout le trafic entre les deux AZs est crypté. Des couches de chiffrement supplémentaires, y compris celles répertoriées dans cette section, peuvent fournir des protections supplémentaires.

Chiffrement fourni par le peering Amazon VPC et le peering interrégional Transit Gateway

Tout le trafic entre régions qui utilise un appairage VPC Amazon et Transit Gateway est automatiquement chiffré en bloc quand il quitte une région. Une couche de cryptage

supplémentaire est automatiquement fournie au niveau de la couche physique pour tout le trafic avant qu'il ne quitte les installations AWS sécurisées.

Chiffrement entre les instances

AWS fournit une connectivité sécurisée et privée entre les instances de base de données de tous types. En outre, certains types d'instances utilisent les capacités de déchargement du matériel du système Nitro sous-jacent pour chiffrer automatiquement le trafic en transit entre instances. Ce chiffrement utilise des algorithmes de chiffrement authentifié avec données associées (AEAD), avec un chiffrement 256 bits. Il n'y a aucun impact sur les performances du réseau. Pour prendre en charge ce chiffrement supplémentaire du trafic en transit entre les instances, les exigences suivantes doivent être satisfaites :

- Les instances utilisent les types d'instance suivants :
  - Usage général : M6i, M6id, M6in, M6idn, M7g
  - Mémoire optimisée : R6i, R6id, R6in, R6idn, R7g, X2iEDN, X2ieZN
- Les instances sont identiques Région AWS.
- Les instances se trouvent dans le même VPC ou peered VPCs, et le trafic ne passe pas par un périphérique ou un service réseau virtuel, tel qu'un équilibreur de charge ou une passerelle de transit.

Limitations des instances de base de données chiffrées Amazon RDS

Les limitations suivantes existent pour les instances de base de données chiffrées Amazon RDS :

- Vous ne pouvez chiffrer une instance de base de données Amazon RDS que lorsque vous la créez, et non après sa création.

  Cependant, parce que vous pouvez chiffrer une copie d'un instantané non chiffré, vous pouvez ajouter le chiffrement efficacement à une instance de base de données non chiffrée. Autrement dit, vous pouvez créer un instantané de votre instance de base de données et ensuite créer une copie chiffrée de l'instantané. Vous pouvez ensuite restaurer une instance de base de données à partir de l'instantané chiffré et vous aurez une copie chiffrée de votre instance de base de données d'origine. Pour de plus amples informations, veuillez consulter Copier un instantané de base de données pour Amazon RDS.
- Vous ne pouvez pas désactiver le chiffrement d'un(e) cluster de bases de données chiffrées.
- Vous ne pouvez pas créer d'instantané chiffré de cluster d'instances.

- Un instantané d'instance de bases de données chiffrées doit être chiffré à l'aide de la même clé KMS que l'instance de bases de données.
- Vous ne pouvez pas avoir un réplica en lecture chiffré d'une instance de base de données non chiffrée ni un réplica en lecture non chiffré d'une instance de base de données chiffrée.
- Les réplicas en lecture chiffrés doivent être chiffrés avec la même clé KMS que l'instance de base de données source lorsque tous deux se trouvent dans la même région AWS .
- Vous ne pouvez pas restaurer un instantané non chiffré ou une sauvegarde non chiffrée vers une instance de base de données chiffrée.
- Pour copier un instantané chiffré d'une AWS région à une autre, vous devez spécifier la clé KMS dans la AWS région de destination. Cela est dû au fait que les clés KMS sont spécifiques à la AWS région dans laquelle elles sont créées.

  L'instantané source reste chiffré pendant tout le processus de copie. Amazon RDS utilise un chiffrement d'enveloppe pour protéger les données pendant le processus de copie. Pour plus d'informations sur le chiffrement d'enveloppe, consultez Chiffrement d'enveloppe dans le Guide du développeur AWS Key Management Service .
- Vous ne pouvez pas déchiffrer un cluster de bases de données chiffrées. Vous pouvez cependant exporter des données à partir d'un cluster de bases de données et importer les données dans un cluster de bases de données non chiffrées.

AWS KMS key gestion

Amazon RDS s'intègre automatiquement à AWS Key Management Service (AWS KMS) pour la gestion des clés. Amazon RDS utilise le chiffrement des enveloppes. Pour plus d'informations sur le chiffrement d'enveloppe, consultez Chiffrement d'enveloppe dans le Guide du développeur AWS Key Management Service .

Vous pouvez utiliser deux types de AWS KMS clés pour chiffrer vos d'instances de base de données.

- Si vous souhaitez un contrôle total sur une clé KMS, vous devez créer une clé gérée par le client. Pour plus d'informations sur les clés gérées par le client, consultez Clés gérées par le client dans le Guide du développeur AWS Key Management Service .
- Clés gérées par AWSsont des clés KMS de votre compte créées, gérées et utilisées en votre nom par un AWS service intégré à AWS KMS. Par défaut, le RDS Clé gérée par AWS (`aws/rds`) est utilisé pour le chiffrement. Vous ne pouvez pas gérer, faire pivoter ou supprimer le RDS. Clé gérée

par AWS Pour plus d'informations à ce sujet Clés gérées par AWS, consultez Clés gérées par AWSle guide du AWS Key Management Service développeur.

Pour gérer les clés KMS utilisées pour les d'instances de base de données chiffrées Amazon RDS , utilisez le AWS Key Management Service (AWS KMS) dans la AWS KMS console AWS CLI, le ou l' AWS KMS API. Pour consulter les journaux d'audit de chaque action effectuée à l'aide d'une clé gérée par le client ou par AWS , utilisez AWS CloudTrail. Pour plus d'informations sur la rotation des clés, consultez Rotation des clés AWS KMS.

Autoriser l'utilisation d'une clé gérée par le client

Lorsque RDS utilise une clé gérée par le client dans le cadre d'opérations cryptographiques, elle agit pour le compte de l'utilisateur qui crée ou modifie la ressource RDS .

Pour créer une ressource RDS à l'aide d'une clé gérée par le client, un utilisateur doit être autorisé à effectuer les opérations suivantes sur la clé gérée par le client :

- `kms:CreateGrant`
- `kms:DescribeKey`

Vous pouvez spécifier les autorisations requises dans une politique de clé ou dans une IAM politique si la politique de clé le permet.

Important

Lorsque vous utilisez des déclarations de refus explicites pour toutes les ressources (*) dans les politiques AWS KMS clés des services gérés tels qu'Amazon RDS, vous devez spécifier une condition pour autoriser le compte propriétaire de la ressource. Les opérations peuvent échouer sans cette condition, même si la règle de refus inclut des exceptions pour votre utilisateur IAM.

Tip

Pour suivre le principe du moindre privilège, n'autorisez pas l'accès complet à `kms:CreateGrant`. Utilisez plutôt la clé de ViaService condition kms : pour permettre à

l'utilisateur de créer des autorisations sur la clé KMS uniquement lorsque l'autorisation est créée en son nom par un AWS service.

Vous pouvez renforcer la politique IAM de différentes manières. Par exemple, si vous souhaitez autoriser l'utilisation de la clé gérée par le client uniquement pour les demandes provenant de RDS , utilisez la clé de kms: ViaService condition avec la `rds.<region>.amazonaws.com` valeur. Vous pouvez également utiliser les clés ou les valeurs du contexte Contexte de chiffrement Amazon RDS comme condition d'utilisation de la clé gérée par le client pour le chiffrement.

Pour plus d'informations, consultez Autoriser des utilisateurs d'autres comptes à utiliser une clé KMS dans le Guide du développeur AWS Key Management Service .

Contexte de chiffrement Amazon RDS

Lorsque RDS utilise votre clé KMS, ou lorsqu'Amazon EBS utilise la clé KMS pour le compte de RDS , le service spécifie un contexte de chiffrement. Le contexte de chiffrement est constitué de données authentifiées supplémentaires (AAD) AWS KMS utilisées pour garantir l'intégrité des données. Autrement dit, quand un contexte de chiffrement est spécifié pour une opération de chiffrement, le service doit spécifier le même contexte de chiffrement pour l'opération de déchiffrement. Dans le cas contraire, le déchiffrement échoue. Le contexte de chiffrement est également écrit dans vos journaux AWS CloudTrail pour vous aider à comprendre pourquoi une clé KMS donnée a été utilisée. Vos CloudTrail journaux peuvent contenir de nombreuses entrées décrivant l'utilisation d'une clé KMS, mais le contexte de chiffrement de chaque entrée de journal peut vous aider à déterminer la raison de cette utilisation particulière.

Au minimum, Amazon RDS utilise toujours l'ID du d'instance de base de données pour le contexte de chiffrement, comme dans l'exemple au format JSON suivant :

```
{ "aws:rds:db-id": "db-CQYSMDPBRZ7BPMH7Y3RTDG5QY" }
```

Ce contexte de chiffrement peut vous aider à identifier le d'instances de base de données pour lequel votre clé KMS a été utilisée.

Lorsque votre clé KMS est utilisée pour un d'instance de base de données et un volume Amazon EBS spécifiques, l'ID de d'instance de base de données et l'ID de volume Amazon EBS sont utilisés pour le contexte de chiffrement, comme dans l'exemple au format JSON suivant :

```
{
```

```
  "aws:rds:db-id": "db-BRG7VYS3SVIFQW7234EJQOM5RQ",
  "aws:ebs:id": "vol-ad8c6542"
}
```

Vous pouvez utiliser le protocole SSL (Secure Socket Layer) ou le protocole TLS (Transport Layer Security) depuis votre application pour chiffrer une connexion à une base de données exécutant Db2, MariaDB, Microsoft SQL Server, MySQL, Oracle ou PostgreSQL.

Votre SSL/TLS connexion peut éventuellement vérifier l'identité du serveur en validant le certificat de serveur installé sur votre base de données. Pour exiger la vérification de l'identité du serveur, suivez ce processus général :

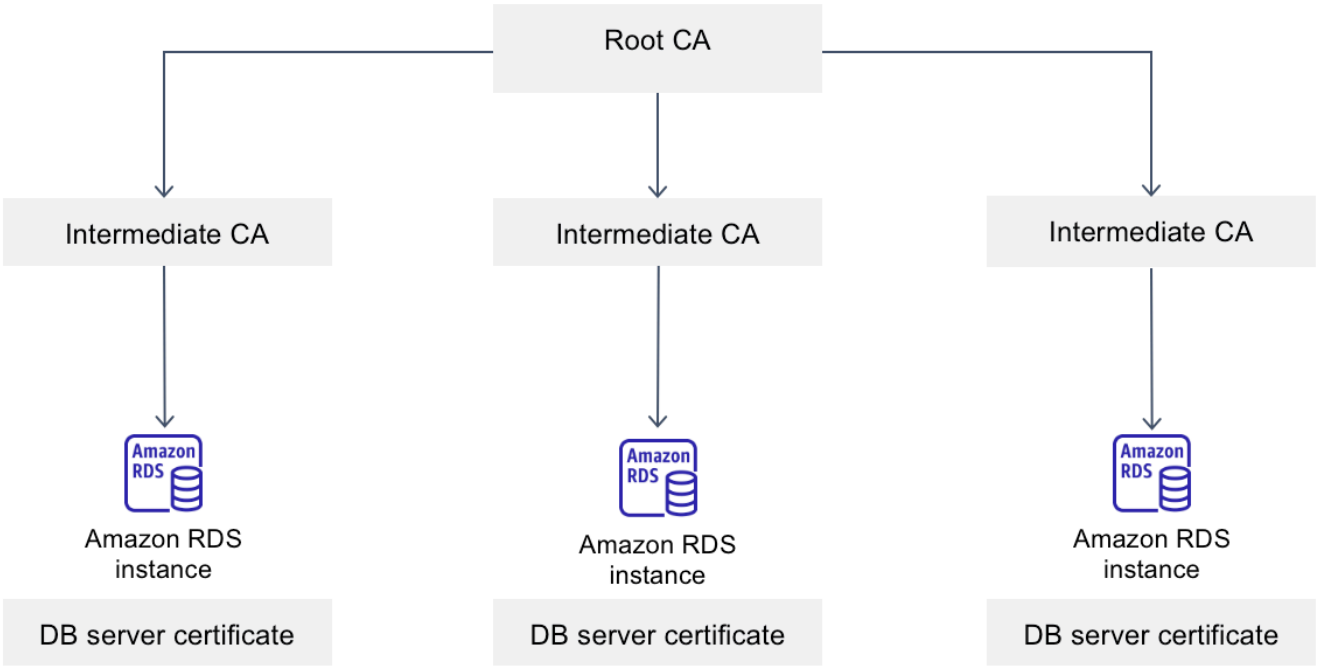
1. Choisissez l'autorité de certification (CA) qui signe le certificat de serveur de base de données pour votre base de données. Pour plus d'informations sur les autorités de certification, consultezAutorités de certification.
2. Téléchargez une offre groupée de certificats à utiliser lorsque vous vous connectez à la base de données. Pour télécharger un ensemble de certificats, consultez Forfaits de certificats par Région AWS.

   > **Note**
   >
   > Tous les certificats sont disponibles uniquement pour le téléchargement sur des connexions SSL/TLS.

3. Connectez-vous à la base de données en utilisant le processus de votre moteur de base de données pour implémenter SSL/TLS les connexions. Chaque moteur de base de données possède son propre processus d'implémentation SSL/TLS. To learn how to implement SSL/TLS pour votre base de données. Suivez le lien correspondant à votre moteur de base de données :
   - Utilisation SSL/TLS avec une instance de base de données Amazon RDS pour DB2
   - SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS
   - Utilisation SSL avec une instance de base de données Microsoft SQL Server
   - SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS
   - Utilisation de SSL avec une instance de base de données RDS for Oracle
   - Utilisation de SSL avec une instance de base de données PostgreSQL

Autorités de certification

L'autorité de certification (CA) est le certificat qui identifie l'autorité de certification racine en haut de la chaîne de certificats. L'autorité de certification signe le certificat de serveur de base de données, qui est installé sur chaque instance de base de données. Le certificat de serveur de base de données identifie l'instance de base de données en tant que serveur approuvé.



Amazon RDS fournit les éléments suivants CAs pour signer le certificat du serveur de base de données d'une base de données.

Autorité de certification (CA)	Description	Nom commun
rds-ca-rsa2048-g1	Utilise une autorité de certification avec l'algorithme de clé privée RSA 2048 et l'algorithme de SHA256 signature dans la plupart des cas. Régions AWS Dans le AWS GovCloud (US) Regions, cette autorité de certification utilise une autorité de certification avec un algorithm e de clé privée et un algorithme de SHA384 signature RSA 2048.	Amazon RDS *region-id entifier* Root CA RSA2 048 G1

Autorité de certification (CA)	Description	Nom commun
	Cette autorité de certification prend en charge la rotation automatique des certificats de serveur.	
rds-ca-rsa4096-g1	Utilise une autorité de certification avec l'algorithme de clé privée et l'algorithme de SHA384 signature RSA 4096. Cette autorité de certification prend en charge la rotation automatique des certificats de serveur.	Amazon RDS *region-id entifier* Root CA RSA4 096 G1
rds-ca-ecc384-g1	Utilise une autorité de certification avec un algorithme de clé privée et un algorithme de SHA384 signature ECC 384. Cette autorité de certification prend en charge la rotation automatique des certificats de serveur.	Amazon RDS *region-id entifier* Root CA G1 ECC384

> **Note**
>
> Si vous utilisez le AWS CLI, vous pouvez vérifier la validité des autorités de certification répertoriées ci-dessus en utilisant describe-certificates.

Ces certificats de CA sont inclus dans la solution groupée de certificats régionaux et mondiaux. Lorsque vous utilisez l'autorité de certification rds-ca-rsa 2048-g1, rds-ca-rsa 4096-g1 ou rds-ca-ecc 384-g1 avec une base de données, RDS gère le certificat du serveur de base de données sur la base de données. RDS effectue automatiquement la rotation du certificat de serveur de base de données avant son expiration.

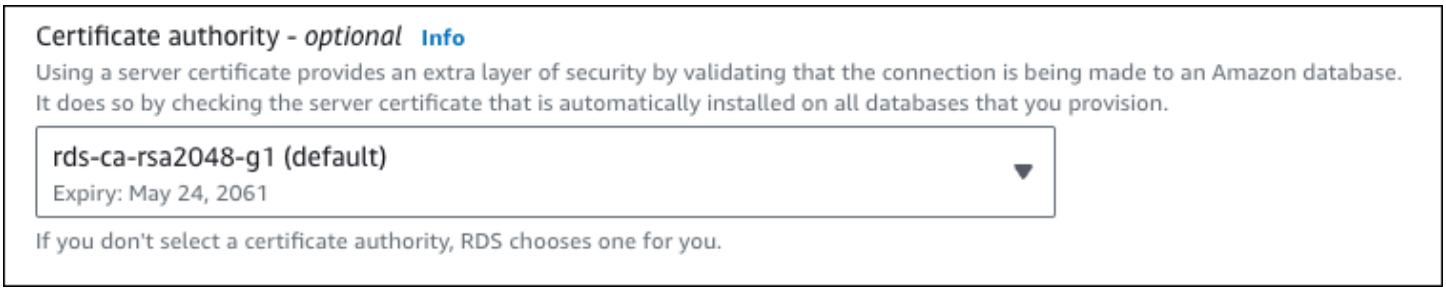
Configuration de l'autorité de certification pour votre base de données

Vous pouvez définir l'autorité de certification pour une base de données lorsque vous effectuez les tâches suivantes :

- Création d'une instance de base de données ou d'un cluster de base de données multi-AZ : vous pouvez définir l'autorité de certification lorsque vous créez une instance ou un cluster de base de données. Pour obtenir des instructions, consultez the section called “Création d'une instance de base de données” outhe section called “Création d'un cluster de base de données multi-AZ”.
- Modifier une instance de base de données ou un cluster de base de données multi-AZ : vous pouvez définir l'autorité de certification pour une instance ou un cluster de base de données en le modifiant. Pour obtenir des instructions, consultez the section called “Modification d'une instance de base de données” outhe section called “Modification d'un cluster de base de données multi-AZ”.

> **Note**
>
> L'autorité de certification par défaut est définie sur rds-ca-rsa 2048-g1. Vous pouvez remplacer l'autorité de certification par défaut pour votre compte Compte AWS en utilisant la commande modify-certificates.

La disponibilité CAs dépend du moteur de base de données et de la version du moteur de base de données. Lorsque vous utilisez le AWS Management Console, vous pouvez choisir l'autorité de certification à l'aide du paramètre Autorité de certification, comme indiqué dans l'image suivante.



La console affiche uniquement ceux CAs qui sont disponibles pour le moteur de base de données et la version du moteur de base de données. Si vous utilisez le AWS CLI, vous pouvez définir l'autorité de certification pour une instance de base de données à l'aide de la modify-db-instancecommande create-db-instanceor. Vous pouvez définir l'autorité de certification pour un cluster de base de données multi-AZ à l'aide de la modify-db-clustercommande create-db-clusteror.

Si vous utilisez le AWS CLI, vous pouvez voir ce qui est disponible CAs pour votre compte en utilisant la commande describe-certificates. Cette commande indique également la date d'expiration de chaque autorité de certification dans `ValidTill`, dans la sortie. Vous pouvez trouver ceux CAs qui sont disponibles pour un moteur de base de données et une version de moteur de base de données spécifiques à l'aide de la describe-db-engine-versionscommande.

L'exemple suivant montre la version CAs disponible pour la version par défaut du moteur de base de données RDS pour PostgreSQL.

```
aws rds describe-db-engine-versions --default-only --engine postgres
```

Votre sortie est similaire à ce qui suit. Les options disponibles CAs sont répertoriées dans`SupportedCACertificateIdentifiers`. La sortie indique également si la version du moteur de base de données prend en charge la rotation du certificat sans redémarrage dans `SupportsCertificateRotationWithoutRestart`.

```
{
    "DBEngineVersions": [
        {
            "Engine": "postgres",
            "MajorEngineVersion": "13",
            "EngineVersion": "13.4",
            "DBParameterGroupFamily": "postgres13",
            "DBEngineDescription": "PostgreSQL",
            "DBEngineVersionDescription": "PostgreSQL 13.4-R1",
            "ValidUpgradeTarget": [],
            "SupportsLogExportsToCloudwatchLogs": false,
            "SupportsReadReplica": true,
            "SupportedFeatureNames": [
                "Lambda"
            ],
            "Status": "available",
            "SupportsParallelQuery": false,
            "SupportsGlobalDatabases": false,
            "SupportsBabelfish": false,
            "SupportsCertificateRotationWithoutRestart": true,
            "SupportedCACertificateIdentifiers": [
                "rds-ca-rsa2048-g1",
                "rds-ca-ecc384-g1",
                "rds-ca-rsa4096-g1"
            ]
        }
    ]
}
```

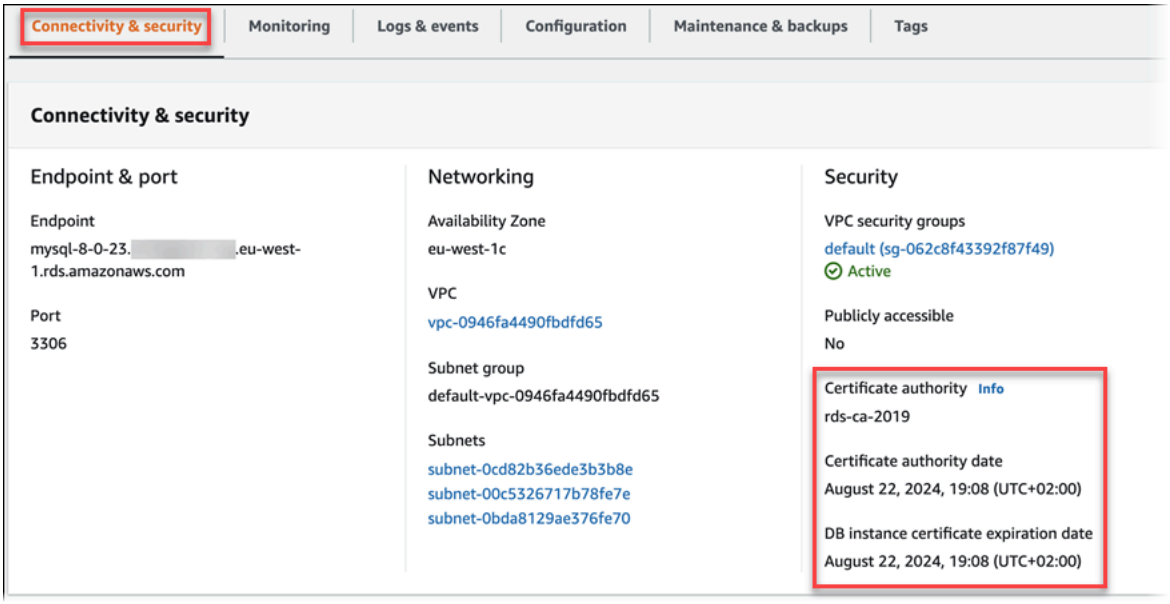
Validité des certificats de serveur de base de données

La validité du certificat de serveur de base de données dépend du moteur de base de données et de la version du moteur de base de données. Si la version du moteur de base de données prend en charge la rotation du certificat sans redémarrage, la validité du certificat de serveur de base de données est de 1 an. Dans le cas contraire, la validité est de 3 ans.

Pour plus d'informations sur la rotation des certificats du serveur de base de données, consultezRotation automatique du certificat de serveur.

Afficher l'autorité de certification de votre instance de base de données

Vous pouvez consulter les détails relatifs à l'autorité de certification d'une base de données en consultant l'onglet Connectivité et sécurité de la console, comme dans l'image suivante.



Si vous utilisez le AWS CLI, vous pouvez afficher les détails de l'autorité de certification pour une instance de base de données à l'aide de la describe-db-instancescommande. Vous pouvez afficher les détails de l'autorité de certification pour un cluster de base de données multi-AZ à l'aide de la describe-db-clusterscommande.

Lorsque vous vous connectez à votre base de données via SSL ou TLS, l'instance de base de données nécessite un certificat de confiance d'Amazon RDS. Sélectionnez le lien approprié dans le tableau suivant pour télécharger le bundle correspondant à l' Région AWS endroit où vous hébergez votre base de données.

Forfaits de certificats par Région AWS

Les ensembles de certificats pour toutes les régions Régions AWS et GovCloud (États-Unis) contiennent les certificats CA racine suivants :

- `rds-ca-rsa2048-g1`
- `rds-ca-rsa4096-g1`
- `rds-ca-ecc384-g1`

Les `rds-ca-ecc384-g1` certificats `rds-ca-rsa4096-g1` et ne sont pas disponibles dans les régions suivantes :

- Asie-Pacifique (Mumbai)
- Asie-Pacifique (Melbourne)
- Canada-Ouest (Calgary)
- Europe (Zurich)
- Europe (Espagne)
- Israël (Tel Aviv)

Le magasin de confiance de votre application doit uniquement enregistrer le certificat CA racine.

> **Note**
>
> Le proxy les certificats du AWS Certificate Manager (ACM). Si vous utilisez le proxy RDS, vous n'avez pas besoin de télécharger les certificats Amazon RDS ni de mettre à jour les applications qui utilisent des connexions au proxy RDS. Pour plus d'informations, consultez UtiliserTLS/SSLavec un RDS proxy.

Pour télécharger un bundle de certificats pour un Région AWS, sélectionnez le lien correspondant à celui Région AWS qui héberge votre base de données dans le tableau suivant.

AWS Région	Solution groupée de certificats (PEM)	Ensemble de certificats (PKCS7)
N'importe quelle publicité Région AWS	global-bundle.pem	global-bundle.p7b
USA Est (Virginie du Nord)	us-east-1-bundle.pem	us-east-1-bundle.p7b
US East (Ohio)	us-east-2-bundle.pem	us-east-2-bundle.p7b
US West (N. California)	us-west-1-bundle.pem	us-west-1-bundle.p7b
US West (Oregon)	us-west-2-bundle.pem	us-west-2-bundle.p7b
Africa (Cape Town)	af-south-1-bundle.pem	af-south-1-bundle.p7b
Asia Pacific (Hong Kong)	ap-east-1-bundle.pem	ap-east-1-bundle.p7b
Asie-Pacifique (Hyderabad)	ap-south-2-bundle.pem	ap-south-2-bundle.p7b
Asie-Pacifique (Jakarta)	ap-southeast-3-bundle.pem	ap-southeast-3-bundle.p7b
Asie-Pacifique (Malaisie)	ap-southeast-5-bundle.pem	ap-southeast-5-bundle.p7b
Asie-Pacifique (Melbourne)	ap-southeast-4-bundle.pem	ap-southeast-4-bundle.p7b
Asia Pacific (Mumbai)	ap-south-1-bundle.pem	ap-south-1-bundle.p7b
Asia Pacific (Osaka)	ap-northeast-3-bundle.pem	ap-northeast-3-bundle.p7b
Asie-Pacifique (Thaïlande)	ap-southeast-7-bundle.pem	ap-southeast-7-bundle.p7b
Asie-Pacifique (Tokyo)	ap-northeast-1-bundle.pem	ap-northeast-1-bundle.p7b
Asia Pacific (Seoul)	ap-northeast-2-bundle.pem	ap-northeast-2-bundle.p7b
Asia Pacific (Singapore)	ap-southeast-1-bundle.pem	ap-southeast-1-bundle.p7b
Asia Pacific (Sydney)	ap-southeast-2-bundle.pem	ap-southeast-2-bundle.p7b
Canada (Central)	ca-central-1-bundle.pem	ca-central-1-bundle.p7b

AWS Région	Solution groupée de certificats (PEM)	Ensemble de certificats (PKCS7)
Canada-Ouest (Calgary)	ca-west-1-bundle.pem	ca-west-1-bundle.p7b
Europe (Frankfurt)	eu-central-1-bundle.pem	eu-central-1-bundle.p7b
Europe (Ireland)	eu-west-1-bundle.pem	eu-west-1-bundle.p7b
Europe (London)	eu-west-2-bundle.pem	eu-west-2-bundle.p7b
Europe (Milan)	eu-south-1-bundle.pem	eu-south-1-bundle.p7b
Europe (Paris)	eu-west-3-bundle.pem	eu-west-3-bundle.p7b
Europe (Espagne)	eu-south-2-bundle.pem	eu-south-2-bundle.p7b
Europe (Stockholm)	eu-north-1-bundle.pem	eu-north-1-bundle.p7b
Europe (Zurich)	eu-central-2-bundle.pem	eu-central-2-bundle.p7b
Israël (Tel Aviv)	il-central-1-bundle.pem	il-central-1-bundle.p7b
Mexique (centre)	mx-central-1-bundle.pem	mx-central-1-bundle.p7b
Middle East (Bahrain)	me-south-1-bundle.pem	me-south-1-bundle.p7b
Moyen-Orient (EAU)	me-central-1-bundle.pem	me-central-1-bundle.p7b
Amérique du Sud (São Paulo)	sa-east-1-bundle.pem	sa-east-1-bundle.p7b
N'importe quel AWS GovCloud (US) Region s	global-bundle.pem	global-bundle.p7b
AWS GovCloud (USA Est)	us-gov-east-1-bundle.pem	us-gov-east-1-bundle.p7b
AWS GovCloud (US-Ouest)	us-gov-west-1-bundle.pem	us-gov-west-1-bundle.p7b

Afficher le contenu de votre certificat CA

Pour vérifier le contenu de votre offre groupée de certificats d'autorité de certification, utilisez la commande suivante :

```
keytool -printcert -v -file global-bundle.pem
```

Rotation de votre certificat SSL/TLS

Les certificats RDS-CA-2019 de l'autorité de certification Amazon RDS ont expiré en août 2024. Si vous utilisez ou prévoyez d'utiliser le protocole SSL (Secure Sockets Layer) ou le protocole Transport Layer Security (TLS) avec vérification des certificats pour vous connecter à vos instances de base de données RDS ou à vos clusters de bases de données multi-AZ, pensez à utiliser l'un des nouveaux certificats CA rds-ca-rsa 2048-g1, 4096-g1 ou 384-g1. rds-ca-rsa rds-ca-ecc Si vous ne l'utilisez pas actuellement SSL/TLS with certificate verification, you might still have an expired CA certificate and must update them to a new CA certificate if you plan to use SSL/TLS avec la vérification des certificats pour vous connecter à vos bases de données RDS.

Amazon RDS fournit de nouveaux certificats CA dans le cadre des meilleures pratiques AWS de sécurité. Pour plus d'informations sur les nouveaux certificats et les AWS régions prises en charge, consultez.

Pour mettre à jour le certificat CA de votre base de données, appliquez les méthodes suivantes :

- Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données
- Mise à jour de votre certificat CA en appliquant la maintenance

Avant de mettre à jour vos instances de base de données ou vos clusters de base de données multi-AZ pour utiliser le nouveau certificat CA, assurez-vous de mettre à jour vos clients ou applications qui se connectent à vos bases de données RDS.

Considérations relatives à la rotation des certificats

Tenez compte des situations suivantes avant de procéder à la rotation de votre certificat :

- Le proxy les certificats du AWS Certificate Manager (ACM). Si vous utilisez un proxy RDS, lorsque vous faites pivoter votre certificat SSL/TLS, vous n'avez pas besoin de mettre à jour les applications qui utilisent des connexions au proxy RDS. Pour plus d'informations, consultez UtiliserTLS/SSLavec un RDS proxy.

- Si vous utilisez une application Go version 1.15 avec une instance de base de données ou un cluster de base de données multi-AZ créé ou mis à jour vers le certificat rds-ca-2019 avant le 28 juillet 2020, vous devez à nouveau mettre à jour le certificat.

  Utilisez la `modify-db-instance` commande pour une instance de base de données ou la `modify-db-cluster` commande pour un cluster de base de données multi-AZ, en utilisant le nouvel identifiant de certificat CA. Vous pouvez trouver ceux CAs qui sont disponibles pour un moteur de base de données et une version de moteur de base de données spécifiques à l'aide de la `describe-db-engine-versions` commande.

  Si vous avez créé votre base de données ou mis à jour son certificat après le 28 juillet 2020, aucune action n'est requise. Pour plus d'informations, consultez Go GitHub issue #39568.

Mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données

L'exemple suivant met à jour votre certificat CA de rds-ca-2019 à 2048-g1. rds-ca-rsa Vous pouvez choisir un autre certificat. Pour plus d'informations, consultezAutorités de certification.

Mettez à jour le magasin de confiance de votre application afin de réduire les temps d'arrêt associés à la mise à jour de votre certificat CA. Pour plus d'informations sur les redémarrages associés à la rotation des certificats CA, consultezRotation automatique du certificat de serveur.

Pour mettre à jour votre certificat CA en modifiant votre instance ou cluster de base de données

1. Téléchargez le nouveau certificat SSL/TLS comme décrit dans.
2. Mettez à jour vos applications de sorte à utiliser le nouveau certificat SSL/TLS.

   Les méthodes de mise à jour des applications pour les nouveaux SSL/TLS certificates depend on your specific applications. Work with your application developers to update the SSL/TLS certificats pour vos applications.

   Pour plus d'informations sur la vérification des connexions SSL/TLS et la mise à jour des applications pour chaque moteur de bases de données, veuillez consulter les rubriques suivantes :

   - Mise à jour des applications pour se connecter aux instances MariaDB à l'aide de nouveaux certificats/SSLTLS
   - Mise à jour des applications pour se connecter aux instances de base de données Microsoft SQL Server à l'aide de nouveaux TLS certificatsSSL/

- Mise à jour des applications pour se connecter aux instances My SQL DB à l'aide de nouveaux TLS certificatsSSL/
- Mise à jour des applications pour se connecter aux instances de bases de données Oracle à l'aide des nouveaux certificats SSL/TLS
- Mise à jour des applications pour se connecter aux SQL instances de base de données Postgre à l'aide de nouveaux certificatsSSL/TLS

Pour un exemple de script mettant à jour un trust store pour un système d'exploitation Linux, consultezExemple de script pour importer les certificats dans votre magasin d'approbations.

> **Note**
>
> L'ensemble de certificats contient des certificats pour le nouveau et l'ancien CA, ce qui signifie que vous pouvez mettre à niveau votre application en toute sécurité et conserver la connectivité pendant la période de transition. Si vous utilisez le AWS Database Migration Service pour migrer une base de données vers une instance de base de données ou un cluster , nous vous recommandons d'utiliser le bundle de certificats pour garantir la connectivité pendant la migration.

3. Modifiez l'instance de base de données ou le cluster de base de données multi-AZ pour faire passer l'autorité de certification de rds-ca-2019 à 2048-g1. rds-ca-rsa Pour vérifier si votre base de données doit être redémarrée pour mettre à jour les certificats CA, utilisez la describe-db-engine-versionscommande et cochez l'`SupportsCertificateRotationWithoutRestart`indicateur.

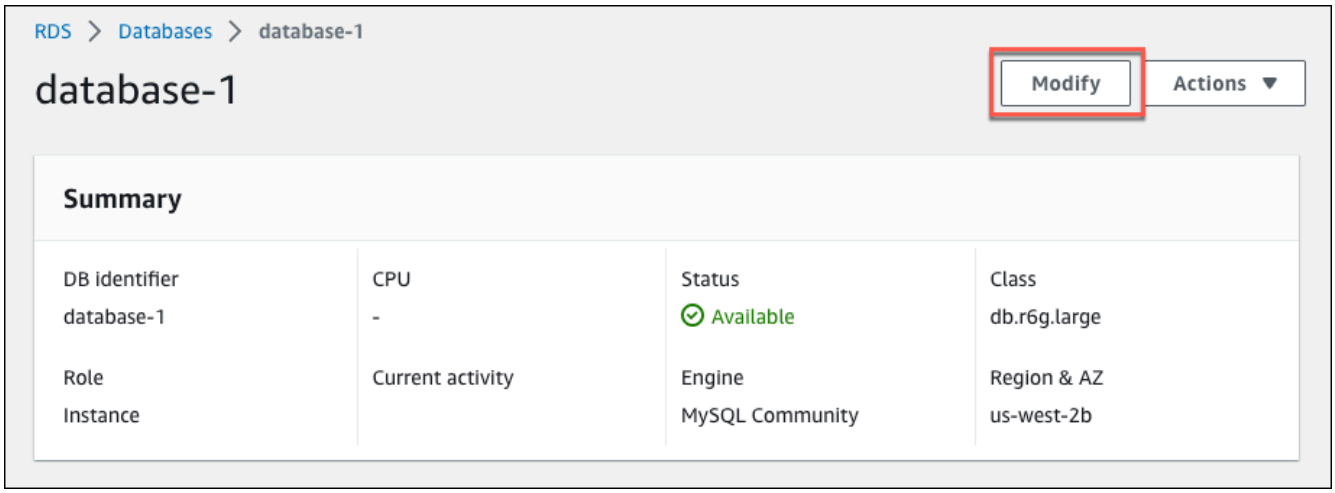
> **Important**
>
> Si vous rencontrez des problèmes de connectivité après l'expiration du certificat, utilisez l'option Appliquer immédiatement en la spécifiant dans la console ou en spécifiant l'option `--apply-immediately` à l'aide d' AWS CLI. Par défaut, il est prévu que cette opération soit exécutée pendant votre prochaine fenêtre de maintenance.
> Pour les instances de base de données RDS pour Oracle, nous vous recommandons de redémarrer votre base de données Oracle afin d'éviter toute erreur de connexion.
> Pour les instances RDS for SQL Server Multi-AZ dont l'option de mise en miroir est activée AlwaysOn ou dont l'option de mise en miroir est activée, un basculement est attendu lorsque l'instance est redémarrée après la rotation des certificats.

Pour définir une dérogation pour votre instance de CA différente de l'autorité de certification RDS par défaut, utilisez la commande de la CLI modify-certificates.

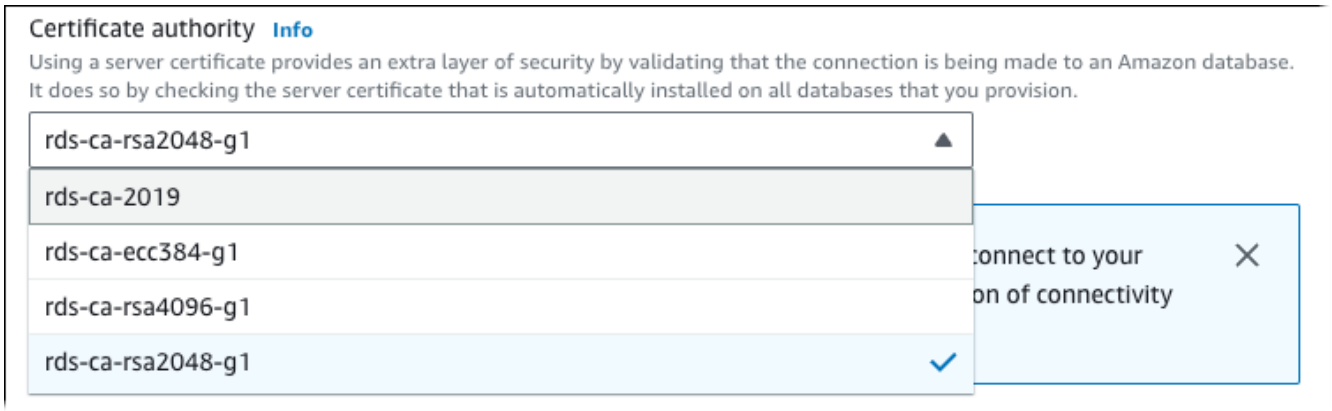
Vous pouvez utiliser le AWS Management Console ou AWS CLI pour modifier le certificat CA de rds-ca-2019 à rds-ca-rsa2048-g1 pour une instance de base de données ou un cluster de base de données multi-AZ.

Console

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, choisissez Databases, puis choisissez l'instance de base de données ou le cluster de base de données multi-AZ que vous souhaitez modifier.
3. Sélectionnez Modifier.



4. Dans la section Connectivité, choisissez rds-ca-rsa2048-g1.



5. Choisissez Continuer et vérifiez le récapitulatif des modifications.

6. Pour appliquer les modifications immédiatement, choisissez Appliquer immédiatement.
7. Sur la page de confirmation, examinez vos modifications. S'ils sont corrects, choisissez Modifier l'instance de base de données ou Modifier le cluster pour enregistrer vos modifications.

> **Important**
>
> Lorsque vous planifiez cette opération, assurez-vous d'avoir mis à jour votre magasin d'approbation côté client au préalable.

Ou choisissez Retour pour revoir vos modifications, ou choisissez Annuler pour les annuler.

AWS CLI

Pour utiliser le pour AWS CLI faire passer l'autorité de certification de rds-ca-2019 à rds-ca-rsa2048-g1 pour une instance de base de données ou un cluster de base de données multi-AZ, appelez la commande or. modify-db-instancemodify-db-cluster Spécifiez l'identifiant de l'instance ou du cluster de base de données et l'`--ca-certificate-identifier`option.

Utilisez le `--apply-immediately` paramètre pour appliquer la mise à jour immédiatement. Par défaut, il est prévu que cette opération soit exécutée pendant votre prochaine fenêtre de maintenance.

> **Important**
>
> Lorsque vous planifiez cette opération, assurez-vous d'avoir mis à jour votre magasin d'approbation côté client au préalable.

Example

Instance DB

L'exemple suivant modifie `mydbinstance` en définissant le certificat CA sur`rds-ca-rsa2048-g1`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --ca-certificate-identifier rds-ca-rsa2048-g1
```

Dans Windows :

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --ca-certificate-identifier rds-ca-rsa2048-g1
```

> **Note**
>
> Si votre instance nécessite un redémarrage, vous pouvez utiliser la commande modify-db-instanceCLI et spécifier l'`--no-certificate-rotation-restart`option.

Example

Cluster de bases de données multi-AZ

L'exemple suivant modifie `mydbcluster` en définissant le certificat CA sur`rds-ca-rsa2048-g1`.

Pour LinuxmacOS, ou Unix :

```
aws rds modify-db-cluster \
    --db-cluster-identifier mydbcluster \
    --ca-certificate-identifier rds-ca-rsa2048-g1
```

Dans Windows :

```
aws rds modify-db-cluster ^
    --db-cluster-identifier mydbcluster ^
    --ca-certificate-identifier rds-ca-rsa2048-g1
```

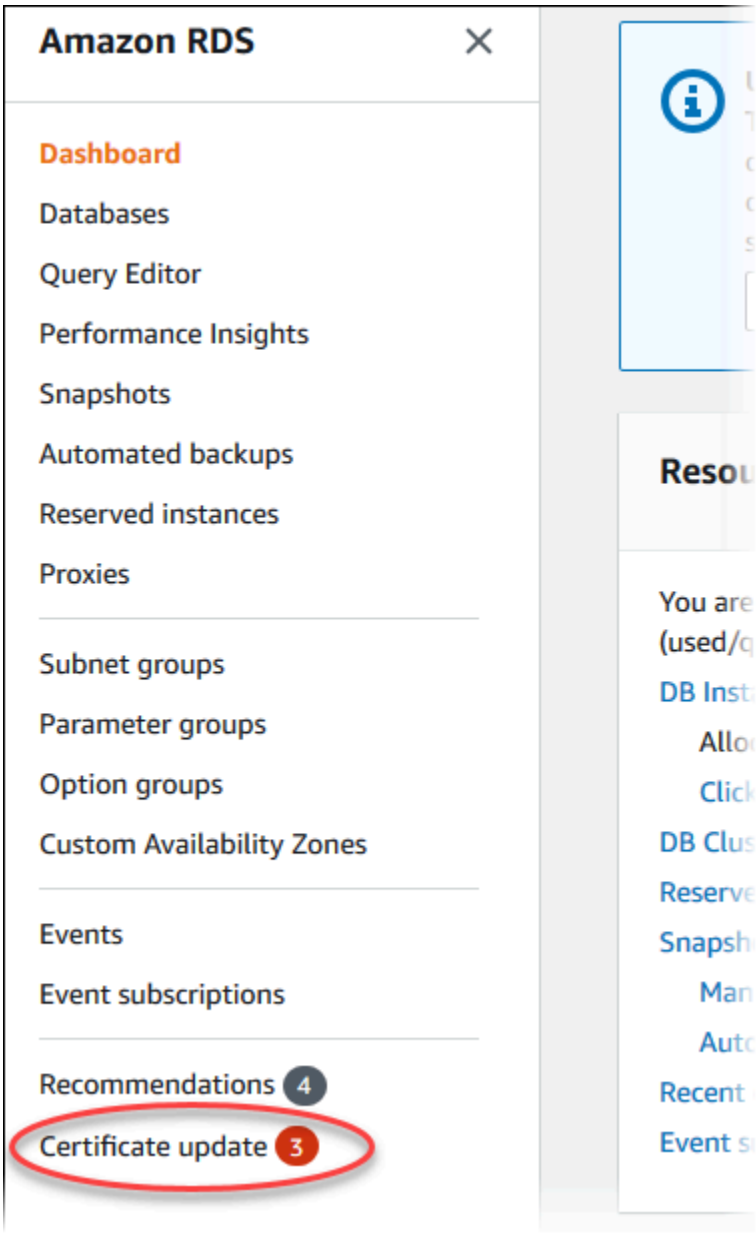
Mise à jour de votre certificat CA en appliquant la maintenance

Procédez comme suit pour mettre à jour votre certificat CA en appliquant la maintenance.

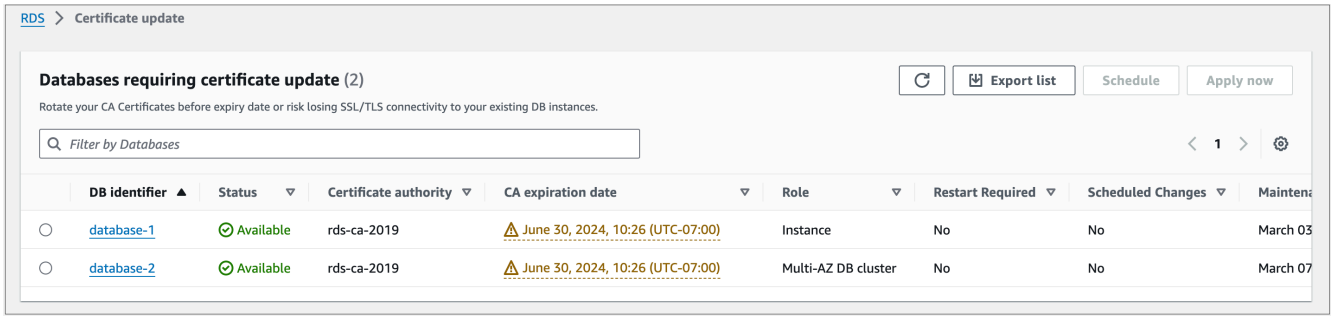
Console

Pour mettre à jour votre certificat CA en appliquant la maintenance

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
2. Dans le volet de navigation, sélectionnez Mise à jour du certificat.



La page Bases de données nécessitant une mise à jour de certificat apparaît.



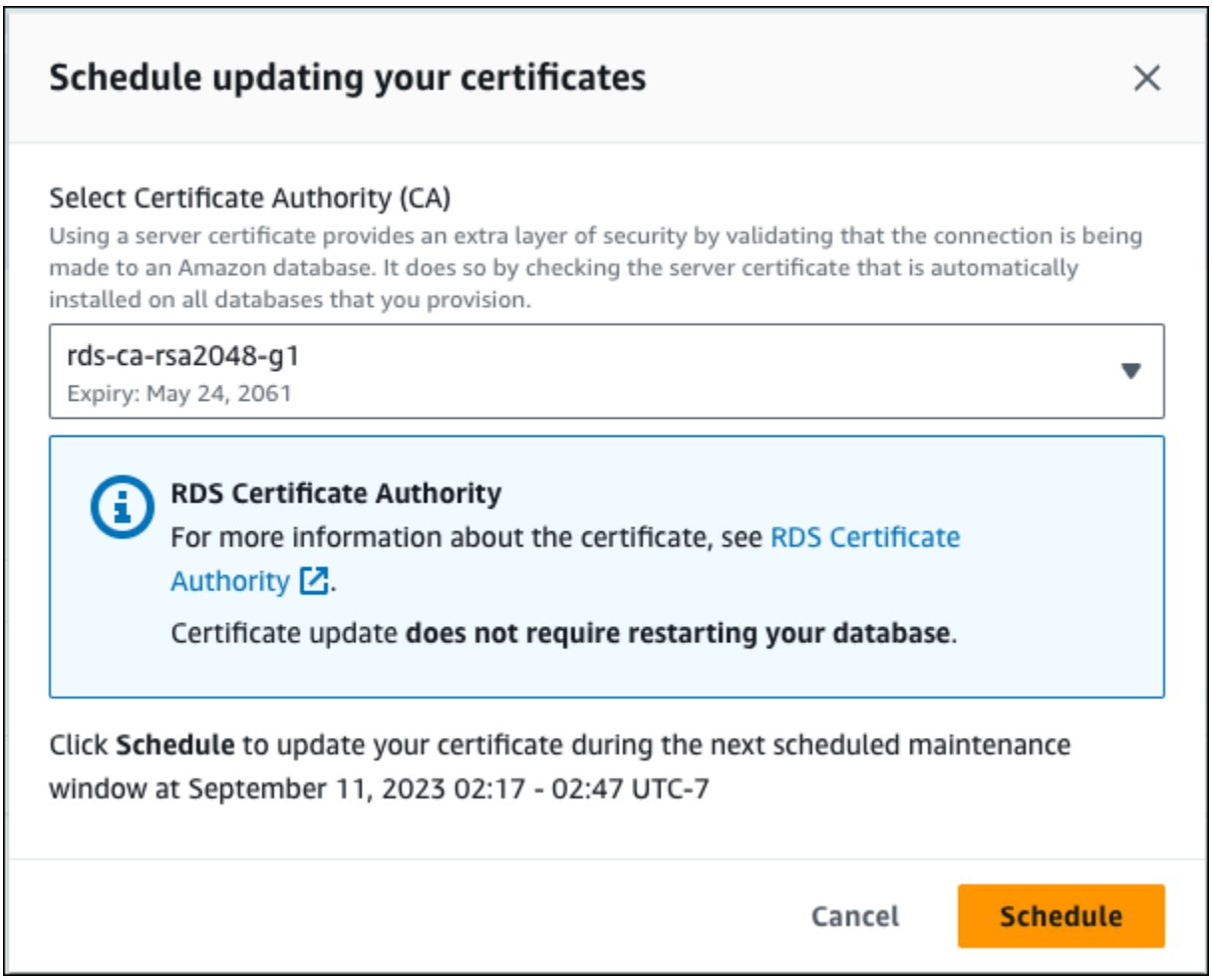
> **Note**
>
> Cette page affiche uniquement les instances de base de données et les clusters actuels Région AWS. Si vous avez des bases de données dans plusieurs d'entre elles Région AWS, consultez cette page Région AWS pour voir toutes les instances de base de données dotées d'anciens certificats SSL/TLS.

3. Choisissez l'instance de base de données ou le cluster de base de données multi-AZ que vous souhaitez mettre à jour.

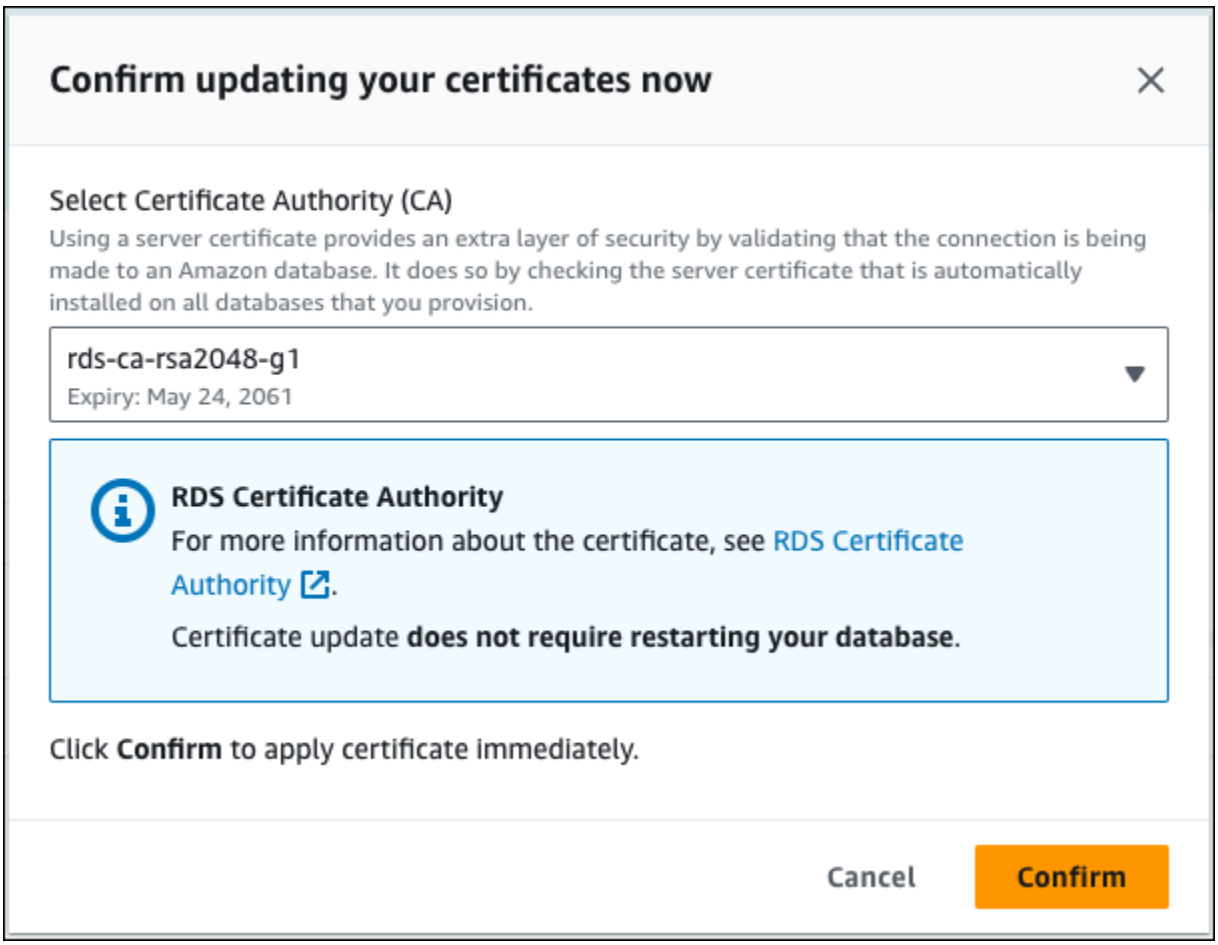
   Vous pouvez planifier la rotation du certificat pour votre prochaine fenêtre de maintenance en choisissant Planification. Appliquez la rotation immédiatement en choisissant Appliquer maintenant.

> **Important**
>
> Si vous rencontrez des problèmes de connectivité après l'expiration du certificat, utilisez l'option Appliquer maintenant.

4. a. Si vous choisissez Planification, vous êtes invité à confirmer la rotation du certificat CA. Cette invite indique également la fenêtre planifiée pour votre mise à jour.



b. Si vous choisissez Appliquer maintenant, vous êtes invité à confirmer la rotation du certificat CA.



> **Important**
>
> Avant de planifier la rotation du certificat CA sur votre base de données, mettez à jour toutes les applications clientes qui utilisent SSL/TLS et le certificat de serveur pour se connecter. Ces mises à jour sont spécifiques à votre moteur de base de données. Après avoir mis à jour ces applications clientes, vous pouvez confirmer la rotation du certificat CA.

Pour continuer, cochez la case, puis cliquez sur Confirmation.

5. Répétez les étapes 3 et 4 pour chaque instance de base de données et cluster que vous souhaitez mettre à jour.

Rotation automatique du certificat de serveur

Si votre autorité de certification racine prend en charge la rotation automatique des certificats de serveur, RDS gère automatiquement la rotation du certificat de serveur de base de données. RDS utilise la même autorité de certification racine pour cette rotation automatique. Vous n'avez donc pas besoin de télécharger une nouvelle offre groupée d'autorités de certification. VoirAutorités de certification.

La rotation et la validité de votre certificat de serveur de base de données dépendent de votre moteur de base de données :

- Si votre moteur de base de données prend en charge la rotation sans redémarrage, RDS effectue automatiquement la rotation du certificat de serveur de base de données sans que vous ayez à intervenir. RDS tente d'effectuer la rotation de votre certificat de serveur de base de données pendant la fenêtre de maintenance de votre choix, à la moitié de la durée de vie du certificat de serveur de base de données. Le nouveau certificat de serveur de base de données est valide pendant 12 mois.
- Si votre moteur de base de données ne prend pas en charge la rotation sans redémarrage, RDS vous informe d'un événement de maintenance au moins 6 mois avant l'expiration du certificat de serveur de base de données. Le nouveau certificat de serveur de base de données est valide pendant 36 mois.

Utilisez la describe-db-engine-versionscommande et inspectez l'`SupportsCertificateRotationWithoutRestart`indicateur pour déterminer si la version du moteur de base de données prend en charge la rotation du certificat sans redémarrage. Pour plus d'informations, consultez Configuration de l'autorité de certification pour votre base de données.

Exemple de script pour importer les certificats dans votre magasin d'approbations

Voici des exemples de scripts shell qui importent le lot de certificats dans un magasin d'approbations.

Chaque exemple de script shell utilise keytool, qui fait partie du kit de développement Java (JDK). Pour plus d'informations sur l'installation du JDK, veuillez consulter le Guide d'installation du JDK.

Linux

Voici un exemple de scripting shell qui importe le lot de certificats vers un magasin d'approbations sur un système d'exploitation Linux.

```
mydir=tmp/certs
if [ ! -e "${mydir}" ]
then
mkdir -p "${mydir}"
fi truststore=${mydir}/rds-truststore.jks storepassword=changeit

curl -sS "https://truststore.pki.rds.amazonaws.com/global/global-bundle.pem">
 ${mydir}/global-bundle.pem
awk 'split_after == 1 {n++;split_after=0} /-----END CERTIFICATE-----/
 {split_after=1}{print > "rds-ca-" n+1 ".pem"}' < ${mydir}/global-bundle.pem

for CERT in rds-ca-*; do alias=$(openssl x509 -noout -text -in $CERT | perl -ne
 'next unless /Subject:/; s/.*(CN=|CN = )//; print')
  echo "Importing $alias"
  keytool -import -file ${CERT} -alias "${alias}" -storepass ${storepassword} -
keystore ${truststore} -noprompt
  rm $CERT
done

rm ${mydir}/global-bundle.pem

echo "Trust store content is: "

keytool -list -v -keystore "$truststore" -storepass ${storepassword} | grep Alias |
 cut -d " " -f3- | while read alias
do expiry=`keytool -list -v -keystore "$truststore" -storepass ${storepassword} -
alias "${alias}" | grep Valid | perl -ne 'if(/until: (.*?)\n/) { print "$1\n"; }'`
   echo " Certificate ${alias} expires in '$expiry'"
done
```

macOS

Voici un exemple de scripting shell qui importe le lot de certificats vers un magasin d'approbations sur macOS.

```
mydir=tmp/certs
if [ ! -e "${mydir}" ]
then
mkdir -p "${mydir}"
fi truststore=${mydir}/rds-truststore.jks storepassword=changeit
```

```
curl -sS "https://truststore.pki.rds.amazonaws.com/global/global-bundle.pem">
 ${mydir}/global-bundle.pem
split -p "-----BEGIN CERTIFICATE-----" ${mydir}/global-bundle.pem rds-ca-

for CERT in rds-ca-*; do alias=$(openssl x509 -noout -text -in $CERT | perl -ne
 'next unless /Subject:/; s/.*(CN=|CN = )//; print')
  echo "Importing $alias"
  keytool -import -file ${CERT} -alias "${alias}" -storepass ${storepassword} -
keystore ${truststore} -noprompt
  rm $CERT
done

rm ${mydir}/global-bundle.pem

echo "Trust store content is: "

keytool -list -v -keystore "$truststore" -storepass ${storepassword} | grep Alias |
 cut -d " " -f3- | while read alias
do expiry=`keytool -list -v -keystore "$truststore" -storepass ${storepassword} -
alias "${alias}" | grep Valid | perl -ne 'if(/until: (.*?)\n/) { print "$1\n"; }'`
   echo " Certificate ${alias} expires in '$expiry'"
done
```

Confidentialité du trafic inter-réseau

Les connexions sont protégées à la fois entre Amazon RDS, et les applications sur site, et entre Amazon RDS, et d'autres AWS ressources au sein de la même région. AWS

Trafic entre les clients de service et sur site et les applications

Vous disposez de deux options de connectivité entre votre réseau privé et AWS :

- Une connexion AWS Site-to-Site VPN. Pour plus d’informations, consultez Qu’est-ce qu’ AWS Site-to-Site VPN ?
- Une AWS Direct Connect connexion. Pour plus d'informations, voir Qu'est-ce que c'est AWS Direct Connect ?

Vous pouvez accéder à Amazon RDS () via le réseau en utilisant des opérations d'API AWS publiées. Les clients doivent prendre en charge les éléments suivants :

- Protocole TLS (Transport Layer Security). Nous exigeons TLS 1.2 et recommandons TLS 1.3.
- Ses suites de chiffrement PFS (Perfect Forward Secrecy) comme DHE (Ephemeral Diffie-Hellman) ou ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un ID de clé d'accès et d'une clé d'accès secrète associée à un principal IAM. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

Gestion des identités et des accès pour Amazon RDS

AWS Identity and Access Management (IAM) est un outil Service AWS qui permet à un administrateur de contrôler en toute sécurité l'accès aux AWS ressources. IAMles administrateurs contrôlent qui peut être authentifié (connecté) et autorisé (autorisé) à utiliser les RDS ressources Amazon. IAMest un Service AWS ventilateur que vous pouvez utiliser sans frais supplémentaires.

Rubriques

- Public ciblé
- Authentification par des identités
- Gestion des accès à l’aide de politiques
- Comment fonctionne RDSAmazon avec IAM
- Exemples de politiques basées sur l'identité pour Amazon RDS
- AWS politiques gérées pour Amazon RDS
- Amazon RDS met à jour les politiques AWS gérées
- Prévention des problèmes d'adjoint confus entre services
- Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL
- Résolution des problèmes liés à Identity and Access Amazon RDS

Public ciblé

La façon dont vous utilisez AWS Identity and Access Management (IAM) varie en fonction du travail que vous effectuez dans Amazon RDS .

Utilisateur du service : si vous utilisez le service Amazon RDS pour effectuer votre travail, votre administrateur vous fournit les informations d'identification et les autorisations dont vous avez besoin. Au fur et à mesure que vous utilisez de plus en plus de fonctionnalités d'Amazon RDS pour effectuer votre travail, il se peut que vous ayez besoin d'autorisations supplémentaires. En comprenant bien la gestion des accès, vous saurez demander les autorisations appropriées à votre administrateur. Si vous ne parvenez pas à accéder à une fonctionnalité d'Amazon RDS , consultezRésolution des problèmes liés à Identity and Access Amazon RDS.

Administrateur du service : si vous êtes responsable des ressources Amazon RDS au sein de votre entreprise, vous disposez probablement d'un accès complet à Amazon RDS . C'est à vous de

déterminer les fonctionnalités et les ressources d'Amazon RDS auxquelles vos employés doivent avoir accès. Vous devez ensuite soumettre les demandes à votre administrateur pour modifier les autorisations des utilisateurs de votre service. Consultez les informations sur cette page pour comprendre les concepts de base d'IAM. Pour en savoir plus sur la manière dont votre entreprise peut utiliser IAM Amazon RDS , consultezComment fonctionne RDSAmazon avec IAM.

Administrateur — Si vous êtes administrateur, vous souhaiterez peut-être en savoir plus sur la manière dont vous pouvez rédiger des politiques pour gérer l'accès à Amazon RDS . Pour consulter des exemples de politiques basées sur l'identité Amazon RDS que vous pouvez utiliserIAM, consultez. Exemples de politiques basées sur l'identité pour Amazon RDS

Authentification par des identités

L'authentification est la façon dont vous vous connectez à AWS l'aide de vos informations d'identification. Vous devez être authentifié (connecté à AWS) en tant que Utilisateur racine d'un compte AWS, en tant qu'IAMutilisateur ou en assumant un IAM rôle.

Vous pouvez vous connecter en AWS tant qu'identité fédérée en utilisant les informations d'identification fournies par le biais d'une source d'identité. AWS IAM Identity Center Les utilisateurs (IAMIdentity Center), l'authentification unique de votre entreprise et vos informations d'identification Google ou Facebook sont des exemples d'identités fédérées. Lorsque vous vous connectez en tant qu'identité fédérée, votre administrateur aura précédemment configuré une fédération d'identités avec des rôles IAM. Lorsque vous accédez à AWS l'aide de la fédération, vous assumez indirectement un rôle.

Selon le type d'utilisateur que vous êtes, vous pouvez vous connecter au portail AWS Management Console ou au portail AWS d'accès. Pour plus d'informations sur la connexion à AWS, consultez la section Comment vous connecter à votre compte Compte AWS dans le guide de Connexion à AWS l'utilisateur.

Si vous y accédez AWS par programmation, AWS fournit un kit de développement logiciel (SDK) et une interface de ligne de commande (CLI) pour signer cryptographiquement vos demandes à l'aide de vos informations d'identification. Si vous n'utilisez pas d' AWS outils, vous devez signer vous-même les demandes. Pour plus d'informations sur l'utilisation de la méthode recommandée pour signer vous-même les demandes, consultez la version 4 de AWS Signature pour les API demandes dans le guide de IAM l'utilisateur.

Quelle que soit la méthode d'authentification que vous utilisez, vous devrez peut-être fournir des informations de sécurité supplémentaires. Par exemple, il vous AWS recommande d'utiliser

l'authentification multifactorielle (MFA) pour renforcer la sécurité de votre compte. Pour en savoir plus, voir Authentification multifactorielle dans le guide de l'AWS IAM Identity Center utilisateur et Authentification AWS multifactorielle IAM dans le guide de l'IAMutilisateur.

AWS utilisateur root du compte

Lorsque vous créez un Compte AWS, vous commencez par une identité de connexion unique qui donne un accès complet à toutes Services AWS les ressources du compte. Cette identité est appelée utilisateur Compte AWS root et est accessible en vous connectant avec l'adresse e-mail et le mot de passe que vous avez utilisés pour créer le compte. Il est vivement recommandé de ne pas utiliser l'utilisateur racine pour vos tâches quotidiennes. Protégez vos informations d'identification d'utilisateur racine et utilisez-les pour effectuer les tâches que seul l'utilisateur racine peut effectuer. Pour obtenir la liste complète des tâches qui nécessitent que vous vous connectiez en tant qu'utilisateur root, consultez la section Tâches nécessitant des informations d'identification utilisateur root dans le Guide de IAM l'utilisateur.

Identité fédérée

La meilleure pratique consiste à obliger les utilisateurs humains, y compris ceux qui ont besoin d'un accès administrateur, à utiliser la fédération avec un fournisseur d'identité pour accéder à l'aide Services AWS d'informations d'identification temporaires.

Une identité fédérée est un utilisateur de l'annuaire des utilisateurs de votre entreprise, d'un fournisseur d'identité Web AWS Directory Service, du répertoire Identity Center ou de tout utilisateur qui y accède à l'aide des informations d'identification fournies Services AWS par le biais d'une source d'identité. Lorsque des identités fédérées y accèdent Comptes AWS, elles assument des rôles, qui fournissent des informations d'identification temporaires.

Pour une gestion des accès centralisée, nous vous recommandons d'utiliser AWS IAM Identity Center. Vous pouvez créer des utilisateurs et des groupes dans IAM Identity Center, ou vous pouvez vous connecter et synchroniser avec un ensemble d'utilisateurs et de groupes dans votre propre source d'identité afin de les utiliser dans toutes vos applications Comptes AWS et applications. Pour plus d'informations sur IAM Identity Center, consultez Qu'est-ce qu'IAMIdentity Center ? dans le guide de AWS IAM Identity Center l'utilisateur.

Utilisateurs et groupes IAM

Un IAMutilisateur est une identité au sein de vous Compte AWS qui possède des autorisations spécifiques pour une seule personne ou une seule application. Dans la mesure du possible, nous

vous recommandons de vous appuyer sur des informations d'identification temporaires plutôt que de créer des IAM utilisateurs dotés d'informations d'identification à long terme, telles que des mots de passe et des clés d'accès. Toutefois, si vous avez des cas d'utilisation spécifiques qui nécessitent des informations d'identification à long terme auprès des IAM utilisateurs, nous vous recommandons de faire pivoter les clés d'accès. Pour plus d'informations, voir Rotation régulière des clés d'accès pour les cas d'utilisation nécessitant des informations d'identification à long terme dans le Guide de IAM l'utilisateur.

Un groupe IAM est une identité qui spécifie un ensemble d'utilisateurs IAM. Vous ne pouvez pas vous connecter en tant que groupe. Vous pouvez utiliser les groupes pour spécifier des autorisations pour plusieurs utilisateurs à la fois. Les groupes permettent de gérer plus facilement les autorisations pour de grands ensembles d'utilisateurs. Par exemple, vous pouvez nommer un groupe IAMAdminset lui donner les autorisations nécessaires pour administrer IAM des ressources.

Les utilisateurs sont différents des rôles. Un utilisateur est associé de manière unique à une personne ou une application, alors qu'un rôle est conçu pour être endossé par tout utilisateur qui en a besoin. Les utilisateurs disposent d'informations d'identification permanentes, mais les rôles fournissent des informations d'identification temporaires. Pour en savoir plus, consultez la section Cas d'utilisation pour IAM les utilisateurs dans le Guide de IAM l'utilisateur.

Vous pouvez vous authentifier auprès de votre d'instances de base de données à l'aide de l'authentification IAM de base de données.

L'authentification de base de données IAM fonctionne avec les moteurs de base de données suivants :

- RDS for MariaDB
- RDSpour My SQL
- RDSpour Postgre SQL

Pour plus d'informations sur l'authentification auprès de votre d'instances de base de données à l'aide IAM de. Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL

Rôles IAM

Un IAMrôle est une identité au sein de Compte AWS vous dotée d'autorisations spécifiques. Le concept ressemble à celui d'un utilisateur, mais un rôle n'est pas associé à une personne en particulier. Vous pouvez assumer temporairement un IAM rôle dans le en AWS Management Console

changeant de rôle. Vous pouvez assumer un rôle en appelant une AWS API opération AWS CLI or ou en utilisant une option personnaliséeURL. Pour plus d'informations sur les méthodes d'utilisation des rôles, consultez la section Utilisation IAM des rôles dans le Guide de IAM l'utilisateur.

Les rôles IAM avec des informations d'identification temporaires sont utiles dans les cas suivants :

- Autorisations utilisateur temporaires : un utilisateur peut assumer un IAM rôle afin d'obtenir temporairement différentes autorisations pour une tâche spécifique.
- Accès utilisateur fédéré : pour attribuer des autorisations à une identité fédérée, vous créez un rôle et définissez des autorisations pour le rôle. Quand une identité externe s'authentifie, l'identité est associée au rôle et reçoit les autorisations qui sont définies par celui-ci. Pour plus d'informations sur les rôles pour la fédération, voir Créer un rôle pour un fournisseur d'identité tiers (fédération) dans le guide de IAM l'utilisateur. Si vous utilisez IAM Identity Center, vous configurez un ensemble d'autorisations. Pour contrôler les accès auxquels vos identités peuvent accéder après leur authentification, IAM Identity Center met en corrélation l'ensemble d'autorisations avec un rôle dans. IAM Pour plus d'informations sur les jeux d'autorisations, consultez Jeux d'autorisations dans le Guide de l'utilisateur AWS IAM Identity Center .
- Accès intercompte : vous pouvez utiliser un rôle IAM pour permettre à un utilisateur (un principal de confiance) d'un compte différent d'accéder aux ressources de votre compte. Les rôles constituent le principal moyen d'accorder l'accès intercompte. Toutefois, dans certains Services AWS cas, vous pouvez associer une politique directement à une ressource (au lieu d'utiliser un rôle comme proxy). Pour connaître la différence entre les rôles et les politiques basées sur les ressources pour l'accès entre comptes, consultez la section En quoi les IAM rôles diffèrent des politiques basées sur les ressources dans le Guide de l'utilisateur. IAM
- Accès multiservices — Certains Services AWS utilisent des fonctionnalités dans d'autres Services AWS. Par exemple, lorsque vous effectuez un appel dans un service, il est courant que ce service exécute des applications dans Amazon EC2 ou stocke des objets dans Amazon S3. Un service peut le faire en utilisant les autorisations d'appel du principal, un rôle de service ou un rôle lié au service.
  - Sessions d'accès transféré : lorsque vous utilisez un IAM utilisateur ou un rôle pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui déclenche ensuite une autre action dans un autre service. FASutilise les autorisations du principal appelant an Service AWS, combinées à la demande Service AWS pour adresser des demandes aux services en aval. FASles demandes ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer

d’autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur les politiques relatives FAS aux demandes, consultez la section Transférer les sessions d'accès.

- Rôle de service — Un rôle de service est un IAMrôle qu'un service assume pour effectuer des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du service à partir de IAM. Pour plus d'informations, consultez la section Créer un rôle pour déléguer des autorisations à un Service AWS dans le guide de IAM l'utilisateur.
- Rôle lié à un service — Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d’effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier les autorisations concernant les rôles liés à un service.

- Applications exécutées sur Amazon EC2 : vous pouvez utiliser un IAM rôle pour gérer les informations d'identification temporaires pour les applications qui s'exécutent sur une EC2 instance et qui soumettent AWS CLI ou AWS API demandent des informations. Cette solution est préférable au stockage des clés d'accès au sein de l'instance EC2. Pour attribuer un AWS rôle à une EC2 instance et le rendre disponible pour toutes ses applications, vous devez créer un profil d'instance attaché à l'instance. Un profil d'instance contient le rôle et permet aux programmes qui s'exécutent sur l'instance EC2 d'obtenir des informations d'identification temporaires. Pour plus d'informations, consultez la section Utiliser un IAM rôle pour accorder des autorisations aux applications exécutées sur des EC2 instances Amazon dans le Guide de IAM l'utilisateur.

Pour savoir s'il faut utiliser IAM des rôles, voir Quand créer un IAM rôle (au lieu d'un utilisateur) dans le guide de IAM l'utilisateur.

Gestion des accès à l’aide de politiques

Vous contrôlez l'accès en AWS créant des politiques et en les associant à IAM des identités ou à AWS des ressources. Une politique est un objet AWS qui, lorsqu'il est associé à une identité ou à une ressource, définit leurs autorisations. AWS évalue ces politiques lorsqu'une entité (utilisateur root, utilisateur ou IAM rôle) fait une demande. Les autorisations dans les politiques déterminent si la demande est autorisée ou refusée. La plupart des politiques sont stockées AWS sous forme de JSON documents. Pour plus d'informations sur la structure et le contenu des documents de JSON politique, voir Présentation des JSON politiques dans le guide de IAM l'utilisateur.

Un administrateur peut utiliser des politiques pour spécifier qui a accès aux AWS ressources et quelles actions il peut effectuer sur ces ressources. Chaque IAM entité (ensemble d'autorisations ou rôle) commence sans aucune autorisation. En d'autres termes, par défaut, les utilisateurs ne

peuvent rien faire, pas même changer leurs propres mots de passe. Pour autoriser un utilisateur à effectuer une opération, un administrateur doit associer une politique d'autorisations à ce dernier. Il peut également ajouter l'utilisateur à un groupe disposant des autorisations prévues. Lorsqu'un administrateur accorde des autorisations à un groupe, tous les utilisateurs de ce groupe se voient octroyer ces autorisations.

Les stratégies IAM définissent les autorisations d'une action quelle que soit la méthode que vous utilisez pour exécuter l'opération. Par exemple, supposons que vous disposiez d'une politique qui autorise l'action `iam:GetRole`. Un utilisateur doté de cette politique peut obtenir des informations sur le rôle auprès du AWS Management Console AWS CLI, ou du AWS API.

Politiques basées sur l'identité

Les politiques basées sur l'identité sont JSON des documents de politique d'autorisation que vous pouvez joindre à une identité, telle qu'un ensemble d'autorisations ou un rôle. Ces politiques contrôlent les actions que peut exécuter cette identité, sur quelles ressources et dans quelles conditions. Pour savoir comment créer une politique basée sur l'identité, consultez la section Création de IAM politiques dans le Guide de l'IAMutilisateur.

Les politiques basées sur l'identité peuvent être classées comme des politiques en ligne ou des politiques gérées. Les politiques en ligne sont intégrées directement à un seul jeu d'autorisations ou rôle. Les politiques gérées sont des politiques autonomes que vous pouvez associer à plusieurs ensembles d'autorisations et rôles dans votre AWS compte. Les politiques gérées incluent les politiques AWS gérées et les politiques gérées par le client. Pour savoir comment choisir entre une politique gérée ou une politique intégrée, voir Choisir entre des politiques gérées et des politiques intégrées dans le Guide de l'IAMutilisateur.

Pour plus d'informations sur les politiques AWS gérées spécifiques à Amazon RDS , consultezAWS politiques gérées pour Amazon RDS.

Autres types de politique

AWS prend en charge d'autres types de politiques moins courants. Ces types de politiques peuvent définir le nombre maximum d'autorisations qui vous sont accordées par des types de politiques plus courants.

- Limites d'autorisations — Une limite d'autorisations est une fonctionnalité avancée dans laquelle vous définissez le maximum d'autorisations qu'une politique basée sur l'identité peut accorder à une IAM entité (ensemble d'autorisations ou rôle). Vous pouvez définir une limite d'autorisations

pour une entité. Les autorisations obtenues représentent la combinaison des politiques basées sur l'identité de l'entité et de ses limites d'autorisations. Les politiques basées sur les ressources qui spécifient le jeu d'autorisations ou le rôle dans le champ `Principal` ne sont pas limitées par les limites des autorisations. Un refus explicite dans l'une de ces politiques annule l'autorisation. Pour plus d'informations sur les limites d'autorisations, consultez la section Limites d'autorisations pour les IAM entités dans le Guide de IAM l'utilisateur.

- Politiques de contrôle des services (SCPs) : SCPs JSON politiques qui spécifient les autorisations maximales pour une organisation ou une unité organisationnelle (UO) dans AWS Organizations. AWS Organizations est un service permettant de regrouper et de gérer de manière centralisée plusieurs AWS comptes détenus par votre entreprise. Si vous activez toutes les fonctionnalités d'une organisation, vous pouvez appliquer des politiques de contrôle des services (SCPs) à l'un ou à l'ensemble de vos comptes. Les SCP limites d'autorisations pour les entités présentes dans les comptes des membres, y compris chacune d'entre elles Utilisateur racine d'un compte AWS. Pour plus d'informations sur les OrganizationsSCPs, voir Comment SCPs travailler dans le Guide de AWS Organizations l'utilisateur.
- Politiques de séance : les politiques de séance sont des politiques avancées que vous utilisez en tant que paramètre lorsque vous créez par programmation une séance temporaire pour un rôle ou un utilisateur fédéré. Les autorisations de la session obtenues sont une combinaison des politiques basées sur l'identité du rôle ou des jeux d'autorisations et des politiques de session. Les autorisations peuvent également provenir d'une politique basée sur les ressources. Un refus explicite dans l'une de ces politiques annule l'autorisation. Pour en savoir plus, consultez Politiques de session dans le Guide de l'utilisateur IAM.

Plusieurs types de politique

Lorsque plusieurs types de politiques s'appliquent à la requête, les autorisations en résultant sont plus compliquées à comprendre. Pour savoir comment AWS déterminer s'il faut autoriser une demande lorsque plusieurs types de politiques sont impliqués, consultez la section Logique d'évaluation des politiques dans le guide de IAM l'utilisateur.

Comment fonctionne RDSAmazon avec IAM

Avant de IAM gérer l'accès à RDSAmazon , vous devez connaître les IAM fonctionnalités disponibles pour Amazon RDS .

Le tableau suivant répertorie les IAM fonctionnalités que vous pouvez utiliser avec Amazon RDS :

Fonctionnalité IAM	Assistance RDSAmazon
Politiques basées sur l’identité	Oui
Politiques basées sur les ressources	Non
Actions de politique	Oui
Ressources de politique	Oui
Clés de condition de politique (spécifiques au service)	Oui
ACLs	Non
Contrôle d'accès basé sur les attributs (ABAC) (balises dans les politiques)	Oui
Informations d’identification temporaires	Oui
Transférer les sessions d'accès	Oui
Rôles de service	Oui
Rôles liés à un service	Oui

Pour obtenir une vue d'ensemble de la façon dont RDSAmazon et les autres AWS services fonctionnent avecIAM, consultez la section AWS Services compatibles IAM dans le Guide de IAM l'utilisateur.

Rubriques

- Politiques basées sur l'identité Amazon RDS
-
- Actions politiques pour Amazon RDS
- Ressources relatives aux politiques pour Amazon RDS
- Clés de conditions de politique pour Amazon RDS
- Listes de contrôle d'accès (ACLs) dans Amazon RDS

-
- Utilisation d'informations d'identification temporaires avec Amazon RDS
- Transférer des sessions d'accès pour Amazon RDS
- Rôles de service pour Amazon RDS
-

Politiques basées sur l'identité Amazon RDS

Prend en charge les politiques basées sur l'identité : Oui.

Les politiques basées sur l'identité sont JSON des documents de politique d'autorisation que vous pouvez joindre à une identité, telle qu'un IAM utilisateur, un groupe d'utilisateurs ou un rôle. Ces politiques contrôlent quel type d’actions des utilisateurs et des rôles peuvent exécuter, sur quelles ressources et dans quelles conditions. Pour savoir comment créer une politique basée sur l'identité, voir Définir des IAM autorisations personnalisées avec des politiques gérées par le client dans le Guide de l'IAMutilisateur.

Avec les stratégies IAM basées sur l’identité, vous pouvez spécifier des actions et ressources autorisées ou refusées, ainsi que les conditions dans lesquelles les actions sont autorisées ou refusées. Vous ne pouvez pas spécifier le principal dans une politique basée sur une identité, car celle-ci s’applique à l’utilisateur ou au rôle auquel elle est attachée. Pour en savoir plus sur tous les éléments que vous pouvez utiliser dans une JSON politique, consultez la référence aux éléments de IAM JSON politique dans le Guide de IAM l'utilisateur.

Pour consulter des exemples de politiques basées sur l'identité Amazon RDS , consultez. Exemples de politiques basées sur l'identité pour Amazon RDS

Prend en charge les politiques basées sur les ressources : Non.

Les politiques basées sur les ressources sont des documents JSON de stratégie que vous attachez à une ressource. Les politiques de confiance dans les IAM rôles et les politiques relatives aux compartiments Amazon S3 sont des exemples de politiques basées sur les ressources. Dans les services qui sont compatibles avec les politiques basées sur les ressources, les administrateurs de service peuvent les utiliser pour contrôler l’accès à une ressource spécifique. Pour la ressource dans laquelle se trouve la politique, cette dernière définit quel type d’actions un principal spécifié peut

effectuer sur cette ressource et dans quelles conditions. Vous devez spécifier un principal dans une politique basée sur les ressources. Les principaux peuvent inclure des comptes, des utilisateurs, des rôles, des utilisateurs fédérés ou. Services AWS

Pour permettre un accès entre comptes, vous pouvez spécifier un compte entier ou des entités IAM dans un autre compte en tant que mandataire dans une stratégie basée sur les ressources. L'ajout d'un principal entre comptes à une politique basée sur les ressources ne représente qu'une partie de l'instauration de la relation d'approbation. Lorsque le principal et la ressource sont différents Comptes AWS, un IAM administrateur du compte de confiance doit également accorder à l'entité principale (utilisateur ou rôle) l'autorisation d'accéder à la ressource. Pour ce faire, il attache une politique basée sur une identité à l'entité. Toutefois, si une politique basée sur des ressources accorde l'accès à un principal dans le même compte, aucune autre politique basée sur l'identité n'est requise. Pour plus d'informations, voir Accès aux ressources entre comptes IAM dans le Guide de IAM l'utilisateur.

Actions politiques pour Amazon RDS

Soutient les actions politiques : Oui.

Les administrateurs peuvent utiliser AWS JSON des politiques pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'`Action`élément d'une JSON politique décrit les actions que vous pouvez utiliser pour autoriser ou refuser l'accès dans une politique. Les actions de stratégie portent généralement le même nom que l' AWS APIopération associée. Il existe certaines exceptions, telles que les actions avec autorisation uniquement qui n'ont pas d'opération correspondante. API Certaines opérations nécessitent également plusieurs actions dans une politique. Ces actions supplémentaires sont nommées actions dépendantes.

Intégration d'actions dans une politique afin d'accorder l'autorisation d'exécuter les opérations associées.

Les actions politiques dans Amazon RDS utilisent le préfixe suivant avant l'action :`rds:`.
Par exemple, pour autoriser quelqu'un à décrire des instances de base de données avec l'RDS`DescribeDBInstances`APIopération Amazon, vous devez inclure l'`rds:DescribeDBInstances`action dans sa politique. Les déclarations de politique doivent inclure un élément `Action` ou `NotAction`. Amazon RDS définit son propre ensemble d'actions décrivant les tâches que vous pouvez effectuer avec ce service.

Pour spécifier plusieurs actions dans une seule instruction, séparez-les par des virgules, comme suit :

```
"Action": [
      "rds:action1",
      "rds:action2"
```

Vous pouvez aussi préciser plusieurs actions à l'aide de caractères génériques (*). Par exemple, pour spécifier toutes les actions qui commencent par le mot `Describe`, incluez l'action suivante.

```
"Action": "rds:Describe*"
```

Pour consulter la liste des actions Amazon RDS , consultez la section Actions définies par Amazon RDS dans le Service Authorization Reference.

Ressources relatives aux politiques pour Amazon RDS

Prend en charge les ressources politiques : Oui.

Les administrateurs peuvent utiliser AWS JSON des politiques pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Resource` JSON de stratégie indique le ou les objets auxquels s'applique l'action. Les instructions doivent inclure un élément `Resource` ou `NotResource`. Il est recommandé de spécifier une ressource en utilisant son Amazon Resource Name (ARN). Vous pouvez le faire pour des actions qui prennent en charge un type de ressource spécifique, connu sous la dénomination autorisations de niveau ressource.

Pour les actions qui ne sont pas compatibles avec les autorisations de niveau ressource, telles que les opérations de liste, utilisez un caractère générique (*) afin d'indiquer que l'instruction s'applique à toutes les ressources.

```
"Resource": "*"
```

La ressource d'instance de base de données possède le nom de ressource Amazon suivant (ARN).

```
arn:${Partition}:rds:${Region}:${Account}:{ResourceType}/${Resource}
```

Pour plus d'informations sur le format deARNs, consultez Amazon Resource Names (ARNs) et espaces de noms AWS de services.

Par exemple, pour spécifier l'`dbtest`instance de base de données dans votre instruction, utilisez ce qui suitARN.

```
"Resource": "arn:aws:rds:us-west-2:123456789012:db:dbtest"
```

Pour spécifier toutes les instances de base de données qui appartiennent à un compte spécifique, utilisez le caractère générique (*).

```
"Resource": "arn:aws:rds:us-east-1:123456789012:db:*"
```

Certaines RDS API opérations, telles que celles relatives à la création de ressources, ne peuvent pas être effectuées sur une ressource spécifique. Dans ces cas-là, utilisez le caractère générique (*).

```
"Resource": "*"
```

De nombreuses RDS API opérations Amazon impliquent plusieurs ressources. Par exemple, `CreateDBInstance` crée une instance de base de données. Vous pouvez spécifier qu'un utilisateur doit utiliser un groupe de sécurité spécifique et un groupe de paramètres lors de la création d'une instance de base de données. Pour spécifier plusieurs ressources dans une seule instruction, séparez-les ARNs par des virgules.

```
"Resource": [
      "resource1",
      "resource2"
```

Pour consulter la liste des types de ressources Amazon RDS et leurs caractéristiquesARNs, consultez la section Ressources définies par Amazon RDS dans le Service Authorization Reference. Pour savoir avec quelles actions vous pouvez spécifier pour chaque ressource, consultez Actions définies par Amazon RDS. ARN

Clés de conditions de politique pour Amazon RDS

Prend en charge les clés de condition de politique spécifiques au service : Oui.

Les administrateurs peuvent utiliser AWS JSON des politiques pour spécifier qui a accès à quoi. C'est-à-dire, quel principal peut effectuer des actions sur quelles ressources et dans quelles conditions.

L'élément `Condition` (ou le bloc `Condition`) vous permet de spécifier des conditions lorsqu'une instruction est appliquée. L'élément `Condition` est facultatif. Vous pouvez créer des expressions conditionnelles qui utilisent des opérateurs de condition, tels que les signes égal ou inférieur à, pour faire correspondre la condition de la politique aux valeurs de la demande.

Si vous spécifiez plusieurs éléments `Condition` dans une instruction, ou plusieurs clés dans un seul élément `Condition`, AWS les évalue à l'aide d'une opération `AND` logique. Si vous spécifiez plusieurs valeurs pour une seule clé de condition, AWS évalue la condition à l'aide d'une `OR` opération logique. Toutes les conditions doivent être remplies avant que les autorisations associées à l'instruction ne soient accordées.

Vous pouvez aussi utiliser des variables d'espace réservé quand vous spécifiez des conditions. Par exemple, vous pouvez accorder à un utilisateur IAM l'autorisation d'accéder à une ressource uniquement si elle est balisée avec son nom d'utilisateur IAM . Pour plus d'informations, consultez IAMla section Éléments de politique : variables et balises dans le Guide de IAM l'utilisateur.

AWS prend en charge les clés de condition globales et les clés de condition spécifiques au service. Pour voir toutes les clés de condition AWS globales, voir les clés contextuelles de condition AWS globales dans le guide de IAM l'utilisateur.

Amazon RDS définit son propre ensemble de clés de condition et prend également en charge l'utilisation de certaines clés de condition globales. Pour voir toutes les clés de condition AWS globales, voir les clés contextuelles de condition AWS globales dans le guide de IAM l'utilisateur.

Toutes les RDS API opérations prennent en charge la clé de `aws:RequestedRegion` condition.

Pour consulter la liste des clés de condition Amazon RDS , consultez la section Clés de condition pour Amazon RDS dans la référence d'autorisation de service. Pour savoir avec quelles actions et ressources vous pouvez utiliser une clé de condition, consultez Actions définies par Amazon RDS.

Listes de contrôle d'accès (ACLs) dans Amazon RDS

Supporte les listes de contrôle d'accès (ACLs) : Non

Les listes de contrôle d'accès (ACLs) contrôlent les principaux (membres du compte, utilisateurs ou rôles) autorisés à accéder à une ressource. ACLssont similaires aux politiques basées sur les ressources, bien qu'elles n'utilisent pas le format du document JSON de stratégie.

Prend en charge les balises de contrôle d'accès basées sur les attributs (ABAC) dans les politiques : Oui

Le contrôle d'accès basé sur les attributs (ABAC) est une stratégie d'autorisation qui définit les autorisations en fonction des attributs. Dans AWS, ces attributs sont appelés balises. Vous pouvez associer des balises à IAM des entités (utilisateurs ou rôles) et à de nombreuses AWS ressources. Le balisage des entités et des ressources est la première étape deABAC. Vous concevez ensuite des ABAC politiques pour autoriser les opérations lorsque le tag du principal correspond à celui de la ressource à laquelle il essaie d'accéder.

ABACest utile dans les environnements qui se développent rapidement et aide dans les situations où la gestion des politiques devient fastidieuse.

Pour contrôler l'accès basé sur des étiquettes, vous devez fournir les informations d'étiquette dans l'élément de condition d'une politique utilisant les clés de condition `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` ou `aws:TagKeys`.

Si un service prend en charge les trois clés de condition pour tous les types de ressources, alors la valeur pour ce service est Oui. Si un service prend en charge les trois clés de condition pour certains types de ressources uniquement, la valeur est Partielle.

Pour plus d'informationsABAC, voir Définir des autorisations avec ABAC autorisation dans le Guide de IAM l'utilisateur. Pour consulter un didacticiel présentant les étapes de configurationABAC, voir Utiliser le contrôle d'accès basé sur les attributs (ABAC) dans le guide de l'IAMutilisateur.

Pour plus d'informations sur le balisage des ressources Amazon RDS , consultezSpécification de conditions : Utilisation de balises personnalisées. Pour visualiser un exemple de politique basée sur l'identité permettant de limiter l'accès à une ressource en fonction des balises de cette ressource, consultez Accorder une autorisation pour des actions sur une ressource à l'aide d'une balise spécifique avec deux valeurs différentes.

Utilisation d'informations d'identification temporaires avec Amazon RDS

Supporte les informations d'identification temporaires : Oui.

Certains Services AWS ne fonctionnent pas lorsque vous vous connectez à l'aide d'informations d'identification temporaires. Pour plus d'informations, y compris celles qui Services AWS fonctionnent avec des informations d'identification temporaires, consultez Services AWS la section relative à l'utilisation IAM dans le Guide de IAM l'utilisateur.

Vous utilisez des informations d'identification temporaires si vous vous connectez à l' AWS Management Console aide d'une méthode autre qu'un nom d'utilisateur et un mot de passe. Par

exemple, lorsque vous accédez à AWS l'aide du lien d'authentification unique (SSO) de votre entreprise, ce processus crée automatiquement des informations d'identification temporaires. Vous créez également automatiquement des informations d'identification temporaires lorsque vous vous connectez à la console en tant qu'utilisateur, puis changez de rôle. Pour plus d'informations sur le changement de rôle, voir Passer d'un utilisateur à un IAM rôle (console) dans le guide de IAM l'utilisateur.

Vous pouvez créer manuellement des informations d'identification temporaires à l'aide du AWS CLI ou AWS API. Vous pouvez ensuite utiliser ces informations d'identification temporaires pour y accéder AWS. AWS recommande de générer dynamiquement des informations d'identification temporaires au lieu d'utiliser des clés d'accès à long terme. Pour plus d'informations, consultez la section Informations d'identification de sécurité temporaires dans IAM.

Transférer des sessions d'accès pour Amazon RDS

Prend en charge les sessions d'accès transféré : Oui.

Lorsque vous utilisez un IAM utilisateur ou un rôle pour effectuer des actions AWS, vous êtes considéré comme un mandant. Lorsque vous utilisez certains services, vous pouvez effectuer une action qui déclenche ensuite une autre action dans un autre service. FASutilise les autorisations du principal appelant an Service AWS, combinées à la demande Service AWS pour adresser des demandes aux services en aval. FASles demandes ne sont effectuées que lorsqu'un service reçoit une demande qui nécessite des interactions avec d'autres personnes Services AWS ou des ressources pour être traitée. Dans ce cas, vous devez disposer d'autorisations nécessaires pour effectuer les deux actions. Pour plus de détails sur les politiques relatives FAS aux demandes, consultez la section Transférer les sessions d'accès.

Rôles de service pour Amazon RDS

Supporte les rôles de service : Oui.

Un rôle de service est un IAMrôle qu'un service assume pour effectuer des actions en votre nom. Un administrateur IAM peut créer, modifier et supprimer une fonction du service à partir de IAM. Pour plus d'informations, consultez la section Créer un rôle pour déléguer des autorisations à un Service AWS dans le guide de IAM l'utilisateur.

> **Warning**
>
> La modification des autorisations associées à un rôle de service peut perturber les fonctionnalités d'Amazon RDS . Modifiez les rôles de service uniquement lorsque Amazon RDS fournit des instructions à cet effet.

Prend en charge les rôles liés aux services : Oui.

Un rôle lié à un service est un type de rôle de service lié à un. Service AWS Le service peut endosser le rôle afin d'effectuer une action en votre nom. Les rôles liés à un service apparaissent dans votre Compte AWS répertoire et appartiennent au service. Un administrateur IAM peut consulter, mais ne peut pas modifier les autorisations concernant les rôles liés à un service.

Pour en savoir plus sur l'utilisation des rôles liés aux services Amazon RDS , consultez. Utilisation des rôles liés à un service pour Amazon RDS

Exemples de politiques basées sur l'identité pour Amazon RDS

Par défaut, les jeux d'autorisations et les rôles ne sont pas autorisés à créer ou modifier des ressources Amazon RDS. Ils ne peuvent pas non plus effectuer de tâches à l'aide de l' AWS API AWS Management Console AWS CLI, ou. Un administrateur doit créer des politiques IAM autorisant les jeux d'autorisations et les rôles à exécuter des opérations d'API spécifiques sur les ressources spécifiées dont ils ont besoin. L'administrateur doit ensuite attacher ces politiques aux jeux d'autorisations et aux rôles qui ont besoin de ces autorisations.

Pour apprendre à créer une politique basée sur l'identité IAM à l'aide de ces exemples de documents de politique JSON, veuillez consulter Création de politiques dans l'onglet JSON dans le Guide de l'utilisateur IAM.

Rubriques

- Bonnes pratiques en matière de politiques
- Utilisation de la console Amazon RDS
- Autorisations requises pour utiliser la console
- Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations
- Politiques d'autorisation pour créer, modifier et supprimer des ressources dans Amazon RDS

- Exemples de politiques : Utilisation des clés de condition
- Spécification de conditions : Utilisation de balises personnalisées
- Accorder l'autorisation de baliser les ressources Amazon RDS lors de leur création

Bonnes pratiques en matière de politiques

Les stratégies basées sur l'identité déterminent si une personne peut créer, consulter ou supprimer des ressources Amazon RDS dans votre compte. Ces actions peuvent entraîner des frais pour votre Compte AWS. Lorsque vous créez ou modifiez des politiques basées sur l'identité, suivez ces instructions et recommandations :

- Commencez AWS par les politiques gérées et passez aux autorisations du moindre privilège : pour commencer à accorder des autorisations à vos utilisateurs et à vos charges de travail, utilisez les politiques AWS gérées qui accordent des autorisations pour de nombreux cas d'utilisation courants. Ils sont disponibles dans votre Compte AWS. Nous vous recommandons de réduire davantage les autorisations en définissant des politiques gérées par les AWS clients spécifiques à vos cas d'utilisation. Pour plus d'informations, consultez politiques gérées par AWS ou politiques gérées par AWS pour les activités professionnelles dans le Guide de l'utilisateur IAM.
- Accordez les autorisations de moindre privilège : lorsque vous définissez des autorisations avec des politiques IAM, accordez uniquement les autorisations nécessaires à l'exécution d'une seule tâche. Pour ce faire, vous définissez les actions qui peuvent être entreprises sur des ressources spécifiques dans des conditions spécifiques, également appelées autorisations de moindre privilège. Pour plus d'informations sur l'utilisation d'IAM pour appliquer des autorisations, consultez politiques et autorisations dans IAM dans le Guide de l'utilisateur IAM.
- Utilisez des conditions dans les politiques IAM pour restreindre davantage l'accès : vous pouvez ajouter une condition à vos politiques afin de limiter l'accès aux actions et aux ressources. Par exemple, vous pouvez écrire une condition de politique pour spécifier que toutes les demandes doivent être envoyées via SSL. Vous pouvez également utiliser des conditions pour accorder l'accès aux actions de service si elles sont utilisées par le biais d'un service spécifique Service AWS, tel que AWS CloudFormation. Pour plus d'informations, consultez Conditions pour éléments de politique JSON IAM dans le Guide de l'utilisateur IAM.
- Utilisez l'Analyseur d'accès IAM pour valider vos politiques IAM afin de garantir des autorisations sécurisées et fonctionnelles : l'Analyseur d'accès IAM valide les politiques nouvelles et existantes de manière à ce que les politiques IAM respectent le langage de politique IAM (JSON) et les bonnes pratiques IAM. IAM Access Analyzer fournit plus de 100 vérifications de politiques et des recommandations exploitables pour vous aider à créer des politiques sécurisées et fonctionnelles.

Pour plus d’informations, consultez Validation de politiques avec IAM Access Analyzer dans le Guide de l’utilisateur IAM.

- Exiger l'authentification multifactorielle (MFA) : si vous avez un scénario qui nécessite des utilisateurs IAM ou un utilisateur root, activez l'authentification MFA pour une sécurité accrue. Compte AWS Pour exiger la MFA lorsque des opérations d’API sont appelées, ajoutez des conditions MFA à vos politiques. Pour plus d’informations, consultez Sécurisation de l’accès aux API avec MFA dans le Guide de l’utilisateur IAM.

Pour plus d’informations sur les bonnes pratiques dans IAM, consultez Bonnes pratiques de sécurité dans IAM dans le Guide de l’utilisateur IAM.

Utilisation de la console Amazon RDS

Pour accéder à la console Amazon RDS, vous devez disposer d'un ensemble minimal d'autorisations. Ces autorisations doivent vous permettre de répertorier et de consulter les informations relatives aux ressources Amazon RDS présentes dans votre Compte AWS. Si vous créez une politique basée sur l’identité qui est plus restrictive que l’ensemble minimum d’autorisations requis, la console ne fonctionnera pas comme prévu pour les entités (utilisateurs ou rôles) tributaires de cette politique.

Il n'est pas nécessaire d'accorder des autorisations de console minimales aux utilisateurs qui appellent uniquement l'API AWS CLI ou l' AWS API. Autorisez plutôt l’accès à uniquement aux actions qui correspondent à l’opération d’API que vous tentez d’effectuer.

Pour garantir que ces entités peuvent toujours utiliser la console Amazon RDS , associez également la politique AWS gérée suivante aux entités.

```
AmazonRDSReadOnlyAccess
```

Pour plus d’informations, consultez Ajout d’autorisations à un utilisateur dans le Guide de l’utilisateur IAM.

Autorisations requises pour utiliser la console

Pour qu'un utilisateur puisse utiliser la console, il doit avoir un ensemble minimal d'autorisations. Ces autorisations permettent à l'utilisateur de décrire les ressources Amazon RDS associées à son AWS compte et de fournir d'autres informations connexes, notamment des informations relatives à EC2 la sécurité et au réseau Amazon.

Si vous créez une politique IAM plus restrictive que les autorisations minimales requises, la console ne fonctionne pas comme prévu pour les utilisateurs dotés de cette politique IAM. Pour garantir que ces utilisateurs puissent continuer à utiliser la console, attachez également la stratégie gérée `AmazonRDSReadOnlyAccess` à l'utilisateur, comme décrit dans Gestion des accès à l’aide de politiques.

Vous n'avez pas besoin d'accorder d'autorisations minimales d'utilisation de la console aux utilisateurs qui effectuent des appels uniquement à l' AWS CLI ou à l'API Amazon RDS.

La politique suivante accorde un accès complet à toutes les ressources Amazon RDS pour le AWS compte racine :

```
AmazonRDSFullAccess
```

Autorisation accordée aux utilisateurs pour afficher leurs propres autorisations

Cet exemple montre comment créer une politique qui permet aux utilisateurs IAM d’afficher les politiques en ligne et gérées attachées à leur identité d’utilisateur. Cette politique inclut les autorisations permettant d'effectuer cette action sur la console ou par programmation à l'aide de l'API AWS CLI or AWS .

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "ViewOwnUserInfo",
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
```

```
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

Politiques d'autorisation pour créer, modifier et supprimer des ressources dans Amazon RDS

Les sections suivantes présentent des exemples de politiques d'autorisation qui accordent et limitent l'accès aux ressources :

Autoriser un utilisateur à créer des instances de base de données dans un AWS compte

Voici un exemple de politique qui permet au compte doté de l'ID de `123456789012` créer des instances de base de données pour votre AWS compte. La stratégie exige que le nom de la nouvelle instance de base de données commence par `test`. La nouvelle instance de base de données doit également utiliser le moteur de base de données MySQL et la classe d'instance de base de données `db.t2.micro`. En outre, la nouvelle instance de base de données doit utiliser un groupe d'options et un groupe de paramètres de base de données commençant par `default`, et elle doit utiliser le groupe de sous-réseaux `default`.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowCreateDBInstanceOnly",
         "Effect": "Allow",
         "Action": [
            "rds:CreateDBInstance"
```

```
            ],
            "Resource": [
               "arn:aws:rds:*:123456789012:db:test*",
               "arn:aws:rds:*:123456789012:og:default*",
               "arn:aws:rds:*:123456789012:pg:default*",
               "arn:aws:rds:*:123456789012:subgrp:default"
            ],
            "Condition": {
               "StringEquals": {
                  "rds:DatabaseEngine": "mysql",
                  "rds:DatabaseClass": "db.t2.micro"
               }
            }
         }
      ]
}
```

La stratégie inclut une instruction unique spécifiant les autorisations suivantes pour l'utilisateur :

- La politique permet au compte de créer une instance de base de données à l'aide de l'opération Create DBInstance API (cela s'applique également à la create-db-instance AWS CLI commande et au AWS Management Console).
- L'élément `Resource` spécifie que l'utilisateur peut effectuer des actions sur et avec des ressources. Vous indiquez des ressources à l'aide d'un nom ARN (Amazon Resources Name). Cet ARN inclut le nom du service auquel appartient la ressource (`rds`), la AWS région (*indique n'importe quelle région dans cet exemple), le numéro de AWS compte (`123456789012`il s'agit du numéro de compte dans cet exemple) et le type de ressource. Pour plus d'informations sur la création ARNs, consultezNoms des ressources Amazon (ARNs) dans Amazon RDS.

  L'élément `Resource` dans l'exemple spécifie les contraintes de stratégie suivantes sur les ressources de l'utilisateur :

  - L'identifiant d'instance de base de données de la nouvelle instance de base de données doit commencer par `test` (par exemple, `testCustomerData1`, `test-region2-data`).
  - Le groupe d'options de la nouvelle instance de base de données doit commencer par `default`.
  - Le groupe de paramètres de base de données de la nouvelle instance de base de données doit commencer par `default`.
  - Le groupe de sous-réseaux de la nouvelle instance de base de données doit être le groupe de sous-réseaux `default`.

- L'élément `Condition` indique que le moteur de base de données doit être MySQL et la classe d'instance de base de données doit être `db.t2.micro`. L'élément `Condition` indique les conditions lorsqu'une stratégie doit entrer en vigueur. Vous pouvez ajouter des autorisations ou des restrictions supplémentaires à l'aide de l'élément `Condition`. Pour plus d'informations sur la spécification de conditions, consultez Clés de conditions de politique pour Amazon RDS. Cet exemple spécifie les conditions `rds:DatabaseEngine` et `rds:DatabaseClass`. Pour plus d'informations sur les valeurs de condition valides pour`rds:DatabaseEngine`, consultez la liste sous le `Engine` paramètre dans Create DBInstance. Pour plus d'informations sur les valeurs de conditions valides pour `rds:DatabaseClass`, veuillez consulter Moteurs de base de données pris en charge pour les classes d'instance de base de données .

La politique ne spécifie pas l'élément `Principal` car, dans une politique basée sur une identité, vous ne spécifiez pas le principal qui obtient l'autorisation. Quand vous attachez une politique à un utilisateur, l'utilisateur est le principal implicite. Lorsque vous attachez une politique d'autorisation à un rôle IAM, le principal identifié dans la politique d'approbation de ce rôle obtient les autorisations.

Pour afficher la liste des actions Amazon RDS, consultez Actions définies par Amazon RDS dans Référence de l'autorisation de service.

Autoriser un utilisateur à effectuer une action Describe sur une ressource RDS

La politique d'autorisation suivante accorde des autorisations à un utilisateur lui permettant d'exécuter toutes les actions commençant par `Describe`. Ces actions affichent des informations sur une ressource RDS, telle qu'une instance de base de données. Le caractère générique (*) figurant dans l'élément `Resource` indique que les actions sont autorisées pour toutes les ressources Amazon RDS détenues par le compte.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowRDSDescribe",
         "Effect": "Allow",
         "Action": "rds:Describe*",
         "Resource": "*"
      }
   ]
```

```
}
```

Autoriser un utilisateur à créer une instance de base de données qui utilise le groupe de paramètres de base de données et le groupe de sous-réseau spécifiés

La politique d'autorisation suivante accorde des autorisations permettant à un utilisateur de créer uniquement une instance de base de données devant utiliser le groupe de paramètres de base de données `mydbpg` et le groupe de sous-réseau de base de données `mydbsubnetgroup`.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "VisualEditor0",
         "Effect": "Allow",
         "Action": "rds:CreateDBInstance",
         "Resource": [
            "arn:aws:rds:*:*:pg:mydbpg",
            "arn:aws:rds:*:*:subgrp:mydbsubnetgroup"
         ]
      }
   ]
}
```

Accorder une autorisation pour des actions sur une ressource à l'aide d'une balise spécifique avec deux valeurs différentes

Vous pouvez utiliser des conditions dans votre stratégie basée sur l'identité pour contrôler l'accès aux ressources Amazon RDS en fonction des balises. La politique suivante accorde l'autorisation d'exécuter l'opération d'API `CreateDBSnapshot` sur les instances de base de données avec la balise `stage` définie sur `development` ou `test`.

JSON

```
{
   "Version":"2012-10-17",
```

```
   "Statement":[
      {
         "Sid":"AllowAnySnapshotName",
         "Effect":"Allow",
         "Action":[
            "rds:CreateDBSnapshot"
         ],
         "Resource":"arn:aws:rds:*:123456789012:snapshot:*"
      },
      {
         "Sid":"AllowDevTestToCreateSnapshot",
         "Effect":"Allow",
         "Action":[
            "rds:CreateDBSnapshot"
         ],
         "Resource":"arn:aws:rds:*:123456789012:db:*",
         "Condition":{
            "StringEquals":{
                "rds:db-tag/stage":[
                  "development",
                  "test"
               ]
            }
         }
      }
   ]
}
```

La politique suivante accorde l'autorisation d'exécuter l'opération d'API `ModifyDBInstance` sur les instances de base de données avec la balise `stage` définie sur `development` ou `test`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowChangingParameterOptionSecurityGroups",
         "Effect":"Allow",
         "Action":[
            "rds:ModifyDBInstance"
```

```
         ],
         "Resource": [
            "arn:aws:rds:*:123456789012:pg:*",
            "arn:aws:rds:*:123456789012:secgrp:*",
            "arn:aws:rds:*:123456789012:og:*"
         ]
      },
      {
         "Sid":"AllowDevTestToModifyInstance",
         "Effect":"Allow",
         "Action":[
            "rds:ModifyDBInstance"
         ],
         "Resource":"arn:aws:rds:*:123456789012:db:*",
         "Condition":{
            "StringEquals":{
                "rds:db-tag/stage":[
                  "development",
                  "test"
               ]
            }
         }
      }
   ]
}
```

Empêcher un utilisateur de supprimer une instance de base de données

La politique d'autorisation suivante accorde des autorisations empêchant un utilisateur de supprimer une instance de base de données spécifique. Par exemple, il est possible de refuser la capacité à supprimer vos instances de base de données de production à un utilisateur quelconque qui n'est pas un administrateur.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
```

```
      {
         "Sid": "DenyDelete1",
         "Effect": "Deny",
         "Action": "rds:DeleteDBInstance",
         "Resource": "arn:aws:rds:us-west-2:123456789012:db:my-mysql-instance"
      }
   ]
}
```

Refuser tout accès à une ressource

Vous pouvez refuser explicitement l'accès à une ressource. Les politiques de refus ont priorité sur les politiques d'autorisation. La politique suivante refuse explicitement à un utilisateur la possibilité de gérer une ressource :

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Deny",
         "Action": "rds:*",
         "Resource": "arn:aws:rds:us-east-1:123456789012:db:mydb"
      }
   ]
}
```

Exemples de politiques : Utilisation des clés de condition

Les exemples suivants montrent comment vous pouvez utiliser des clés de condition dans les stratégies d'autorisation IAM Amazon RDS.

Exemple 1 : Accorder l'autorisation de créer une instance de base de données qui utilise un moteur de base de données spécifique et n'est pas Multi-AZ

La politique suivante utilise une clé de condition RDS et autorise un utilisateur à créer seulement des instances de bases de données qui utilisent le moteur de base de données MySQL et n'utilisent

pas la configuration Multi-AZ. L'élément `Condition` indique l'exigence que le moteur de base de données soit MySQL.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "AllowMySQLCreate",
         "Effect": "Allow",
         "Action": "rds:CreateDBInstance",
         "Resource": "*",
         "Condition": {
            "StringEquals": {
               "rds:DatabaseEngine": "mysql"
            },
            "Bool": {
               "rds:MultiAz": false
            }
         }
      }
   ]
}
```

Exemple 2 : Refuser explicitement l'autorisation de créer des instances de bases de données pour certaines classes d'instance de base de données et de créer des instances de bases de données qui utilisent les IOPS provisionnées

La stratégie suivante refuse explicitement l'autorisation de créer des instances de bases de données qui utilisent les classes d'instance de base de données `r3.8xlarge` et `m4.10xlarge`, lesquelles représentent les classes d'instances de base de données les plus grandes et les plus onéreuses. Cette politique empêche également les utilisateurs de créer des instances de bases de données qui utilisent les IOPS provisionnées, ce qui génère un coût additionnel.

Le refus explicite d'une autorisation a priorité sur toutes les autres autorisations accordées. Cela garantit que des identités n'obtiendront pas par erreur une autorisation que vous ne souhaitez pas accorder.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "DenyLargeCreate",
         "Effect": "Deny",
         "Action": "rds:CreateDBInstance",
         "Resource": "*",
         "Condition": {
            "StringEquals": {
               "rds:DatabaseClass": [
                  "db.r3.8xlarge",
                  "db.m4.10xlarge"
               ]
            }
         }
      },
      {
         "Sid": "DenyPIOPSCreate",
         "Effect": "Deny",
         "Action": "rds:CreateDBInstance",
         "Resource": "*",
         "Condition": {
            "NumericNotEquals": {
               "rds:Piops": "0"
            }
         }
      }
   ]
}
```

Exemple 3 : Limiter l'ensemble de clés et de valeurs de balise pouvant être utilisées pour baliser une ressource

La politique suivante utilise une clé de condition RDS et autorise l'ajout d'une balise avec la clé `stage` à une ressource avec les valeurs `test`, `qa` et `production`.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowTagEdits",
      "Effect": "Allow",
      "Action": [
        "rds:AddTagsToResource",
        "rds:RemoveTagsFromResource"
      ],
      "Resource": "arn:aws:rds:us-east-1:123456789012:db:db-123456",
      "Condition": {
        "StringEquals": {
          "rds:req-tag/stage": [
            "test",
            "qa",
            "production"
          ]
        }
      }
    }
  ]
}
```

Spécification de conditions : Utilisation de balises personnalisées

Amazon RDS prend en charge la spécification de conditions dans une stratégie IAM à l'aide de balises personnalisées.

Par exemple, supposons que vous ajoutiez une balise nommée `environment` à vos instances de base de données avec des valeurs telles que `beta`, `staging`, `production`, etc. Dans ce cas, vous pouvez créer une stratégie qui limite certains utilisateurs aux instances de base de données fondées sur la valeur de balise `environment`.

> **Note**
>
> Les identifiants des balises personnalisées sont sensibles à la casse.

Le tableau suivant répertorie les identifiants des balises RDS que vous pouvez utiliser dans un élément `Condition`.

Identifiant de balise RDS	S'applique à
`db-tag`	Instances de base de données, y compris les réplicas en lecture
`snapshot-tag`	Instantanés de base de données
`ri-tag`	Instances de base de données réservées
`og-tag`	Groupes d'options DB
`pg-tag`	Groupes de paramètres DB
`subgrp-tag`	Groupes de sous-réseaux DB
`es-tag`	Abonnements aux événements
`cluster-tag`	Clusters DB
`cluster-pg-tag`	Groupes de paramètres de cluster DB
`cluster-snapshot-tag`	Instantanés de cluster DB

La syntaxe d'une condition de balise personnalisée est la suivante :

`"Condition":{"StringEquals":{"rds:`*`rds-tag-identifier`*`/`*`tag-name`*`": ["`*`value`*`"]} }`

Par exemple, l'élément `Condition` suivant s'applique aux instances de bases de données avec une balise nommée `environment` et la valeur de balise `production`.

`"Condition":{"StringEquals":{"rds:db-tag/`**`environment`**`": ["`**`production`**`"]} }`

Pour plus d'informations sur la création de balises, consultez Marquage d' Amazon RDS.

Important

Si vous gérez l'accès à vos ressources RDS à l'aide du balisage, nous vous recommandons de sécuriser l'accès aux balises pour vos ressources RDS. Vous pouvez gérer l'accès aux balises en créant des stratégies pour les actions `AddTagsToResource` et `RemoveTagsFromResource`. Par exemple, la politique suivante refuse aux utilisateurs la capacité à ajouter ou supprimer des balises pour toutes les ressources. Vous pouvez alors créer des politiques pour autoriser des utilisateurs spécifiques à ajouter ou supprimer des balises.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"DenyTagUpdates",
         "Effect":"Deny",
         "Action":[
            "rds:AddTagsToResource",
            "rds:RemoveTagsFromResource"
         ],
         "Resource":"*"
      }
   ]
}
```

Pour afficher la liste des actions Amazon RDS, consultez Actions définies par Amazon RDS dans Référence de l'autorisation de service.

Exemples de politiques : Utilisation de balises personnalisées

Les exemples suivants montrent comment vous pouvez utiliser des balises personnalisées dans les stratégies d'autorisation IAM Amazon RDS. Pour plus d'informations sur l'ajout de balises à une ressource Amazon RDS, consultez Noms des ressources Amazon (ARNs) dans Amazon RDS.

> **Note**
>
> Tous les exemples utilisent la région us-west-2 et contiennent un compte fictif. IDs

Exemple 1 : Accorder une autorisation pour des actions sur une ressource à l'aide d'une balise spécifique avec deux valeurs différentes

La politique suivante accorde l'autorisation d'exécuter l'opération d'API `CreateDBSnapshot` sur les instances de base de données avec la balise `stage` définie sur `development` ou `test`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowAnySnapshotName",
         "Effect":"Allow",
         "Action":[
            "rds:CreateDBSnapshot"
         ],
         "Resource":"arn:aws:rds:*:123456789012:snapshot:*"
      },
      {
         "Sid":"AllowDevTestToCreateSnapshot",
         "Effect":"Allow",
         "Action":[
            "rds:CreateDBSnapshot"
         ],
         "Resource":"arn:aws:rds:*:123456789012:db:*",
         "Condition":{
            "StringEquals":{
                "rds:db-tag/stage":[
                  "development",
                  "test"
               ]
            }
         }
      }
   ]
```

```
}
```

La politique suivante accorde l'autorisation d'exécuter l'opération d'API `ModifyDBInstance` sur les instances de base de données avec la balise `stage` définie sur `development` ou `test`.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowChangingParameterOptionSecurityGroups",
         "Effect":"Allow",
         "Action":[
            "rds:ModifyDBInstance"
            ],
          "Resource": [
            "arn:aws:rds:*:123456789012:pg:*",
            "arn:aws:rds:*:123456789012:secgrp:*",
            "arn:aws:rds:*:123456789012:og:*"
            ]
       },
       {
         "Sid":"AllowDevTestToModifyInstance",
         "Effect":"Allow",
         "Action":[
            "rds:ModifyDBInstance"
            ],
         "Resource":"arn:aws:rds:*:123456789012:db:*",
         "Condition":{
            "StringEquals":{
               "rds:db-tag/stage":[
                  "development",
                  "test"
                  ]
               }
            }
       }
    ]
}
```

Exemple 2 : Refuser explicitement l'autorisation de créer une instance de base de données qui utilise les groupes de paramètres DB spécifiés

La politique suivante refuse explicitement l'autorisation de créer une instance de base de données qui utilise les groupes de paramètres DB avec des valeurs de balise spécifiques. Vous pouvez appliquer cette politique si vous avez besoin qu'un groupe de paramètres DB créé par le client soit toujours utilisé lors de la création des instances de bases de données. Notez que les stratégies qui utilisent `Deny` sont le plus souvent utilisées pour limiter un accès accordé par une stratégie plus large.

Le refus explicite d'une autorisation a priorité sur toutes les autres autorisations accordées. Cela garantit que des identités n'obtiendront pas par erreur une autorisation que vous ne souhaitez pas accorder.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"DenyProductionCreate",
         "Effect":"Deny",
         "Action":"rds:CreateDBInstance",
         "Resource":"arn:aws:rds:*:123456789012:pg:*",
         "Condition":{
            "StringEquals":{
               "rds:pg-tag/usage":"prod"
            }
         }
      }
   ]
}
```

Exemple 3 : Accorder une autorisation pour des actions sur une instance de base de données dont le nom d'instance a un nom d'utilisateur comme préfixe

La stratégie suivante accorde l'autorisation d'appeler une API quelconque (à l'exception de `AddTagsToResource` et de `RemoveTagsFromResource`) sur une instance de base de données dont le nom d'instance de base de données a comme préfixe le nom de l'utilisateur et a une balise nommée `stage` égale à `devo` ou qui n'a pas de balise nommée `stage`.

La ligne `Resource` dans la stratégie identifie une ressource par son Amazon Resource Name (ARN). Pour plus d'informations sur l'utilisation ARNs des ressources Amazon RDS, consultezNoms des ressources Amazon (ARNs) dans Amazon RDS.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Sid":"AllowFullDevAccessNoTags",
         "Effect":"Allow",
         "NotAction":[
            "rds:AddTagsToResource",
            "rds:RemoveTagsFromResource"
         ],
         "Resource":"arn:aws:rds:*:123456789012:db:${aws:username}*",
         "Condition":{
            "StringEqualsIfExists":{
               "rds:db-tag/stage":"devo"
            }
         }
      }
   ]
}
```

Accorder l'autorisation de baliser les ressources Amazon RDS lors de leur création

Certaines opérations de l'API RDS vous permettent de spécifier des balises lorsque vous créez la ressource. Vous pouvez utiliser des balises de ressource pour implémenter le contrôle basé sur les attributs (ABAC). Pour plus d'informations, voir À quoi sert ABAC ? AWS et Contrôle de l'accès aux AWS ressources à l'aide de balises.

Pour permettre aux utilisateurs de baliser les ressources lors de leur création, ils doivent être autorisés à utiliser l'action qui crée la ressource, telle que `rds:CreateDBInstance`. Si des balises sont spécifiées dans l'action de création, RDS accorde une autorisation supplémentaire à l'`rds:AddTagsToResource`action afin de vérifier si les utilisateurs sont autorisés à créer des balises. Par conséquent, les utilisateurs doivent également avoir des autorisations explicites d'utiliser l'action `rds:AddTagsToResource`.

Dans la définition de la politique IAM pour l'`rds:AddTagsToResource`action, vous pouvez utiliser la clé de `aws:RequestTag` condition pour exiger des balises dans une demande de balise pour une ressource.

Par exemple, la politique suivante permet aux utilisateurs de créer des instances de base de données et d'appliquer des balises lors de la création d'instances de base de données, mais uniquement avec des clés de balise spécifiques (`environment`ou`project`) :

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "rds:CreateDBInstance"
           ],
           "Resource": "*"
       },
       {
           "Effect": "Allow",
           "Action": [
               "rds:AddTagsToResource"
           ],
           "Resource": "*",
           "Condition": {
               "StringEquals": {
                   "aws:RequestTag/environment": ["production", "development"],
                   "aws:RequestTag/project": ["dataanalytics", "webapp"]
               },
               "ForAllValues:StringEquals": {
                   "aws:TagKeys": ["environment", "project"]
               }
           }
       }
   ]
}
```

Cette politique refuse toute demande de création d'instance de base de données qui inclut des `environment` `project` balises autres que les balises ou, ou qui ne spécifie aucune de ces balises.

En outre, les utilisateurs doivent spécifier des valeurs pour les balises qui correspondent aux valeurs autorisées dans la politique.

La politique suivante permet aux utilisateurs de créer des clusters de base de données et d'appliquer toutes les balises lors de la création, à l'exception de la `environment=prod` balise :

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
       {
           "Effect": "Allow",
           "Action": [
               "rds:CreateDBCluster"
           ],
           "Resource": "*"
       },
       {
           "Effect": "Allow",
           "Action": [
               "rds:AddTagsToResource"
           ],
           "Resource": "*",
           "Condition": {
               "StringNotEquals": {
                   "aws:RequestTag/environment": "prod"
               }
           }
       }
   ]
}
```

Actions d'API RDS prises en charge pour le balisage lors de la création

Les actions d'API RDS suivantes prennent en charge le balisage lorsque vous créez une ressource. Pour ces actions, vous pouvez spécifier des balises lors de la création de la ressource :

- `CreateBlueGreenDeployment`
- `CreateCustomDBEngineVersion`

- CreateDBCluster
- CreateDBClusterEndpoint
- CreateDBClusterParameterGroup
- CreateDBClusterSnapshot
- CreateDBInstance
- CreateDBInstanceReadReplica
- CreateDBParameterGroup
- CreateDBProxy
- CreateDBProxyEndpoint
- CreateDBSecurityGroup
- CreateDBShardGroup
- CreateDBSnapshot
- CreateDBSubnetGroup
- CreateEventSubscription
- CreateGlobalCluster
- CreateIntegration
- CreateOptionGroup
- CreateTenantDatabase
- CopyDBClusterParameterGroup
- CopyDBClusterSnapshot
- CopyDBParameterGroup
- CopyDBSnapshot
- CopyOptionGroup
- RestoreDBClusterFromS3
- RestoreDBClusterFromSnapshot
- RestoreDBClusterToPointInTime
- RestoreDBInstanceFromDBSnapshot
- RestoreDBInstanceFromS3
- RestoreDBInstanceToPointInTime
- PurchaseReservedDBInstancesOffering

Si vous utilisez l'API AWS CLI or pour créer une ressource avec des balises, le `Tags` paramètre est utilisé pour appliquer des balises aux ressources lors de la création.

Pour ces actions d'API, si le balisage échoue, la ressource n'est pas créée et la demande échoue avec une erreur. Cela garantit que les ressources sont créées avec des balises ou qu'elles ne sont pas créées du tout, empêchant ainsi la création de ressources sans les balises prévues.

AWS politiques gérées pour Amazon RDS

Pour ajouter des autorisations aux ensembles d'autorisations et aux rôles, il est plus facile d'utiliser des politiques AWS gérées que de les rédiger vous-même. Il faut du temps et de l'expertise pour créer des politiques gérées par le client IAM qui ne fournissent à votre équipe que les autorisations dont elle a besoin. Pour démarrer rapidement, vous pouvez utiliser nos politiques AWS gérées. Ces politiques couvrent des cas d'utilisation courants et sont disponibles dans votre Compte AWS. Pour plus d'informations sur les politiques AWS gérées, voir les politiques AWS gérées dans le guide de l'utilisateur IAM.

Services AWS maintenir et mettre à jour les politiques AWS gérées. Vous ne pouvez pas modifier les autorisations dans les politiques AWS gérées. Les services ajoutent parfois des autorisations supplémentaires à une politique AWS gérée pour prendre en charge de nouvelles fonctionnalités. Ce type de mise à jour affecte toutes les identités (jeux d'autorisations et rôles) auxquelles la politique est attachée. Les services sont plus susceptibles de mettre à jour une politique AWS gérée lorsqu'une nouvelle fonctionnalité est lancée ou lorsque de nouvelles opérations sont disponibles. Les services ne suppriment pas les autorisations d'une politique AWS gérée, de sorte que les mises à jour des politiques n'enfreignent pas vos autorisations existantes.

En outre, AWS prend en charge les politiques gérées pour les fonctions professionnelles qui couvrent plusieurs services. Par exemple, la politique `ReadOnlyAccess` AWS gérée fournit un accès en lecture seule à toutes Services AWS les ressources. Lorsqu'un service lance une nouvelle fonctionnalité, il AWS ajoute des autorisations en lecture seule pour les nouvelles opérations et ressources. Pour obtenir la liste des politiques de fonctions professionnelles et leurs descriptions, consultez la page politiques gérées par AWS pour les fonctions de tâche dans le Guide de l'utilisateur IAM.

Rubriques

- AWS politique gérée : Amazon RDSRead OnlyAccess
- AWS politique gérée : Amazon RDSFull Access
- AWS politique gérée : Amazon RDSData FullAccess
- AWS politique gérée : Amazon RDSEnhanced MonitoringRole
- AWS politique gérée : Amazon RDSPerformance InsightsReadOnly
- AWS politique gérée : Amazon RDSPerformance InsightsFullAccess
- AWS politique gérée : Amazon RDSDirectory ServiceAccess
- AWS politique gérée : Amazon RDSService RolePolicy

- AWS politique gérée : Amazon RDSCustom ServiceRolePolicy
- AWS politique gérée : Amazon RDSCustom Instance ProfileRolePolicy
- AWS politique gérée : Amazon RDSPreview ServiceRolePolicy
- AWS politique gérée : Amazon RDSBeta ServiceRolePolicy

AWS politique gérée : Amazon RDSRead OnlyAccess

Cette politique autorise l'accès en lecture seule à Amazon RDS via le. AWS Management Console

Détails de l'autorisation

Cette politique inclut les autorisations suivantes :

- `rds` : permet aux principaux de décrire les ressources Amazon RDS et de dresser la liste des balises pour les ressources Amazon RDS.
- `cloudwatch`— Permet aux principaux d'obtenir les statistiques CloudWatch métriques d'Amazon.
- `ec2` : permet aux principaux de décrire les zones de disponibilité et les ressources de réseaux.
- `logs`— Permet aux directeurs de décrire les flux de CloudWatch journaux des groupes de journaux et d'obtenir les événements du journal CloudWatch des journaux.
- `devops-guru`— Permet aux responsables de décrire les ressources couvertes par Amazon DevOps Guru, qui sont spécifiées soit par des noms de CloudFormation pile, soit par des balises de ressources.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSRead OnlyAccess dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSFull Access

Cette politique fournit un accès complet à Amazon RDS via le AWS Management Console.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes :

- `rds` : donne aux principaux un accès complet à Amazon RDS.
- `application-autoscaling` : permet aux principaux de décrire et de gérer les cibles et les politiques de scalabilité automatique des applications.

- `cloudwatch`— Permet aux directeurs d'obtenir des statistiques CloudWatch métriques et de gérer les CloudWatch alarmes.
- `ec2` : permet aux principaux de décrire les zones de disponibilité et les ressources de réseaux.
- `logs`— Permet aux directeurs de décrire les flux de CloudWatch journaux des groupes de journaux et d'obtenir les événements du journal CloudWatch des journaux.
- `outposts`— Permet aux principaux d'obtenir des types d' AWS Outposts instances.
- `pi` : permet aux principaux d'obtenir les métriques de Performance Insights.
- `sns` : permet aux principaux de s'abonner à Amazon Simple Notification Service (Amazon SNS) et à ses rubriques, et de publier des messages Amazon SNS.
- `devops-guru`— Permet aux responsables de décrire les ressources couvertes par Amazon DevOps Guru, qui sont spécifiées soit par des noms de CloudFormation pile, soit par des balises de ressources.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSFull Access dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSData FullAccess

Cette politique permet un accès complet à l'utilisation de l'API de données et de l'éditeur de requêtes sur des Aurora Serverless clusters spécifiques Compte AWS. Cette politique permet d' Compte AWS obtenir la valeur d'un secret auprès de AWS Secrets Manager.

Vous pouvez associer la politique `AmazonRDSDataFullAccess` à vos identités IAM.

Détails de l’autorisation

Cette politique inclut les autorisations suivantes :

- `dbqms` : permet aux principaux d'accéder, de créer, de supprimer, de décrire et de mettre à jour des requêtes. Le service `dbqms` (Database Query Metadata Service, service de métadonnées de requête de base de données) est un service interne uniquement. Il fournit vos requêtes récentes et enregistrées pour l'éditeur de requêtes sur le AWS Management Console for multiple Services AWS, y compris Amazon RDS.
- `rds-data` : permet aux principaux d'exécuter des instructions SQL sur les bases de données Aurora Serverless.
- `secretsmanager`— Permet aux principaux d'obtenir la valeur d'un secret auprès de AWS Secrets Manager.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSData FullAccess dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSEnhanced MonitoringRole

Cette politique donne accès à Amazon CloudWatch Logs pour Amazon RDS Enhanced Monitoring.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes :

- `logs`— Permet aux responsables de créer des groupes de CloudWatch journaux et des politiques de conservation, ainsi que de créer et de décrire CloudWatch les flux de journaux des groupes de journaux. Il permet également aux directeurs de mettre et d'obtenir les événements du journal CloudWatch Logs.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSEnhanced MonitoringRole dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSPerformance InsightsReadOnly

Cette politique fournit un accès en lecture seule à l'analyse des performances d'Amazon RDS pour les instances Amazon de base de données RDS et les clusters de base de données Amazon Aurora.

Cette politique inclut désormais `Sid` (ID d'instruction) comme identifiant pour l'instruction de la politique.

Détails de l'autorisation

Cette politique inclut les autorisations suivantes :

- `rds` : permet aux principaux de décrire des instances de base de données Amazon RDS et des clusters de base de données Amazon Aurora.
- `pi` : permet aux principaux de faire des appels à l'API Analyse des performances d'Amazon RDS et d'accéder aux métriques de Performance Insights.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSPerformance InsightsReadOnly dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSPerformance InsightsFullAccess

Cette politique fournit un accès complet à l'analyse des performances d'Amazon RDS pour les instances de base de données Amazon RDS et les clusters de base de données Amazon Aurora.

Cette politique inclut désormais `Sid` (ID d'instruction) comme identifiant pour l'instruction de la politique.

Détails de l’autorisation

Cette politique inclut les autorisations suivantes :

- `rds` : permet aux principaux de décrire des instances de base de données Amazon RDS et des clusters de base de données Amazon Aurora.
- `pi` – Permet aux principaux d'appeler l'API Analyse des performances d'Amazon RDS et de créer, d'afficher et de supprimer des rapports d'analyse des performances.
- `cloudwatch`— Permet aux principaux de répertorier toutes les CloudWatch métriques Amazon et d'obtenir des données et des statistiques métriques.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSPerformance InsightsFullAccess dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSDirectory ServiceAccess

Cette politique permet à Amazon RDS d'effectuer des appels vers AWS Directory Service.

Détails de l’autorisation

Cette politique inclut l'autorisation suivante :

- `ds`— Permet aux principaux de décrire les AWS Directory Service répertoires et de contrôler les autorisations accordées aux AWS Directory Service annuaires.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSDirectory ServiceAccess dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSService RolePolicy

Vous ne pouvez pas attacher `AmazonRDSServiceRolePolicy` à vos entités IAM. Cette politique est attachée à un rôle lié à un service qui permet à Amazon RDS d'effectuer des actions en votre

nom. Pour de plus amples informations, veuillez consulter Autorisations des rôles liés à un service pour Amazon RDS.

AWS politique gérée : Amazon RDSCustom ServiceRolePolicy

Vous ne pouvez pas associer `AmazonRDSCustomServiceRolePolicy` à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à Amazon RDS d'appeler des AWS services au nom de vos ressources de base de données RDS.

Cette politique inclut les autorisations suivantes :

- `ec2`- Permet à RDS Custom d'effectuer des opérations de sauvegarde sur l'instance de base de données qui fournit des fonctionnalités de point-in-time restauration.
- `secretsmanager`- Permet à RDS Custom de gérer les secrets spécifiques aux instances de base de données créés par RDS Custom.
- `cloudwatch`- Permet à RDS Custom de télécharger les métriques et les journaux de l'instance de base de données CloudWatch via CloudWatch un agent.
- `events`, `sqs` - Permet à RDS Custom d'envoyer et de recevoir des informations d'état concernant l'instance de base de données.
- `cloudtrail`- Permet à RDS Custom de recevoir des événements de modification concernant l'instance de base de données
- `servicequotas`- Permet à RDS Custom de lire les quotas de service liés à l'instance de base de données
- `ssm`- Permet à RDS Custom de gérer l'instance sous-jacente EC2 de l'instance de base de données.
- `rds`- Permet à RDS Custom de gérer les ressources RDS pour votre instance de base de données
- `iam`- Permet à RDS Custom de valider et d'associer le profil d'instance à l'instance sous-jacente EC2 d'une instance de base de données.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSCustom ServiceRolePolicy dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSCustom Instance ProfileRolePolicy

Ne joignez pas `AmazonRDSCustomInstanceProfileRolePolicy` à vos entités IAM. Il ne doit être associé qu'à un rôle de profil d'instance utilisé pour accorder des autorisations à votre instance de base de données personnalisée Amazon RDS afin d'effectuer diverses actions d'automatisation

et tâches de gestion de base de données. Passez le profil d'instance en tant que `custom-iam-instance-profile` paramètre lors de la création de l'instance RDS Custom et RDS Custom associe ce profil d'instance à votre instance de base de données.

Détails de l’autorisation

Cette politique inclut les autorisations suivantes :

- `ssmssmmessages`, `ec2messages` - Permet à RDS Custom de communiquer, d'exécuter l'automatisation et de gérer les agents sur l'instance de base de données via Systems Manager.
- `ec2`, `s3` - Permet à RDS Custom d'effectuer des opérations de sauvegarde sur l'instance de base de données qui fournit des fonctionnalités de point-in-time restauration.
- `secretsmanager`- Permet à RDS Custom de gérer les secrets spécifiques aux instances de base de données créés par RDS Custom.
- `cloudwatch`, `logs` - Permet à RDS Custom de télécharger les métriques et les journaux de l'instance de base de données CloudWatch via CloudWatch un agent.
- `events`, `sqs` - Permet à RDS Custom d'envoyer et de recevoir des informations d'état concernant l'instance de base de données.
- `kms`- Permet à RDS Custom d'utiliser une clé KMS spécifique à l'instance pour chiffrer les secrets et les objets S3 gérés par RDS Custom.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSCustom Instance ProfileRolePolicy dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSPreview ServiceRolePolicy

Ne joignez pas `AmazonRDSPreviewServiceRolePolicy` à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à Amazon RDS d'appeler des AWS services au nom de vos ressources de base de données RDS. Pour de plus amples informations, veuillez consulter Rôle lié à un service pour Amazon RDS Preview.

Détails de l’autorisation

Cette politique inclut les autorisations suivantes :

- `ec2`- Permet aux principaux de décrire les zones de disponibilité et les ressources réseau.
- `secretsmanager`— Permet aux principaux d'obtenir la valeur d'un secret auprès de AWS Secrets Manager.

- `cloudwatch`, `logs` - Permet à Amazon RDS de télécharger les métriques et les journaux des instances de base de données CloudWatch via CloudWatch un agent.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSPreview ServiceRolePolicy dans le AWS Managed Policy Reference Guide.

AWS politique gérée : Amazon RDSBeta ServiceRolePolicy

Ne joignez pas `AmazonRDSBetaServiceRolePolicy` à vos entités IAM. Cette politique est associée à un rôle lié à un service qui permet à Amazon RDS d'appeler des AWS services au nom de vos ressources de base de données RDS. Pour de plus amples informations, veuillez consulter Autorisations de rôle liées à un service pour Amazon RDS Beta.

Détails de l’autorisation

Cette politique inclut les autorisations suivantes :

- `ec2`- Permet à Amazon RDS d'effectuer des opérations de sauvegarde sur l'instance de base de données qui fournit des fonctionnalités de point-in-time restauration.
- `secretsmanager`- Permet à Amazon RDS de gérer les secrets spécifiques aux instances de base de données créés par Amazon RDS.
- `cloudwatch`, `logs` - Permet à Amazon RDS de télécharger les métriques et les journaux des instances de base de données CloudWatch via CloudWatch un agent.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSBeta ServiceRolePolicy dans le AWS Managed Policy Reference Guide.

Amazon RDS met à jour les politiques AWS gérées

Consultez les informations relatives aux mises à jour des politiques AWS gérées pour Amazon RDS depuis que ce service a commencé à suivre ces modifications. Pour recevoir des alertes automatiques sur les modifications apportées à cette page, abonnez-vous au flux RSS de la page Document history (Historique des documents) d'Amazon RDS.

Modification	Description	Date
Autorisations du rôle lié à un service pour Amazon RDS Custom – Mise à jour de la politique existante	Amazon RDS a mis à jour les autorisations relatives `AmazonRDSCustomServiceRolePolicy` au rôle lié au `AWSServiceRoleForRDSCustom` service. La mise à jour `ec2:CopySnapshot` supprime une instruction et ajoute deux nouvelles instructions pour les autorisations de capture d'écran source et de destination. Ces mises à jour sont conformes à une modification du comportement CopySnapshot d'autorisation EBS tout en préservant les autorisations effectives inchangées. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	7 août 2025
Autorisations du rôle lié à un service pour Amazon RDS	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDS`	25 mars 2025

Modification	Description	Date
Custom – Mise à jour de la politique existante	`CustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces autorisations permettent à RDS Custom de gérer les EC2 paires de clés et à RDS Custom de s'intégrer à Amazon SQS. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	
AWS politique gérée : Amazon RDSCustom Instance ProfileRolePolicy – Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique gérée `AmazonRDSCustomInstanceProfileRolePolicy` afin de permettre l'utilisation de secrets gérés RDS Custom sur une instance personnalisée RDS. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSCustom Instance ProfileRolePolicy.	20 mars 2025

Modification	Description	Date
Autorisations du rôle lié à un service pour Amazon RDS Custom – Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom de répertorier et de restaurer les secrets de Secrets Manager. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	6 mars 2025
AWS politique gérée : Amazon RDSPreview ServiceRolePolicy – Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSPreviewServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDSPreview` service. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSPreview ServiceRolePolicy.	7 août 2024

Modification	Description	Date
AWS politique gérée : Amazon RDSBeta ServiceRolePolicy – Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSBetaServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDSBeta` service. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSBeta ServiceRolePolicy.	7 août 2024
Autorisations du rôle lié à un service pour Amazon RDS Custom – Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Les autorisations permettent à RDS Custom de communiquer avec les services Amazon RDS via un autre service Région AWS et de copier EC2 des images. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	18 juillet 2024

Modification	Description	Date
AWS politique gérée : Amazon RDSService RolePolicy – Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDS` service. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSService RolePolicy.	2 juillet 2024
Autorisations du rôle lié à un service pour Amazon RDS Custom – Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Cette nouvelle autorisation permet à RDS Custom d'associer un rôle de service en tant que profil d'instance à une instance RDS Custom. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	19 avril 2024

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Mise à jour de la politique existante	Amazon RDS a ajouté une nouvelle autorisation au rôle `AWSServiceRoleForRDSCustom` lié au service afin `AmazonRDSCustomServiceRolePolicy` de permettre à RDS Custom for SQL Server de modifier le type d'instance hôte de base de données sous-jacent. RDS a également ajouté l'`ec2:DescribeInstanceTypes` autorisation d'obtenir des informations sur le type d'instance pour l'hôte de base de données. Pour de plus amples informations, veuillez consulter AWS politiques gérées pour Amazon RDS.	8 avril 2024
AWS politiques gérées pour Amazon RDS : nouvelle politique	Amazon RDS a ajouté une nouvelle politique gérée nommée `AmazonRDSCustomInstanceProfileRolePolicy` pour permettre à RDS Custom d'effectuer des actions d'automatisation et des tâches de gestion de base de données via un profil d' EC2 instance. Pour de plus amples informations, veuillez consulter AWS politiques gérées pour Amazon RDS.	27 février 2024

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté une nouvelle déclaration IDs concernant le `AmazonRDSServiceRolePolicy` rôle lié au `AWSServiceRoleForRDS` service. Pour de plus amples informations, veuillez consulter Autorisations des rôles liés à un service pour Amazon RDS.	19 janvier 2024
AWS politiques gérées pour Amazon RDS – Mise à jour des politiques existantes	Les politiques gérées par `AmazonRDSPerformanceInsightsReadOnly` et `AmazonRDSPerformanceInsightsFullAccess` incluent désormais `Sid` (ID d'instruction) comme identifiant dans l'instruction de la politique. Pour plus d'informations, consultez AWS politique gérée : Amazon RDSPerformance InsightsReadOnly et AWS politique gérée : Amazon RDSPerformance InsightsFullAccess.	23 octobre 2023

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom for Oracle de créer, de modifier et de supprimer des règles EventBridge gérées. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	20 septembre 2023

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique gérée `AmazonRDSFullAccess`. Les autorisations vous permettent de générer, d'afficher et de supprimer le rapport d'analyse des performances pendant une période donnée. Pour plus d'informations sur la configuration de stratégies d'accès pour l'analyse des performances, consultez Configuration des politiques d'accès pour Performance Insights	17 août 2023

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Nouvelle politique et mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique gérée `AmazonRDSPerformanceInsightsReadOnly` et une nouvelle politique gérée nommée `AmazonRDSPerformanceInsightsFullAccess`. Ces autorisations vous permettent d'analyser les informations de performances pour une période donnée, de consulter les résultats d'analyse ainsi que les recommandations, et de supprimer les rapports. Pour plus d'informations sur la configuration de stratégies d'accès pour l'analyse des performances, consultez Configuration des politiques d'accès pour Performance Insights	16 août 2023

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom for Oracle d'utiliser des instantanés de base de données. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	23 juin 2023
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom for Oracle d'utiliser des instantanés de base de données. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	23 juin 2023

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom de créer des interfaces réseau. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	30 mai 2023
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. Ces nouvelles autorisations permettent à RDS Custom d'appeler Amazon EBS pour vérifier le quota de stockage. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	18 avril 2023

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS Custom a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service pour l'intégration avec Amazon SQS. RDS Custom nécessite une intégration à Amazon SQS pour créer et gérer des files d'attente SQS dans le compte client. Les noms des files d'attente SQS suivent le format `do-not-delete-rds-custom-[identifier]` et sont balisés avec `Amazon RDS Custom`. L'autorisation pour `ec2:CreateSnapshot` a également été ajoutée pour permettre à RDS Custom de créer des sauvegardes pour les volumes attachés à l'instance. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	6 avril 2023

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté un nouvel espace de CloudWatch noms Amazon `ListMetrics` à `AmazonRDSFullAccess` et. `AmazonRDSReadOnlyAccess` Cet espace de nom est nécessaire à Amazon RDS pour répertorier des métriques spécifiques sur l'utilisation des ressources. Pour plus d'informations, consultez la section Présentation de la gestion des autorisations d'accès à vos CloudWatch ressources dans le guide de CloudWatch l'utilisateur Amazon.	4 avril 2023

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté une nouvelle autorisation `AmazonRDSFullAccess` et des politiques `AmazonRDSReadOnlyAccess` gérées pour permettre l'affichage des résultats d'Amazon DevOps Guru dans la console RDS. Cette autorisation est requise pour permettre l'affichage des découvertes du DevOps Guru. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	30 mars 2023

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations au rôle `AWSServiceRoleForRDS` lié au service à `AmazonRDS ServiceRolePolicy` des fins d'intégration avec. AWS Secrets Manager RDS nécessite une intégration à Secrets Manager pour gérer les mots de passe des utilisateurs principaux dans Secrets Manager. Le secret utilise une convention de dénomination réservée et restreint les mises à jour des clients. Pour de plus amples informations, veuillez consulter Gestion des mots de passe avec Amazon RDS, et AWS Secrets Manager.	22 décembre 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique `AmazonRDSCustomServiceRolePolicy` du rôle `AWSServiceRoleForRDSCustom` lié au service. RDS Custom prend en charge les clusters de base de données. Ces nouvelles autorisations incluses dans la politique permettent à RDS Custom d'appeler au Services AWS nom de vos clusters de base de données. Pour de plus amples informations, veuillez consulter Autorisations du rôle lié à un service pour Amazon RDS Custom.	9 novembre 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations au rôle `AWSServiceRoleForRDS` lié au service pour l'intégration avec AWS Secrets Manager. L'intégration avec Secrets Manager est nécessaire pour que SQL Server Reporting Services (SSRS) Email fonctionne sur RDS. SSRS Email crée un secret au nom du client. Le secret utilise une convention de dénomination réservée et restreint les mises à jour des clients. Pour de plus amples informations, veuillez consulter Utilisation de SSRS Email pour envoyer des rapports.	26 août 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté un nouvel espace de CloudWatch noms Amazon à `AmazonRDS PreviewServiceRolePolicy` for. `PutMetricData` Cet espace de nom est nécessaire à Amazon RDS pour publier des métriques sur l'utilisation des ressources. Pour plus d'informations, consultez la section Utilisation de clés de condition pour limiter l'accès aux CloudWatch espaces de noms dans le guide de CloudWatch l'utilisateur Amazon.	7 juin 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté un nouvel espace de CloudWatch noms Amazon à `AmazonRDSBetaServiceRolePolicy` for. `PutMetricData` Cet espace de nom est nécessaire à Amazon RDS pour publier des métriques sur l'utilisation des ressources. Pour plus d'informations, consultez la section Utilisation de clés de condition pour limiter l'accès aux CloudWatch espaces de noms dans le guide de CloudWatch l'utilisateur Amazon.	7 juin 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté un nouvel espace de CloudWatch noms Amazon à `AWSServiceRoleForRDS` for. `PutMetricData` Cet espace de nom est nécessaire à Amazon RDS pour publier des métriques sur l'utilisation des ressources. Pour plus d'informations, consultez la section Utilisation de clés de condition pour limiter l'accès aux CloudWatch espaces de noms dans le guide de CloudWatch l'utilisateur Amazon.	22 avril 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouvelles autorisations au rôle `AWSServiceRoleForRDS` lié au service afin de gérer les autorisations pour les pools d'adresses IP appartenant au client et les tables de routage des passerelles locales (LGW-). RTBs Ces autorisations sont nécessaires pour que RDS on Outposts effectue une réplication Multi-AZ sur le réseau local des Outposts. Pour de plus amples informations, veuillez consulter Utilisation des déploiements multi-AZ pour Amazon RDS on AWS Outposts.	19 avril 2022

Modification	Description	Date
Politiques basées sur l'identité – Mise à jour d'une politique existante	Amazon RDS a ajouté une nouvelle autorisation à la politique `AmazonRDSFullAccess` gérée pour décrire les autorisations sur LGW-. RTBs Cette autorisation est nécessaire pour décrire les autorisations permettant à RDS on Outposts d'effectuer une réplication Multi-AZ sur le réseau local des Outposts. Pour de plus amples informations, veuillez consulter Utilisation des déploiements multi-AZ pour Amazon RDS on AWS Outposts.	19 avril 2022

Modification	Description	Date
AWS politiques gérées pour Amazon RDS – Nouvelle politique	Amazon RDS a ajouté une nouvelle politique gérée nommée `AmazonRDSPerformanceInsightsReadOnly` pour permettre à Amazon RDS d'appeler des AWS services pour le compte de vos instances de base de données. Pour plus d'informations sur la configuration de stratégies d'accès pour l'analyse des performances, consultez Configuration des politiques d'accès pour Performance Insights	10 mars 2022

Modification	Description	Date
Autorisations des rôles liés à un service pour Amazon RDS – Mise à jour d'une politique existante	Amazon RDS a ajouté de nouveaux CloudWatch espaces de noms Amazon à `AWSServiceRoleForRDS` for. `PutMetricData` Ces espaces de noms sont nécessaires pour qu'Amazon DocumentDB (compatible avec MongoDB) et Amazon Neptune puissent publier des métriques. CloudWatch Pour plus d'informations, consultez la section Utilisation de clés de condition pour limiter l'accès aux CloudWatch espaces de noms dans le guide de CloudWatch l'utilisateur Amazon.	4 mars 2022
Autorisations du rôle lié à un service pour Amazon RDS Custom : nouvelle politique	Amazon RDS a ajouté un nouveau rôle lié au service nommé `AWSServiceRoleForRDSCustom` pour permettre à RDS Custom d'appeler Services AWS au nom de vos instances de base de données.	26 octobre 2021
Amazon RDS a commencé à assurer le suivi des modifications	Amazon RDS a commencé à suivre les modifications apportées à ses politiques AWS gérées.	26 octobre 2021

Prévention des problèmes d'adjoint confus entre services

Le problème de l’adjoint confus est un problème de sécurité dans lequel une entité qui n’a pas l’autorisation d’effectuer une action peut contraindre une entité plus privilégiée à effectuer cette action. En AWS, l'usurpation d'identité interservices peut entraîner un problème de confusion chez les adjoints.

L’usurpation d’identité entre services peut se produire lorsqu’un service (le service appelant) appelle un autre service (le service appelé). Le service appelant peut être manipulé pour utiliser ses autorisations et agir sur les ressources d'un autre client, d'une manière dont il ne devrait pas avoir accès. Pour éviter cela, AWS fournit des outils qui peuvent vous aider à protéger vos données pour tous les services dont les responsables ont obtenu l'accès aux ressources de votre compte. Pour de plus amples informations, veuillez consulter Le problème du député confus dans le Guide de l'utilisateur IAM.

Afin de limiter les autorisations octroyées par Amazon RDS à un autre service pour une ressource spécifique, nous vous recommandons d'utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` dans les politiques de ressources.

Dans certains cas, la valeur `aws:SourceArn` ne contient pas l'ID du compte, par exemple lorsque vous utilisez l'Amazon Resource Name (ARN) pour un compartiment Amazon S3. Dans ces cas, veillez à utiliser les deux clés de contexte de condition globale pour limiter les autorisations. Dans certains cas, vous utilisez les deux clés de contexte de condition globale et la valeur `aws:SourceArn` contient l'ID du compte. Dans ces cas, assurez-vous que la valeur `aws:SourceAccount` et le compte dans le `aws:SourceArn` utilisent le même ID de compte lorsqu'ils sont utilisés dans la même instruction de politique. Utilisez `aws:SourceArn` si vous souhaitez qu'une seule ressource soit associée à l'accès entre services. Si vous souhaitez autoriser l'association d'une ressource du AWS compte spécifié à l'utilisation interservices, utilisez`aws:SourceAccount`.

Assurez-vous que la valeur de `aws:SourceArn` est un ARN d'un type de ressource Amazon RDS. Pour de plus amples informations, veuillez consulter Noms des ressources Amazon (ARNs) dans Amazon RDS.

Le moyen le plus efficace de se protéger contre le problème de député confus consiste à utiliser la clé de contexte de condition globale `aws:SourceArn` avec l’ARN complet de la ressource. Dans certains cas, vous ne connaissez pas l'ARN complet de la ressource ou vous spécifiez plusieurs ressources. Dans ces cas, utilisez la clé de contexte de condition globale

`aws:SourceArn` avec des caractères génériques (*) pour les parties inconnues de l'ARN. Par exemple : `arn:aws:rds:*:123456789012:*`.

L'exemple suivant montre comment utiliser les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount` pour dans Amazon RDS afin d'éviter le problème de l'adjoint confus.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "ConfusedDeputyPreventionExamplePolicy",
    "Effect": "Allow",
    "Principal": {
      "Service": "rds.amazonaws.com"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
      "ArnLike": {
        "aws:SourceArn": "arn:aws:rds:us-east-1:123456789012:db:mydbinstance"
      },
      "StringEquals": {
        "aws:SourceAccount": "123456789012"
      }
    }
  }
}
```

Pour obtenir d'autres exemples de politiques qui utilisent les clés de contexte de condition globale `aws:SourceArn` et `aws:SourceAccount`, veuillez consulter les sections suivantes :

- Octroi d'autorisations de publication de notifications dans une rubrique Amazon SNS
- Création manuelle d'un rôle IAM pour les sauvegarde et restauration natives
- Configuration de l'authentification Windows pour les instances de base de données SQL Server
- Prérequis pour l'intégration de RDS for SQL Server avec S3
- Création manuelle d'un rôle IAM pour SQL Server Audit
- Configuration des autorisations IAM pour l'intégration de RDS for Oracle à Amazon S3

- Configuration de l'accès à un compartiment Amazon S3 (importation PostgreSQL)
- Configuration de l'accès à un compartiment Amazon S3 (exportation PostgreSQL)

Authentification de base de données IAM pour MariaDB, MySQL et PostgreSQL

Vous pouvez vous authentifier auprès de votre d'instances de base de données à l'aide de l'authentification de base de données AWS Identity and Access Management (IAM). L'authentification de base de données IAM fonctionne avec MariaDB, MySQL et PostgreSQL. Grâce à cette méthode d'authentification, vous n'avez plus besoin de mot de passe pour vous connecter à une instance de base de données. En revanche, un jeton d'authentification est nécessaire.

Un jeton d'authentification est une chaîne de caractères unique générée par Amazon RDS sur demande. Les jetons d'authentification sont générés à l'aide de AWS la version 4 de Signature. Chaque jeton a une durée de vie de 15 minutes. Il n'est pas nécessaire de stocker les informations d'identification des utilisateurs dans la base de données, car l'authentification est gérée de manière externe avec IAM. Vous pouvez aussi toujours utiliser l'authentification de base de données standard. Le jeton est uniquement utilisé pour l'authentification et n'affecte pas la session une fois qu'il est établi.

L'authentification de base de données IAM offre les avantages suivants :

- Le trafic réseau à destination et en provenance de la base de données est chiffré à l'aide de Secure Socket Layer (SSL) ou de Transport Layer Security (TLS). Pour plus d'informations sur l'utilisation SSL/TLS d'Amazon RDS, consultez.
- Vous pouvez utiliser IAM pour gérer de façon centralisée l'accès à vos ressources de base de données, au lieu de gérer l'accès de manière individuelle sur chaque instance de bases de données.
- Pour les applications exécutées sur Amazon EC2, vous pouvez utiliser les informations d'identification de profil spécifiques à votre EC2 instance pour accéder à votre base de données au lieu d'un mot de passe, afin de renforcer la sécurité.

En règle générale, envisagez d'utiliser l'authentification de base de données IAM lorsque vos applications créent moins de 200 connexions par seconde, et que vous ne souhaitez pas gérer les noms d'utilisateur et les mots de passe directement dans le code de votre application.

Le pilote JDBC Amazon Web Services (AWS) prend en charge l'authentification de base de données IAM. Pour plus d'informations, consultez la section Plug-in d'authentification AWS IAM dans le référentiel de pilotes JDBC Amazon Web Services (AWS). GitHub

Le pilote Python Amazon Web Services (AWS) prend en charge l'authentification de base de données IAM. Pour plus d'informations, consultez la section Plug-in d'authentification AWS IAM dans le GitHubréférentiel de pilotes Python Amazon Web Services (AWS).

Parcourez les rubriques suivantes pour découvrir le processus de configuration de l'IAM pour l'authentification de base de données :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM
- Connexion à votre instance de base de données à l'aide de l'authentification IAM.

Disponibilité des régions et des versions

La disponibilité et le support des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données. Pour plus d'informations sur la disponibilité du moteur, de la version et de la région avec Amazon RDS et l'authentification de base de données IAM, consultez. Régions et moteurs de base de données pris en charge pour l'authentification de base de données IAM dans Amazon RDS

Support CLI et kit SDK

L'authentification de base de données IAM est disponible pour AWS CLIet pour les langues AWS SDKs spécifiques suivantes :

- AWS SDK pour .NET
- AWS SDK pour C++
- AWS SDK pour Go
- AWS SDK pour Java
- AWS SDK pour JavaScript
- AWS SDK pour PHP
- AWS SDK pour Python (Boto3)
- AWS SDK pour Ruby

Limites de l'authentification de base de données IAM

Les limitations suivantes s'appliquent lors de l'utilisation de l'authentification de base de données IAM :

- L'authentification de base de données IAM limite les connexions à 200 connexions par seconde.

  Les connexions qui utilisent le même jeton d'authentification ne sont pas limitées. Nous vous recommandons de réutiliser les jetons d'authentification dans la mesure du possible.
- Actuellement, l'authentification de base de données IAM ne prend pas en charge toutes les clés de contexte de condition globale.

  Pour plus d'informations sur les clés de contexte de condition globale, veuillez consulter Clés de contexte de condition globales AWS dans le Guide de l'utilisateur IAM.
- Pour PostgreSQL, si le rôle IAM (`rds_iam`) est ajouté à un utilisateur (y compris à l'utilisateur principal RDS), l'authentification IAM a priorité sur l'authentification par mot de passe, de sorte que l'utilisateur doit se connecter en tant qu'utilisateur IAM.
- Pour PostgreSQL, Amazon RDS ne prend pas en charge l'activation simultanée des méthodes d'authentification IAM et Kerberos.
- Pour PostgreSQL, vous ne pouvez pas utiliser l'authentification IAM pour établir une connexion de réplication.
- Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.
- CloudWatch et CloudTrail n'enregistrez pas l'authentification IAM. Ces services ne suivent pas les appels `generate-db-auth-token` d'API qui autorisent le rôle IAM à activer la connexion à la base de données.
- L'authentification IAM DB nécessite des ressources de calcul sur le d'instances de base de données. Pour garantir une connectivité fiable, votre base de données doit disposer de 300 à 1 000 Mo de mémoire supplémentaire. Pour connaître la mémoire nécessaire à votre charge de travail, comparez la colonne RES pour les processus RDS dans la liste des processus de surveillance améliorée avant et après l'activation de l'authentification IAM DB. Consultez Afficher les métriques du système d'exploitation dans la RDS console.

  Si vous utilisez une instance de classe burstable, évitez de manquer de mémoire en réduisant d'autant la mémoire utilisée par d'autres paramètres tels que les tampons et le cache.

- Pour RDS for MySQL , vous ne pouvez pas utiliser l'authentification par mot de passe pour un utilisateur de base de données que vous configurez avec l'authentification IAM.
- L'authentification IAM DB n'est prise en charge pour RDS on Outposts, quel que soit le moteur.

Recommandations pour l'authentification de base de données IAM

Nous recommandons les pratiques suivantes lors de l'utilisation de l'authentification de base de données IAM :

- Utilisez l'authentification de base de données IAM si votre application exige moins de 200 nouvelles connexions d'authentification de base de données IAM par seconde.

  Les moteurs de base de données qui fonctionnent avec Amazon RDS n'imposent pas de limites de tentatives d'authentification par seconde. Néanmoins, lorsque vous utilisez l'authentification de base de données IAM, votre application doit générer un jeton d'authentification. Votre application emploie ensuite ce jeton pour la connexion à l'instance de base de données. Si vous dépassez la limite maximale de nouvelles connexions par seconde, le traitement supplémentaire d'authentification de base de données IAM peut entraîner une limitation de la connexion.

  Envisagez d'utiliser le regroupement de connexions dans vos applications pour limiter la création constante de connexions. Cela peut réduire les frais généraux liés à l'authentification de base de données IAM et permettre à vos applications de réutiliser les connexions existantes. Vous pouvez également envisager d'utiliser le proxy RDS pour ces cas d'utilisation. Le proxy RDS entraîne des coûts supplémentaires. Consultez Tarification de Proxy Amazon RDS.
- La taille d'un jeton d'authentification de base de données IAM dépend de nombreux facteurs, notamment du nombre de balises IAM, des politiques de service IAM, de la longueur des ARN, ainsi que d'autres propriétés IAM et de base de données. La taille minimale de ce jeton est généralement d'environ 1 Ko, mais elle peut être plus grande. Étant donné que ce jeton est utilisé comme mot de passe dans la chaîne de connexion à la base de données à l'aide de l'authentification IAM, vous devez vous assurer que les outils de votre pilote de base de données (par exemple, ODBC) and/or ne limitent ni ne tronquent ce jeton en raison de sa taille. Un jeton tronqué provoquera l'échec de la validation d'authentification effectuée par la base de données et IAM.
- Si vous utilisez des informations d'identification temporaires lors de la création d'un jeton d'authentification d'une base de données IAM, les informations d'identification temporaires doivent toujours être valides lorsque vous utilisez le jeton d'authentification d'une base de données IAM pour effectuer une demande de connexion.

Clés contextuelles de condition AWS globale non prises en charge

L'authentification de base de données IAM ne prend pas en charge le sous-ensemble suivant de clés AWS contextuelles de conditions globales.

- `aws:Referer`
- `aws:SourceIp`
- `aws:SourceVpc`
- `aws:SourceVpce`
- `aws:UserAgent`
- `aws:VpcSourceIp`

Pour plus d'informations, consultez Clés de contexte de condition globales AWS dans le Guide de l'utilisateur IAM.

Activation et désactivation de l'authentification de base de données IAM

Par défaut, l'authentification de base de données IAM est désactivée sur les instances et de bases de données. Vous pouvez activer ou désactiver l'authentification de base de données IAM à l'aide de AWS Management Console AWS CLI, ou de l'API.

Vous pouvez activer l'authentification de base de données IAM lorsque vous effectuez une des actions suivantes :

- Pour créer une nouvelle instance de base de données avec l'authentification de base de données IAM activée, veuillez consulter Création d'une instance de base de données Amazon RDS.
- Pour modifier une instance de base de données afin d'activer l'authentification de base de données IAM, veuillez consulter Modification d'une RDS instance de base de données Amazon.
- Pour restaurer une instance de base de données à partir d'un instantané avec l'authentification de base de données IAM activée, veuillez consulter Restauration vers une instance de base de données.
- Pour restaurer une instance de base de données à un instant dans le passé avec l'authentification de base de données IAM activée, veuillez consulter Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

L'authentification IAM pour les des instances de base de données PostgreSQL nécessite que la valeur SSL soit égale à 1. Vous ne pouvez pas activer l'authentification IAM pour une instance de base de données PostgreSQL si la valeur SSL est égale à 0. Vous ne pouvez pas modifier et définir la valeur SSL sur 0 si l'authentification IAM est activée pour une instance de base de données PostgreSQL.

console

Chaque flux de travail de création ou de modification comporte une section Authentification de base de données dans laquelle vous pouvez activer ou désactiver l'authentification de base de données IAM. Dans cette section, choisissez Authentification de base de données par mot de passe et IAM pour activer l'authentification de base de données IAM.

Pour activer ou désactiver l'authentification de base de données IAM pour une instance de base de données

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez instance de base de données que vous souhaitez modifier.

   > **Note**
   >
   > Assurez-vous que l'instance de base de données est compatible avec l'authentification IAM. Consultez les exigences de compatibilité présentées dans Disponibilité des régions et des versions.

4. Sélectionnez Modify.
5. Dans la section Database authentication (Authentification de base de données), cliquez sur Password and IAM database authentication (Authentification par mot de passe et IAM) pour activer l'authentification de base de données IAM. Choisissez Authentification par mot de passe ou Authentification par mot de passe et Kerberos pour désactiver l'authentification IAM.
6. Vous pouvez également choisir d'activer la publication des journaux d'authentification IAM DB dans CloudWatch Logs. Sous Exportations de journaux, choisissez l'option de iam-db-auth-error journal. La publication de vos CloudWatch journaux dans Logs consomme de l'espace de stockage et vous devez payer des frais pour ce stockage. Assurez-vous de supprimer tous les CloudWatch journaux dont vous n'avez plus besoin.
7. Choisissez Continuer.

8. Pour appliquer immédiatement les modifications, choisissez Immédiatement dans la section Planification des modifications.
9. Choisissez Modifier l'instance de base de données ou .

AWS CLI

Pour créer une nouvelle instance de base de données avec authentification IAM à l'aide de AWS CLI, utilisez la `create-db-instance`commande. Spécifiez l'option `--enable-iam-database-authentication`, comme indiqué dans l'exemple suivant.

```
aws rds create-db-instance \
    --db-instance-identifier mydbinstance \
    --db-instance-class db.m3.medium \
    --engine MySQL \
    --allocated-storage 20 \
    --master-username masterawsuser \
    --manage-master-user-password \
    --enable-iam-database-authentication
```

Pour mettre à jour une instance de bases de données existante de manière à activer ou non l'authentification IAM, utilisez la commande de l' AWS CLI `modify-db-instance`. Spécifiez l'option `--enable-iam-database-authentication` ou `--no-enable-iam-database-authentication`, selon le cas.

> **Note**
>
> Assurez-vous que l'instance de base de données est compatible avec l'authentification IAM. Consultez les exigences de compatibilité présentées dans Disponibilité des régions et des versions.

Par défaut, Amazon RDS procède à la modification pendant la fenêtre de maintenance suivante. Si vous souhaitez ignorer ceci et activer l'authentification de bases de données IAM dès que possible, utilisez le paramètre `--apply-immediately`.

L'exemple suivant montre comment activer immédiatement l'authentification IAM pour une instance de base de données existante.

```
aws rds modify-db-instance \
```

```
    --db-instance-identifier mydbinstance \
    --apply-immediately \
    --enable-iam-database-authentication
```

Si vous restaurez un d'instances de base de données, utilisez l'une des AWS CLI commandes suivantes :

- `restore-db-instance-to-point-in-time`
- `restore-db-instance-from-db-snapshot`

Le paramètre d'authentification de base de données IAM par défaut est celui de l'instantané source. Pour le modifier, spécifiez l'option `--enable-iam-database-authentication` ou `--no-enable-iam-database-authentication`, selon le cas.

API RDS

Pour créer une nouvelle instance de base de données avec authentification IAM par l'intermédiaire de l'API, utilisez l'opération d'API `CreateDBInstance`. Définissez le paramètre `EnableIAMDatabaseAuthentication` sur `true`.

Pour mettre à jour une instance de base de données existante de manière à activer l'authentification IAM, utilisez l'opération d'API `ModifyDBInstance`. Définissez le paramètre `EnableIAMDatabaseAuthentication` sur `true` pour activer l'authentification IAM ou sur `false` pour la désactiver.

> **Note**
>
> Assurez-vous que l'instance de base de données est compatible avec l'authentification IAM. Consultez les exigences de compatibilité présentées dans Disponibilité des régions et des versions.

Si vous restaurez un ou une instance de base de données, utilisez l'une des opérations d'API suivantes :

- `RestoreDBInstanceFromDBSnapshot`
- `RestoreDBInstanceToPointInTime`

Le paramètre d'authentification de base de données IAM par défaut est celui de l'instantané source. Pour modifier ce paramètre, définissez le paramètre `EnableIAMDatabaseAuthentication` sur `true` pour activer l'authentification IAM ou sur `false` pour la désactiver.

Création et utilisation d'une politique IAM pour l'accès à une base de données IAM

Pour autoriser un utilisateur ou un rôle à se connecter à votre instance de bases de données, vous devez créer une politique IAM. Vous attachez ensuite la politique à un jeu d'autorisations ou à un rôle.

> **Note**
>
> Pour en savoir plus sur les stratégies IAM, consultez Gestion des identités et des accès pour Amazon RDS.

L'exemple de politique suivant autorise un utilisateur à se connecter à une instance de bases de données en utilisant l'authentification de base de données IAM.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds-db:connect"
            ],
            "Resource": [
                "arn:aws:rds-db:us-east-2:111122223333:dbuser:db-
ABCDEFGHIJKL01234/db_user"
            ]
        }
    ]
}
```

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds-db:connect"
            ],
            "Resource": [
                "arn:aws:rds-db:us-east-2:111122223333:dbuser:cluster-
ABCDEFGHIJKL01234/db_user"
            ]
        }
    ]
}
```

Important

Un utilisateur doté d'autorisations d'administrateur peut accéder aux instances de base de données sans autorisations explicites dans une politique IAM. Si vous souhaitez restreindre l'accès de l'administrateur aux instances et aux de base de données, vous pouvez créer un rôle IAM avec les autorisations appropriées accordant moins de privilèges, puis les assigner à l'administrateur.

Note

Ne confondez pas le préfixe `rds-db:` avec d'autres préfixes d'opération d'API RDS; qui commencent par `rds:`. Vous utilisez le préfixe `rds-db:` et l'action `rds-db:connect` uniquement pour l'authentification de base de données IAM. Ils ne sont valides que dans ce contexte.

L'exemple de politique inclut une instruction unique avec les éléments suivants :

- `Effect` – Spécifiez `Allow` pour octroyer l'accès à l'instance de base de données. Si vous n'autorisez pas explicitement l'accès, celui-ci est refusé par défaut.
- `Action` – Spécifiez `rds-db:connect` pour autoriser les connexions à l'instance de base de données.
- `Resource` – Spécifiez un ARN (Amazon Resource Name) qui décrit un compte de base de données dans une instance de base de données. Le format de l'ARN est le suivant.

```
arn:aws:rds-db:region:account-id:dbuser:DbiResourceId/db-user-name
```

Dans ce format, remplacez les variables suivantes :

- *`region`*est la AWS région du d'instances de base de données. Dans l'exemple de politique, la AWS région est`us-east-2`.
- *`account-id`*est le numéro de AWS compte du d'instances de base de données. Dans l'exemple de stratégie, le numéro de compte est `1234567890`. L'utilisateur doit figurer dans le même compte que le compte de l'instance de base de données.

  Pour bénéficier d'un accès intercompte, créez un rôle IAM avec la politique décrite ci-dessus dans le compte de l'instance de base de données et autorisez votre autre compte à endosser ce rôle.
- *`DbiResourceId`* correspond à l'identifiant de l'instance de base de données. Cet identifiant est propre à une AWS région et ne change jamais. Dans cet exemple de stratégie, l'identifiant est `db-ABCDEFGHIJKL01234`.

  Pour trouver un ID de ressource de d'instance de base de AWS Management Console données dans Amazon RDS , choisissez le d'instance de base de données pour en voir les détails. Choisissez ensuite l'onglet Configuration. L'ID de ressource est indiqué dans la section Configuration.

  Vous pouvez également utiliser la AWS CLI commande pour répertorier les identifiants et les ressources IDs de l'ensemble de votre d'instances de base de données dans la AWS région actuelle, comme indiqué ci-dessous.

```
aws rds describe-db-instances --query "DBInstances[*].
[DBInstanceIdentifier,DbiResourceId]"
```

Si vous utilisez Amazon Aurora, spécifiez `DbClusterResourceId` au lieu de `DbiResourceId`. Pour plus d'informations, consultez la section Creating and using an IAM policy for IAM database access (Création et utilisation d'une politique IAM pour l'accès à la base de données IAM) dans le Guide de l'utilisateur d'Amazon Aurora.

> **Note**
>
> Si vous vous connectez à une base de données via le proxy RDS, spécifiez l'ID de ressource de proxy, par exemple `prx-ABCDEFGHIJKL01234`. Pour plus d'informations sur l'utilisation de l'authentification de base de données IAM avec le proxy RDS, consultez Connexion à un proxy à l'aide de l'authentification IAM.

- *`db-user-name`* correspond au nom du compte de base de données à associer à l'authentification IAM. Dans cet exemple de stratégie, le compte de la base de données est `db_user`.

Vous pouvez en créer d'autres ARNs pour prendre en charge différents modèles d'accès. La stratégie suivante permet d'accéder à deux comptes de base de données différents dans une instance de base de données.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
             "rds-db:connect"
         ],
         "Resource": [
             "arn:aws:rds-db:us-east-2:123456789012:dbuser:db-ABCDEFGHIJKL01234/
jane_doe",
             "arn:aws:rds-db:us-east-2:123456789012:dbuser:db-ABCDEFGHIJKL01234/
mary_roe"
         ]
      }
```

```
   ]
}
```

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
             "rds-db:connect"
         ],
         "Resource": [
             "arn:aws:rds-db:us-east-2:123456789012:dbuser:cluster-
ABCDEFGHIJKL01234/jane_doe",
             "arn:aws:rds-db:us-east-2:123456789012:dbuser:cluster-
ABCDEFGHIJKL01234/mary_roe"
         ]
      }
   ]
}
```

La politique suivante utilise le caractère « * » pour faire correspondre tous les d'instances de base de données et les comptes de base de données pour un AWS compte et une AWS région spécifiques.

JSON

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds-db:connect"
            ],
            "Resource": [
```

```
                "arn:aws:rds-db:us-east-2:111122223333:dbuser:*/*"
            ]
        }
    ]
}
```

La politique suivante correspond à tous les d'instances de base de données pour un AWS compte et une AWS région particuliers. Néanmoins, cette stratégie n'octroie l'accès qu'aux instances et de bases de données qui ont un compte de base de données `jane_doe`.

JSON

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
             "rds-db:connect"
         ],
         "Resource": [
             "arn:aws:rds-db:us-east-2:123456789012:dbuser:*/jane_doe"
         ]
      }
   ]
}
```

L'utilisateur ou le rôle a uniquement accès aux bases de données auxquelles l'utilisateur de base de données a accès. Supposons par exemple que votre instance de base de donnés possède une base de données nommée dev et une autre base de données nommée test. Si l'utilisateur de base de données `jane_doe` a uniquement accès à dev, tous les rôles ou utilisateurs qui accèdent à cette instance de bases de données avec l'utilisateur `jane_doe` ont aussi uniquement accès à dev. Cette restriction d'accès s'applique également aux autres objets de bases de données, tels que les tables, les vues, etc.

Un administrateur doit créer des politiques IAM autorisant les entités à exécuter des opérations d'API spécifiques sur les ressources spécifiées dont ils ont besoin. L'administrateur doit ensuite attacher

ces politiques aux jeux d'autorisations ou aux rôles qui ont besoin de ces autorisations. Pour obtenir des exemples de stratégies, consultez la section Exemples de politiques basées sur l'identité pour Amazon RDS.

Attacher une politique IAM à un jeu d'autorisations ou à un rôle

Après avoir créé une politique IAM pour permettre l'authentification d'une base de données, il convient d'attacher la politique à un jeu d'autorisations ou à un rôle. Pour accéder à un didacticiel sur ce sujet, veuillez consulter Créer et attacher votre première politique gérée par le client dans le Guide de l'utilisateur IAM.

Tandis que vous parcourez ce didacticiel, vous pouvez utiliser un exemple de politique illustré dans cette section comme point de départ afin de le personnaliser en fonction de vos besoins. À la fin de ce didacticiel, vous obtenez un jeu d'autorisations avec une politique attachée qui peut utiliser l'action `rds-db:connect`.

> **Note**
>
> Vous pouvez mapper plusieurs jeux d'autorisations ou rôles au même compte d'utilisateur de base de données. Supposons par exemple que votre politique IAM a spécifié l'ARN de ressource suivant.
>
> ```
> arn:aws:rds-db:us-east-2:123456789012:dbuser:db-12ABC34DEFG5HIJ6KLMNOP78QR/
> jane_doe
> ```
>
> Si vous attachez la politique à Jane, Bob et Diego, chacun de ces utilisateurs peut se connecter à l'instance de bases de données en utilisant le compte de base de données `jane_doe`.

Création d'un compte de base de données à l'aide de l'authentification IAM

Avec l'authentification de base de données IAM, vous n'avez pas besoin d'associer de mots de passe de base de données aux comptes d'utilisateurs que vous créez. Si vous supprimez un utilisateur qui est mappé à un compte de base de données, vous devez également supprimer le compte de base de données avec l'instruction `DROP USER`.

> **Note**
>
> Le nom d'utilisateur utilisé pour l'authentification IAM doit correspondre à la casse du nom d'utilisateur dans la base de données.

Rubriques

- Utilisation de l'authentification IAM avec MariaDB et MySQL
- Utilisation de l'authentification IAM avec PostgreSQL

Utilisation de l'authentification IAM avec MariaDB et MySQL

Avec MariaDB et MySQL, l'authentification est `AWSAuthenticationPlugin` gérée AWS par un plugin fourni qui fonctionne parfaitement avec IAM pour authentifier vos utilisateurs. Connectez-vous à l'instance de base de données en tant qu'utilisateur principal ou autre utilisateur qui peut créer des utilisateurs et accorder des privilèges. Après vous être connecté, exécutez l'instruction `CREATE USER`, comme indiqué dans l'exemple suivant.

```
CREATE USER 'jane_doe' IDENTIFIED WITH AWSAuthenticationPlugin AS 'RDS';
```

La clause `IDENTIFIED WITH` permet à MariaDB et MySQL d'utiliser `AWSAuthenticationPlugin` pour authentifier le compte de base de données (`jane_doe`). La clause `AS 'RDS'` fait référence à la méthode d'authentification. Assurez-vous que le nom d'utilisateur de base de données spécifié est identique à une ressource dans la politique IAM pour l'accès à la base de données IAM. Pour plus d'informations, consultez Création et utilisation d'une politique IAM pour l'accès à une base de données IAM.

> **Note**
>
> Si le message suivant s'affiche, cela signifie que le plug-in AWS fourni n'est pas disponible pour le d'instances de base de données actuel.
> `ERROR 1524 (HY000): Plugin 'AWSAuthenticationPlugin' is not loaded`
> Pour remédier à cette erreur, vérifiez si vous utilisez une configuration prise en charge et si vous avez activé l'authentification de base de données IAM sur votre instance de base de données. Pour plus d'informations, veuillez consulter Disponibilité des régions et des versions et Activation et désactivation de l'authentification de base de données IAM.

Après avoir créé un compte à l'aide de `AWSAuthenticationPlugin`, vous pouvez le gérer de la même manière que les autres comptes de base de données. Vous pouvez par exemple modifier les privilèges de compte avec `GRANT` et `REVOKE`, ou changer divers attributs de compte avec l'instruction `ALTER USER`.

Le trafic réseau de base de données est chiffré à l'aide de SSL/TLS lors de l'utilisation d'IAM. Pour autoriser les connexions SSL, modifiez le compte d'utilisateur à l'aide de la commande suivante.

```
ALTER USER 'jane_doe'@'%' REQUIRE SSL;
```

Utilisation de l'authentification IAM avec PostgreSQL

Pour utiliser l'authentification IAM avec PostgreSQL, connectez-vous à l'instance de base de données en tant qu'utilisateur principal ou autre utilisateur qui peut créer des utilisateurs et accorder des privilèges. Après vous être connecté, créez des utilisateurs de base de données, puis accordez-leur le rôle `rds_iam`, comme indiqué dans l'exemple suivant.

```
CREATE USER db_userx;
GRANT rds_iam TO db_userx;
```

Assurez-vous que le nom d'utilisateur de base de données spécifié est identique à une ressource dans la politique IAM pour l'accès à la base de données IAM. Pour plus d'informations, consultez Création et utilisation d'une politique IAM pour l'accès à une base de données IAM. Vous devez accorder le `rds_iam` rôle pour utiliser l'authentification IAM. Vous pouvez également utiliser des adhésions imbriquées ou des attributions indirectes du rôle.

Connexion à votre instance de base de données à l'aide de l'authentification IAM.

Avec l'authentification de base de données IAM, vous utilisez un jeton d'identification lors de la connexion à votre instance de base de données. Un jeton d'authentification constitue une chaîne de caractères unique qui remplace un mot de passe. Après avoir été créé, un jeton d'authentification est valable pendant 15 minutes avant d'expirer. Si vous tentez de vous connecter alors que le jeton expiré, la demande de connexion est rejetée.

Chaque jeton d'authentification doit être accompagné d'une signature valide, en utilisant AWS Signature Version 4. (Pour plus d'informations, consultez le processus de signature de la version 4 de Signature dans le Références générales AWS.) Le AWS CLI et un AWS SDK, tel que le AWS

SDK pour Java or AWS SDK pour Python (Boto3), peuvent signer automatiquement chaque jeton que vous créez.

Vous pouvez utiliser un jeton d'authentification lorsque vous vous connectez à Amazon RDS depuis un autre AWS service, tel que AWS Lambda. L'utilisation d'un jeton vous évite d'avoir à placer un mot de passe dans votre code. Vous pouvez également utiliser un AWS SDK pour créer et signer par programmation un jeton d'authentification.

Après avoir obtenu un jeton d'authentification IAM signé, vous pouvez vous connecter à une instance de base de données Amazon RDS. Vous trouverez ci-dessous comment procéder à l'aide d'un outil de ligne de commande ou d'un AWS SDK, tel que le AWS SDK pour Java ou AWS SDK pour Python (Boto3).

Pour plus d'informations, consultez les billets de blog suivants :

- Utilisation de l'authentification IAM pour se connecter avec SQL Workbench/J à Aurora MySQL ou Amazon RDS for MySQL
- Utilisation de l'authentification IAM pour se connecter à PgAdmin Amazon Aurora PostgreSQL ou Amazon RDS for PostgreSQL

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

Rubriques

- Connexion à votre d'instances de base de données à l'aide de l'authentification IAM avec les pilotes AWS
- Connexion à votre d'instances de base de données à l'aide de l'authentification IAM à partir de la ligne de commande : AWS CLI et du client MySQL
- Connexion à votre instance de base de données à l'aide de l'authentification IAM à partir de la ligne de commande : AWS CLI et client psql

- Connexion à votre instance de base de données à l'aide de l'authentification IAM et de AWS SDK pour .NET
- Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Go
- Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Java
- Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Python (Boto3)

Connexion à votre d'instances de base de données à l'aide de l'authentification IAM avec les pilotes AWS

La AWS suite de pilotes a été conçue pour accélérer les temps de basculement et de basculement, ainsi que pour l'authentification avec AWS Secrets Manager, AWS Identity and Access Management (IAM) et l'identité fédérée. Les AWS pilotes s'appuient sur la surveillance de l'état de l'instance de base de données du de bases de données et sur la connaissance de la topologie de l'instance de pour déterminer le nouveau rédacteur. Cette approche réduit les temps de basculement et de basculement à un chiffre, contre des dizaines de secondes pour les pilotes open source.

Pour plus d'informations sur les AWS pilotes, consultez le pilote de langue correspondant à votre instance de base de données RDS pour MariaDB, RDS pour MySQL ou RDS pour PostgreSQL.

> **Note**
>
> Les seules fonctionnalités prises en charge par RDS pour MariaDB sont l'authentification AWS Secrets Manager avec AWS Identity and Access Management , (IAM) et l'identité fédérée.

Connexion à votre d'instances de base de données à l'aide de l'authentification IAM à partir de la ligne de commande : AWS CLI et du client MySQL

Vous pouvez vous connecter depuis la ligne de commande à un Amazon RDS à l'aide de l'outil de ligne de `mysql` commande AWS CLI et comme décrit ci-dessous.

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

> **Note**
>
> Pour plus d'informations sur la connexion à votre base de données à l'aide de SQL Workbench/J avec authentification IAM, lisez le billet de blog Utilisation de l'authentification IAM pour se connecter avec SQL Workbench/J à Aurora MySQL ou Amazon RDS for MySQL.

Rubriques

- Création d'un jeton d'authentification IAM
- Connexion à votre instance de base de données

Création d'un jeton d'authentification IAM

L'exemple suivant illustre comment obtenir un jeton d'identification signé à l'aide d AWS CLI.

```
aws rds generate-db-auth-token \
   --hostname rdsmysql.123456789012.us-west-2.rds.amazonaws.com \
   --port 3306 \
   --region us-west-2 \
   --username jane_doe
```

Dans cet exemple, les paramètres sont les suivants :

- `--hostname` – Le nom d'hôte de l'instance de base de données auquel vous souhaitez accéder.
- `--port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `--region`— La AWS région dans laquelle le d'instances de base de données est exécuté
- `--username` – Le compte de base de données auquel vous souhaitez accéder.

Les premiers caractères du jeton ressemblent à l'exemple suivant.

```
rdsmysql.123456789012.us-west-2.rds.amazonaws.com:3306/?
Action=connect&DBUser=jane_doe&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Expires=900...
```

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

Connexion à votre instance de base de données

Le format général de connexion est illustré ci-dessous.

```
mysql --host=hostName --port=portNumber --ssl-ca=full_path_to_ssl_certificate --enable-
cleartext-plugin --user=userName --password=authToken
```

Les paramètres sont les suivants :

- `--host` – Le nom d'hôte de l'instance de base de données auquel vous souhaitez accéder.
- `--port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `--ssl-ca` – Le chemin d'accès complet vers le fichier de certificat SSL contenant la clé publique.

  Pour de plus amples informations sur la prise en charge SSL/TLS pour MariaDB, veuillez consulter SSL/TLSsupport pour les instances de base de données MariaDB sur Amazon RDS.

  Pour de plus amples informations sur la prise en charge SSL/TLS pour MySQL, veuillez consulter SSL/TLSsupport pour les instances My SQL DB sur Amazon RDS.

  Pour télécharger un certificat SSL, consultez .
- `--enable-cleartext-plugin` – Une valeur qui spécifie que `AWSAuthenticationPlugin` doit être utilisé pour cette connexion.

  Si vous utilisez un client MariaDB, l'option `--enable-cleartext-plugin` n'est pas requise.
- `--user` – Le compte de base de données auquel vous souhaitez accéder.
- `--password` – Un jeton d'authentification IAM signé.

Le jeton d'authentification est composé de plusieurs centaines de caractères. Il peut être encombrant sur la ligne de commande. Pour contourner ce problème, vous pouvez enregistrer le jeton dans une variable d'environnement, puis utiliser cette variable pour la connexion. L'exemple suivant illustre une manière de contourner ce problème. Dans cet exemple, */sample_dir/* est le chemin d'accès complet au fichier de certificat SSL contenant la clé publique.

```
RDSHOST="mysqldb.123456789012.us-east-1.rds.amazonaws.com"
TOKEN="$(aws rds generate-db-auth-token --hostname $RDSHOST --port 3306 --region us-
west-2 --username jane_doe )"

mysql --host=$RDSHOST --port=3306 --ssl-ca=/sample_dir/global-bundle.pem --enable-
cleartext-plugin --user=jane_doe --password=$TOKEN
```

Lorsque vous vous connectez avec `AWSAuthenticationPlugin`, la connexion est sécurisée par SSL. Pour le vérifier, tapez la commande suivante à l'invite de commande `mysql>`.

```
show status like 'Ssl%';
```

Les lignes suivantes de l'affichage obtenu fournissent plus de détails.

```
+---------------+-------------+
| Variable_name | Value

                                                                    |
+---------------+-------------+
| ...           | ...
| Ssl_cipher    | AES256-SHA

                                                                    |
| ...           | ...
| Ssl_version   | TLSv1.1

                                                                    |
| ...           | ...
+-----------------------------+
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à votre instance de base de données à l'aide de l'authentification IAM à partir de la ligne de commande : AWS CLI et client psql

À partir de la ligne de commande, vous pouvez vous connecter à une instance de base de données Amazon RDS for PostgreSQL avec AWS CLIIl'outil de ligne de commande psql comme décrit ci-après.

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

> **Note**
>
> Pour plus d'informations sur la connexion à votre base de données à l'aide de pgAdmin avec authentification IAM, consultez le billet de blog Utilisation de l'authentification IAM pour se connecter à PgAdmin Amazon Aurora PostgreSQL ou Amazon RDS for PostgreSQL

Rubriques

- Création d'un jeton d'authentification IAM
- Connexion à une instance PostgreSQL Amazon RDS

Création d'un jeton d'authentification IAM

Le jeton d'authentification se compose de plusieurs centaines de caractères ; il peut donc être complexe à manipuler sur la ligne de commande. Pour contourner ce problème, vous pouvez enregistrer le jeton dans une variable d'environnement, puis utiliser cette variable pour la connexion. L'exemple de code suivant montre comment utiliser l'AWS CLI pour obtenir un jeton d'authentification signé à l'aide de la commande `generate-db-auth-token` et le stocker dans une variable d'environnement `PGPASSWORD`.

```
export RDSHOST="rdspostgres.123456789012.us-west-2.rds.amazonaws.com"
```

```
export PGPASSWORD="$(aws rds generate-db-auth-token --hostname $RDSHOST --port 5432 --
region us-west-2 --username jane_doe )"
```

Dans cet exemple, les paramètres de la commande `generate-db-auth-token` sont les suivants :

- `--hostname` – Nom d'hôte de l'instance de base de données auquel vous souhaitez accéder.
- `--port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `--region` – La région AWS où l'instance de base de données s'exécute.
- `--username` – Le compte de base de données auquel vous souhaitez accéder.

Les premiers caractères du jeton généré ressemblent à l'exemple suivant.

```
rdspostgres.123456789012.us-west-2.rds.amazonaws.com:5432/?
Action=connect&DBUser=jane_doe&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Expires=900...
```

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

Connexion à une instance PostgreSQL Amazon RDS

Le format général pour utiliser psql pour la connexion est illustré ci-dessous.

```
psql "host=hostName port=portNumber sslmode=verify-full
 sslrootcert=full_path_to_ssl_certificate dbname=DBName user=userName
 password=authToken"
```

Les paramètres sont les suivants :

- `host` – Nom d'hôte de l'instance de base de données auquel vous souhaitez accéder.
- `port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `sslmode` – Le mode SSL à utiliser.

Lorsque vous utilisez `sslmode=verify-full`, la connexion SSL vérifie le point de terminaison de l'instance de base de données par rapport au point de terminaison dans le certificat SSL.

- `sslrootcert` – Le chemin d'accès complet vers le fichier de certificat SSL contenant la clé publique.

  Pour de plus amples informations, veuillez consulter Utilisation de SSL avec une instance de base de données PostgreSQL.

  Pour télécharger un certificat SSL, consultez .

- `dbname` – La base de données à laquelle vous souhaitez accéder.
- `user` – Le compte de base de données auquel vous souhaitez accéder.
- `password` – Un jeton d'authentification IAM signé.

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

L'exemple suivant montre l'utilisation de psql pour se connecter. Dans cet exemple, psql utilise la variable d'environnement `RDSHOST` pour l'hôte et la variable d'environnement `PGPASSWORD` pour le jeton généré. Par ailleurs, *`/sample_dir/`* est le chemin d'accès complet au fichier de certificat SSL contenant la clé publique.

```
export RDSHOST="rdspostgres.123456789012.us-west-2.rds.amazonaws.com"
export PGPASSWORD="$(aws rds generate-db-auth-token --hostname $RDSHOST --port 5432 --
region us-west-2 --username jane_doe )"

psql "host=$RDSHOST port=5432 sslmode=verify-full sslrootcert=/sample_dir/global-
bundle.pem dbname=DBName user=jane_doe password=$PGPASSWORD"
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à votre instance de base de données à l'aide de l'authentification IAM et de AWS SDK pour .NET

Vous pouvez vous connecter à une instance de base de données RDS pour MariaDB, MySQL ou PostgreSQL avec l'AWS SDK pour .NET, comme décrit ci-après.

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

Exemples

Les exemples de code suivants montre comment générer un jeton d'authentification, puis comment l'utiliser pour se connecter à une instance de base de données.

Pour exécuter cet exemple de code, vous avez besoin de AWS SDK pour .NET, disponible sur le site AWS. Les paquets `AWSSDK.CORE` et `AWSSDK.RDS` sont requis. Pour vous connecter à un(e) instance de base de données, utilisez le connecteur de base de données .NET pour le moteur de base de données, tel que MySQLConnector pour MariaDB ou MySQL, ou Npgsql pour PostgreSQL.

Ce code se connecte à une instance de base de données MariaDB ou MySQL. Modifiez la valeur des variables suivantes selon les besoins :

- `server` – Le point de terminaison de l'instance de base de données à laquelle vous souhaitez accéder.
- `user` – Le compte de base de données auquel vous souhaitez accéder.
- `database` – La base de données à laquelle vous souhaitez accéder.
- `port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `SslMode` – Le mode SSL à utiliser.

  Lorsque vous utilisez `SslMode=Required`, la connexion SSL vérifie le point de terminaison de l'instance de base de données par rapport au point de terminaison dans le certificat SSL.
- `SslCa` – Le chemin d'accès complet au certificat SSL pour Amazon RDS

Pour télécharger un certificat, consultez .

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

```
using System;
using System.Data;
using MySql.Data;
using MySql.Data.MySqlClient;
using Amazon;

namespace ubuntu
{
  class Program
  {
    static void Main(string[] args)
    {
      var pwd =
 Amazon.RDS.Util.RDSAuthTokenGenerator.GenerateAuthToken(RegionEndpoint.USEast1,
 "mysqldb.123456789012.us-east-1.rds.amazonaws.com", 3306, "jane_doe");
      // for debug only Console.Write("{0}\n", pwd);  //this verifies the token is
 generated

      MySqlConnection conn = new MySqlConnection($"server=mysqldb.123456789012.us-
east-1.rds.amazonaws.com;user=jane_doe;database=mydB;port=3306;password={pwd};SslMode=Required;
      conn.Open();

      // Define a query
      MySqlCommand sampleCommand = new MySqlCommand("SHOW DATABASES;", conn);

      // Execute a query
      MySqlDataReader mysqlDataRdr = sampleCommand.ExecuteReader();

      // Read all rows and output the first column in each row
      while (mysqlDataRdr.Read())
        Console.WriteLine(mysqlDataRdr[0]);
```

```
      mysqlDataRdr.Close();
      // Close connection
      conn.Close();
    }
  }
}
```

Ce code se connecte à une instance de base de données PostgreSQL.

Modifiez la valeur des variables suivantes selon les besoins :

- `Server` – Le point de terminaison de l'instance de base de données à laquelle vous souhaitez accéder.
- `User ID` – Le compte de base de données auquel vous souhaitez accéder.
- `Database` – La base de données à laquelle vous souhaitez accéder.
- `Port` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `SSL Mode` – Le mode SSL à utiliser.

  Lorsque vous utilisez `SSL Mode=Required`, la connexion SSL vérifie le point de terminaison de l'instance de base de données par rapport au point de terminaison dans le certificat SSL.
- `Root Certificate` – Le chemin d'accès complet au certificat SSL pour Amazon RDS

  Pour télécharger un certificat, consultez .

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

```
using System;
using Npgsql;
using Amazon.RDS.Util;

namespace ConsoleApp1
{
```

```
    class Program
    {
        static void Main(string[] args)
        {
            var pwd =
 RDSAuthTokenGenerator.GenerateAuthToken("postgresmydb.123456789012.us-
east-1.rds.amazonaws.com", 5432, "jane_doe");
// for debug only Console.Write("{0}\n", pwd);  //this verifies the token is generated

            NpgsqlConnection conn = new
 NpgsqlConnection($"Server=postgresmydb.123456789012.us-east-1.rds.amazonaws.com;User
 Id=jane_doe;Password={pwd};Database=mydb;SSL Mode=Require;Root
 Certificate=full_path_to_ssl_certificate");
            conn.Open();

            // Define a query
                   NpgsqlCommand cmd = new NpgsqlCommand("select count(*) FROM
 pg_user", conn);

            // Execute a query
            NpgsqlDataReader dr = cmd.ExecuteReader();

            // Read all rows and output the first column in each row
            while (dr.Read())
                Console.Write("{0}\n", dr[0]);

            // Close connection
            conn.Close();
        }
    }
}
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Go

Vous pouvez vous connecter à une instance de base de données RDS pour MariaDB, MySQL ou PostgreSQL Aurora MySQL ou à un cluster de base de données Aurora ci-dessous. AWS SDK pour Go

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

Exemples

Pour exécuter ces exemples de code, vous avez besoin du AWS SDK pour Go, disponible sur le AWS site.

Modifiez la valeur des variables suivantes selon les besoins :

- `dbName` – La base de données à laquelle vous souhaitez accéder.
- `dbUser` – Le compte de base de données auquel vous souhaitez accéder.
- `dbHost` – Le point de terminaison de l'instance de base de données à laquelle vous souhaitez accéder.

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

- `dbPort` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `region`— La AWS région dans laquelle le d'instances de base de données est exécuté

En outre, assurez-vous que les bibliothèques importées dans l'exemple de code existent sur votre système.

> **Important**
>
> Les exemples de cette section utilisent le code suivant pour fournir des informations d'identification qui accèdent à une base de données à partir d'un environnement local : `creds := credentials.NewEnvCredentials()`

Si vous accédez à une base de données depuis un AWS service tel qu'Amazon EC2 ou Amazon ECS, vous pouvez remplacer le code par le code suivant :

```
sess := session.Must(session.NewSession())
creds := sess.Config.Credentials
```

Si vous effectuez cette modification, assurez-vous d'ajouter l'importation suivante :

```
"github.com/aws/aws-sdk-go/aws/session"
```

Rubriques

- Connexion à l'aide de l'authentification IAM et de la V2 AWS SDK pour Go
- Connexion à l'aide de l'authentification IAM et de la AWS SDK pour Go V1.

Connexion à l'aide de l'authentification IAM et de la V2 AWS SDK pour Go

Vous pouvez vous connecter à un d'instances de base de données à l'aide de l'authentification IAM et de la AWS SDK pour Go V2.

Les exemples de code suivants montre comment générer un jeton d'authentification, puis comment l'utiliser pour se connecter à une instance de base de données.

Ce code se connecte à une instance de base de données MariaDB ou MySQL.

```
package main

import (
     "context"
     "database/sql"
     "fmt"

     "github.com/aws/aws-sdk-go-v2/config"
     "github.com/aws/aws-sdk-go-v2/feature/rds/auth"
     _ "github.com/go-sql-driver/mysql"
)

func main() {

     var dbName string = "DatabaseName"
     var dbUser string = "DatabaseUser"
     var dbHost string = "mysqldb.123456789012.us-east-1.rds.amazonaws.com"
     var dbPort int = 3306
```

```
     var dbEndpoint string = fmt.Sprintf("%s:%d", dbHost, dbPort)
     var region string = "us-east-1"

    cfg, err := config.LoadDefaultConfig(context.TODO())
    if err != nil {
     panic("configuration error: " + err.Error())
    }

    authenticationToken, err := auth.BuildAuthToken(
     context.TODO(), dbEndpoint, region, dbUser, cfg.Credentials)
    if err != nil {
     panic("failed to create authentication token: " + err.Error())
    }

    dsn := fmt.Sprintf("%s:%s@tcp(%s)/%s?tls=true&allowCleartextPasswords=true",
        dbUser, authenticationToken, dbEndpoint, dbName,
    )

    db, err := sql.Open("mysql", dsn)
    if err != nil {
        panic(err)
    }

    err = db.Ping()
    if err != nil {
        panic(err)
    }
}
```

Ce code se connecte à une instance de base de données PostgreSQL.

```
package main

import (
     "context"
     "database/sql"
     "fmt"

     "github.com/aws/aws-sdk-go-v2/config"
     "github.com/aws/aws-sdk-go-v2/feature/rds/auth"
     _ "github.com/lib/pq"
)
```

```
func main() {

     var dbName string = "DatabaseName"
     var dbUser string = "DatabaseUser"
     var dbHost string = "postgresmydb.123456789012.us-east-1.rds.amazonaws.com"
     var dbPort int = 5432
     var dbEndpoint string = fmt.Sprintf("%s:%d", dbHost, dbPort)
     var region string = "us-east-1"

    cfg, err := config.LoadDefaultConfig(context.TODO())
    if err != nil {
     panic("configuration error: " + err.Error())
    }

    authenticationToken, err := auth.BuildAuthToken(
     context.TODO(), dbEndpoint, region, dbUser, cfg.Credentials)
    if err != nil {
     panic("failed to create authentication token: " + err.Error())
    }

    dsn := fmt.Sprintf("host=%s port=%d user=%s password=%s dbname=%s",
        dbHost, dbPort, dbUser, authenticationToken, dbName,
    )

    db, err := sql.Open("postgres", dsn)
    if err != nil {
        panic(err)
    }

    err = db.Ping()
    if err != nil {
        panic(err)
    }
}
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à l'aide de l'authentification IAM et de la AWS SDK pour Go V1.

Vous pouvez vous connecter à un d'instances de base de données à l'aide de l'authentification IAM et du V1 AWS SDK pour Go

Les exemples de code suivants montre comment générer un jeton d'authentification, puis comment l'utiliser pour se connecter à une instance de base de données.

Ce code se connecte à une instance de base de données MariaDB ou MySQL.

```
package main

import (
    "database/sql"
    "fmt"
    "log"

    "github.com/aws/aws-sdk-go/aws/credentials"
    "github.com/aws/aws-sdk-go/service/rds/rdsutils"
    _ "github.com/go-sql-driver/mysql"
)

func main() {
    dbName := "app"
    dbUser := "jane_doe"
    dbHost := "mysqldb.123456789012.us-east-1.rds.amazonaws.com"
    dbPort := 3306
    dbEndpoint := fmt.Sprintf("%s:%d", dbHost, dbPort)
    region := "us-east-1"

    creds := credentials.NewEnvCredentials()
    authToken, err := rdsutils.BuildAuthToken(dbEndpoint, region, dbUser, creds)
    if err != nil {
        panic(err)
    }

    dsn := fmt.Sprintf("%s:%s@tcp(%s)/%s?tls=true&allowCleartextPasswords=true",
        dbUser, authToken, dbEndpoint, dbName,
    )

    db, err := sql.Open("mysql", dsn)
    if err != nil {
        panic(err)
    }

    err = db.Ping()
    if err != nil {
        panic(err)
    }
```

```
}
```

Ce code se connecte à une instance de base de données PostgreSQL.

```
package main

import (
 "database/sql"
 "fmt"

 "github.com/aws/aws-sdk-go/aws/credentials"
 "github.com/aws/aws-sdk-go/service/rds/rdsutils"
 _ "github.com/lib/pq"
)

func main() {
    dbName := "app"
    dbUser := "jane_doe"
    dbHost := "postgresmydb.123456789012.us-east-1.rds.amazonaws.com"
    dbPort := 5432
    dbEndpoint := fmt.Sprintf("%s:%d", dbHost, dbPort)
    region := "us-east-1"

    creds := credentials.NewEnvCredentials()
    authToken, err := rdsutils.BuildAuthToken(dbEndpoint, region, dbUser, creds)
    if err != nil {
        panic(err)
    }

    dsn := fmt.Sprintf("host=%s port=%d user=%s password=%s dbname=%s",
        dbHost, dbPort, dbUser, authToken, dbName,
    )

    db, err := sql.Open("postgres", dsn)
    if err != nil {
        panic(err)
    }

    err = db.Ping()
    if err != nil {
        panic(err)
    }
}
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Java

Vous pouvez vous connecter à une instance de base de données RDS pour MariaDB, MySQL ou PostgreSQL Aurora MySQL ou à un cluster de base de données Aurora ci-dessous. AWS SDK pour Java

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM
- Configuration du AWS SDK pour Java

Pour des exemples d'utilisation du SDK pour Java 2.x, consultez les exemples Amazon RDS utilisant le SDK pour Java 2.x. Vous pouvez également utiliser l' AWS Advanced JDBC Wrapper, voir la documentation AWS Advanced JDBC Wrapper.

Rubriques

- Création d'un jeton d'authentification IAM
- Construction manuelle d'un jeton d'authentification IAM
- Connexion à votre instance de base de données

Création d'un jeton d'authentification IAM

Si vous écrivez des programmes à l'aide de AWS SDK pour Java, vous pouvez obtenir un jeton d'authentification signé à l'aide de la `RdsIamAuthTokenGenerator` classe. L'utilisation de cette classe nécessite que vous fournissiez des AWS informations d'identification. Pour ce faire, vous devez créer une instance de la `DefaultAWSCredentialsProviderChain` classe. `DefaultAWSCredentialsProviderChain`utilise la première clé AWS d'accès et la première clé secrète qu'il trouve dans la chaîne de fournisseurs d'informations d'identification par défaut. Pour plus

d'informations sur les clés d'accès AWS , consultez Gestion des clés d'accès pour les utilisateurs IAM.

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

Après avoir créé une instance de `RdsIamAuthTokenGenerator`, vous pouvez appeler la méthode `getAuthToken` pour obtenir un jeton signé. Fournissez la région AWS , le nom d'hôte, le numéro de port et le nom d'utilisateur. L'exemple de code suivant montre comment procéder.

```
package com.amazonaws.codesamples;

import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.services.rds.auth.GetIamAuthTokenRequest;
import com.amazonaws.services.rds.auth.RdsIamAuthTokenGenerator;

public class GenerateRDSAuthToken {

    public static void main(String[] args) {

     String region = "us-west-2";
     String hostname = "rdsmysql.123456789012.us-west-2.rds.amazonaws.com";
     String port = "3306";
     String username = "jane_doe";

     System.out.println(generateAuthToken(region, hostname, port, username));
    }

    static String generateAuthToken(String region, String hostName, String port, String
 username) {

     RdsIamAuthTokenGenerator generator = RdsIamAuthTokenGenerator.builder()
      .credentials(new DefaultAWSCredentialsProviderChain())
      .region(region)
      .build();

     String authToken = generator.getAuthToken(
      GetIamAuthTokenRequest.builder()
```

```
      .hostname(hostName)
      .port(Integer.parseInt(port))
      .userName(username)
      .build());

     return authToken;
    }

}
```

Construction manuelle d'un jeton d'authentification IAM

Dans Java, la manière la plus facile de créer un jeton d'authentification est d'utiliser `RdsIamAuthTokenGenerator`. Cette classe crée un jeton d'authentification pour vous, puis le signe à l'aide de AWS la version de signature 4. Pour de plus amples informations, veuillez consulter Processus de signature Signature Version 4 dans le Références générales AWS.

Vous pouvez néanmoins aussi construire et signer un jeton d'authentification manuellement, comme indiqué dans l'exemple de code suivant.

```
package com.amazonaws.codesamples;

import com.amazonaws.SdkClientException;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.auth.SigningAlgorithm;
import com.amazonaws.util.BinaryUtils;
import org.apache.commons.lang3.StringUtils;

import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import java.nio.charset.Charset;
import java.security.MessageDigest;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.SortedMap;
import java.util.TreeMap;

import static com.amazonaws.auth.internal.SignerConstants.AWS4_TERMINATOR;
import static com.amazonaws.util.StringUtils.UTF8;

public class CreateRDSAuthTokenManually {
    public static String httpMethod = "GET";
    public static String action = "connect";
```

```
    public static String canonicalURIParameter = "/";
    public static SortedMap<String, String> canonicalQueryParameters = new TreeMap();
    public static String payload = StringUtils.EMPTY;
    public static String signedHeader = "host";
    public static String algorithm = "AWS4-HMAC-SHA256";
    public static String serviceName = "rds-db";
    public static String requestWithoutSignature;

    public static void main(String[] args) throws Exception {

        String region = "us-west-2";
        String instanceName = "rdsmysql.123456789012.us-west-2.rds.amazonaws.com";
        String port = "3306";
        String username = "jane_doe";

        Date now = new Date();
        String date = new SimpleDateFormat("yyyyMMdd").format(now);
        String dateTimeStamp = new
SimpleDateFormat("yyyyMMdd'T'HHmmss'Z'").format(now);
        DefaultAWSCredentialsProviderChain creds = new
DefaultAWSCredentialsProviderChain();
    String awsAccessKey = creds.getCredentials().getAWSAccessKeyId();
    String awsSecretKey = creds.getCredentials().getAWSSecretKey();
        String expiryMinutes = "900";

        System.out.println("Step 1:  Create a canonical request:");
        String canonicalString = createCanonicalString(username, awsAccessKey, date,
dateTimeStamp, region, expiryMinutes, instanceName, port);
        System.out.println(canonicalString);
        System.out.println();

        System.out.println("Step 2:  Create a string to sign:");
        String stringToSign = createStringToSign(dateTimeStamp, canonicalString,
awsAccessKey, date, region);
        System.out.println(stringToSign);
        System.out.println();

        System.out.println("Step 3:  Calculate the signature:");
        String signature = BinaryUtils.toHex(calculateSignature(stringToSign,
newSigningKey(awsSecretKey, date, region, serviceName)));
        System.out.println(signature);
        System.out.println();
```

```
        System.out.println("Step 4:  Add the signing info to the request");

        System.out.println(appendSignature(signature));
        System.out.println();

    }

    //Step 1: Create a canonical request date should be in format YYYYMMDD and dateTime
should be in format YYYYMMDDTHHMMSSZ
    public static String createCanonicalString(String user, String accessKey, String
date, String dateTime, String region, String expiryPeriod, String hostName, String
port) throws Exception {
        canonicalQueryParameters.put("Action", action);
        canonicalQueryParameters.put("DBUser", user);
        canonicalQueryParameters.put("X-Amz-Algorithm", "AWS4-HMAC-SHA256");
        canonicalQueryParameters.put("X-Amz-Credential", accessKey + "%2F" + date +
"%2F" + region + "%2F" + serviceName + "%2Faws4_request");
        canonicalQueryParameters.put("X-Amz-Date", dateTime);
        canonicalQueryParameters.put("X-Amz-Expires", expiryPeriod);
        canonicalQueryParameters.put("X-Amz-SignedHeaders", signedHeader);
        String canonicalQueryString = "";
        while(!canonicalQueryParameters.isEmpty()) {
            String currentQueryParameter = canonicalQueryParameters.firstKey();
            String currentQueryParameterValue =
canonicalQueryParameters.remove(currentQueryParameter);
            canonicalQueryString = canonicalQueryString + currentQueryParameter + "=" +
currentQueryParameterValue;
            if (!currentQueryParameter.equals("X-Amz-SignedHeaders")) {
                canonicalQueryString += "&";
            }
        }
        String canonicalHeaders = "host:" + hostName + ":" + port + '\n';
        requestWithoutSignature = hostName + ":" + port + "/?" + canonicalQueryString;

        String hashedPayload = BinaryUtils.toHex(hash(payload));
        return httpMethod + '\n' + canonicalURIParameter + '\n' + canonicalQueryString
+ '\n' + canonicalHeaders + '\n' + signedHeader + '\n' + hashedPayload;

    }

    //Step 2: Create a string to sign using sig v4
    public static String createStringToSign(String dateTime, String canonicalRequest,
String accessKey, String date, String region) throws Exception {
```

```
        String credentialScope = date + "/" + region + "/" + serviceName + "/
aws4_request";
        return algorithm + '\n' + dateTime + '\n' + credentialScope + '\n' +
 BinaryUtils.toHex(hash(canonicalRequest));

    }

    //Step 3: Calculate signature
    /**
     * Step 3 of the &AWS; Signature version 4 calculation. It involves deriving
     * the signing key and computing the signature. Refer to
     * http://docs.aws.amazon
     * .com/general/latest/gr/sigv4-calculate-signature.html
     */
    public static byte[] calculateSignature(String stringToSign,
                                            byte[] signingKey) {
        return sign(stringToSign.getBytes(Charset.forName("UTF-8")), signingKey,
                SigningAlgorithm.HmacSHA256);
    }

    public static byte[] sign(byte[] data, byte[] key,
                          SigningAlgorithm algorithm) throws SdkClientException {
        try {
            Mac mac = algorithm.getMac();
            mac.init(new SecretKeySpec(key, algorithm.toString()));
            return mac.doFinal(data);
        } catch (Exception e) {
            throw new SdkClientException(
                    "Unable to calculate a request signature: "
                            + e.getMessage(), e);
        }
    }

    public static byte[] newSigningKey(String secretKey,
                                   String dateStamp, String regionName, String
 serviceName) {
        byte[] kSecret = ("AWS4" + secretKey).getBytes(Charset.forName("UTF-8"));
        byte[] kDate = sign(dateStamp, kSecret, SigningAlgorithm.HmacSHA256);
        byte[] kRegion = sign(regionName, kDate, SigningAlgorithm.HmacSHA256);
        byte[] kService = sign(serviceName, kRegion,
                SigningAlgorithm.HmacSHA256);
        return sign(AWS4_TERMINATOR, kService, SigningAlgorithm.HmacSHA256);
    }
```

```
    public static byte[] sign(String stringData, byte[] key,
                      SigningAlgorithm algorithm) throws SdkClientException {
        try {
            byte[] data = stringData.getBytes(UTF8);
            return sign(data, key, algorithm);
        } catch (Exception e) {
            throw new SdkClientException(
                    "Unable to calculate a request signature: "
                            + e.getMessage(), e);
        }
    }

    //Step 4: append the signature
    public static String appendSignature(String signature) {
        return requestWithoutSignature + "&X-Amz-Signature=" + signature;
    }

    public static byte[] hash(String s) throws Exception {
        try {
            MessageDigest md = MessageDigest.getInstance("SHA-256");
            md.update(s.getBytes(UTF8));
            return md.digest();
        } catch (Exception e) {
            throw new SdkClientException(
                    "Unable to compute hash while signing request: "
                            + e.getMessage(), e);
        }
    }
}
```

Connexion à votre instance de base de données

L'exemple de code suivant montre comment créer un jeton d'authentification, puis comment l'utiliser pour se connecter à une instance exécutant MariaDB ou MySQL.

Pour exécuter cet exemple de code, vous avez besoin du AWS SDK pour Java, qui se trouve sur le AWS site. En outre, vous avez besoin des éléments suivants :

- MySQL Connector/J. Cet exemple de code a été testé avec `mysql-connector-java-5.1.33-bin.jar`.

- Certificat intermédiaire pour Amazon RDS () spécifique à une AWS région. (Pour en savoir plus, consultez .) À l'exécution, le chargeur de classe recherche le certificat dans le même annuaire que celui de cet exemple de code Java, afin de pouvoir le trouver.
- Modifiez la valeur des variables suivantes selon les besoins :
  - `RDS_INSTANCE_HOSTNAME` – Le nom d'hôte de l'instance de base de données auquel vous souhaitez accéder.
  - `RDS_INSTANCE_PORT` – Le numéro du port utilisé pour la connexion à votre instance de base de données PostgreSQL.
  - `REGION_NAME`— La AWS région dans laquelle le d'instances de base de données est exécuté.
  - `DB_USER` – Le compte de base de données auquel vous souhaitez accéder.
  - `SSL_CERTIFICATE`— Un certificat SSL pour Amazon RDS spécifique à une AWS région.

    Pour télécharger un certificat pour votre région AWS , veuillez consulter . Placez le certificat SSL dans le même annuaire que ce fichier de programme Java, afin que le chargeur de classe puisse le trouver à l'exécution.

Cet exemple de code permet d'obtenir des AWS informations d'identification à partir de la chaîne de fournisseurs d'informations d'identification par défaut.

> **Note**
>
> Spécifiez un mot de passe pour `DEFAULT_KEY_STORE_PASSWORD` différent de celui indiqué ici, en tant que bonne pratique de sécurité.

```
package com.amazonaws.samples;

import com.amazonaws.services.rds.auth.RdsIamAuthTokenGenerator;
import com.amazonaws.services.rds.auth.GetIamAuthTokenRequest;
import com.amazonaws.auth.BasicAWSCredentials;
import com.amazonaws.auth.DefaultAWSCredentialsProviderChain;
import com.amazonaws.auth.AWSStaticCredentialsProvider;

import java.io.File;
import java.io.FileOutputStream;
import java.io.InputStream;
import java.security.KeyStore;
import java.security.cert.CertificateFactory;
```

```
import java.security.cert.X509Certificate;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Properties;

import java.net.URL;

public class IAMDatabaseAuthenticationTester {
    //&AWS; Credentials of the IAM user with policy enabling IAM Database Authenticated
 access to the db by the db user.
    private static final DefaultAWSCredentialsProviderChain creds = new
 DefaultAWSCredentialsProviderChain();
    private static final String AWS_ACCESS_KEY =
 creds.getCredentials().getAWSAccessKeyId();
    private static final String AWS_SECRET_KEY =
 creds.getCredentials().getAWSSecretKey();

    //Configuration parameters for the generation of the IAM Database Authentication
 token
    private static final String RDS_INSTANCE_HOSTNAME = "rdsmysql.123456789012.us-
west-2.rds.amazonaws.com";
    private static final int RDS_INSTANCE_PORT = 3306;
    private static final String REGION_NAME = "us-west-2";
    private static final String DB_USER = "jane_doe";
    private static final String JDBC_URL = "jdbc:mysql://" + RDS_INSTANCE_HOSTNAME +
 ":" + RDS_INSTANCE_PORT;

    private static final String SSL_CERTIFICATE = "rds-ca-2019-us-west-2.pem";

    private static final String KEY_STORE_TYPE = "JKS";
    private static final String KEY_STORE_PROVIDER = "SUN";
    private static final String KEY_STORE_FILE_PREFIX = "sys-connect-via-ssl-test-
cacerts";
    private static final String KEY_STORE_FILE_SUFFIX = ".jks";
    private static final String DEFAULT_KEY_STORE_PASSWORD = "changeit";

    public static void main(String[] args) throws Exception {
        //get the connection
        Connection connection = getDBConnectionUsingIam();

        //verify the connection is successful
```

```
        Statement stmt= connection.createStatement();
        ResultSet rs=stmt.executeQuery("SELECT 'Success!' FROM DUAL;");
        while (rs.next()) {
             String id = rs.getString(1);
            System.out.println(id); //Should print "Success!"
        }

        //close the connection
        stmt.close();
        connection.close();

        clearSslProperties();

    }

    /**
     * This method returns a connection to the db instance authenticated using IAM
Database Authentication
     * @return
     * @throws Exception
     */
    private static Connection getDBConnectionUsingIam() throws Exception {
        setSslProperties();
        return DriverManager.getConnection(JDBC_URL, setMySqlConnectionProperties());
    }

    /**
     * This method sets the mysql connection properties which includes the IAM Database
Authentication token
     * as the password. It also specifies that SSL verification is required.
     * @return
     */
    private static Properties setMySqlConnectionProperties() {
        Properties mysqlConnectionProperties = new Properties();
        mysqlConnectionProperties.setProperty("verifyServerCertificate","true");
        mysqlConnectionProperties.setProperty("useSSL", "true");
        mysqlConnectionProperties.setProperty("user",DB_USER);
        mysqlConnectionProperties.setProperty("password",generateAuthToken());
        return mysqlConnectionProperties;
    }

    /**
     * This method generates the IAM Auth Token.
     * An example IAM Auth Token would look like follows:
```

```
     * btusi123.cmz7kenwo2ye.rds.cn-north-1.amazonaws.com.cn:3306/?
Action=connect&DBUser=iamtestuser&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-
Date=20171003T010726Z&X-Amz-SignedHeaders=host&X-Amz-Expires=899&X-Amz-
Credential=AKIAPFXHGVDI5RNFO4AQ%2F20171003%2Fcn-north-1%2Frds-db%2Faws4_request&X-Amz-
Signature=f9f45ef96c1f770cdad11a53e33ffa4c3730bc03fdee820cfdf1322eed15483b
     * @return
     */
    private static String generateAuthToken() {
        BasicAWSCredentials awsCredentials = new BasicAWSCredentials(AWS_ACCESS_KEY,
 AWS_SECRET_KEY);

        RdsIamAuthTokenGenerator generator = RdsIamAuthTokenGenerator.builder()
                .credentials(new
 AWSStaticCredentialsProvider(awsCredentials)).region(REGION_NAME).build();
        return generator.getAuthToken(GetIamAuthTokenRequest.builder()

 .hostname(RDS_INSTANCE_HOSTNAME).port(RDS_INSTANCE_PORT).userName(DB_USER).build());
    }

    /**
     * This method sets the SSL properties which specify the key store file, its type
 and password:
     * @throws Exception
     */
    private static void setSslProperties() throws Exception {
        System.setProperty("javax.net.ssl.trustStore", createKeyStoreFile());
        System.setProperty("javax.net.ssl.trustStoreType", KEY_STORE_TYPE);
        System.setProperty("javax.net.ssl.trustStorePassword",
 DEFAULT_KEY_STORE_PASSWORD);
    }

    /**
     * This method returns the path of the Key Store File needed for the SSL
 verification during the IAM Database Authentication to
     * the db instance.
     * @return
     * @throws Exception
     */
    private static String createKeyStoreFile() throws Exception {
        return createKeyStoreFile(createCertificate()).getPath();
    }

    /**
     *  This method generates the SSL certificate
```

```
     * @return
     * @throws Exception
     */
    private static X509Certificate createCertificate() throws Exception {
        CertificateFactory certFactory = CertificateFactory.getInstance("X.509");
        URL url = new File(SSL_CERTIFICATE).toURI().toURL();
        if (url == null) {
            throw new Exception();
        }
        try (InputStream certInputStream = url.openStream()) {
            return (X509Certificate) certFactory.generateCertificate(certInputStream);
        }
    }

    /**
     * This method creates the Key Store File
     * @param rootX509Certificate - the SSL certificate to be stored in the KeyStore
     * @return
     * @throws Exception
     */
    private static File createKeyStoreFile(X509Certificate rootX509Certificate) throws
Exception {
        File keyStoreFile = File.createTempFile(KEY_STORE_FILE_PREFIX,
KEY_STORE_FILE_SUFFIX);
        try (FileOutputStream fos = new FileOutputStream(keyStoreFile.getPath())) {
            KeyStore ks = KeyStore.getInstance(KEY_STORE_TYPE, KEY_STORE_PROVIDER);
            ks.load(null);
            ks.setCertificateEntry("rootCaCertificate", rootX509Certificate);
            ks.store(fos, DEFAULT_KEY_STORE_PASSWORD.toCharArray());
        }
        return keyStoreFile;
    }

    /**
     * This method clears the SSL properties.
     * @throws Exception
     */
    private static void clearSslProperties() throws Exception {
           System.clearProperty("javax.net.ssl.trustStore");
           System.clearProperty("javax.net.ssl.trustStoreType");
           System.clearProperty("javax.net.ssl.trustStorePassword");
    }
```

```
}
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Connexion à votre d'instances de base de données à l'aide de l'authentification IAM et du AWS SDK pour Python (Boto3)

Vous pouvez vous connecter à une instance de base de données RDS pour MariaDB, MySQL ou PostgreSQL Aurora MySQL ou à un cluster de base de données Aurora PostgreSQL ci-dessous. AWS SDK pour Python (Boto3)

Prérequis

Les conditions préalables à la connexion à votre instance de base de données à l'aide de l'authentification IAM sont les suivantes :

- Activation et désactivation de l'authentification de base de données IAM
- Création et utilisation d'une politique IAM pour l'accès à une base de données IAM
- Création d'un compte de base de données à l'aide de l'authentification IAM

En outre, assurez-vous que les bibliothèques importées dans l'exemple de code existent sur votre système.

Exemples

Les exemples de code utilisent des profils pour les informations d'identification partagées. Pour plus d'informations sur la spécification des informations d'identification, consultez la section Informations d'identification dans la AWS SDK pour Python (Boto3) documentation.

Les exemples de code suivants montre comment générer un jeton d'authentification, puis comment l'utiliser pour se connecter à une instance de base de données.

Pour exécuter cet exemple de code, vous avez besoin du AWS SDK pour Python (Boto3), qui se trouve sur le AWS site.

Modifiez la valeur des variables suivantes selon les besoins :

- `ENDPOINT` – Le point de terminaison de l'instance de base de données à laquelle vous souhaitez accéder.

- `PORT` – Le numéro du port utilisé lors de la connexion au d'instances de base de données.
- `USER` – Le compte de base de données auquel vous souhaitez accéder.
- `REGION`— La AWS région dans laquelle le d'instances de base de données est exécuté
- `DBNAME` – La base de données à laquelle vous souhaitez accéder.
- `SSLCERTIFICATE` – Le chemin d'accès complet au certificat SSL pour Amazon RDS

  Pour `ssl_ca`, spécifiez un certificat SSL. Pour télécharger un certificat SSL, consultez .

> **Note**
>
> Vous ne pouvez pas utiliser un enregistrement DNS Route 53 personnalisé à la place du point de terminaison de l'instance de base de données pour générer le jeton d'authentification.

Ce code se connecte à une instance de base de données MariaDB ou MySQL.

Avant d'exécuter ce code, installez le pilote PyMy SQL en suivant les instructions du Python Package Index.

```
import pymysql
import sys
import boto3
import os

ENDPOINT="mysqldb.123456789012.us-east-1.rds.amazonaws.com"
PORT="3306"
USER="jane_doe"
REGION="us-east-1"
DBNAME="mydb"
os.environ['LIBMYSQL_ENABLE_CLEARTEXT_PLUGIN'] = '1'

#gets the credentials from .aws/credentials
session = boto3.Session(profile_name='default')
client = session.client('rds')

token = client.generate_db_auth_token(DBHostname=ENDPOINT, Port=PORT, DBUsername=USER,
 Region=REGION)
```

```
try:
    conn =
 pymysql.connect(auth_plugin_map={'mysql_clear_password':None},host=ENDPOINT,
 user=USER, password=token, port=PORT, database=DBNAME, ssl_ca='SSLCERTIFICATE',
 ssl_verify_identity=True, ssl_verify_cert=True)
    cur = conn.cursor()
    cur.execute("""SELECT now()""")
    query_results = cur.fetchall()
    print(query_results)
except Exception as e:
    print("Database connection failed due to {}".format(e))
```

Ce code se connecte à une instance de base de données PostgreSQL.

Avant d'exécuter ce code, installez `psycopg2` en suivant les instructions de la documentation de Psycopg.

```
import psycopg2
import sys
import boto3
import os

ENDPOINT="postgresmydb.123456789012.us-east-1.rds.amazonaws.com"
PORT="5432"
USER="jane_doe"
REGION="us-east-1"
DBNAME="mydb"

#gets the credentials from .aws/credentials
session = boto3.Session(profile_name='RDSCreds')
client = session.client('rds')

token = client.generate_db_auth_token(DBHostname=ENDPOINT, Port=PORT, DBUsername=USER,
 Region=REGION)

try:
    conn = psycopg2.connect(host=ENDPOINT, port=PORT, database=DBNAME, user=USER,
 password=token, sslrootcert="SSLCERTIFICATE")
    cur = conn.cursor()
    cur.execute("""SELECT now()""")
    query_results = cur.fetchall()
```

```
    print(query_results)
except Exception as e:
    print("Database connection failed due to {}".format(e))
```

Si vous souhaitez vous connecter à une instance de bases de données via un proxy, consultez Connexion à un proxy à l'aide de l'authentification IAM.

Résolution des problèmes liés à l'authentification IAM DB

Vous trouverez ci-dessous des idées pour résoudre certains problèmes courants d'authentification de base de données IAM, ainsi que des informations sur les CloudWatch journaux et les métriques relatifs à l'authentification de base de données IAM.

Exportation des journaux d'erreurs d'authentification IAM DB vers CloudWatch des journaux

Les journaux des erreurs d'authentification IAM DB sont stockés sur l'hôte de la base de données, et vous pouvez les exporter dans votre compte CloudWatch Logs. Utilisez les journaux et les méthodes de correction de cette page pour résoudre les problèmes d'authentification de base de données IAM.

Vous pouvez activer les exportations de CloudWatch journaux vers Logs à partir de la console et de l'API RDS. AWS CLI Pour les instructions relatives à la console, voirPublication des journaux de base de données sur Amazon CloudWatch Logs.

Pour exporter vos journaux d'erreurs d'authentification de base de données IAM vers CloudWatch Logs lors de la création d'un de base de données à partir du AWS CLI, utilisez la commande suivante :

```
aws rds create-db-instance --db-instance-identifier mydbinstance \
--region us-east-1 \
--db-instance-class db.t3.large \
--allocated-storage 50 \
--engine postgres \
--engine-version 16 \
--port 5432 \
--master-username master \
--master-user-password password \
--publicly-accessible \
--enable-iam-database-authentication \
--enable-cloudwatch-logs-exports=iam-db-auth-error
```

Pour exporter vos journaux d'erreurs d'authentification de base de données IAM vers CloudWatch Logs lorsque vous modifiez un de base de données à partir du AWS CLI, utilisez la commande suivante :

```
aws rds modify-db-instance --db-instance-identifier mydbinstance \
--region us-east-1 \
--cloudwatch-logs-export-configuration '{"EnableLogTypes":["iam-db-auth-error"]}'
```

Pour vérifier si le de votre instance de base de données exporte les journaux d'authentification IAM DB vers CloudWatch Logs, vérifiez si le `EnabledCloudwatchLogsExports` paramètre est défini sur `iam-db-auth-error` dans la sortie de la `describe-db-instances` commande.

```
aws rds describe-db-instances --region us-east-1 --db-instance-identifier mydbinstance
            ...

             "EnabledCloudwatchLogsExports": [
                "iam-db-auth-error"
            ],
            ...
```

Métriques d'authentification CloudWatch IAM DB

Amazon RDS fournit des statistiques en temps quasi réel concernant l'authentification de base de données IAM sur votre compte Amazon. CloudWatch Le tableau suivant répertorie les métriques d'authentification IAM DB disponibles à l'aide CloudWatch de :

Métrique	Description
`IamDbAuthConnectionRequests`	Nombre total de demandes de connexion effectuées avec l'authentification IAM DB.
`IamDbAuthConnectionSuccess`	Nombre total de demandes d'authentification IAM DB réussies.
`IamDbAuthConnectionFailure`	Nombre total de demandes d'authentification IAM DB ayant échoué.

Métrique	Description
`IamDbAuthConnectionFailureInvalidToken`	Nombre total de demandes d'authentification IAM DB ayant échoué en raison d'un jeton non valide.
`IamDbAuthConnectionFailureInsufficientPermissions`	Nombre total de demandes d'authentification IAM DB ayant échoué en raison de politiques ou d'autorisations incorrectes.
`IamDbAuthConnectionFailureThrottling`	Nombre total de demandes d'authentification de base de données IAM ayant échoué en raison de la limitation de l'authentification de base de données IAM.
`IamDbAuthConnectionFailureServerError`	Nombre total de demandes d'authentification de base de données IAM ayant échoué en raison d'une erreur interne du serveur dans la fonctionnalité d'authentification de base de données IAM.

Problèmes courants et solutions correspondantes

Vous pouvez rencontrer les problèmes suivants lors de l'utilisation de l'authentification IAM DB. Suivez les étapes de correction indiquées dans le tableau pour résoudre les problèmes :

Erreur	Métrique (s)	Cause	Solution

`[ERROR] Failed to authenticate the connection request for user` *`db_user`* `because the provided token is malformed or otherwise`	`IamDbAuthConnectionFailure`
	
	`IamDbAuthConnectionFailureInvalidToken`
		Le jeton d'authentification IAM DB figurant dans la demande de connexion n'est pas un jeton SigV4A valide ou il n'est pas formaté correctement.	Vérifiez votre stratégie de génération de jetons dans votre application. Dans certains cas, assurez-vous de transmettre le jeton avec un formatage valide. Le fait de tronquer

Erreur	Métrique (s)	Cause	Solution
`invalid. (Status Code: 400, Error Code: InvalidToken)`			le jeton (ou un formatage de chaîne incorrect) le rendra invalide.

`[ERROR] Failed to authenticate the connection request for user` *`db_user`* `because the token age is longer than 15 minutes. (Status Code: 400, Error Code:ExpiredToken)`	`IamDbAuthConnectionFailure`
	
	`IamDbAuthConnectionFailureInvalidToken`
		Le jeton d'authentification IAM DB a expiré. Les jetons ne sont valables que 15 minutes.	Vérifiez la logique de réutilisation des and/or jetons de mise en cache dans votre application. Vous ne devez pas réutiliser des jetons datant de plus de 15 minutes.

Erreur	Métrique (s)	Cause	Solution

`[ERROR] Failed to authorize the connection request for user` *`db_user`* `because the IAM policy assumed by the caller 'arn:aws:sts::123456789012:assumed-role/ <RoleName>/ <RoleSession>' is not authorized to perform `rds-db:connect` on the DB instance. (Status Code: 403, Error Code:NotAuthorized)`	`IamDbAuthConnectionFailure`
	
	`IamDbAuthConnectionFailureInsufficientPermissions`
		Cette erreur peut être due aux raisons suivantes :
		• La politique IAM adoptée par l'application n'autorise pas l'`rds-db:connect` action.
		• Vous supposez que le format de connexion role/policy *`db_user`* à la base de données est incorrect.
		• Vous utilisez la bonne politique pour*`db_user`*, mais vous ne vous connectez pas à la bonne base de données.
			Vérifiez que la and/or politique de rôle IAM que vous adoptez est bien celle que vous adoptez dans votre application. Assurez-vous de suivre la même politique pour générer le jeton que pour vous connecter à la base de données.

Erreur	Métrique (s)	Cause	Solution

`[ERROR] Failed to authorize the connection request for user db_user due to IAM DB authentication throttling. (Status Code: 429, Error Code: ThrottlingException)`	`IamDbAuthConnectionFailure`
	
	`IamDbAuthConnectionFailureThrottling`
		Vous envoyez trop de demandes de connexion à votre base de données en peu de temps. La limite de limitation de l'authentification IAM DB est de 200 connexions par seconde.	Réduisez le taux d'établissement de nouvelles connexions grâce à l'authentification IAM. Envisagez d'implémenter le regroupement de connexions à l'aide du proxy RDS afin de réutiliser les connexions établies dans votre application.

`[ERROR] Failed to authorize the connection request for user db_user due to an internal IAM DB authentication error. (Status Code: 500, Error Code: InternalError)`	`IamDbAuthConnectionFailure`
	
	`IamDbAuthConnectionFailureThrottling`
		Une erreur interne s'est produite lors de l'autorisation de la connexion à la base de données avec l'authentification IAM DB.	Communiquez avec https://aws.amazon.com/premiumsupport/ nous pour étudier le problème.

Résolution des problèmes liés à Identity and Access Amazon RDS

Utilisez les informations suivantes pour identifier et résoudre les problèmes courants que vous pouvez rencontrer lorsque vous travaillez avec Amazon RDS et IAM.

Rubriques

- Je ne suis pas autorisé à effectuer une action dans Amazon RDS
- Je ne suis pas autorisé à exécuter iam:PassRole

- Je souhaite autoriser des personnes extérieures à mon compte AWS, à accéder à mes ressources Amazon RDS.

Je ne suis pas autorisé à effectuer une action dans Amazon RDS

Si la AWS Management Console indique que vous n'êtes pas autorisé à exécuter une action, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni vos informations de connexion.

L'exemple d'erreur suivant se produit lorsque l'utilisateur `mateojackson` tente d'utiliser la console pour afficher des informations détaillées concernant un *`widget`*, mais ne dispose pas d'autorisations `rds:`*`GetWidget`*.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 rds:GetWidget on resource: my-example-widget
```

Le cas échéant, Mateo doit demander à son administrateur de mettre à jour ses politiques pour lui permettre d'accéder à la ressource *`my-example-widget`* à l'aide de l'action `rds:`*`GetWidget`*.

Je ne suis pas autorisé à exécuter iam:PassRole

Si vous recevez un message d'erreur selon lequel vous n'êtes pas autorisé à exécuter l'action `iam:PassRole`, vous devez contacter votre administrateur pour obtenir de l'aide. Votre administrateur est la personne qui vous a fourni vos informations de connexion. Demandez à cette personne de mettre à jour vos stratégies pour vous permettre de transmettre un rôle à Amazon RDS.

Certains services AWS vous permettent de transmettre un rôle existant à ce service, au lieu de créer un nouveau rôle de service ou rôle lié à un service. Pour ce faire, un utilisateur doit disposer des autorisations nécessaires pour transmettre le rôle au service.

L'exemple d'erreur suivant se produit lorsqu'un utilisateur nommé `marymajor` essaie d'utiliser la console pour exécuter une action dans Amazon RDS. Toutefois, l'action nécessite que le service ait des autorisations accordées par une fonction du service. Mary ne dispose pas des autorisations nécessaires pour transférer le rôle au service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

Dans ce cas, Mary demande à son administrateur de mettre à jour ses politiques pour lui permettre d'exécuter l'action `iam:PassRole`.

Je souhaite autoriser des personnes extérieures à mon compte AWS, à accéder à mes ressources Amazon RDS.

Vous pouvez créer un rôle que les utilisateurs provenant d'autres comptes ou les personnes extérieures à votre organisation pourront utiliser pour accéder à vos ressources. Vous pouvez spécifier la personne à qui vous souhaitez confier le rôle. Pour les services qui prennent en charge les politiques basées sur les ressources ou les listes de contrôle d'accès (ACL), vous pouvez utiliser ces politiques pour donner l'accès à vos ressources.

Pour en savoir plus, consultez les éléments suivants :

- Pour savoir si Amazon RDS prend en charge ces fonctionnalités, consultez Comment fonctionne RDSAmazon avec IAM.
- Pour savoir comment octroyer l'accès à vos ressources à des comptes AWS dont vous êtes propriétaire, veuillez consulter la section Fournir l'accès à un utilisateur IAM dans un autre compte AWS que vous possédez dans le Guide de l'utilisateur IAM.
- Pour savoir comment octroyer accès à vos ressources à des comptes AWS tiers, veuillez consulter Fournir l'accès aux comptes AWS appartenant à des tiers dans le Guide de l'utilisateur IAM.
- Pour savoir comment fournir un accès par le biais de la fédération d'identité, veuillez consulter Fournir un accès à des utilisateurs authentifiés en externe (fédération d'identité) dans le Guide de l'utilisateur IAM.
- Pour découvrir quelle est la différence entre l'utilisation des rôles et l'utilisation des politiques basées sur les ressources pour l'accès entre comptes, consultez Différence entre les rôles IAM et les politiques basées sur les ressources dans le Guide de l'utilisateur IAM.

Journalisation et surveillance dans Amazon RDS

La surveillance joue un rôle important dans le maintien de la fiabilité, de la disponibilité et des performances d'Amazon RDS, d' et de vos AWS solutions. Vous devez collecter des données de surveillance provenant de toutes les parties de votre AWS solution afin de pouvoir corriger plus facilement une défaillance multipoint, le cas échéant. AWS fournit plusieurs outils pour surveiller vos ressources Amazon RDS et répondre aux incidents potentiels :

CloudWatch Alarmes Amazon

À l'aide des CloudWatch alarmes Amazon, vous observez une seule métrique sur une période que vous spécifiez. Si la métrique dépasse un seuil donné, une notification est envoyée à une rubrique ou AWS Auto Scaling à une politique Amazon SNS. CloudWatch les alarmes n'appellent pas d'actions car elles se trouvent dans un état particulier. L'état doit avoir changé et avoir été conservé pendant un nombre de périodes spécifié.

AWS CloudTrail Journaux

CloudTrail fournit un enregistrement des actions entreprises par un utilisateur, un rôle ou un AWS service dans Amazon RDS . CloudTrail capture tous les appels d'API pour Amazon RDS sous forme d'événements, y compris les appels depuis la console et les appels de code vers les opérations d'API Amazon RDS. À l'aide des informations collectées par CloudTrail, vous pouvez déterminer la demande qui a été envoyée à Amazon RDS Amazon , l'adresse IP à partir de laquelle la demande a été faite, l'auteur de la demande, la date à laquelle elle a été faite, ainsi que des informations supplémentaires. Pour plus d'informations, consultez Surveillance (Amazon RDS API appelle) AWS CloudTrail.

Surveillance améliorée

Amazon RDS fournit des métriques en temps réel pour le système d'exploitation (OS) sur lequel s'exécute votre d'instances de base de données. Vous pouvez consulter les métriques de votre d'instances de base de données à l'aide de la console ou utiliser la sortie JSON Enhanced Monitoring d'Amazon CloudWatch Logs dans le système de surveillance de votre choix. Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.

Analyse des performances d'Amazon RDS

Performance Insights développe les fonctionnalités de surveillance Amazon RDS existantes pour illustrer les performances de votre base de données et vous aider à analyser les problèmes qui l'affectent. Grâce au tableau de bord de Performance Insights, vous pouvez visualiser la charge de la base de données et la filtrer par attentes, instructions SQL, hôtes ou utilisateurs. Pour plus d'informations, consultez Surveillance de la charge de la base de données avec Performance Insights sur Amazon RDS.

Journaux de base de données

Vous pouvez consulter, télécharger et consulter les journaux de base de données à l'aide de l'API AWS Management Console AWS CLI, ou RDS. Pour plus d'informations, consultez Surveillance des fichiers journaux RDSAmazon.

Amazon RDS : recommandations relatives à

Amazon RDS fournit des recommandations automatisées pour les ressources de base de données. Ces recommandations fournissent des conseils sur les meilleures pratiques en analysant les données de configuration, d'utilisation et de performance des d'instances de base de données. Pour plus d’informations, consultez Recommandations d'Amazon RDS.

Notification d'événement Amazon RDS

Amazon RDS Ces notifications peuvent se présenter sous n'importe quel format de notification pris en charge par Amazon SNS pour une AWS région, tel qu'un e-mail, un SMS ou un appel vers un point de terminaison HTTP. Pour plus d’informations, consultez Utilisation des notifications d'RDSévénements Amazon.

AWS Trusted Advisor

Trusted Advisor s'appuie sur les meilleures pratiques apprises en servant des centaines de milliers de AWS clients. Trusted Advisor inspecte votre AWS environnement, puis émet des recommandations lorsque des opportunités se présentent pour économiser de l'argent, améliorer la disponibilité et les performances du système ou contribuer à combler les failles de sécurité. Tous les AWS clients ont accès à cinq Trusted Advisor chèques. Les clients disposant d'un plan de support Business ou Enterprise peuvent consulter tous les Trusted Advisor chèques.

Trusted Advisor possède les vérifications suivantes relatives à Amazon RDS et :

- Instances de base de données Amazon RDS inactives
- Risque d'accès au groupe de sécurité Amazon RDS
- Sauvegardes Amazon RDS
- Amazon RDS

Pour plus d'informations sur ces vérifications, consultez Bonnes pratiques Trusted Advisor (Checks).

Pour plus d'informations sur la surveillance d'Amazon RDS (), consultezSurveillance des métriques dans une RDS instance Amazon.

Validation de conformité pour Amazon RDS

Des auditeurs tiers évaluent la sécurité et la conformité d'Amazon RDS dans le cadre de plusieurs programmes de AWS conformité. Il s'agit SOC PCI notamment de RAMP la Fed HIPAA et d'autres.

Pour une liste des AWS services concernés par des programmes de conformité spécifiques, voir AWS Services concernés par programme de conformité. Pour obtenir des informations générales, veuillez consulter Programmes de conformité d'AWS.

Vous pouvez télécharger des rapports d'audit tiers à l'aide de AWS Artifact. Pour plus d'informations, consultez la section Téléchargement de rapports dans AWS Artifact.

Votre responsabilité en matière de conformité lors de l'utilisation d'RDSAmazon est déterminée par la sensibilité de vos données, les objectifs de conformité de votre organisation et les lois et réglementations applicables. AWS fournit les ressources suivantes pour faciliter la mise en conformité :

- Guides de démarrage rapide sur la sécurité et la conformité : ces guides de déploiement abordent les considérations architecturales et indiquent les étapes à suivre pour déployer des environnements de base axés sur la sécurité et la conformité sur. AWS
- Architecture axée sur la HIPAA sécurité et la conformité sur Amazon Web Services : ce livre blanc décrit comment les entreprises peuvent AWS créer HIPAA des applications conformes.
- AWS ressources de conformité : collection de classeurs et de guides susceptibles de s'appliquer à votre secteur d'activité et à votre région.
- AWS Config— Ce AWS service évalue dans quelle mesure les configurations de vos ressources sont conformes aux pratiques internes, aux directives du secteur et aux réglementations.
- AWS Security Hub— Cela Service AWS fournit une vue complète de votre état de sécurité interne AWS. Security Hub utilise des contrôles de sécurité pour évaluer vos ressources AWS et vérifier votre conformité par rapport aux normes et aux bonnes pratiques du secteur de la sécurité. Pour obtenir la liste des services et des contrôles pris en charge, consultez Référence des contrôles Security Hub.

Résilience dans Amazon RDS

L'infrastructure mondiale d'AWS s'articule autours de régions et de zones de disponibilité AWS.AWS Les Régions fournissent plusieurs zones de disponibilité physiquement séparées et isolées, reliées par un réseau à latence faible, à haut débit et hautement redondant. Avec les zones de disponibilité, vous pouvez concevoir et exploiter des applications et des bases de données qui basculent automatiquement d'une zone de disponibilité à l'autre sans interruption. Les zones de disponibilité sont plus hautement disponibles, tolérantes aux pannes et évolutives que les infrastructures traditionnelles à un ou plusieurs centres de données.

Pour plus d'informations sur les régions et les zones de disponibilité AWS, veuillez consulter Infrastructure mondiale AWS.

Outre l'infrastructure globale d'AWS, Amazon RDS offrent différentes fonctions qui contribuent à satisfaire vos besoins en matière de résilience et de sauvegarde de données.

Sauvegarde et restauration

Amazon RDS crée et enregistre des sauvegardes automatiques de votre instance de base de données. Amazon RDS crée un instantané du volume de stockage de votre instance de base de données, en sauvegardant l'intégralité de cette dernière et pas seulement les bases de données.

Amazon RDS crée et des sauvegardes automatiques de votre instance de base de données pendant la fenêtre de sauvegarde de celle-ci. Amazon RDS enregistre les sauvegardes automatiques de votre instance de base de données selon la période de rétention des sauvegardes que vous spécifiez. Le cas échéant, vous pouvez récupérer votre base de données à tout moment pendant la période de rétention des sauvegardes. Vous pouvez également sauvegarder votre instance de base de données manuellement, en créant manuellement un instantané de bases de données.

Vous pouvez créer une instance de base de données en restaurant à partir de cet instantané de base de données comme solution de récupération après sinistre si l'instance de base de données source échoue.

Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

Réplication

Amazon RDS utilise aussi la fonctionnalité de réplication intégrée des moteurs de base de données PostgreSQL, MySQL et MariaDB pour créer un type particulier d'instance de base de données appelé

réplica en lecture à partir d'une instance de base de données source. Les mises à jour apportées à l'instance de base de données source sont copiées de façon asynchrone sur le réplica en lecture. Vous pouvez réduire la charge sur votre instance de base de données source en acheminant les requêtes en lecture depuis vos applications vers le réplica en lecture. Les réplicas en lecture permettent une montée en puissance basée sur Elastic au-delà des contraintes de capacité d'une seule instance de base de données dans le cas de charges de travail de base de données à lecture intensive. Vous pouvez effectuez la promotion d'un réplica en lecture en instance autonome comme plan de reprise après sinistre en cas de défaillance de l'instance de base de données source. Pour certains moteurs de base de données, Amazon RDS prend aussi en charge d'autres options de réplication.

Pour plus d'informations, consultez Utilisation des réplicas en lecture d'instance de base de données.

Basculement

Amazon RDS fournit une haute disponibilité et une prise en charge du basculement pour les instances de base de données grâce aux déploiements Multi-AZ. Amazon RDS utilise plusieurs technologies distinctes pour fournir la prise en charge du basculement. Les déploiements multi-AZ pour les instances de base de données Oracle, PostgreSQL, MySQL et MariaDB utilisent la technologie de basculement d'Amazon. Les instances de base de données SQL Server utilisent la mise en miroir de bases de données SQL Server (DBM).

Pour plus d'informations, consultez Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.

Sécurité de l'infrastructure dans Amazon RDS

En tant que service géré, Amazon Relational Database Service est protégé AWS par la sécurité du réseau mondial. Pour plus d'informations sur les services AWS de sécurité et sur la manière dont AWS l'infrastructure est protégée, consultez la section Sécurité du AWS cloud. Pour concevoir votre AWS environnement en utilisant les meilleures pratiques en matière de sécurité de l'infrastructure, consultez la section Protection de l'infrastructure dans le cadre AWS bien architecturé du pilier de sécurité.

Vous utilisez les API appels AWS publiés pour accéder à Amazon RDS via le réseau. Les clients doivent prendre en charge les éléments suivants :

- Sécurité de la couche de transport (TLS). Nous avons besoin de la TLS version 1.2 et recommandons la TLS version 1.3.
- Des suites de chiffrement parfaitement confidentielles (PFS) telles que (Ephemeral Diffie-Hellman) ou DHE ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). La plupart des systèmes modernes tels que Java 7 et les versions ultérieures prennent en charge ces modes.

En outre, les demandes doivent être signées à l'aide d'un identifiant de clé d'accès et d'une clé d'accès secrète associés à un IAM principal. Vous pouvez également utiliser AWS Security Token Service (AWS STS) pour générer des informations d'identification de sécurité temporaires et signer les demandes.

En outre, Amazon RDS propose des fonctionnalités qui contribuent à renforcer la sécurité de l'infrastructure.

Groupes de sécurité

Les groupes de sécurité contrôlent l'accès dont dispose le trafic entrant et sortant d'une instance de base de données. Par défaut, l'accès au réseau est désactivé sur une instance de base de données. Vous pouvez spécifier des règles dans un groupe de sécurité qui autorisent l'accès depuis une plage d'adresses IP, un port ou un groupe de sécurité. Une fois les règles de trafic entrant configurées, les mêmes règles s'appliquent à tou(te)s les instances de base de données qui sont associé(e)s à ce groupe de sécurité.

Pour de plus amples informations, veuillez consulter Contrôle d'accès par groupe de sécurité.

Accessible publiquement

Lorsque vous lancez une instance de base de données dans un cloud privé virtuel (VPC) basé sur le VPC service Amazon, vous pouvez activer ou désactiver l'accessibilité publique pour cette instance de base de données. Pour indiquer si le DNS nom de l'instance de base de données que vous créez correspond à une adresse IP publique, vous utilisez le paramètre d'accessibilité public. Ce paramètre vous permet de définir s'il existe un accès publique à l'instance de base de données. Vous pouvez modifier une instance de base de données pour activer ou désactiver l'accessibilité publique en modifiant le paramètre Public accessibility (Accessibilité publique).

Pour de plus amples informations, veuillez consulter Masquer un(e) instance de base de données dans un VPC depuis Internet.

> **Note**
>
> Si votre instance de base de données se trouve dans un VPC mais n'est pas accessible au public, vous pouvez également utiliser une AWS VPN connexion site à site ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour de plus amples informations, veuillez consulter Confidentialité du trafic inter-réseau.

API Amazon RDS et points de terminaison d'un VPC d'interface (AWS PrivateLink)

Vous pouvez établir une connexion privée entre votre VPC et vos points de terminaison d'API Amazon RDS en créant un point de terminaison de VPC d'interface. Les points de terminaison d'interface sont alimentés par AWS PrivateLink.

AWS PrivateLink vous permet d'accéder en privé aux opérations de l'API Amazon RDS sans passerelle Internet, appareil NAT, connexion VPN ou AWS Direct Connect connexion. Les instances de base de données de votre VPC n'ont pas besoin d'adresses IP publiques pour communiquer avec les points de terminaison d'API Amazon RDS for lancer, modifier ou mettre fin à des instances et. Vos instances de base de données n'ont pas non plus besoin d'adresses IP publiques pour utiliser une des opérations d'API RDS disponibles. Le trafic entre votre VPC et Amazon RDS ne quitte pas le réseau Amazon.

Chaque point de terminaison d'interface est représenté par une ou plusieurs interfaces réseau Elastic dans vos sous-réseaux. Pour plus d'informations sur les interfaces réseau Elastic, veuillez consulter Interfaces réseau Elastic dans le Guide de l'utilisateur Amazon EC2.

Pour plus d'informations sur les points de terminaison VPC, consultez la section Interface VPC endpoints () dans AWS PrivateLink le guide de l'utilisateur Amazon VPC. Pour plus d'informations sur les opérations d'API RDS, consultez Référence d'API Amazon RDS.

Vous n'avez pas besoin d'un point de terminaison VPC d'interface pour vous connecter à un(e) instance de base de données. Pour plus d’informations, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.

Considérations relatives aux points de terminaison d'un VPC

Avant de configurer un point de terminaison d'un VPC d'interface pour les points de terminaison d'API Amazon RDS, assurez-vous de vérifier les propriétés et limitations du point de terminaison d'interface dans le Amazon VPC Guide de l'utilisateur.

Toutes les opérations d'API RDS pertinentes pour gestion de ressources Amazon RDS sont disponibles à partir de votre VPC à l'aide d' AWS PrivateLink.

Les politiques de point de terminaison d'un VPC sont prises en charge pour les points de terminaison de l'API RDS. Par défaut, l'accès complet aux opérations de l'API RDS est autorisé via le point de

terminaison. Pour plus d'informations, veuillez consulter Contrôle de l'accès aux services avec points de terminaison d'un VPC dans le Amazon VPC Guide de l'utilisateur.

Disponibilité

L'API Amazon RDS prend actuellement en charge les points de terminaison VPC dans les régions suivantes : AWS

- USA Est (Ohio)
- USA Est (Virginie du Nord)
- USA Ouest (Californie du Nord)
- USA Ouest (Oregon)
- Afrique (Le Cap)
- Asie-Pacifique (Hong Kong)
- Asia Pacific (Mumbai)
- Asie-Pacifique (Osaka)
- Asia Pacific (Seoul)
- Asie-Pacifique (Singapour)
- Asie-Pacifique (Sydney)
- Asie-Pacifique (Tokyo)
- Canada (Centre)
- Canada Ouest (Calgary)
- Chine (Beijing)
- China (Ningxia)
- Europe (Francfort)
- Europe (Zurich)
- Europe (Irlande)
- Europe (Londres)
- Europe (Paris)
- Europe (Stockholm)
- Europe (Milan)
- Israël (Tel Aviv)
- Moyen-Orient (Bahreïn)

- Amérique du Sud (Sao Paulo)
- AWS GovCloud (USA Est)
- AWS GovCloud (US-Ouest)

Création d'un point de terminaison de VPC d'interface pour l'API Amazon RDS

Vous pouvez créer un point de terminaison VPC pour l'API Amazon RDS à l'aide de la console Amazon VPC ou du (). AWS Command Line Interface AWS CLI Pour plus d'informations, consultez Création d'un point de terminaison d'interface dans le Guide de l'utilisateur Amazon VPC.

Créez un point de terminaison de VPC pour l'API Amazon RDS à l'aide du nom de service `com.amazonaws.region.rds`.

À l'exception des AWS régions de Chine, si vous activez le DNS privé pour le point de terminaison, vous pouvez envoyer des demandes d'API à Amazon RDS avec le point de terminaison VPC en utilisant son nom DNS par défaut pour AWS la région, par exemple. `rds.us-east-1.amazonaws.com` Pour les AWS régions de Chine (Pékin) et de Chine (Ningxia), vous pouvez effectuer des demandes d'API avec le point de terminaison VPC `rds-api.cn-north-1.amazonaws.com.cn` en utilisant `rds-api.cn-northwest-1.amazonaws.com.cn` et, respectivement.

Pour plus d'informations, consultez Accès à un service via un point de terminaison d'interface dans le Guide de l'utilisateur Amazon VPC.

Création d'une politique de point de terminaison de VPC pour l'API Amazon RDS

Vous pouvez attacher une politique de point de terminaison à votre point de terminaison de VPC qui contrôle l'accès à l'API Amazon RDS. La politique spécifie les informations suivantes :

- Le principal qui peut exécuter des actions.
- Les actions qui peuvent être effectuées.
- Les ressources sur lesquelles les actions peuvent être exécutées.

Pour plus d'informations, veuillez consulter Contrôle de l'accès aux services avec points de terminaison d'un VPC dans le Amazon VPC Guide de l'utilisateur.

Exemple : politique de point de terminaison de VPC pour les actions de l'API Amazon RDS

Voici un exemple de politique de point de terminaison pour l'API Amazon RDS. Lorsqu'elle est attachée à un point de terminaison, cette politique accorde l'accès aux actions de l'API Amazon RDS répertoriées pour tous les principaux sur toutes les ressources.

```
{
   "Statement":[
      {
         "Principal":"*",
         "Effect":"Allow",
         "Action":[
            "rds:CreateDBInstance",
            "rds:ModifyDBInstance",
            "rds:CreateDBSnapshot"
         ],
         "Resource":"*"
      }
   ]
}
```

Exemple : politique de point de terminaison VPC qui refuse tout accès depuis un compte spécifié AWS

La politique de point de terminaison VPC suivante refuse au AWS compte `123456789012` tout accès aux ressources utilisant le point de terminaison. La politique autorise toutes les actions provenant d'autres comptes.

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Resource": "*",
      "Principal": "*"
    },
    {
      "Action": "*",
      "Effect": "Deny",
      "Resource": "*",
      "Principal": { "AWS": [ "123456789012" ] }
     }
```

```
    ]
}
```

Bonnes pratiques de sécurité pour Amazon RDS

Utilisez des comptes AWS Identity and Access Management (IAM) pour contrôler l'accès aux opérations de l'API Amazon RDS, en particulier aux opérations qui créent, modifient ou suppriment des ressources Amazon RDS Aurora. Ces ressources incluent les instances de base de données, les , les groupes de sécurité et les groupes de paramètres. Utilisez également IAM pour contrôler les actions qui exécutent des actions administratives courantes telles que la sauvegarde et la restauration de d'instances de base de données.

- Créez un utilisateur individuel pour chaque personne qui gère les ressources Amazon RDS , y compris vous-même. N'utilisez pas les informations d'identification AWS root pour gérer les ressources Amazon RDS .
- Accordez à chaque utilisateur un ensemble minimum d'autorisations requises pour exécuter ses tâches.
- Utilisez des groupes IAM pour gérer efficacement des autorisations pour plusieurs utilisateurs.
- Effectuer une rotation régulière des informations d'identification IAM.
- Configurez AWS Secrets Manager pour alterner automatiquement les secrets pour Amazon RDS . Pour plus d'informations, consultez Rotation de vos secrets AWS Secrets Manager dans le Guide de l'utilisateur AWS Secrets Manager . Vous pouvez également récupérer les informations d'identification par AWS Secrets Manager programmation. Pour plus d'informations, consultez Récupération de la valeur du secret dans le Guide de l'utilisateur AWS Secrets Manager .

Pour plus d'informations sur la sécurité Amazon RDS et , consultezSécurité dans Amazon RDS. Pour plus d'informations sur IAM, consultez AWS Identity and Access Management. Pour plus d'informations sur les bonnes pratiques IAM, consultez Bonnes pratiques IAM.

AWS Security Hub utilise des contrôles de sécurité pour évaluer les configurations des ressources et les normes de sécurité afin de vous aider à vous conformer aux différents cadres de conformité. Pour plus d'informations sur l'utilisation de Security Hub pour évaluer les ressources RDS, consultez Amazon Relational Database Service controls (Contrôles d'Amazon Relational Database Service) dans le Guide de l'utilisateur AWS Security Hub .

Vous pouvez surveiller votre utilisation de RDS, conformément aux bonnes pratiques de sécurité, avec Security Hub. Pour plus d'informations, voir Qu'est-ce que c'est AWS Security Hub ? .

Utilisez l'API AWS Management Console AWS CLI, la ou l'API RDS pour modifier le mot de passe de votre utilisateur principal. Si vous utilisez un autre outil, comme SQL client, pour modifier le mot de passe de l'utilisateur principal, cela pourrait finir par la révocation involontaire des privilèges de l'utilisateur.

Contrôle d'accès par groupe de sécurité

Par défaut, l'accès au réseau est désactivé pour un d'instances de base de données. Vous pouvez spécifier des règles dans un groupe de sécurité qui autorisent l'accès depuis une plage d'adresses IP, un port ou un groupe de sécurité. Une fois les règles d'entrée configurées, les mêmes règles s'appliquent à tous les d'instances de base de données associés à ce groupe de sécurité. Vous pouvez spécifier jusqu'à 20 règles dans un groupe de sécurité.

Présentation des groupes de sécurité VPC

Chaque règle de groupe de sécurité VPC permet à une source spécifique d'accéder à un d'instances de base de données dans un VPC associé à ce groupe de sécurité VPC. Cette source peut être une plage d'adresses (par exemple, 203.0.113.0/24) ou un autre groupe de sécurité VPC. En spécifiant un groupe de sécurité VPC en tant que source, vous autorisez le trafic entrant provenant de toutes les instances (généralement les serveurs d'application) qui utilisent le groupe de sécurité VPC source. Les groupes de sécurité du VPC peuvent avoir des règles qui régissent à la fois le trafic entrant et sortant. Cependant, les règles de trafic sortant ne s'appliquent généralement pas aux d'instances de base de données. Les règles de trafic sortant s'appliquent uniquement si le d'instances de base de données agit en tant que client. Par exemple, dès règles de trafic sortant s'appliquent à une instance de base de données Oracle DB avec des liens de base de données sortants. Vous devez utiliser l' EC2API Amazon ou l'option Security Group sur la console VPC pour créer des groupes de sécurité VPC.

Lorsque vous créez des règles pour votre groupe de sécurité VPC qui autorisent l'accès aux d'instances de votre VPC, vous devez spécifier un port pour chaque plage d'adresses à laquelle la règle autorise l'accès. Par exemple, si vous souhaitez activer l'accès Secure Shell (SSH) pour les instances du VPC, créez une règle autorisant l'accès au port TCP 22 pour la plage d'adresses spécifiée.

Vous pouvez configurer plusieurs groupes de sécurité VPC qui permettent d'accéder à des ports différents pour différentes instances dans votre VPC. Par exemple, vous pouvez créer un groupe de sécurité VPC qui autorise l'accès au port TCP 80 pour les serveurs Web de votre VPC. Vous

pouvez ensuite créer un autre groupe de sécurité VPC qui autorise l'accès au port TCP 3306 pour les instances de base de données RDS for MySQL Aurora de votre VPC.

Pour plus d'informations sur les groupes de sécurité VPC, consultez Groupes de sécurité dans le Guide de l'utilisateur Amazon Virtual Private Cloud.

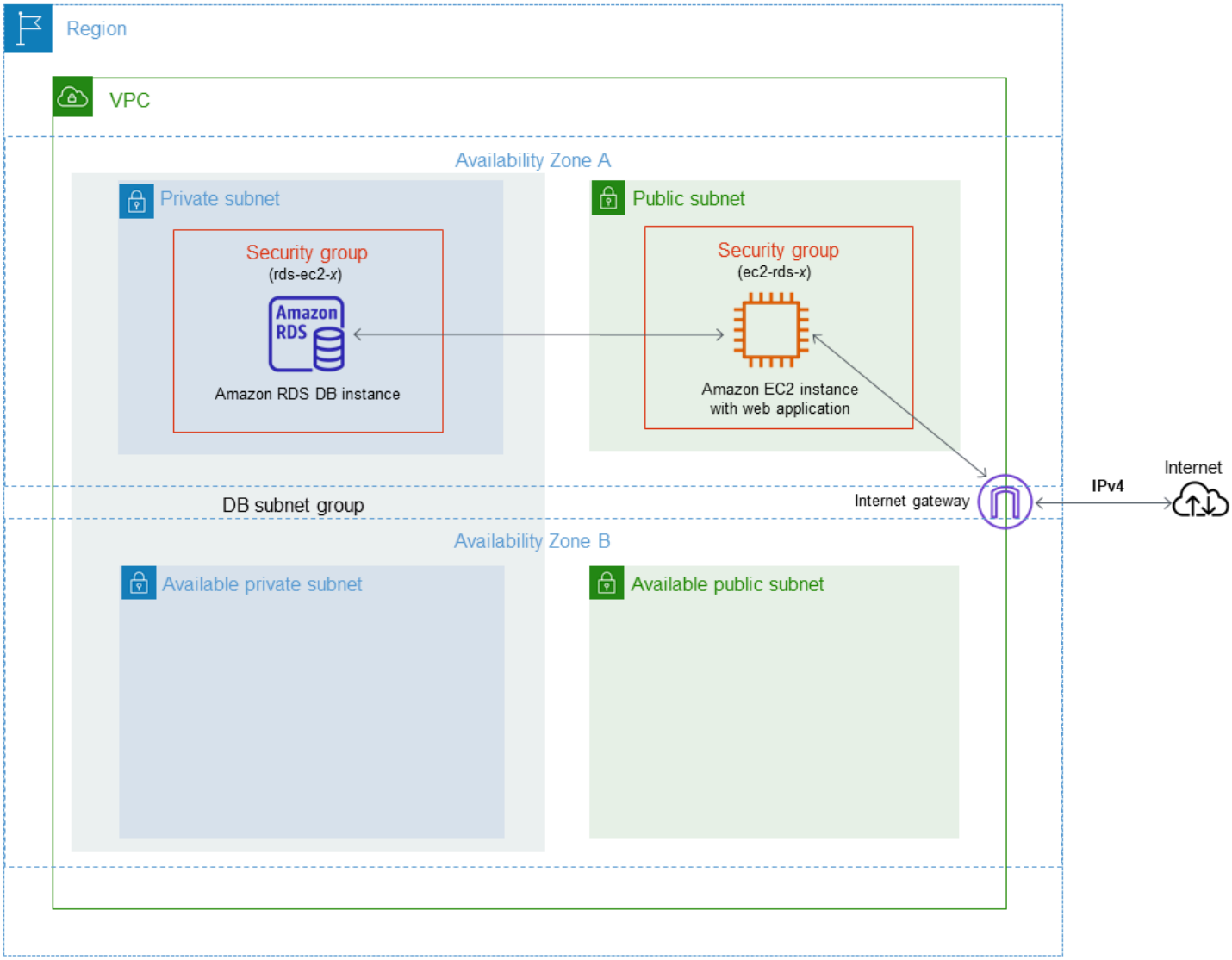
> **Note**
>
> Si votre d'instances de base de données se trouve dans un VPC mais n'est pas accessible au public, vous pouvez également utiliser une connexion AWS Site-to-Site VPN ou une AWS Direct Connect connexion pour y accéder depuis un réseau privé. Pour plus d'informations, consultez Confidentialité du trafic inter-réseau.

Scénario de groupes de sécurité

Une utilisation courante d'un d'instances de base de données dans un VPC consiste à partager des données avec un serveur d'applications exécuté dans une EC2 instance Amazon du même VPC, à laquelle accède une application cliente extérieure au VPC. Dans ce scénario, vous devez utiliser les pages RDS et VPC des opérations RDS EC2 et API pour créer les instances et les groupes de sécurité nécessaires : AWS Management Console

1. Créez un groupe de sécurité VPC (par exemple, `sg-0123ec2example`) et définissez des règles entrantes qui utilisent les adresses IP de l'application cliente comme source. Ce groupe de sécurité permet à votre application cliente de se connecter aux EC2 instances d'un VPC qui utilise ce groupe de sécurité.
2. Créez une EC2 instance pour l'application et ajoutez-la EC2 au groupe de sécurité VPC (`sg-0123ec2example`) que vous avez créé à l'étape précédente.
3. Créez un second groupe de sécurité VPC (par exemple, `sg-6789rdsexample`) et créez une nouvelle règle en spécifiant le groupe de sécurité VPC que vous avez créé à l'étape 1 (`sg-0123ec2example`) en tant que source.
4. Lorsque vous créez le d'instances de base de données, utilisez le même numéro de port que celui spécifié pour la règle du groupe de sécurité VPC (`sg-6789rdsexample`) que vous avez créée à l'étape 3.

Le schéma suivant illustre ce scénario.



Pour obtenir des instructions détaillées sur la configuration d'un VPC pour ce scénario, consultez. Tutoriel : Création d'un VPC à utiliser avec un) IPv4 Pour plus d'informations sur l'utilisation d'un VPC, consultez. Amazon VPC et Amazon RDS

Création d'un groupe de sécurité VPC

Vous pouvez créer un groupe de sécurité VPC pour une instance de base de données à l'aide de la console VPC. Pour plus d'informations sur la création d'un groupe de sécurité, consultez Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC et Groupes de sécurité dans le Guide de l'utilisateur Amazon Virtual Private Cloud.

Association d'un groupe de sécurité à une instance de base de données

Vous pouvez associer un groupe de sécurité à une instance de base de données en utilisant Modify sur la console RDS, l'API `ModifyDBInstance` Amazon RDS ou la `modify-db-instance` AWS CLI commande.

L'exemple de CLI suivant associe un groupe de sécurité VPC spécifique et supprime les groupes de sécurité de base de données de l'instance de base de données.

```
aws rds modify-db-instance --db-instance-identifier dbName --vpc-security-group-ids sg-ID
```

Pour plus d'informations sur la modification d'une instance de base de données, consultezModification d'une RDS instance de base de données Amazon. Pour les considérations relatives aux groupes de sécurité lorsque vous restaurez une instance de base de données à partir d'un instantané de base de données, consultezConsidérations relatives aux groupes de sécurité.

> **Note**
>
> La console RDS affiche différents noms de règles de groupe de sécurité pour votre base de données si la valeur Port est configurée sur une valeur autre que celle par défaut.

Pour les instances de base de données RDS pour Oracle, des groupes de sécurité supplémentaires peuvent être associés en renseignant le paramètre des options de groupe de sécurité pour les options Oracle Enterprise Manager Database Express (OEM), Oracle Management Agent for Enterprise Manager Cloud Control (OEM Agent) et Oracle Secure Sockets Layer. Dans ce cas, les groupes de sécurité associés à l'instance de base de données et les paramètres d'options s'appliquent à l'instance de base de données. Pour plus d'informations sur ces groupes d'options, reportez-vous aux Oracle Management Agent pour Enterprise Manager Cloud Control sectionsOracle Enterprise Manager, etOracle Secure Sockets Layer (SSL).

Privilèges du compte utilisateur principal

Lorsque vous créez un nouveau d'instances de base de données, l'utilisateur principal par défaut que vous utilisez obtient certains privilèges pour ce d'instances de base de données. Vous ne pouvez pas modifier le nom d'utilisateur principal une fois le d'instances de base de données créé.

Important

Nous vous recommandons vivement de ne pas avoir recours au rôle d'utilisateur principal directement dans vos applications. Au lieu de cela, respectez la bonne pratique qui consiste à avoir recours à un utilisateur de base de données doté des privilèges minimum requis pour votre application.

Note

Si vous supprimez accidentellement les autorisations de l'utilisateur principal, vous pouvez les restaurer en modifiant le d'instances de base de données et en définissant un nouveau mot de passe utilisateur principal. Pour plus d'informations sur la modification d'un d'instances de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Le tableau suivant montre les privilèges et les rôles de base de données que l'utilisateur principal obtient pour chacun des moteurs de base de données.

Moteur de base de données	Privilège système	Rôle de base de données
RDS pour Db2	L'utilisateur principal est affecté au `masterdba` groupe et le`master_user_role` . `SYSMON`, `DBADM` avec `DATAACCESS` `ACCCESSCTRL` `ANDBINDADD`,`CONNECT`,`CREATETAB` , `CREATE_SECURE_OBJECT` `EXPLAIN`,`IMPLICIT_SCHEMA` ,`LOAD`,`SQLADM`, `WLMADM`	`DBA`,`DBA_RESTRICTED` , `DEVELOPER` ,`ROLE_NULLID_PACKAGES` , `ROLE_PROCEDURES` ,`ROLE_TABLESPACES` Pour plus d'informations, consultez Rôles par défaut RDS d'Amazon pour DB2.

Moteur de base de données	Privilège système	Rôle de base de données
RDS for MariaDB	`SELECT,INSERT,UPDATE,DELETE, CREATE,DROP,RELOAD, PROCESS,REFERENCES ,INDEX, ALTER,SHOW DATABASES ,CREATE TEMPORARY TABLES ,LOCK TABLES, EXECUTE,REPLICATION CLIENT,CREATE VIEW ,SHOW VIEW,CREATE ROUTINE, ALTER ROUTINE,CREATE USER, EVENT,TRIGGER,REPLICATION SLAVE` À partir de la version 11.4 de RDS pour MariaDB, l'utilisateur principal obtient également le privilège. `SHOW CREATE ROUTINE`	—
RDS pour MySQL 8.0.36 et versions ultérieures	`SELECT,INSERT,UPDATE, DELETE,CREATE,DROP, RELOAD,PROCESS, REFERENCES ,INDEX,ALTER, SHOW DATABASES ,CREATE TEMPORARY TABLES,LOCK TABLES,EXECUTE, REPLICATION SLAVE ,REPLICATION CLIENT , CREATE VIEW,SHOW VIEW,CREATE ROUTINE,ALTER ROUTINE,CREATE USER,EVENT,TRIGGER, CREATE ROLE,DROP ROLE, APPLICATION_PASSWORD_ADMIN , ROLE_ADMIN ,SET_USER_ID , XA_RECOVER_ADMIN`	`rds_superuser_role` Pour plus d'informations sur `rds_superuser_role` , voir Modèle de privilèges basé sur les rôles pour for My RDS SQL.
RDS pour les versions de MySQL inférieures à 8.0.36	`SELECT,INSERT,UPDATE, DELETE,CREATE,DROP, RELOAD,PROCESS, REFERENCES ,INDEX,ALTER, SHOW DATABASES ,CREATE TEMPORARY TABLES,LOCK TABLES,EXECUTE, REPLICATION CLIENT ,CREATE VIEW, SHOW VIEW,CREATE ROUTINE,ALTER ROUTINE,CREATE USER,EVENT, TRIGGER,REPLICATION SLAVE`	—

Moteur de base de données	Privilège système	Rôle de base de données
RDS for PostgreSQL	`CREATE ROLE,CREATE DB, PASSWORD VALID UNTIL INFINITY,CREATE EXTENSION ,ALTER EXTENSION ,DROP EXTENSION ,CREATE TABLESPACE ,ALTER <OBJECT> OWNER,CHECKPOINT , PG_CANCEL_BACKEND( ) , PG_TERMINATE_BACKEND() ,SELECT PG_STAT_REPLICATION ,EXECUTE PG_STAT_STATEMENTS_RESET() ,OWN POSTGRES_ FDW_HANDLER() ,OWN POSTGRES_FDW_VALIDATOR() ,OWN POSTGRES_FDW , EXECUTE PG_BUFFERCACHE_PAGES() ,SELECT PG_BUFFERCACHE`	`RDS_SUPERUSER` Pour plus d'informations sur RDS_SUPERUSER, consultez . Comprendre les rôles et les autorisations PostgreSQL

Moteur de base de données	Privilège système	Rôle de base de données
RDS for Oracle	ADMINISTER DATABASE TRIGGER ,ALTER DATABASE LINK,ALTER PUBLIC DATABASE LINK, AUDIT SYSTEM,CHANGE NOTIFICATION , DROP ANY DIRECTORY ,EXEMPT ACCESS POLICY,EXEMPT IDENTITY POLICY,EXEMPT REDACTION POLICY,FLASHBACK ANY TABLE, GRANT ANY OBJECT PRIVILEGE ,RESTRICTED SESSION ,SELECT ANY TABLE,UNLIMITED TABLESPACE	DBA **Note** Le DBA rôle est exempté des privilèges suivants : ALTER DATABASE,ALTER SYSTEM, CREATE ANY DIRECTORY ,CREATE EXTERNAL JOB,CREATE PLUGGABLE DATABASE, GRANT ANY PRIVILEGE ,GRANT ANY ROLE,READ ANY FILE GROUP
Amazon RDS for Microsoft SQL Server	ADMINISTER BULK OPERATIONS ,ALTER ANY CONNECTION ,ALTER ANY CREDENTIAL , ALTER ANY EVENT SESSION,ALTER ANY LINKED SERVER,ALTER ANY LOGIN,ALTER ANY SERVER AUDIT,ALTER ANY SERVER ROLE, ALTER SERVER STATE,ALTER TRACE, CONNECT SQL,CREATE ANY DATABASE, VIEW ANY DATABASE,VIEW ANY DEFINITION ,VIEW SERVER STATE,ALTER ON ROLE SQLAgentOperatorRole	DB_OWNER (rôle au niveau de la base de données), PROCESSADMIN (rôle au niveau du serveur), SETUPADMIN (rôle au niveau du serveur),SQLAgentUserRole (rôle au niveau de la base de données)

Utilisation des rôles liés à un service pour Amazon RDS

Amazon RDS utilise des rôles liés à un service AWS Identity and Access Management (IAM). Un rôle lié à un service est un type unique de rôle IAM lié directement à Amazon RDS. Les rôles liés à un service sont prédéfinis par Amazon RDS () et incluent toutes les autorisations requises par le service pour appeler d'autres AWS services en votre nom.

Un rôle lié à un service simplifie l'utilisation d'Amazon RDS, car vous n'avez pas besoin d'ajouter manuellement les autorisations requises. Amazon RDS définit les autorisations de ses rôles liés à un service et, sauf définition contraire, seul Amazon RDS peut endosser ses rôles. Les autorisations définies comprennent la politique d'approbation et la politique d'autorisation. De plus, cette politique d'autorisation ne peut pas être attachée à une autre entité IAM.

Vous pouvez supprimer les rôles uniquement après la suppression préalable de leurs ressources connexes. Vos ressources Amazon RDS sont ainsi protégées, car vous ne pouvez pas involontairement supprimer l'autorisation d'accéder aux ressources.

Pour plus d'informations sur les autres services qui prennent en charge les rôles liés à un service, consultez AWS services that work with IAM (Services AWS qui fonctionnent avec IAM) et recherchez les services avec un Yes (Oui) dans la colonne Service-Linked Role (Rôle lié à un service). Choisissez un Yes (oui) ayant un lien permettant de consulter les détails du rôle pour ce service.

Autorisations des rôles liés à un service pour Amazon RDS

Amazon RDS

Le rôle lié à un service AWSService RoleFor RDS fait confiance aux services suivants pour assumer le rôle :

- `rds.amazonaws.com`

Ce rôle lié à un service est associé à une politique appelée `AmazonRDSServiceRolePolicy` qui lui accorde l'autorisation d'opérer dans votre compte.

Pour plus d'informations sur cette politique, y compris le document de politique JSON, consultez Amazon RDSService RolePolicy dans le AWS Managed Policy Reference Guide.

Note

Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, groupe ou rôle) de créer, modifier ou supprimer un rôle lié à un service. Si vous rencontrez le message d'erreur suivant :
Impossible de créer la ressource. Vérifiez que vous détenez l'autorisation de créer un rôle lié au service. Dans le cas contraire, attendez et réessayez ultérieurement.
Vérifiez que les autorisations suivantes sont activées :

```
{
    "Action": "iam:CreateServiceLinkedRole",
    "Effect": "Allow",
    "Resource": "arn:aws:iam::*:role/aws-service-role/rds.amazonaws.com/
AWSServiceRoleForRDS",
    "Condition": {
        "StringLike": {
            "iam:AWSServiceName":"rds.amazonaws.com"
        }
    }
}
```

Pour plus d'informations, veuillez consulter Autorisations de rôles liés à un service dans le IAM Guide de l'utilisateur.

Création d'un rôle lié à un service pour Amazon RDS

Vous n'avez pas besoin de créer manuellement un rôle lié à un service. Lorsque vous créez une instance de base de données, Amazon RDS crée le rôle lié à un service pour vous.

Important

Si vous utilisiez le service Amazon RDS avant le 1er décembre 2017, date à laquelle il a commencé à prendre en charge les rôles liés au service, Amazon RDS Amazon a créé AWSService RoleFor le rôle RDS dans votre compte. Pour en savoir plus, consultez l'article Un nouveau rôle est apparu dans mon AWS compte.

Si vous supprimez ce rôle lié à un service et que vous avez ensuite besoin de le recréer, vous pouvez utiliser la même procédure pour recréer le rôle dans votre compte. Lorsque vous créez une instance de base de données, Amazon RDS crée de nouveau le rôle lié à un service pour vous.

Modification d'un rôle lié à un service pour Amazon RDS

Amazon RDS ne vous permet pas de modifier le rôle lié au service AWSService RoleFor RDS. Une fois que vous avez créé un rôle lié à un service, vous ne pouvez pas changer le nom du rôle, car plusieurs entités peuvent faire référence à ce rôle. Néanmoins, vous pouvez modifier la description du rôle à l'aide d'IAM. Pour plus d'informations, consultez Modification d'un rôle lié à un service dans le IAM Guide de l'utilisateur.

Suppression d'un rôle lié à un service pour Amazon RDS

Si vous n'avez plus besoin d'utiliser une fonction ou un service qui nécessite un rôle lié à un service, nous vous recommandons de supprimer ce rôle. De cette façon, vous n'avez aucune entité inutilisée qui n'est pas surveillée ou gérée activement. Cependant, vous devez supprimer toutes vos instances et de bases de données avant de pouvoir supprimer le rôle lié à un service.

Nettoyage d'un rôle lié à un service

Avant de pouvoir utiliser IAM pour supprimer un rôle lié à un service, vous devez d'abord vérifier qu'aucune session n'est active pour le rôle et supprimer toutes les ressources utilisées par le rôle.

Pour vérifier si une session est active pour le rôle lié à un service dans la console IAM

1. Connectez-vous à la console IAM AWS Management Console et ouvrez-la à https://console.aws.amazon.com/iam/l'adresse.
2. Dans le panneau de navigation de la console IAM, sélectionnez Roles (Rôles). Choisissez ensuite le nom (et non la case à cocher) du rôle AWSService RoleFor RDS.
3. Sur la page Résumé du rôle choisi, choisissez l'onglet Dernier accès.
4. Dans l'onglet Dernier accès, passez en revue l'activité récente du rôle lié au service.

> **Note**
>
> Si vous ne savez pas si Amazon RDS utilise le rôle AWSService RoleFor RDS, vous pouvez essayer de le supprimer. Si le service utilise le rôle, la suppression échoue et vous avez accès aux régions AWS dans lesquelles le rôle est utilisé. Si le rôle est utilisé,

vous devez attendre que la session se termine avant de pouvoir le supprimer. Vous ne pouvez pas révoquer la session d'un rôle lié à un service.

Si vous souhaitez supprimer le rôle AWSService RoleFor RDS, vous devez d'abord supprimer tous vos d'instances de base de données.

Suppression de toutes vos instances

Utilisez l'une des procédures suivantes pour supprimer chacune de vos instances.

Pour supprimer une instance (console)

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans la panneau de navigation, choisissez Databases (Bases de données).
3. Choisissez l'instance que vous voulez supprimer.
4. Pour Actions, choisissez Supprimer.
5. Si vous on vous demande de Créer un aperçu final ?, sélectionnez Oui ou Non.
6. Si vous avez choisi Oui à l'étape précédente, dans le champ Nom de l'instantané final, saisissez le nom de votre instantané final.
7. Sélectionnez Delete.

Pour supprimer une instance (CLI)

Consultez `delete-db-instance` dans la Référence de commande AWS CLI .

Pour supprimer une instance (API)

Voir `DeleteDBInstance` dans le Amazon RDS API Reference.

Vous pouvez utiliser la console IAM, la CLI IAM ou l'API IAM pour supprimer le rôle lié au service AWSService RoleFor RDS. Pour plus d'informations, consultez Suppression d'un rôle lié à un service dans le Guide de l'utilisateur IAM.

Autorisations du rôle lié à un service pour Amazon RDS Custom

Amazon RDS Custom utilise le rôle lié au service nommé `AWSServiceRoleForRDSCustom` pour permettre à RDS Custom d'appeler des AWS services au nom de vos ressources de base de données RDS.

Le rôle AWSService RoleFor RDSCustom lié à un service fait confiance aux services suivants pour assumer le rôle :

- `custom.rds.amazonaws.com`

Ce rôle lié à un service est associé à une politique appelée `AmazonRDSCustomServiceRolePolicy` qui lui accorde l'autorisation d'opérer dans votre compte.

La création, la modification ou la suppression du rôle lié au service pour RDS Custom fonctionne de la même manière que pour Amazon RDS. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSCustom ServiceRolePolicy.

> **Note**
>
> Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Si vous rencontrez le message d'erreur suivant :
> Impossible de créer la ressource. Vérifiez que vous détenez l'autorisation de créer un rôle lié au service. Dans le cas contraire, attendez et réessayez ultérieurement.
> Vérifiez que les autorisations suivantes sont activées :
>
> ```
> {
>     "Action": "iam:CreateServiceLinkedRole",
>     "Effect": "Allow",
>     "Resource": "arn:aws:iam::*:role/aws-service-role/custom.rds.amazonaws.com/
> AmazonRDSCustomServiceRolePolicy",
>     "Condition": {
>         "StringLike": {
>             "iam:AWSServiceName":"custom.rds.amazonaws.com"
>         }
>     }
> }
> ```
>
> Pour plus d'informations, veuillez consulter Autorisations de rôles liés à un service dans le IAM Guide de l'utilisateur.

Autorisations de rôle liées à un service pour Amazon RDS Beta

Amazon RDS utilise le rôle lié au service nommé pour `AWSServiceRoleForRDSBeta` permettre à Amazon RDS Amazon d' AWS appeler des services au nom de vos ressources de base de données RDS.

Le rôle AWSService RoleFor RDSBeta lié à un service fait confiance aux services suivants pour assumer le rôle :

- `rds.amazonaws.com`

Ce rôle lié à un service est associé à une politique appelée `AmazonRDSBetaServiceRolePolicy` qui lui accorde l'autorisation d'opérer dans votre compte. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSBeta ServiceRolePolicy.

> **Note**
>
> Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Si vous rencontrez le message d'erreur suivant :
> Impossible de créer la ressource. Vérifiez que vous détenez l'autorisation de créer un rôle lié au service. Dans le cas contraire, attendez et réessayez ultérieurement.
> Vérifiez que les autorisations suivantes sont activées :
>
> ```
> {
>     "Action": "iam:CreateServiceLinkedRole",
>     "Effect": "Allow",
>     "Resource": "arn:aws:iam::*:role/aws-service-role/custom.rds.amazonaws.com/
> AmazonRDSBetaServiceRolePolicy",
>     "Condition": {
>         "StringLike": {
>             "iam:AWSServiceName":"custom.rds.amazonaws.com"
>         }
>     }
> }
> ```
>
> Pour plus d'informations, veuillez consulter Autorisations de rôles liés à un service dans le IAM Guide de l'utilisateur.

Rôle lié à un service pour Amazon RDS Preview

Amazon RDS utilise le rôle lié au service nommé pour `AWSServiceRoleForRDSPreview` permettre à Amazon RDS Amazon d' AWS appeler des services au nom de vos ressources de base de données RDS.

Le rôle AWSService RoleFor RDSPreview lié à un service fait confiance aux services suivants pour assumer le rôle :

- `rds.amazonaws.com`

Ce rôle lié à un service est associé à une politique appelée `AmazonRDSPreviewServiceRolePolicy` qui lui accorde l'autorisation d'opérer dans votre compte. Pour de plus amples informations, veuillez consulter AWS politique gérée : Amazon RDSPreview ServiceRolePolicy.

> **Note**
>
> Vous devez configurer les autorisations de manière à permettre à une entité IAM (comme un utilisateur, un groupe ou un rôle) de créer, modifier ou supprimer un rôle lié à un service. Si vous rencontrez le message d'erreur suivant :
> Impossible de créer la ressource. Vérifiez que vous détenez l'autorisation de créer un rôle lié au service. Dans le cas contraire, attendez et réessayez ultérieurement.
> Vérifiez que les autorisations suivantes sont activées :
>
> ```
> {
>     "Action": "iam:CreateServiceLinkedRole",
>     "Effect": "Allow",
>     "Resource": "arn:aws:iam::*:role/aws-service-role/custom.rds.amazonaws.com/
> AmazonRDSPreviewServiceRolePolicy",
>     "Condition": {
>         "StringLike": {
>             "iam:AWSServiceName":"custom.rds.amazonaws.com"
>         }
>     }
> }
> ```

Pour plus d'informations, veuillez consulter Autorisations de rôles liés à un service dans le IAM Guide de l'utilisateur.

Amazon VPC et Amazon RDS

Amazon Virtual Private Cloud (AmazonVPC) vous permet de lancer AWS des ressources, telles que des instances Amazon RDS DB, dans un cloud privé virtuel (VPC).

Lorsque vous utilisez unVPC, vous contrôlez votre environnement réseau virtuel. Vous pouvez choisir votre propre plage d'adresses IP, créer des sous-réseaux et configurer le routage et les listes de contrôle d'accès. Il n'y a aucun coût supplémentaire pour exécuter votre d'instances de base de données dans unVPC.

Les comptes ont une valeur par défautVPC. Tous les nouveaux d'instances de base de données sont créés par défaut, VPC sauf indication contraire de votre part.

Rubriques

- Utilisation d'un(e) instance de base de données dans un VPC
- Mettre à jour le VPC pour une instance de base de données
- Scénarios d'accès à un d'instances de base de données dans un VPC
- Tutoriel : Création d'un VPC à utiliser avec un) IPv4
- Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile)
- Déplacement vers un VPC d'une instance de base de données n'appartenant pas à un VPC

Vous trouverez ci-dessous une discussion sur les VPC fonctionnalités relatives aux instances de base de RDS données Amazon et aux . Pour plus d'informations sur AmazonVPC, consultez le guide de VPCdémarrage Amazon et le guide de VPCl'utilisateur Amazon.

Utilisation d'un(e) instance de base de données dans un VPC

Votre instance de base de données se trouve dans un cloud privé virtuel (VPC). Un VPC est un réseau virtuel isolé logiquement des autres réseaux virtuels du cloud. AWS Amazon VPC vous permet de lancer des AWS ressources, telles qu'un cluster d'instances de base de données Amazon RDS une EC2 instance Amazon, dans un VPC. Le VPC peut être un VPC par défaut fourni avec votre compte ou un VPC que vous créez. Ils VPCs sont tous associés à votre AWS compte.

Votre VPC par défaut a trois sous-réseaux que vous pouvez utiliser pour isoler les ressources à l'intérieur du VPC. Le VPC par défaut possède aussi une passerelle Internet qui peut être utilisée pour fournir l'accès aux ressources à l'intérieur du VPC depuis l'extérieur du VPC.

Pour obtenir une liste des scénarios impliquant des instances de base de données Amazon RDS dans un VPC et en dehors d'un VPC, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.

Rubriques

- Utilisation d'un(e) instance de base de données dans un VPC
- Utilisation de groupes de sous-réseaux DB
- Sous-réseaux partagés
- Adressage IP Amazon RDS
- Masquer un(e) instance de base de données dans un VPC depuis Internet
- Création d'un(e) instance de base de données dans un VPC

Dans les tutoriels suivants, vous apprendrez à créer un VPC que vous pouvez utiliser pour un scénario commun Amazon RDS :

- Tutoriel : Création d'un VPC à utiliser avec un) IPv4
- Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile)

Utilisation d'un(e) instance de base de données dans un VPC

Voici quelques conseils d'utilisation d'un(e) instance de base de données dans un VPC :

- Votre VPC doit avoir au moins deux sous-réseaux. Ces sous-réseaux doivent se trouver dans deux zones de disponibilité différentes dans l' Région AWS endroit où vous souhaitez déployer votre d'instances de base de données. Un sous-réseau est un segment de la plage d'adresses IP d'un VPC que vous pouvez spécifier et que vous pouvez utiliser pour regrouper des instances de base de données en fonction de vos besoins en matière de sécurité et de fonctionnement.

  Pour les déploiements multi-AZ, la définition d'un sous-réseau pour deux zones de disponibilité ou plus dans un permet à Région AWS Amazon RDS de créer un nouveau serveur de secours dans une autre zone de disponibilité selon les besoins. Procédez de la sorte même pour les déploiements Single-AZ, au cas où vous souhaitiez les convertir en déploiements multi-AZ par la suite.

> **Note**
>
> Le groupe de sous-réseaux de base de données d'une zone locale ne peut avoir qu'un seul sous-réseau.

- Si vous voulez que votre instance de base de données dans le VPC soit publiquement accessible, assurez-vous d'activer les attributs VPC DNS hostnames (Noms d'hôtes DNS) et DNS resolution (Résolution DNS).
- Votre VPC doit disposer d'un groupe de sous-réseau de base de données que vous créez. Vous créez un groupe de sous-réseaux de base de données en spécifiant les sous-réseaux que vous avez créés. Amazon RDS choisit dans ce groupe de sous-réseaux un sous-réseau et une adresse IP à associer à votre instance de base de données. L'instance de base de données utilise la zone de disponibilité contenant le sous-réseau.
- Votre VPC doit avoir un groupe de sécurité VPC qui autorise l'accès à l'instance de base de données.

  Pour de plus amples informations, veuillez consulter Scénarios d'accès à un d'instances de base de données dans un VPC.
- Les blocs d'adresse CIDR de chacun de vos sous-réseaux doivent être assez grands pour accueillir les adresses IP de rechange utilisées par Amazon RDS pendant les activités de maintenance, y compris le basculement et le dimensionnement du calcul. Par exemple, une plage telle que 10.0.0.0/24 et 10.0.1.0/24 est généralement suffisante.
- Un VPC peut avoir un attribut instance tenancy (location d'instance) ayant la valeur par défaut ou dédiée. Tous les paramètres par défaut VPCs ont l'attribut de location d'instance défini sur défaut, et un VPC par défaut peut prendre en charge n'importe quelle classe d'instance de base de données.

  Si vous choisissez de placer votre d'instance de base de données dans un VPC dédié où l'attribut de location d'instance est défini sur dedicated, la classe d'instance de votre d'instance de base de données doit être l'un des types d'instance EC2 dédiés Amazon approuvés. Par exemple, l'instance EC2 dédiée r5.large correspond à la classe d'instance de base de données db.r5.large. Pour plus d'informations sur la location d'instance dans un VPC, consultez Instances dédiées dans le Guide de l'utilisateur Amazon Elastic Compute Cloud.

  Pour plus d'informations sur les types d'instances qui peuvent figurer dans une instance dédiée, consultez la section Instances EC2 dédiées Amazon sur la page de EC2 tarification Amazon.

> **Note**
>
> Lorsque vous définissez l'attribut de location d'instance sur dédié pour un(e) instance de base de données, cela ne garantit pas que l'instance de base de données fonctionnera sur un hôte dédié.

- Lorsqu'un groupe d'options est attribué à une instance de base de données, il est associé au VPC de l'instance de base de données. Cette liaison signifie que vous ne pouvez pas utiliser le groupe d'options assigné à une instance de base de données si vous tentez de restaurer l'instance de base de données dans un VPC différent.
- Si vous restaurez une instance de base de données dans un VPC différent, veillez à affecter le groupe d'options par défaut à l'instance de base de données, à affecter un groupe d'options lié à ce VPC ou à créer un nouveau groupe d'options et à l'affecter à l'instance de base de données. Avec les options permanentes ou persistantes, telles qu'Oracle TDE, vous devez créer un nouveau groupe d'options incluant l'option persistante ou permanente lorsque vous restaurez une instance de base de données dans un VPC différent.

Utilisation de groupes de sous-réseaux DB

Les sous-réseaux sont des segments d'une plage d'adresses IP d'un VPC que vous définissez pour regrouper vos ressources en fonction de vos besoins de sécurité et de fonctionnement. Un groupe de sous-réseaux de base de données est une collection de sous-réseaux (généralement privés) que vous créez dans un VPC et que vous spécifiez alors pour vos instances de base de données. En utilisant un groupe de sous-réseaux de base de données, vous pouvez spécifier un VPC particulier lors de la création de clusters d' de base de données à l'aide de l'API ou AWS CLI RDS. Si vous utilisez la console, vous pouvez choisir le VPC et les groupes de sous-réseaux que vous voulez utiliser.

Chaque groupe de sous-réseaux DB doit avoir des sous-réseaux dans au moins deux zones de disponibilité d'une Région AWS donnée. Lorsque vous créez un(e) instance de base de données dans un VPC, vous choisissez un groupe de sous-réseau de base de données pour celui-ci. Dans le groupe de sous-réseaux de base de données, Amazon RDS choisit un sous-réseau et une adresse IP dans ce sous-réseau pour les employer avec l'instance de base de données . La base de données utilise la zone de disponibilité contenant le sous-réseau. Amazon RDS attribue toujours une adresse IP à partir d'un sous-réseau disposant d'un espace d'adressage IP libre.

Si l'instance de base de données principale d'un déploiement multi-AZ échoue, Amazon RDS peut promouvoir l'instance de secours et par la suite créer une nouvelle instance de secours à l'aide d'une adresse IP du sous-réseau dans l'une des autres zones de disponibilité.

Les sous-réseaux d'un groupe de sous-réseaux de base de données sont publics ou privés. Les sous-réseaux sont publics ou privés, selon la configuration que vous définissez pour leurs listes de contrôle d'accès réseau (réseau ACLs) et leurs tables de routage. Pour qu'un(e) instance de base de données soit accessible au public, tous les sous-réseaux de son groupe de sous-réseaux de base de données doivent être publics. Si un sous-réseau associé à un(e) instance de base de données accessible au public passe de public à privé, cela peut affecter la disponibilité de l'instance de base de données.

Pour créer un groupe de sous-réseaux de base de données prenant en charge le mode double pile, assurez-vous qu'un bloc CIDR du protocole Internet version 6 (IPv6) est associé à chaque sous-réseau que vous ajoutez au groupe de sous-réseaux de base de données. Pour plus d'informations, consultez Adressage IP Amazon RDS la section « Migration vers » IPv6 dans le guide de l'utilisateur Amazon VPC.

> **Note**
>
> Le groupe de sous-réseaux de base de données d'une zone locale ne peut avoir qu'un seul sous-réseau.

Lorsque Amazon RDS crée un(e) instance de base de données dans un VPC, il attribue une interface réseau à votre instance de base de données en utilisant une adresse IP de votre groupe de sous-réseau de base de données. Toutefois, nous vous recommandons vivement d'utiliser le nom du système de nom de domaine (DNS) pour vous connecter à votre instance de base de données. Nous le recommandons car l'adresse IP sous-jacente change pendant le basculement.

> **Note**
>
> Pour chaque instance de base de données que vous exécutez dans un VPC, assurez-vous de réserver au moins une adresse dans chaque sous-réseau du groupe de sous-réseaux de base de données qui sera utilisée par Amazon RDS pour les actions de récupération.

Sous-réseaux partagés

Vous pouvez créer un cluster de bases de données dans un VPC partagé.

Voici quelques points à prendre en compte lors de l'utilisation du partage VPCs :

- Vous pouvez déplacer une instance de base de données d'un sous-réseau VPC partagé vers un sous-réseau VPC non partagé et vice-versa.
- Les participants à un VPC partagé doivent créer un groupe de sécurité dans le VPC pour pouvoir créer une instance de base de données.
- Les propriétaires et les participants d'un VPC partagé peuvent accéder à la base de données à l'aide de requêtes SQL. Toutefois, seul le créateur d'une ressource peut effectuer des appels d'API sur cette ressource.

Adressage IP Amazon RDS

Les adresses IP permettent aux ressources de votre VPC de communiquer entre elles et avec les ressources sur Internet. Amazon RDS prend en charge les deux protocoles ainsi que les protocoles IPv4 d' IPv6 adressage. Par défaut, Amazon RDS, et Amazon VPC utilisent IPv4 le protocole d'adressage. Vous ne pouvez pas désactiver ce comportement. Lorsque vous créez un VPC, assurez-vous de spécifier un bloc IPv4 CIDR (une plage d'adresses privées IPv4). Vous pouvez éventuellement attribuer un bloc IPv6 CIDR à votre VPC et à vos sous-réseaux, et IPv6 attribuer les adresses de ce bloc aux clusters d' de base de données de votre sous-réseau.

Support du IPv6 protocole augmente le nombre d'adresses IP prises en charge. En utilisant le IPv6 protocole, vous vous assurez de disposer de suffisamment d'adresses disponibles pour la future croissance d'Internet. Les ressources et IPv6 adresses RDS nouvelles et existantes peuvent être utilisées IPv4 au sein de votre VPC. La configuration, la sécurisation et la traduction du trafic réseau entre les deux protocoles utilisés dans les différentes parties d'une application peuvent entraîner une surcharge opérationnelle. Vous pouvez standardiser le IPv6 protocole des ressources Amazon RDS afin de simplifier la configuration de votre réseau.

Rubriques

- IPv4 adresses
- IPv6 adresses
- Mode double pile

IPv4 adresses

Lorsque vous créez un VPC, vous devez spécifier une plage d' IPv4 adresses pour le VPC sous la forme d'un bloc CIDR, tel que. `10.0.0.0/16` Un groupe de sous-réseau de base de données définit la plage d'adresses IP de ce bloc CIDR qu'un(e) instance de base de données peut utiliser. Ces adresses IP peuvent être privées ou publiques.

Une IPv4 adresse privée est une adresse IP qui n'est pas accessible via Internet. Vous pouvez utiliser IPv4 des adresses privées pour la communication entre votre d'instances de base de données et d'autres ressources, telles que EC2 les instances Amazon, dans le même VPC. Chaque instance de base de données dispose d'une adresse IP privée pour la communication dans le VPC.

Une adresse IP publique est une IPv4 adresse accessible depuis Internet. Vous pouvez utiliser des adresses publiques pour la communication entre votre instance de base de données et des ressources sur Internet, comme un client SQL. Vous contrôlez si votre instance de base de données reçoit une adresse IP publique.

Pour un didacticiel expliquant comment créer un VPC avec uniquement des IPv4 adresses privées que vous pouvez utiliser pour un scénario Amazon RDS courant, consultez. Tutoriel : Création d'un VPC à utiliser avec un) IPv4

IPv6 adresses

Vous pouvez éventuellement associer un bloc IPv6 CIDR à votre VPC et à vos sous-réseaux, et IPv6 attribuer des adresses de ce bloc aux ressources de votre VPC. Chaque IPv6 adresse est unique au monde.

Le bloc IPv6 CIDR pour votre VPC est automatiquement attribué à partir du pool IPv6 d'adresses d'Amazon. Vous ne pouvez pas choisir la plage vous-même.

Lorsque vous vous connectez à une IPv6 adresse, assurez-vous que les conditions suivantes sont remplies :

- Le client est configuré de telle sorte que le trafic entre le client et la base de données IPv6 soit autorisé.
- Les groupes de sécurité RDS utilisés par l'instance de base de données sont correctement configurés afin que le trafic client-base de données IPv6 soit autorisé.
- La pile du système d'exploitation client autorise le trafic sur l' IPv6 adresse, et les pilotes et bibliothèques du système d'exploitation sont configurés pour choisir le point de terminaison d'instance de base de données par défaut correct (IPv4 ou non IPv6).

Pour plus d'informations IPv6, consultez la section Adressage IP dans le guide de l'utilisateur Amazon VPC.

Mode double pile

Lorsqu'un d'instances de base de données peut communiquer à la fois via les protocoles IPv4 et d'IPv6 adressage, il s'exécute en mode double pile. Ainsi, les ressources peuvent communiquer avec le d'instances de base de IPv4 données IPv6 via ou les deux. RDS désactive l'accès à Internet Gateway pour les IPv6 points de terminaison des instances de base de données privées en mode double pile. RDS fait cela pour garantir que vos IPv6 points de terminaison sont privés et ne sont accessibles que depuis votre VPC.

Rubriques

- Mode double pile et groupes de sous-réseaux de base de données
- Utilisation d'instances de base de données en mode double pile
- Modification de IPv4 d'instances de base de données réservés uniquement pour utiliser le mode double pile
- Disponibilité des régions et des versions
- Limitations pour les instances de base de données en réseau à double pile

Pour un didacticiel expliquant comment créer un VPC à la fois avec IPv4 des IPv6 adresses que vous pouvez utiliser pour un scénario Amazon RDS courant, consultez. Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile)

Mode double pile et groupes de sous-réseaux de base de données

Pour utiliser le mode double pile, assurez-vous qu'un bloc IPv6 CIDR est associé à chaque sous-réseau du groupe de sous-réseaux de base de données que vous associez au d'instances de base de données. Vous pouvez créer un nouveau groupe de sous-réseau de base de données ou modifier un groupe de sous-réseau de base de données existant pour répondre à cette exigence. Une fois qu'une instance de base de données est en mode double pile, les clients peuvent s'y connecter normalement. Assurez-vous que les pare-feux de sécurité du client et les groupes de sécurité des instances de base de données RDS sont correctement configurés pour autoriser le transfert du trafic. IPv6 Pour se connecter, les clients utilisent le point de terminaison de l'instance de base de données. Les applications client peuvent spécifier quel protocole est préféré lors de la connexion à une base de données. En mode double pile, le d'instances de base de données détecte le protocole réseau préféré du client, soit IPv6, IPv4 soit, et utilise ce protocole pour la connexion.

Si un groupe de sous-réseaux de base de données cesse de prendre en charge le mode double pile en raison de la suppression d'un sous-réseau ou d'une dissociation CIDR, il existe un risque d'incompatibilité de l'état du réseau pour les instances de base de données associées au groupe de sous-réseaux de base de données. De même, vous ne pouvez pas utiliser le groupe de sous-réseau de base de données lorsque vous créez une instance de base de données en mode double pile.

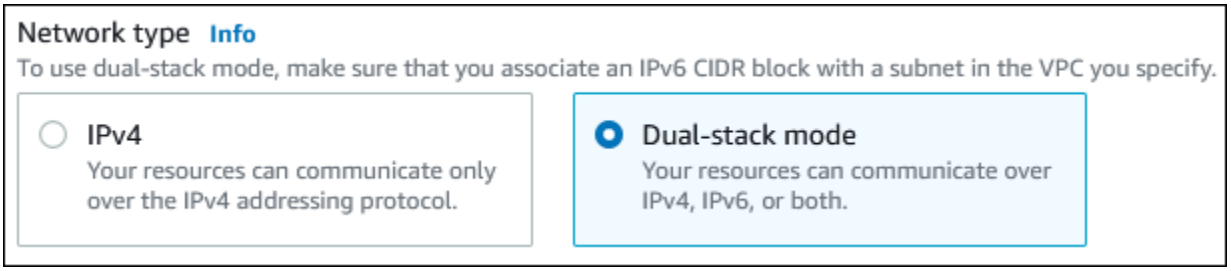
Pour déterminer si un groupe de sous-réseaux de base de données prend en charge le mode double pile à l'aide du AWS Management Console, consultez le type de réseau sur la page de détails du groupe de sous-réseaux de base de données. Pour déterminer si un groupe de sous-réseaux de base de données prend en charge le mode double pile à l'aide de AWS CLI, exécutez la describe-db-subnet-groupscommande et visualisez `SupportedNetworkTypes` la sortie.

Les réplicas en lecture sont traités comme des instances de base de données indépendantes et peuvent avoir un type de réseau différent de celui de l'instance de base de données principale. Si vous modifiez le type de réseau de l'instance de base de données principale d'un réplica en lecture, le réplica en lecture n'est pas affecté. Lorsque vous restaurez une instance de base de données, vous pouvez la restaurer sur tout type de réseau pris en charge.

Utilisation d'instances de base de données en mode double pile

Lorsque vous créez ou modifiez un d'instances de base de données, vous pouvez spécifier le mode double pile pour permettre à vos ressources de communiquer avec votre d'instances de base de données IPv4 via ou IPv6 les deux.

Lorsque vous utilisez le AWS Management Console pour créer ou modifier une instance de base de données, vous pouvez spécifier le mode double pile dans la section Type de réseau. L'image suivante présente la section Network type (Type de réseau) dans la console.



Lorsque vous utilisez le AWS CLI pour créer ou modifier un d'instances de base de données, définissez l'`--network-type`option `DUAL` pour utiliser le mode double pile. Lorsque vous utilisez l'API RDS pour créer ou modifier une instance de base de données, définissez le paramètre `NetworkType` sur `DUAL` pour utiliser le mode double pile. Lorsque vous modifiez le type de réseau d'une instance de base de données, un temps d'arrêt est possible. Si le mode double pile n'est pas

pris en charge par la version du moteur de base de données ou le groupe de sous-réseau de base de données spécifié, l'erreur `NetworkTypeNotSupported` est renvoyée.

Pour plus d'informations sur la création d'une instance de base de données, consultez Création d'une instance de base de données Amazon RDS. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Pour déterminer si un(e) instance de base de données est en mode double pile en utilisant la console, affichez Network type (Type de réseau) dans l'onglet Connectivity & security (Connectivité et sécurité) pour l'instance de la base de données.

Modification de IPv4 d'instances de base de données réservés uniquement pour utiliser le mode double pile

Vous pouvez modifier un d'instances de base de données réservé IPv4 uniquement pour utiliser le mode double pile. Pour ce faire, modifiez le type de réseau de l'instance de base de données. La modification peut entraîner un temps d'arrêt.

Nous vous recommandons de modifier le type de réseau de vos instances de base de données Amazon RDS au cours d'une fenêtre de maintenance. Pour l'heure, il n'est pas possible de définir le type de réseau des nouvelles instances sur le mode double pile. Vous pouvez définir le type de réseau manuellement à l'aide de la commande `modify-db-instance` .

Avant de modifier un(e) instance de base de données pour utiliser le mode double pile, assurez-vous que son groupe de sous-réseau de base de données prend en charge le mode double pile. Si le groupe de sous-réseau de base de données associé à l'instance de base de données ne prend pas en charge le mode double pile, spécifiez un autre groupe de sous-réseau de base de données qui le prend en charge lorsque vous modifiez l'instance de base de données. La modification du groupe de sous-réseaux de base de données d'une instance de bases de données peut entraîner une interruption de service.

Si vous modifiez le groupe de sous-réseau de base de données d'une instance de bases de données avant de modifier l'instance de bases de données pour utiliser le mode double pile, assurez-vous que le groupe de sous-réseau de base de données est valide pour l'instance de bases de données avant et après la modification.

Pour les instances mono-AZ RDS pour PostgreSQL, RDS pour MySQL, RDS pour Oracle et RDS pour MariaDB, nous vous recommandons d'exécuter la commande avec uniquement le paramètre

défini sur pour faire passer modify-db-instancele réseau en mode double pile. `--network-type` DUAL L'ajout d'autres paramètres en même temps que le paramètre `--network-type` dans le même appel d'API peut entraîner des temps d'arrêt. Pour modifier plusieurs paramètres, vérifiez que le modification du type de réseau a bien abouti avant d'envoyer une autre demande `modify-db-instance` avec d'autres paramètres.

Les modifications du type de réseau pour les instances de base de données multi-AZ RDS pour PostgreSQL, RDS pour MySQL, RDS pour Oracle et RDS pour MariaDB entraînent un bref temps d'arrêt et déclenchent un basculement si vous utilisez uniquement le paramètre ou si vous combinez des paramètres dans une commande. `--network-type` modify-db-instance

Les modifications du type de réseau sur les instances de base de données RDS for SQL Server mono-AZ ou multi-AZ provoquent une interruption si vous utilisez uniquement le paramètre `--network-type` ou si vous combinez des paramètres dans une commande `modify-db-instance`. Les modifications du type de réseau entraînent un basculement dans une instance SQL Server multi-AZ.

Si vous ne parvenez pas à vous connecter au d'instances de base de données après la modification, assurez-vous que les pare-feux de sécurité et les tables de routage du client et de la base de données sont correctement configurés pour autoriser le trafic vers la base de données sur le réseau sélectionné (IPv4 ou non IPv6). Vous devrez peut-être également modifier les paramètres du système d'exploitation, les bibliothèques ou les pilotes pour vous connecter à l'aide d'une IPv6 adresse.

Lorsque vous modifiez une instance de base de données pour qu'elle utilise le mode double pile, aucune modification (passage du déploiement mono-AZ au déploiement multi-AZ ou du déploiement multi-AZ au déploiement mono-AZ) ne doit être en cours.

Pour modifier un d'instances de base de données réservé IPv4 uniquement afin d'utiliser le mode double pile

1. Modifiez un groupe de sous-réseaux de base de données pour prendre en charge le mode double pile ou créez un groupe de sous-réseaux de base de données qui prend en charge le mode double pile :

   a. Associez un bloc IPv6 CIDR à votre VPC.

      Pour obtenir des instructions, consultez la section Ajouter un bloc IPv6 CIDR à votre VPC dans le guide de l'utilisateur Amazon VPC.

b. Attachez le bloc IPv6 CIDR à tous les sous-réseaux de votre groupe de sous-réseaux de base de données.

   Pour obtenir des instructions, consultez la section Ajouter un bloc IPv6 CIDR à votre sous-réseau dans le guide de l'utilisateur Amazon VPC.

c. Confirmez que le groupe de sous-réseaux de base de données prend en charge le mode double pile.

   Si vous utilisez le AWS Management Console, sélectionnez le groupe de sous-réseaux de base de données et assurez-vous que la valeur des types de réseau pris en charge est Dual, IPv4.

   Si vous utilisez le AWS CLI, exécutez la describe-db-subnet-groupscommande et assurez-vous que la `SupportedNetworkType` valeur de l'instance de base de données est`Dual, IPv4`.

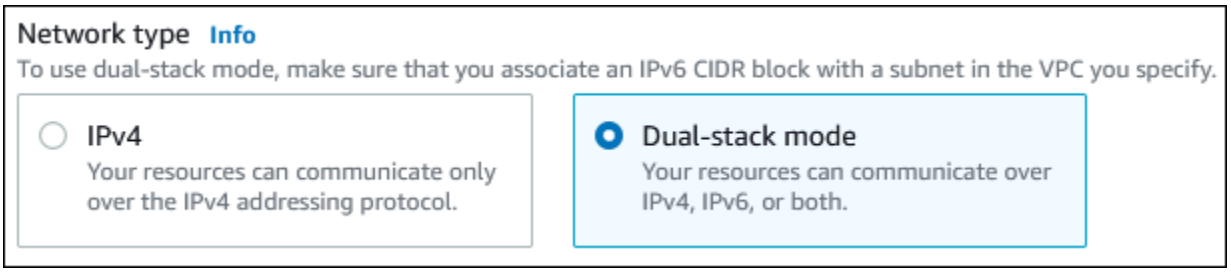
2. Modifiez le groupe de sécurité associé au d'instances de base de données pour autoriser IPv6 les connexions à la base de données, ou créez un nouveau groupe de sécurité qui autorise IPv6 les connexions.

   Pour obtenir des instructions, consultez la section Security group rules (Règles des groupes de sécurité) dans le Guide de l'utilisateur Amazon VPC.

3. Modifiez l'instance de la base de données pour qu'il prenne en charge le mode double pile. Pour ce faire, réglez le Network type (Type de réseau) sur Dual-stack mode (Mode double pile).

   Si vous utilisez la console, assurez-vous que les paramètres suivants sont corrects :

   - Network type (Type de réseau) – Dual-stack mode (Mode double pile)



   - DB subnet group (Groupe de sous-réseau de base de données) : le groupe de sous-réseau de base de données que vous avez configuré à l'étape précédente.
   - Security group (Groupe de sécurité) – la sécurité que vous avez configurée dans une étape précédente.

Si vous utilisez le AWS CLI, assurez-vous que les paramètres suivants sont corrects :

- `--network-type` – `dual`
- `--db-subnet-group-name` — le groupe de sous-réseau de base de données que vous avez configuré à l'étape précédente.
- `--vpc-security-group-ids` : le groupe de sécurité du VPC que vous avez configuré à l'étape précédente.

Par exemple :

```
aws rds modify-db-instance --db-instance-identifier my-instance --network-type
 "DUAL"
```

4. Confirmez que l'instance de base de données prend en charge le mode double pile.

   Si vous utilisez la console, choisissez l'onglet Connectivity & security (Connectivité et sécurité) pour l'instance de la base de données. Dans cet onglet, assurez-vous que la valeur de Network type (Type de réseau) est Dual-stack mode (Mode double pile).

   Si vous utilisez le AWS CLI, exécutez la describe-db-instancescommande et assurez-vous que la `NetworkType` valeur de l'instance de base de données est`dual`.

   Exécutez la `dig` commande sur le point de terminaison de l'instance de base de données du pour identifier l' IPv6adresse qui lui est associée.

   ```
   dig db-instance-endpoint AAAA
   ```

   Utilisez le point de terminaison de l'instance de base de données du , et non l' IPv6 adresse, pour vous connecter au d'instances de base de données.

Disponibilité des régions et des versions

La disponibilité et la prise en charge des fonctionnalités varient selon les versions spécifiques de chaque moteur de base de données, et selon les Régions AWS. Pour en savoir plus sur la disponibilité des versions et des régions avec le mode double pile, consultez Régions et moteurs de base de données pris en charge pour le mode double pile dans Amazon RDS.

Limitations pour les instances de base de données en réseau à double pile

Les limitations suivantes s'appliquent aux instances de base de données en réseau à double pile :

- Les d'instances de base de données ne peuvent pas utiliser le IPv6 protocole exclusivement. Ils peuvent utiliser IPv4 exclusivement, ou ils peuvent utiliser le IPv6 protocole IPv4 and (mode double pile).
- Amazon RDS ne prend pas en charge les IPv6 sous-réseaux natifs.
- Les instances de bases de données qui utilisent le mode double pile doivent être privé(e)s. Ils/elles ne peuvent pas être publiquement accessibles.
- Pour RDS for SQL Server, les instances de base de données en mode double pile qui utilisent les points de terminaison du récepteur du groupe de AGs disponibilité Always On présentent uniquement des adresses. IPv4
- Vous ne pouvez pas utiliser RDS Proxy avec des instances de base de données en mode double pile.
- Vous ne pouvez pas utiliser le mode double pile avec RDS sur les instances de AWS Outposts base de données.
- Vous ne pouvez pas utiliser le mode double pile avec des instances de base de données dans une zone locale.

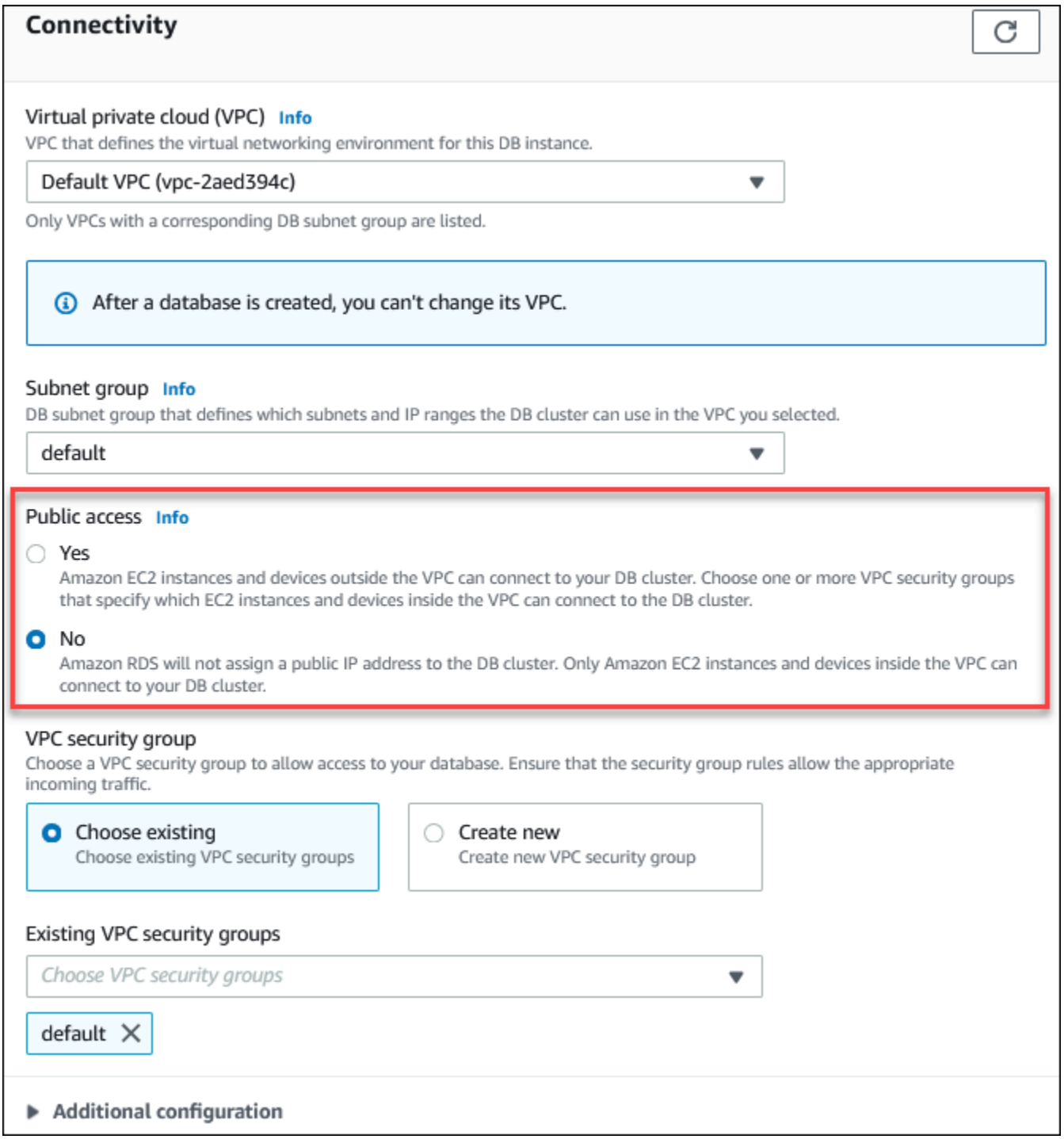
Masquer un(e) instance de base de données dans un VPC depuis Internet

L'un des scénarios Amazon RDS les plus courants consiste à avoir un VPC dans lequel vous avez une instance EC2 Amazon avec une application Web destinée au public et un cluster d' de base de données avec une base de données non accessible au public. Par exemple, vous pouvez créer un VPC doté d'un sous-réseau public et d'un sous-réseau privé. EC2 les instances qui fonctionnent comme des serveurs Web peuvent être déployées dans le sous-réseau public. Les instances de base de données sont déployés dans le sous-réseau privé. Dans un tel déploiement, seuls les serveurs web ont accès aux instances de base de données. Pour obtenir une illustration de ce scénario, consultez Un d'instances de base de données dans une EC2 instance VPC accessible par Amazon dans le même VPC.

Lorsque vous lancez un(e) instance de base de données dans un VPC, l'instance de base de données possède une adresse IP privée pour le trafic à l'intérieur du VPC. Cette adresse IP privée n'est pas accessible au public. Vous pouvez utiliser l'option Public access (Accès public) pour indiquer si l'instance de base de données possède également une adresse IP publique en plus de

l'adresse IP privée. Si l'instance de la base de données est désigné comme publiquement accessible, son point de terminaison DNS se résout à l'adresse IP privée à partir du VPC. Il renvoit à l'adresse IP publique depuis l'extérieur du VPC. L'accès à l'instance de la base de données est contrôlé en dernier ressort par le groupe de sécurité qu'il utilise. Cet accès public n'est pas autorisé si le groupe de sécurité attribué à l'instance de la base de données ne comprend pas de règles d'entrée qui l'autorisent. En outre, pour qu'un(e) instance de base de données soit publiquement accessible, les sous-réseaux de son groupe de sous-réseaux de base de données doivent avoir une passerelle Internet. Pour plus d'informations, consultez Impossible de se connecter à l'instance de base de données Amazon RDS.

Vous pouvez modifier un(e) instance de base de données pour activer ou désactiver l'accessibilité publique en modifiant l'option Public access (Accès public). L'illustration suivante présente l'option Public Access (Accès public) dans la section Additional connectivity configuration (Configuration de connectivité supplémentaire). Pour définir cette option, ouvrez la section Additional connectivity configuration (Configuration de connectivité supplémentaire) dans la section Connectivity (Connectivité).



Pour plus d'informations sur la modification d'une instance de base de données afin de définir l'option Public access (Accès public), veuillez consulter Modification d'une RDS instance de base de données Amazon.

Création d'un(e) instance de base de données dans un VPC

Les procédure suivantes vous aident à créer un(e) instance de base de données dans un VPC. Pour utiliser le VPC par défaut, vous pouvez commencer par l'étape 2, et utiliser le groupe VPC et sous-réseau de base de données qui ont déjà été créés pour vous. Si vous souhaitez créer un VPC supplémentaire, vous pouvez créer un nouveau VPC.

> **Note**
>
> Si vous voulez que votre instance de base de données du VPC soit publiquement accessible, vous devez mettre à jour les informations DNS pour le VPC en activant les attributs VPC DNS hostnames (noms d'hôtes DNS) et DNS resolution (Résolution DNS). Pour plus d'informations sur la mise à jour des informations DNS pour une instance VPC, consultez Mise à jour de la prise en charge DNS pour votre VPC.

Suivez les étapes ci-après pour créer une instance de base de données dans un VPC:

- Étape 1 : Création d'un VPC
- Étape 2 : créer un groupe de sous-réseaux de base de données
- Étape 3 : créer un groupe de sécurité VPC
- Étape 4 : créer une instance de base de données dans le VPC

Étape 1 : Création d'un VPC

Créez un VPC avec des sous-réseaux dans au moins deux zones de disponibilité. Vous utilisez ces sous-réseaux lorsque vous créerez un groupe de sous-réseaux de base de données. Si vous avez un VPC par défaut, un sous-réseau est automatiquement créé pour vous dans chaque zone de disponibilité de la Région AWS.

Pour obtenir plus d'informations, consultez Créer un VPC avec des sous-réseaux publics et privés, ou Create a VPC (Créer un VPC) dans le Guide de l'utilisateur Amazon VPC.

Étape 2 : créer un groupe de sous-réseaux de base de données

Un groupe de sous-réseaux DB est une collection de sous-réseaux (généralement privés) que vous créez pour un VPC et que vous spécifiez alors pour vos instances de base de données. Un groupe de sous-réseaux de base de données vous permet de spécifier un VPC particulier lorsque vous créez

des clusters d' de base de données à l'aide de l'API ou AWS CLI RDS. Si vous utilisez la console, vous pouvez simplement choisir le VPC et les sous-réseaux que vous voulez utiliser. Chaque groupe de sous-réseaux DB doit avoir au moins un sous-réseau dans au moins deux zones de disponibilité de la Région AWS. La bonne pratique est la suivante : chaque groupe de sous-réseaux de base de données doit être constitué d'au moins un sous-réseau pour chaque zone de disponibilité dans la Région AWS.

Pour les déploiements multi-AZ, la définition d'un sous-réseau pour toutes les zones de disponibilité d'un permet à Région AWS Amazon RDS de créer une nouvelle réplique de secours dans une autre zone de disponibilité si nécessaire. Vous pouvez également suivre cette bonne pratique pour les déploiements mono-AZ, au cas où vous seriez amené à les convertir en déploiements multi-AZ à l'avenir.

Pour qu'un(e) instance de base de données soit publiquement accessible, les sous-réseaux du groupe de sous-réseaux de base de données doivent avoir une passerelle Internet. Pour obtenir plus d'informations sur les passerelles Internet pour les sous-réseaux, consultez la section Connect to the internet using an internet gateway (Se connecter à Internet à l'aide d'une passerelle Internet) dans le Guide de l'utilisateur Amazon VPC.

> **Note**
>
> Le groupe de sous-réseaux de base de données d'une zone locale ne peut avoir qu'un seul sous-réseau.

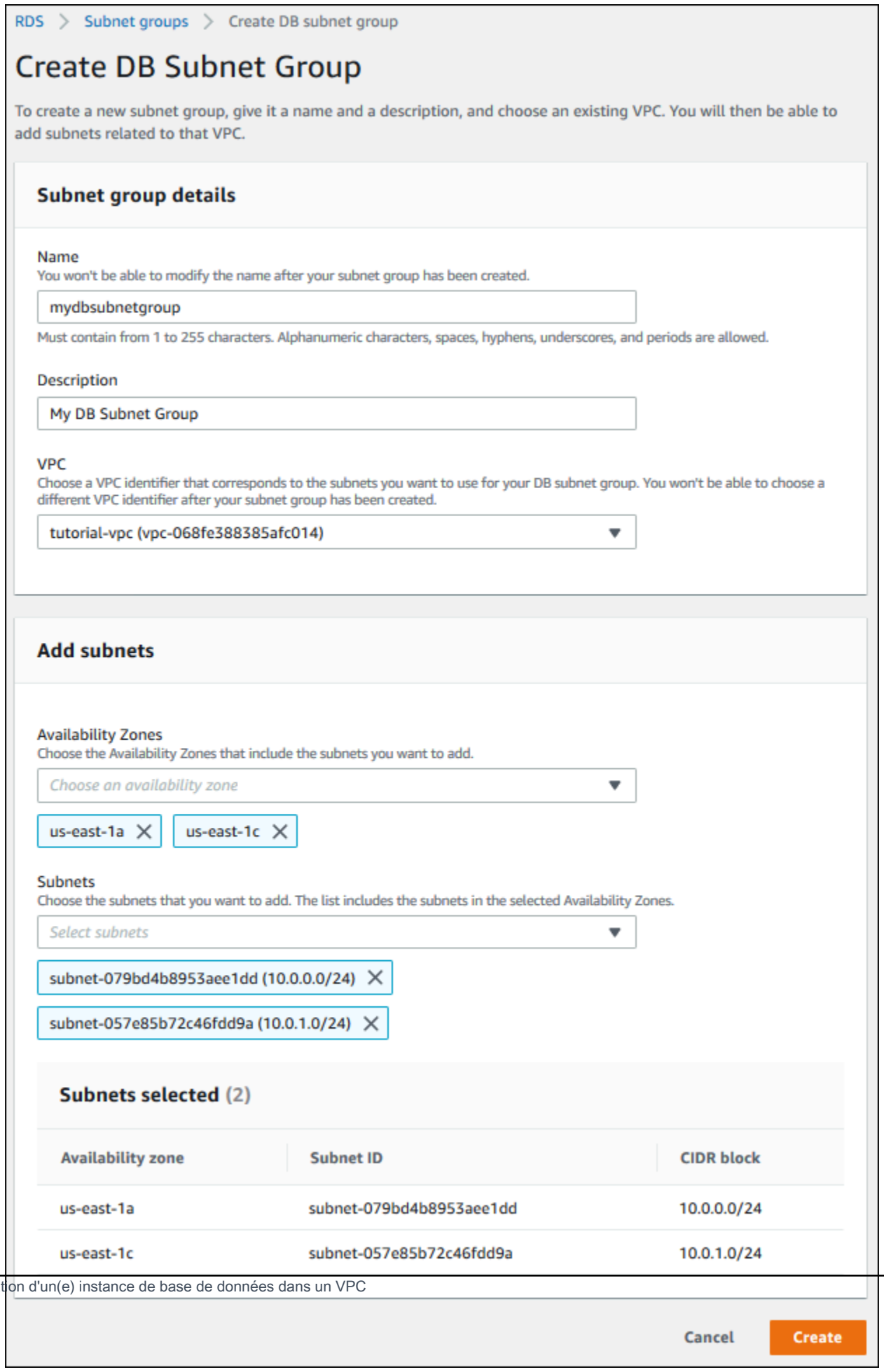
Lorsque vous créez un(e) instance de base de données dans un VPC, vous pouvez choisir un groupe de sous-réseau de base de données. Amazon RDS choisit dans ce sous-réseau un sous-réseau et une adresse IP à associer à votre instance de base de données. Si aucun groupe de sous-réseau de base de données n'existe, Amazon RDS crée un groupe de sous-réseau par défaut lorsque vous créez un(e) instance de base de données. Amazon RDS crée une interface réseau Elastic pour votre instance de base de données, et l'associe à cette adresse IP. L'instance de base de données utilise la zone de disponibilité contenant le sous-réseau.

Pour les déploiements multi-AZ, la définition d'un sous-réseau pour deux zones de disponibilité ou plus dans un permet à Région AWS Amazon RDS de créer un nouveau serveur de secours dans une autre zone de disponibilité en cas de besoin. Vous devez procéder de la sorte même pour les déploiements Single-AZ, au cas où vous souhaitiez les convertir en déploiements multi-AZ par la suite.

Dans cette étape, vous créez un groupe de sous-réseaux de base de données et ajoutez les sous-réseaux que vous avez créés pour votre VPC.

Pour créer un groupe de sous-réseaux

1. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
2. Dans le panneau de navigation, choisissez Subnet groups (Groupes de sous-réseaux).
3. Choisissez Create DB Subnet Group (Créer groupe de sous-réseaux de base de données).
4. Dans Nom, saisissez le nom de votre nouveau groupe de sous-réseaux de base de données.
5. Dans le champ Description, saisissez une description de votre groupe de sous-réseaux de base de données.
6. Pour le champ VPC, choisissez le VPC par défaut ou le VPC que vous avez créé.
7. Dans la section Ajouter des sous-réseaux, choisissez les zones de disponibilité qui incluent les sous-réseaux à partir de Zones de disponibilité, puis choisissez les sous-réseaux à partir de Sous-réseaux.


RDS > Subnet groups > Create DB subnet group
Create DB Subnet Group
To create a new subnet group, give it a name and a description, and choose an existing VPC. You will then be able to add subnets related to that VPC.
Subnet group details
Name
You won't be able to modify the name after your subnet group has been created.
mydbsubnetgroup
Must contain from 1 to 255 characters. Alphanumeric characters, spaces, hyphens, underscores, and periods are allowed.
Description
My DB Subnet Group
VPC
Choose a VPC identifier that corresponds to the subnets you want to use for your DB subnet group. You won't be able to choose a different VPC identifier after your subnet group has been created.
tutorial-vpc (vpc-068fe388385afc014)
Add subnets
Availability Zones
Choose the Availability Zones that include the subnets you want to add.
Choose an availability zone
us-east-1a
us-east-1c
Subnets
Choose the subnets that you want to add. The list includes the subnets in the selected Availability Zones.
Select subnets
subnet-079bd4b8953aee1dd (10.0.0.0/24)
subnet-057e85b72c46fdd9a (10.0.1.0/24)
Subnets selected (2)
Availability zone
Subnet ID
CIDR block
us-east-1a
subnet-079bd4b8953aee1dd
10.0.0.0/24
us-east-1c
subnet-057e85b72c46fdd9a
10.0.1.0/24
Cancel
Create

> **Note**
>
> Si vous avez activé une zone locale, vous pouvez choisir un groupe de zone de disponibilité sur la page Créer un groupe de sous-réseaux DB. Dans ce cas, choisissez Groupe de zone de disponibilité, Zones de disponibilité et Sous-réseaux.

8. Sélectionnez Créer.

   Votre nouveau groupe de sous-réseaux DB apparaît dans la liste des groupes de sous-réseaux sur la console RDS. Vous pouvez choisir le groupe de sous-réseaux DB pour afficher les détails, y compris l'ensemble des sous-réseaux associés au groupe, dans le volet des détails en bas de la fenêtre.

Étape 3 : créer un groupe de sécurité VPC

Avant de créer votre instance de base de données, vous pouvez créer un groupe de sécurité VPC à associer à votre instance de base de données. Si vous ne créez pas de groupe de sécurité VPC, vous pouvez utiliser le groupe de sécurité par défaut lorsque vous créez un(e) instance de base de données. Pour obtenir des instructions sur la création d'un groupe de sécurité pour votre instance de base de données, consultez Créer un groupe de sécurité VPC pour une instance de base de données privé(e), ou Control traffic to resources using security groups (Contrôler le trafic vers les ressources à l'aide de groupes de sécurité) dans le Guide de l'utilisateur Amazon VPC.

Étape 4 : créer une instance de base de données dans le VPC

Dans cette étape, vous créez un(e) instance de base de données et utilisez le nom du VPC, le groupe de sous-réseaux de base de données et le groupe de sécurité VPC que vous avez créés dans les étapes précédentes.

> **Note**
>
> Si vous voulez que votre instance de base de données du VPC soit publiquement accessible, vous devez activer les attributs du VPC DNS hostnames (Noms d'hôte DNS) et DNS resolution (Résolution DNS). Pour plus d'informations, consultez DNS attributes for your VPC (Attributs DNS pour votre VPC) dans le Guide de l'utilisateur d'Amazon VPC.

Pour plus d'informations sur la création d'une instance de bases de données, consultez Création d'une instance de base de données Amazon RDS.

Lorsque vous y êtes invité dans la section Connectivity (Connectivité), saisissez le nom du VPC, le groupe de sous-réseaux de base de données et le groupe de sécurité VPC.

Mettre à jour le VPC pour une instance de base de données

Vous pouvez utiliser le AWS Management Console pour déplacer votre instance de base de données vers une autre instanceVPC.

Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon. Dans la section Connectivity (Connectivité) de la page de modification, illustrée ci-dessous, entrez le nouveau groupe de sous-réseau de base de données pour DB subnet group (Groupe de sous-réseau de base de données). Le nouveau groupe de sous-réseaux doit être un groupe de sous-réseaux dans un nouveau. VPC



Vous ne pouvez pas modifier le VPC pour une instance de base de données si les conditions suivantes s'appliquent :

- L'instance de base de données se trouve dans plusieurs zones de disponibilité. Vous pouvez convertir l'instance de base de données en une seule zone de disponibilité, la déplacer vers une nouvelleVPC, puis la reconvertir en instance de base de données multi-AZ. Pour de plus amples informations, veuillez consulter Configuration et gestion d'un déploiement multi-AZ pour Amazon RDS.
- L'instance de base de données possède un ou plusieurs réplicas en lecture. Vous pouvez supprimer les répliques lues, déplacer l'instance de base de données vers une nouvelle instanceVPC, puis ajouter à nouveau les répliques lues. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture d'instance de base de données.

- L'instance de base de données est un réplica en lecture. Vous pouvez promouvoir la réplique en lecture, puis déplacer l'instance de base de données autonome vers une nouvelleVPC. Pour de plus amples informations, veuillez consulter Promotion d'un réplica en lecture en instance de bases de données autonome.
- Le groupe de sous-réseaux de la cible VPC ne possède pas de sous-réseaux dans la zone de disponibilité de l'instance de base de données. Vous pouvez ajouter des sous-réseaux dans la zone de disponibilité de l'instance de base de données au groupe de sous-réseaux de base de données, puis déplacer l'instance de base de données vers le nouveau. VPC Pour de plus amples informations, veuillez consulter Utilisation de groupes de sous-réseaux DB.

Scénarios d'accès à un d'instances de base de données dans un VPC

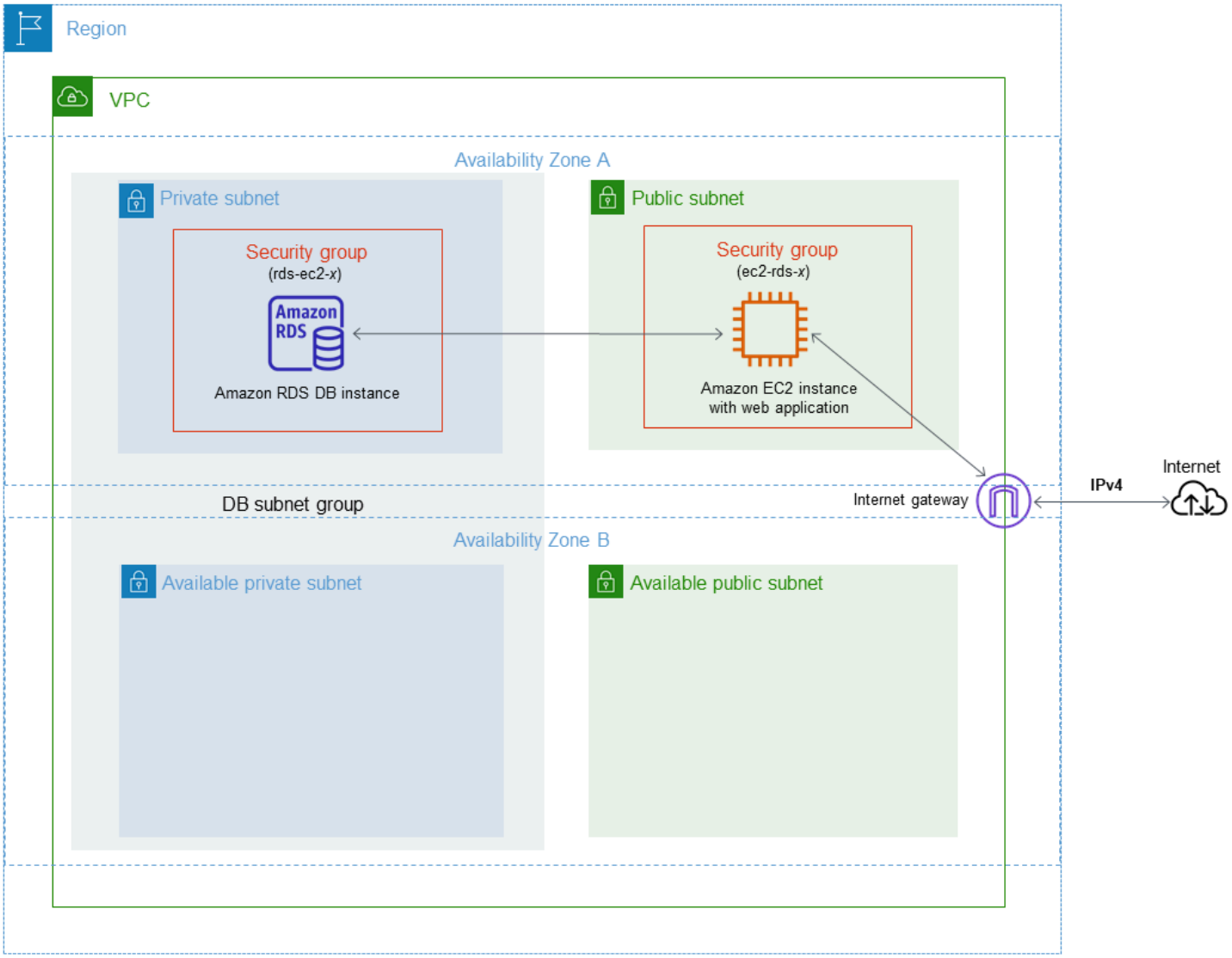
Amazon RDS Amazon prend en charge les scénarios suivants pour accéder à un d'instances de base de données dans un VPC :

- Une EC2 instance Amazon dans le même VPC
- Une EC2 instance dans un autre VPC
- Une application cliente via Internet
- Un réseau privé

Un d'instances de base de données dans une EC2 instance VPC accessible par Amazon dans le même VPC

Une utilisation courante d'un d'instances de base de données dans un VPC est de partager des données avec un serveur d'applications qui s'exécute dans une EC2 instance Amazon de ce même clusterVPC.

Le schéma suivant illustre ce scénario.



Le moyen le plus simple de gérer l'accès entre les EC2 instances et les d'instances de base de données dans le même environnement VPC est de procéder comme suit :

- Créez un groupe VPC de sécurité dans lequel vos d'instances de base de données seront intégrés. Ce groupe de sécurité peut être utilisé pour restreindre l'accès aux instances de base de données. Par exemple, vous pouvez créer une règle personnalisée pour ce groupe de sécurité. Cela peut autoriser l'TCPaccès à l'aide du port que vous avez attribué au d'instances de base de données lorsque vous l'avez créé et d'une adresse IP que vous utilisez pour accéder au d'instances de base de données à des fins de développement ou à d'autres fins.
- Créez un groupe VPC de sécurité EC2 dans lequel vos instances (serveurs Web et clients) seront intégrées. Ce groupe de sécurité peut, si nécessaire, autoriser l'accès à l'EC2instance depuis

Internet en utilisant la table VPC de routage. Par exemple, vous pouvez définir des règles sur ce groupe de sécurité pour autoriser l'TCPaccès à l'EC2instance via le port 22.

- Créez des règles personnalisées dans le groupe de sécurité pour vos d'instances de base de données qui autorisent les connexions à partir du groupe de sécurité que vous avez créé pour vos EC2 instances. Ces règles peuvent permettre à tout membre du groupe de sécurité d'accéder aux instances de la base de données.

Il existe un sous-réseau public et privé supplémentaire dans une zone de disponibilité distincte. Un groupe RDS de sous-réseaux de base de données nécessite un sous-réseau dans au moins deux zones de disponibilité. Le sous-réseau supplémentaire permet de passer facilement à un déploiement d'instance de base de données Multi-AZ à l'avenir.

Pour un didacticiel qui explique comment créer un VPC avec des sous-réseaux publics et privés pour ce scénario, voirTutoriel : Création d'un VPC à utiliser avec un) IPv4 .

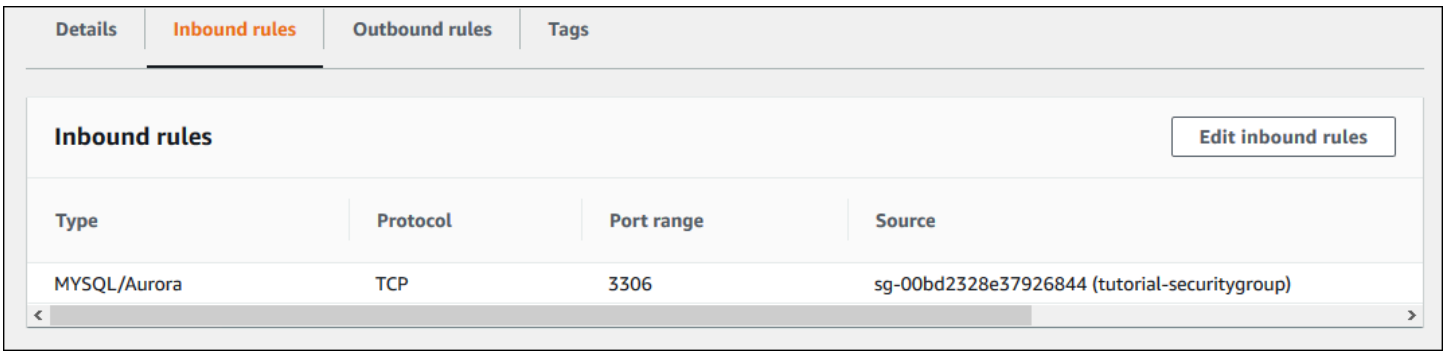
> **Tip**
>
> Pour plus d'informations, consultez Configuration de la connectivité réseau automatique avec une EC2 instance.

Pour créer une règle dans un groupe VPC de sécurité qui autorise les connexions depuis un autre groupe de sécurité, procédez comme suit :

1. Connectez-vous à la VPC console Amazon AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/vpc.
2. Dans le panneau de navigation, choisissez Groupes de sécurité.
3. Choisissez ou créez un groupe de sécurité auquel vous voulez autoriser les membres d'un autre groupe de sécurité à accéder. Dans le scénario précédent, il s'agit du groupe de sécurité que vous utilisez pour vos instances de base de données. Sélectionnez l'onglet Inbound Rules (Règles entrantes), puis Edit inbound rules (Modifier les règles entrantes).
4. Sur la page Edit inbound rules (Modifier les règles entrantes), cliquez sur Add Rule (Ajouter une règle).
5. Pour Type, choisissez l'entrée qui correspond au port que vous avez utilisé lors de la création de votre d'instances de base de données, tel que MYSQL/Aurora.

6. Dans la zone Source, commencez à taper l'ID du groupe de sécurité, qui répertorie les groupes de sécurité correspondants. Choisissez le groupe de sécurité dont vous voulez autoriser les membres à accéder aux ressources protégées par ce groupe de sécurité. Dans le scénario précédent, il s'agit du groupe de sécurité que vous utilisez pour votre EC2 instance.
7. Si nécessaire, répétez les étapes du TCP protocole en créant une règle avec All TCP comme type et votre groupe de sécurité dans le champ Source. Si vous avez l'intention d'utiliser le UDP protocole, créez une règle avec All UDP comme type et votre groupe de sécurité dans Source.
8. Sélectionnez Enregistrer les règles.

L'écran suivant affiche une règle entrante, ainsi qu'un groupe de sécurité pour sa source.

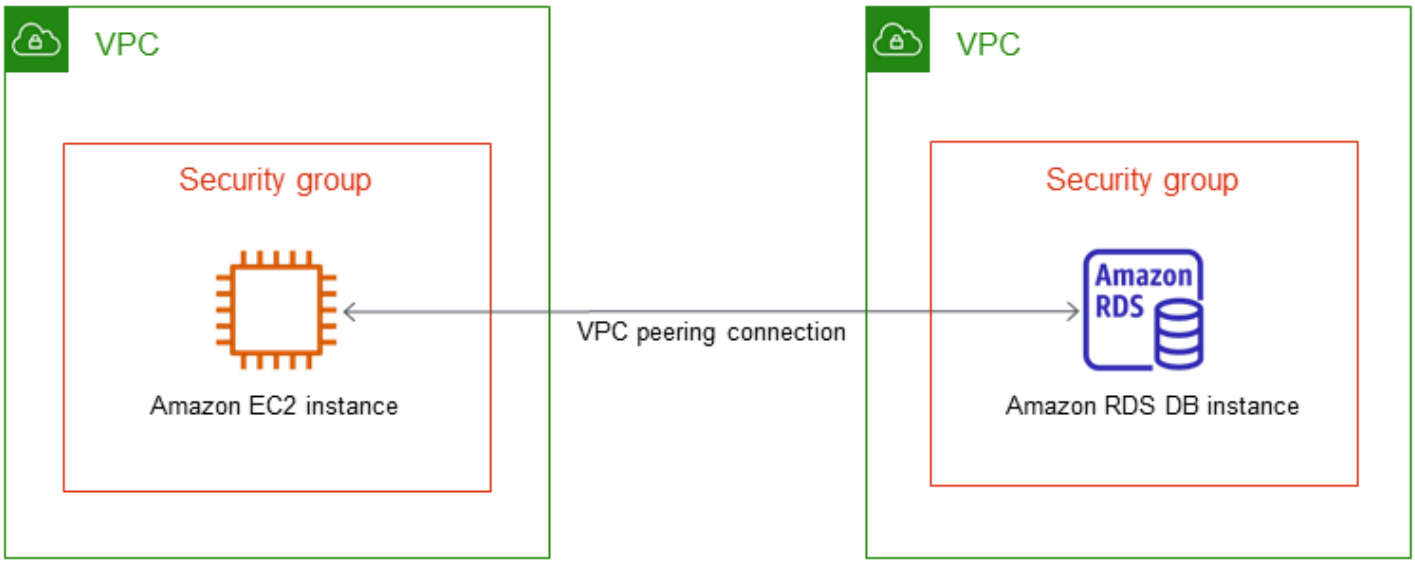


Pour plus d'informations sur la connexion à l'instance de de base de données depuis votre EC2 instance, consultez Connexion à une RDS instance de base de données Amazon.

Un d'instances de base de données dans un VPC accès par une EC2 instance dans un autre VPC

Lorsque vos d'instances de base de données se trouvent dans une EC2 instance VPC différente de celle que vous utilisez pour y accéder, vous pouvez utiliser le VPC peering pour accéder au d'instances de base de données.

Le schéma suivant illustre ce scénario.

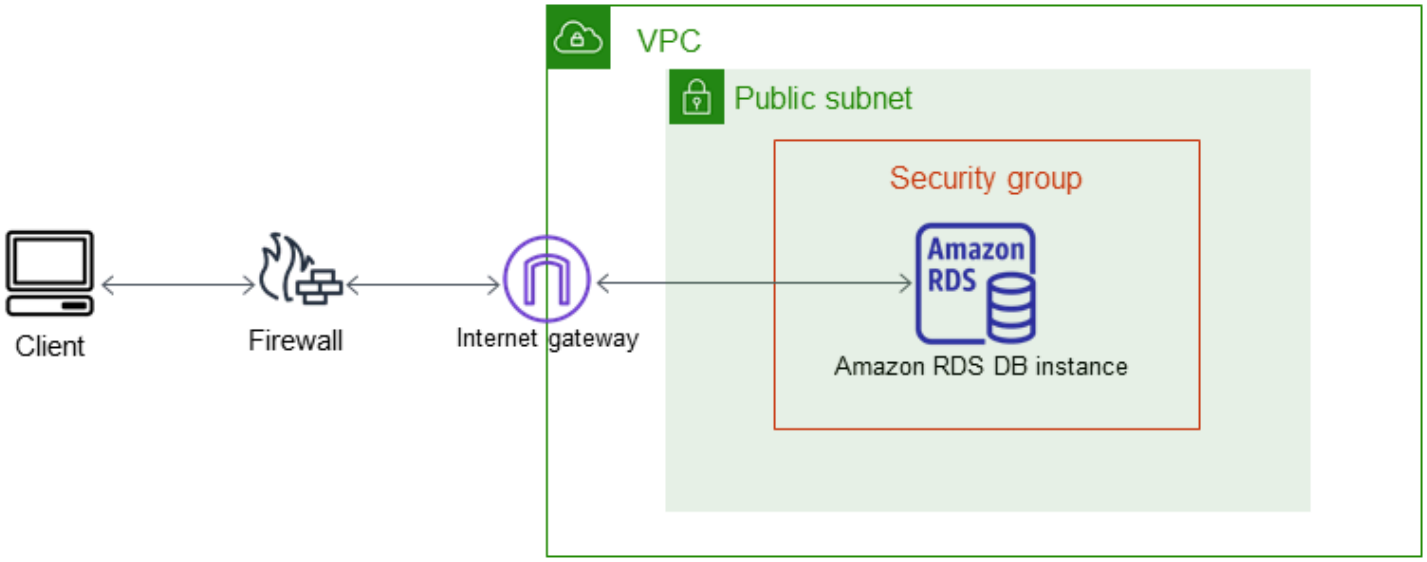


Une connexion de VPC peering est une connexion réseau entre deux personnes VPCs qui vous permet d'acheminer le trafic entre elles à l'aide d'adresses IP privées. Les ressources de l'un ou de l'autre VPC peuvent communiquer entre elles comme si elles se trouvaient au sein du même réseau. Vous pouvez créer une connexion de VPC peering entre le vôtreVPCs, avec un VPC compte sur un autre AWS compte ou avec un VPC compte différent Région AWS. Pour en savoir plus sur le VPC peering, consultez le VPCguide de l'utilisateur d'Amazon Virtual Private Cloud.

Un d'instances de base de données dans une application cliente VPC accessible via Internet

Pour accéder à des d'instances de base de données VPC depuis une application cliente via Internet, vous configurez une VPC avec un sous-réseau public unique et une passerelle Internet pour permettre la communication sur Internet.

Le schéma suivant illustre ce scénario.



Nous recommandons la configuration suivante :

- A VPC de taille /16 (par exemple CIDR : 10.0.0.0/16). Cette taille fournit 65 536 adresses IP privées.
- Un sous-réseau de taille /24 (par exemple CIDR : 10.0.0.0/24). Cette taille fournit 256 adresses IP privées.
- Un d'instances de base de données RDSAmazon associé au sous-réseau VPC et au sous-réseau.
- Passerelle Internet qui se connecte VPC à Internet et à d'autres AWS produits.
- Groupe de sécurité associé à l'instance de base de données. Les règles de trafic entrant de votre groupe de sécurité permettent à votre application client d'accéder à votre instance de base de données.

Pour plus d'informations sur la création de d'instances de base de données dans unVPC, consultezCréation d'un(e) instance de base de données dans un VPC.

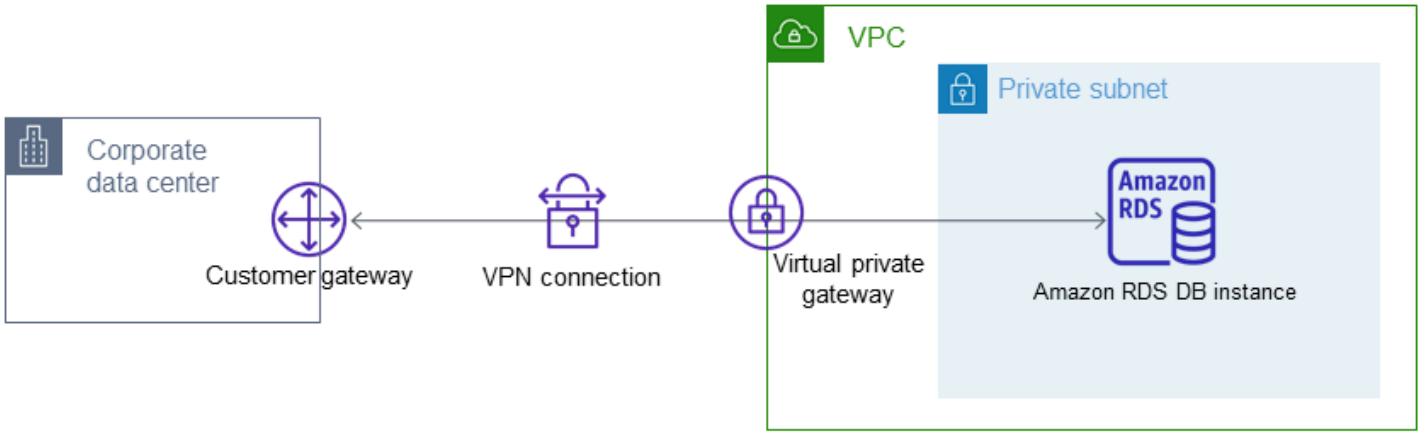
Un d'instances de base de données dans un VPC accès par un réseau privé

Si votre instance de base de données n'est pas accessible publiquement, les options suivantes vous permettent d'y accéder à partir d'un réseau privé :

- Une AWS Site-to-Site VPN connexion. Pour plus d'informations, consultez Qu'est-ce que AWS Site-to-Site VPN ?

- Une AWS Direct Connect connexion. Pour plus d'informations, voir Qu'est-ce que c'est AWS Direct Connect ?
- Une AWS Client VPN connexion. Pour plus d'informations, consultez Qu'est-ce que AWS Client VPN ?

Le schéma suivant illustre un scénario avec une AWS Site-to-Site VPN connexion.

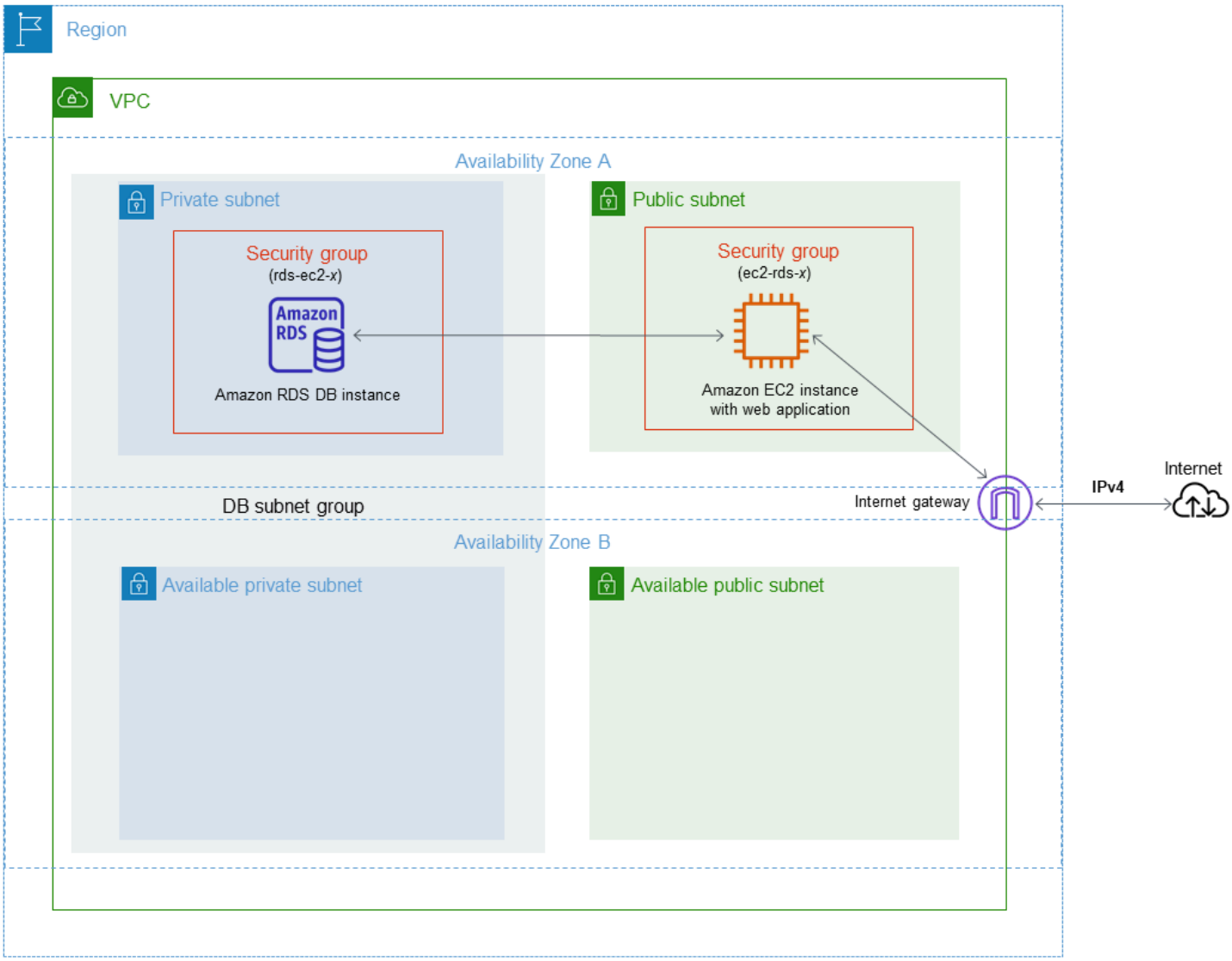


Pour de plus amples informations, veuillez consulter Confidentialité du trafic inter-réseau.

Tutoriel : Création d'un VPC à utiliser avec un) IPv4

Un scénario courant comprend une instance de base de données dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC. Ce VPC partage des données avec un serveur web qui fonctionne dans le même VPC. Dans ce didacticiel, vous créez le VPC pour ce scénario.

Le schéma suivant illustre ce scénario. Pour plus d'informations sur d'autres scénarios, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.



Votre instance de bases de données doit être disponible uniquement pour votre serveur Web, et non pour l'Internet public. Vous créez ainsi un VPC avec des sous-réseaux publics et privés. Le serveur web étant hébergé dans le sous-réseau public, il peut atteindre Internet. L'instance de base de données est hébergé(e) dans un sous-réseau privé. Le serveur web peut se connecter à l'instance

de base de données car il est hébergé dans le même VPC. Mais l'instance de base de données n'est pas accessible à l'Internet public, ce qui assure une plus grande sécurité.

Ce tutoriel configure un sous-réseau public et privé supplémentaire dans une zone de disponibilité séparée. Ces sous-réseaux ne sont pas utilisés par le tutoriel. Un groupe de sous-réseaux de base de données RDS nécessite un sous-réseau dans au moins deux zones de disponibilité. Le sous-réseau supplémentaire facilite le passage à un déploiement d'instances de base de données multi-AZ à l'avenir.

Ce didacticiel décrit la configuration d'un VPC pour les instances de base de données Amazon RDS. Pour obtenir un didacticiel qui vous montre comment créer un serveur Web pour ce scénario VPC, consultez Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS. Pour plus d'informations sur Amazon VPC, consultez le Guide de mise en route Amazon VPC et le Guide de l'utilisateur Amazon VPC.

> **Tip**
>
> La configuration du réseau est similaire à celle décrite dans ce tutoriel. Pour plus d'informations, consultez Configuration de la connectivité réseau automatique avec une EC2 instance.

Créer un VPC avec des sous-réseaux publics et privés

Utilisez la procédure suivante pour créer un VPC avec des sous-réseaux publics et privés.

Pour créer un VPC et des sous-réseaux

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le coin supérieur droit du AWS Management Console, choisissez la région dans laquelle créer votre VPC. Cet exemple utilise la région USA Ouest (Oregon).
3. Dans le coin supérieur gauche, choisissez VPC Dashboard (Tableau de bord VPC). Pour commencer à créer un VPC, sélectionnez Create VPC (Créer un VPC).
4. Pour Resources to create (Ressources à créer) sous VPC settings (Paramètres VPC), choisissez VPC and more (VPC et plus).
5. Pour VPC settings (Paramètres de VPC), définissez les valeurs suivantes :

   - Name tag auto-generation (Génération automatique de balise de nom) : **`tutorial`**

- IPv4 Bloc CIDR — **`10.0.0.0/16`**
- IPv6 Bloc CIDR — Aucun bloc IPv6 CIDR
- Tenancy (Location) : Default (Par défaut)
- Nombre de zones de disponibilité (AZs) — 2
- Personnaliser AZs : conservez les valeurs par défaut.
- Number of public subnet (Nombre de sous-réseaux publics) : 2
- Number of private subnets (Nombre de sous-réseaux privés) : 2
- Customize subnets CIDR blocks (Personnaliser les blocs CIDR des sous-réseaux) : conserver les valeurs par défaut.
- NAT gateways ($) [Passerelles NAT ($)] : None (Aucune)
- VPC endpoints (Points de terminaison VPC) : None (Aucun)
- DNS options (Options DNS) : conservez les valeurs par défaut.

> **Note**
>
> Amazon RDS nécessite au moins deux sous-réseaux dans deux zones de disponibilité différentes pour prendre en charge les déploiements d'instances de base de données Multi-AZ. Ce tutoriel crée un déploiement Mono-AZ, mais l'exigence facilite la conversion vers un déploiement d'instance de base de données Multi-AZ dans le futur.

6. Sélectionnez Create VPC (Créer un VPC).

Créer un groupe de sécurité VPC pour un serveur web public

Ensuite, vous créez un groupe de sécurité pour l'accès public. Pour vous connecter aux EC2 instances publiques de votre VPC, vous devez ajouter des règles entrantes à votre groupe de sécurité VPC. Elles permettent au trafic de se connecter depuis Internet.

Pour créer un groupe de sécurité VPC

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Choisissez successivement VPC Dashboard (Tableau de bord VPC), Security Groups (Groupes de sécurité) et Create Security Group (Créer un groupe de sécurité).

3. Sur la page Create Security Group (Créer un groupe de sécurité), définissez les valeurs suivantes :

   - Nom du groupe de sécurité : **`tutorial-securitygroup`**
   - Description : **`Tutorial Security Group`**
   - VPC : Choisissez le VPC que vous avez créé précédemment, par exemple : vpc- (tutorial-vpc) *`identifier`*

4. Ajoutez des règles entrantes au groupe de sécurité.

   a. Déterminez l'adresse IP à utiliser pour vous connecter aux EC2 instances de votre VPC à l'aide de Secure Shell (SSH). Pour déterminer votre adresse IP publique, dans une autre fenêtre ou un autre onglet du navigateur, vous pouvez utiliser le service à l'adresse https://checkip.amazonaws.com. Exemple d'adresse IP : `203.0.113.25/32`.

      Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Dans ce cas, trouvez la plage d'adresses IP utilisées par les ordinateurs clients.

      > ⚠ Warning
      >
      > Si vous utilisez `0.0.0.0/0` pour l'accès SSH, vous permettez à toutes les adresses IP d'accéder à vos instances publiques par SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. Dans un environnement de production, autorisez uniquement l'accès à vos instances à l'aide de SSH pour une adresse IP ou une plage d'adresses spécifique.

   b. Dans la section Règles entrantes, choisissez Ajouter une règle.

   c. Définissez les valeurs suivantes pour votre nouvelle règle d'entrée afin d'autoriser l'accès SSH à votre instance Amazon EC2 . Dans ce cas, vous pouvez vous connecter à votre EC2 instance Amazon pour installer le serveur Web et d'autres utilitaires. Vous vous connectez également à votre EC2 instance pour télécharger du contenu pour votre serveur Web.

      - Type: **`SSH`**
      - Source : l'adresse IP ou la plage d'adresses IP de l'étape a ; par exemple : **`203.0.113.25/32`**.

   d. Choisissez Ajouter une règle.

e. Définissez les valeurs suivantes pour que votre nouvelle règle entrante autorise HTTP à accéder à votre serveur Web :

- Type : **HTTP**
- Source : **0.0.0.0/0**

5. Choisissez Create security group (Créer un groupe de sécurité) pour créer le groupe de sécurité.

   Notez l'ID du groupe de sécurité, car vous en aurez besoin ultérieurement dans ce didacticiel.

Créer un groupe de sécurité VPC pour une instance de base de données privé(e)

Pour que votre instance de base de données demeure privé(e), créez un deuxième groupe de sécurité pour l'accès privé. Pour vous connecter à des d'instances de base de données privées dans votre VPC, vous ajoutez des règles entrantes à votre groupe de sécurité VPC qui autorisent uniquement le trafic provenant de votre serveur Web.

Pour créer un groupe de sécurité VPC

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Choisissez successivement VPC Dashboard (Tableau de bord VPC), Security Groups (Groupes de sécurité) et Create Security Group (Créer un groupe de sécurité).
3. Sur la page Create Security Group (Créer un groupe de sécurité), définissez les valeurs suivantes :

   - Nom du groupe de sécurité : **tutorial-db-securitygroup**
   - Description : **Tutorial DB Instance Security Group**
   - VPC : Choisissez le VPC que vous avez créé précédemment, par exemple : vpc- (tutorial-vpc) *identifier*

4. Ajoutez des règles entrantes au groupe de sécurité.

   a. Dans la section Règles entrantes, choisissez Ajouter une règle.

   b. Définissez les valeurs suivantes pour votre nouvelle règle entrante afin d'autoriser le trafic MySQL sur le port 3306 depuis votre instance Amazon EC2 . Dans ce cas, vous pouvez vous connecter du serveur Web à votre instance de bases de données. Pour ce faire, vous pouvez stocker et extraire les données entre votre application web et votre base de données.

- Type : **MySQL/Aurora**
- Source : identifiant du groupe de sécurité tutorial-securitygroup que vous avez créé précédemment dans ce tutoriel, par exemple : sg-9edd5cfb.

5. Choisissez Create security group (Créer un groupe de sécurité) pour créer le groupe de sécurité.

Création d'un groupe de sous-réseaux DB

Un groupe de sous-réseaux de base de données désigne une collection de sous-réseaux que vous créez dans un VPC et que vous spécifiez alors pour vos instances de bases de données. Un groupe de sous-réseaux de base de données vous permet de spécifier un VPC particulier lors de la création d'instances de base de données.

Pour créer un groupe de sous-réseaux

1. Identifiez les sous-réseaux privés pour votre base de données dans le VPC.

   a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
   b. Choisissez VPC Dashboard (Tableau de bord du VPC), puis Subnets (Sous-réseaux).
   c. Notez le sous-réseau IDs des sous-réseaux nommés tutorial-subnet-private1-us-west-2a et 2-us-west-2b. tutorial-subnet-private

      Vous avez besoin du sous-réseau IDs lorsque vous créez votre groupe de sous-réseaux de base de données.

2. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.

   Assurez-vous de vous connecter à la console Amazon RDS et non à la console Amazon VPC.

3. Dans le panneau de navigation, choisissez Subnet groups (Groupes de sous-réseaux).
4. Choisissez Create DB Subnet Group (Créer groupe de sous-réseaux de base de données).
5. Sur la page Create DB subnet group (Créer groupe de sous-réseaux de base de données), définissez ces valeurs dans Subnet group details (Détails de groupe de sous-réseaux) :

   - Nom: **tutorial-db-subnet-group**
   - Description: **Tutorial DB Subnet Group**
   - VPC : didacticiel-vpc (vpc -) *identifier*

6. Dans la section Ajouter des sous-réseaux, choisissez les zones de disponibilité et les sous-réseaux.

Pour ce tutoriel, choisissez us-west-2a et us-west-2b pour les Availability Zones (Zones de disponibilité). Pour Subnets (Sous-réseaux), choisissez les sous-réseaux privés que vous avez identifiés à l'étape précédente.

7. Choisissez Créer.

   Votre nouveau groupe de sous-réseaux DB apparaît dans la liste des groupes de sous-réseaux sur la console RDS. Vous pouvez choisir le groupe de sous-réseaux DB pour afficher les détails dans le volet des détails en bas de la fenêtre. Ces détails comprennent tous les sous-réseaux employés par le groupe.

> **Note**
>
> Si vous avez créé ce VPC pour effectuer Didacticiel : Créer un serveur web et une instance de base de données Amazon RDS, créez l'instance de base de données en suivant les instructions fournies dans Créer une instance de base de données Amazon RDS.

Suppression du VPC

Après avoir créé le VPC et d'autres ressources pour ce didacticiel, vous pouvez les supprimer si elles ne sont plus nécessaires.

> **Note**
>
> Si vous avez ajouté des ressources dans le VPC que vous avez créé pour ce tutoriel, vous devrez peut-être les supprimer avant de pouvoir supprimer le VPC. Par exemple, ces ressources peuvent inclure des EC2 instances Amazon ou des d'instances de base de données Amazon RDS. Pour plus d'informations, consultez Supprimer votre VPC dans le Guide de l'utilisateur Amazon VPC.

Pour supprimer un VPC et les ressources associées

1. Supprimez le groupe de sous-réseaux de base de données.

   a. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.

   b. Dans le panneau de navigation, choisissez Subnet groups (Groupes de sous-réseaux).

c. Sélectionnez le groupe de sous-réseaux de base de données que vous souhaitez supprimer, par exemple tutorial-db-subnet-group.

d. Choisissez Supprimer, puis Supprimer dans la fenêtre de confirmation.

2. Notez l'ID du VPC.

a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.

b. Choisissez VPC Dashboard, puis choisissez. VPCs

c. Dans la liste, identifiez le VPC que vous avez créé, tel que tutorial-vpc.

d. Notez le VPC ID (ID de VPC) du VPC que vous avez créé. Vous aurez besoin de l'ID de VPC dans les étapes suivantes.

3. Suppression du groupe de sécurité

a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.

b. Choisissez Tableau de bord du VPC, puis Groupes de sécurité.

c. Sélectionnez le groupe de sécurité pour l'instance de base de données Amazon RDS, tel que tutorial-db-securitygroup.

d. Pour Actions, choisissez Delete security groups (Supprimer des groupes de sécurité), puis Delete (Supprimer) sur la page de confirmation.

e. Sur la page Groupes de sécurité, sélectionnez le groupe de sécurité pour l' EC2 instance Amazon, tel que tutorial-securitygroup.

f. Pour Actions, choisissez Delete security groups (Supprimer des groupes de sécurité), puis Delete (Supprimer) sur la page de confirmation.

4. Supprimer le VPC.

a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.

b. Choisissez VPC Dashboard, puis choisissez. VPCs

c. Sélectionnez le VPC que vous voulez supprimer, tel que tutorial-vpc.

d. Pour Actions, choisissez Supprimer le VPC.

La page de confirmation affiche les autres ressources associées au VPC qui seront également supprimées, y compris les sous-réseaux qui lui sont associés.

e. Sur la page de confirmation, entrez **`delete`** et choisissez Supprimer.

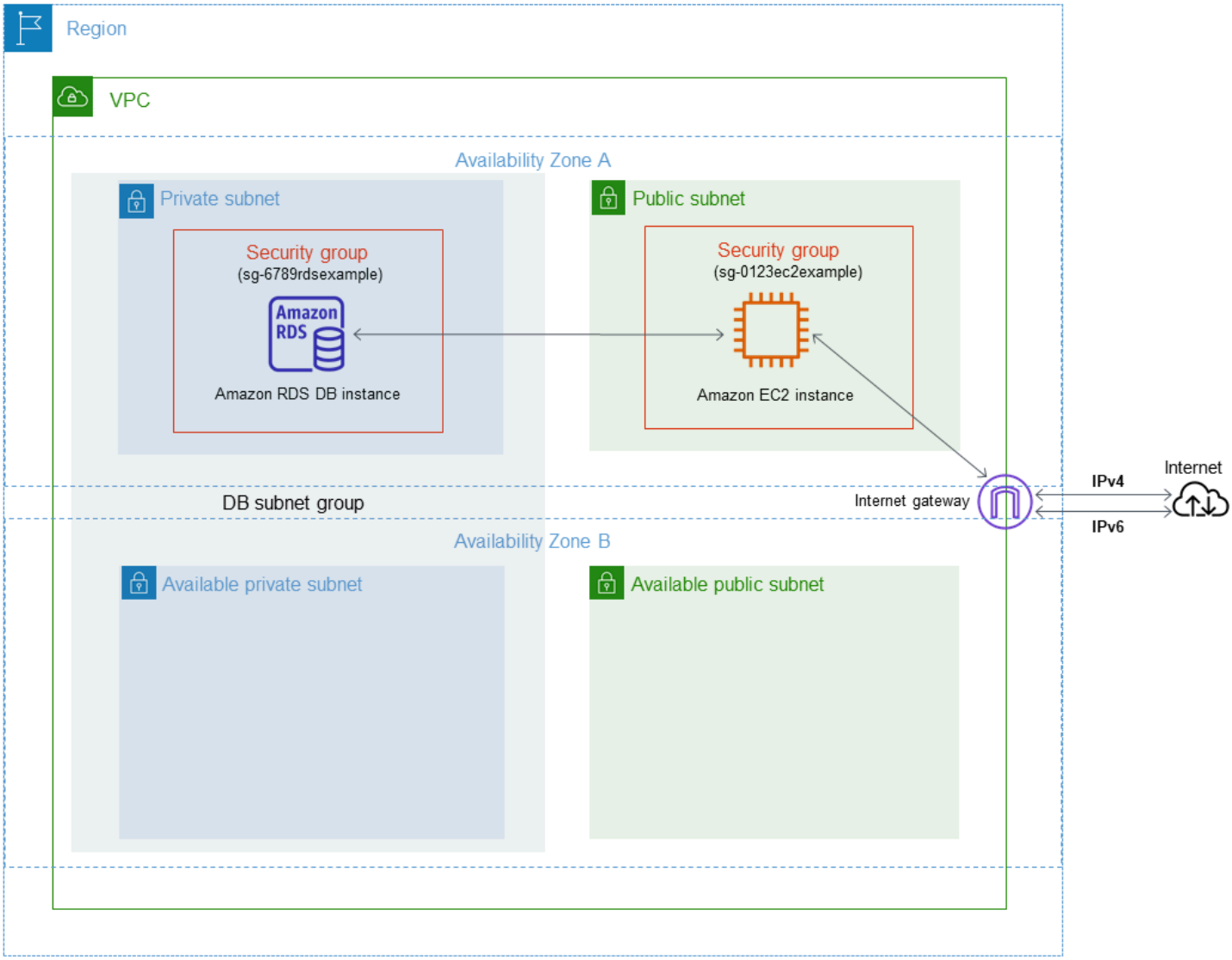
Tutoriel : Créer un VPC à utiliser avec une instance de base de données (mode double-pile)

Un scénario courant comprend une instance de base de données dans un cloud privé virtuel (VPC) basé sur le service Amazon VPC. Ce VPC partage des données avec une instance publique Amazon EC2 qui fonctionne dans le même VPC.

Dans ce tutoriel, vous créez le VPC pour ce scénario qui fonctionne avec une base de données en mode double pile. Le mode double pile active la connexion via le protocole d'adressage IPv6. Pour plus d'informations sur les adresses IP, consultez Adressage IP Amazon RDS.

Les instances de réseau à double pile sont pris en charge dans la plupart des régions. Pour plus d’informations, consultez Disponibilité des régions et des versions. Pour connaître les limites du mode à double pile, consultez Limitations pour les instances de base de données en réseau à double pile.

Le schéma suivant illustre ce scénario.



Pour plus d'informations sur d'autres scénarios, consultez Scénarios d'accès à un d'instances de base de données dans un VPC.

Votre instance de bases de données doit être disponible uniquement pour votre instance Amazon EC2, et non pour l'Internet public. Vous créez ainsi un VPC avec des sous-réseaux publics et privés. L'instance Amazon EC2 est hébergée dans le sous-réseau public, de sorte qu'elle peut accéder à l'Internet public. L'instance de base de données est hébergé(e) dans un sous-réseau privé. L'instance Amazon EC2 peut se connecter à l'instance de base de données car elle est hébergée dans le même VPC. Cependant, l'instance de base de données n'est pas accessible à l'Internet public, ce qui assure une plus grande sécurité.

Ce tutoriel configure un sous-réseau public et privé supplémentaire dans une zone de disponibilité séparée. Ces sous-réseaux ne sont pas utilisés par le tutoriel. Un groupe de sous-réseaux de

base de données RDS nécessite un sous-réseau dans au moins deux zones de disponibilité. Le sous-réseau supplémentaire permet de passer facilement à un déploiement d'instance de base de données Multi-AZ à l'avenir.

Pour créer une instance de base de données qui utilise le mode double pile, spécifiez Dual-stack mode (mode double pile) pour le paramètre Network type (Type de réseau). Vous pouvez également modifier une instance de base de données avec le même paramètre. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS et Modification d'une RDS instance de base de données Amazon.

Ce didacticiel décrit la configuration d'un VPC pour les instances de base de données Amazon RDS. Pour en savoir plus sur Amazon VPC, consultez le Guide de l'utilisateur Amazon VPC.

Créer un VPC avec des sous-réseaux publics et privés

Utilisez la procédure suivante pour créer un VPC avec des sous-réseaux publics et privés.

Pour créer un VPC et des sous-réseaux

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Dans le coin supérieur droit du AWS Management Console, choisissez la région dans laquelle créer votre VPC. L'exemple utilise la région USA Est (Ohio).
3. Dans le coin supérieur gauche, choisissez VPC Dashboard (Tableau de bord VPC). Pour commencer à créer un VPC, sélectionnez Create VPC (Créer un VPC).
4. Pour Resources to create (Ressources à créer) sous VPC settings (Paramètres VPC), choisissez VPC and more (VPC et plus).
5. Pour les valeurs VPC settings (Paramètres VPC) restantes, définissez ce qui suit :

   - Name tag auto-generation (Génération automatique de balise de nom) : **`tutorial-dual-stack`**
   - IPv4 CIDR block (Bloc d'adresse CIDR IPv4) : **`10.0.0.0/16`**
   - IPv6 CIDR block (Bloc d'adresse CIDR IPv6) : Amazon-provided IPv6 CIDR block (Bloc d'adresse CIDR IPv6 fourni par Amazon)
   - Tenancy (Location) : Default (Par défaut)
   - Number of Availability Zones (AZs) [Nombre de zones de disponibilité (AZ)] : 2
   - Customize AZs (Personnaliser les AZ) : conserver les valeurs par défaut.
   - Number of public subnet (Nombre de sous-réseaux publics) : 2

- Number of private subnets (Nombre de sous-réseaux privés) : 2
- Customize subnets CIDR blocks (Personnaliser les blocs CIDR des sous-réseaux) : conserver les valeurs par défaut.
- NAT gateways ($) [Passerelles NAT ($)] : None (Aucune)
- Egress only internet gateway (Passerelle Internet de sortie uniquement) : No (Non)
- VPC endpoints (Points de terminaison VPC) : None (Aucun)
- DNS options (Options DNS) : conservez les valeurs par défaut.

> **Note**
>
> Amazon RDS nécessite au moins deux sous-réseaux dans deux zones de disponibilité différentes pour prendre en charge les déploiements d'instances de base de données Multi-AZ. Ce tutoriel crée un déploiement Mono-AZ, mais l'exigence permet de le convertir facilement en un déploiement d'instance de base de données Multi-AZ à l'avenir.

6. Sélectionnez Create VPC (Créer un VPC).

Créer un groupe de sécurité VPC pour une instance publique Amazon EC2

Ensuite, vous créez un groupe de sécurité pour l'accès public. Pour vous connecter aux instance EC2 publiques de votre VPC, ajoutez des règles entrantes à votre groupe de sécurité VPC qui autorisent le trafic à se connecter depuis Internet.

Pour créer un groupe de sécurité VPC

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Choisissez successivement VPC Dashboard (Tableau de bord VPC), Security Groups (Groupes de sécurité) et Create Security Group (Créer un groupe de sécurité).
3. Sur la page Create Security Group (Créer un groupe de sécurité), définissez les valeurs suivantes :
   - Nom du groupe de sécurité : **`tutorial-dual-stack-securitygroup`**
   - Description : **`Tutorial Dual-Stack Security Group`**

- VPC : choisissez le VPC que vous avez créé précédemment, par exemple : vpc-***identifier*** (tutorial-dual-stack-vpc)

4. Ajoutez des règles entrantes au groupe de sécurité.

   a. Déterminez l'adresse IP à utiliser pour vous connecter aux instances EC2 de votre VPC à l'aide de Secure Shell (SSH).

      `203.0.113.25/32` est un exemple d'adresse IPv4 (Internet Protocol version 4). `2001:db8:1234:1a00::/64` est un exemple de plage d'adresses IPv6 (Internet Protocol version 6).

      Dans de nombreux cas, votre connexion s'effectue via un fournisseur de services Internet (FSI) ou derrière votre pare-feu sans adresse IP statique. Dans ce cas, trouvez la plage d'adresses IP utilisées par les ordinateurs clients.

      > ⚠ Warning
      >
      > Si vous utilisez `0.0.0.0/0` pour IPv4 ou `::0` pour IPv6, vous permettez à toutes les adresses IP d'accéder à vos instances publiques à l'aide de SSH. Cette approche est acceptable pour une brève durée dans un environnement de test, mais n'est pas sécurisée pour les environnements de production. Dans un environnement de production, autorisez uniquement une adresse IP ou une plage d'adresses IP spécifiques à accéder à vos instances.

   b. Dans la section Règles entrantes, choisissez Ajouter une règle.

   c. Définissez les valeurs suivantes pour que votre nouvelle règle entrante autorise l'accès Secure Shell (SSH) à votre instance Amazon EC2. Si vous faites cela, vous pouvez vous connecter à votre instance EC2 pour installer des clients SQL et d'autres applications. Indiquez une adresse IP pour accéder à votre instance EC2 :

      - Type : **`SSH`**
      - Source : l'adresse IP ou la plage de l'étape a. Exemple d'adresse IP IPv4 : **`203.0.113.25/32`**. Exemple d'adresse IP IPv6 : **`2001:DB8::/32`**.

5. Choisissez Create security group (Créer un groupe de sécurité) pour créer le groupe de sécurité.

   Notez l'ID du groupe de sécurité, car vous en aurez besoin ultérieurement dans ce didacticiel.

Créer un groupe de sécurité VPC pour une instance de base de données privé(e)

Pour que votre instance de base de données demeure privé(e), créez un deuxième groupe de sécurité pour l'accès privé. Pour vous connecter aux instances de bases de données privé(e)s de votre VPC, ajoutez des règles entrantes à votre groupe de sécurité VPC. Elles autorisent le trafic provenant de votre instance Amazon EC2 uniquement.

Pour créer un groupe de sécurité VPC

1. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
2. Choisissez successivement VPC Dashboard (Tableau de bord VPC), Security Groups (Groupes de sécurité) et Create Security Group (Créer un groupe de sécurité).
3. Sur la page Create Security Group (Créer un groupe de sécurité), définissez les valeurs suivantes :
   - Nom du groupe de sécurité : **`tutorial-dual-stack-db-securitygroup`**
   - Description : **`Tutorial Dual-Stack DB Instance Security Group`**
   - VPC : choisissez le VPC que vous avez créé précédemment, par exemple : vpc-***`identifier`*** (tutorial-dual-stack-vpc)
4. Ajoutez des règles entrantes au groupe de sécurité :
   a. Dans la section Règles entrantes, choisissez Ajouter une règle.
   b. Définissez les valeurs suivantes pour que votre nouvelle règle entrante autorise le trafic MySQL sur le port 3306 à partir de votre instance Amazon EC2. Dans ce cas, vous pouvez vous connecter de votre instance EC2 à votre instance de bases de données. Cela signifie que vous pouvez envoyer des données de votre instance EC2 vers votre base de données.
      - Type : MySQL/Aurora
      - Source : identifiant du groupe de sécurité tutorial-dual-stack-securitygroup que vous avez créé précédemment dans ce tutoriel, par exemple : sg-9edd5cfb.
5. Pour créer le groupe de sécurité, choisissez Créer un groupe de sécurité.

Création d'un groupe de sous-réseaux DB

Un groupe de sous-réseaux de base de données désigne une collection de sous-réseaux que vous créez dans un VPC et que vous spécifiez alors pour vos instances de bases de données. En utilisant un groupe de sous-réseau de base de données, vous pouvez spécifier un VPC particulier lors de

la création d'instances de base de données. Pour créer un groupe de sous-réseaux de la base de données qui soit compatible DUAL, tous les sous-réseaux doivent être compatibles DUAL. Pour être compatible DUAL, un sous-réseau doit être associé à un CIDR IPv6.

Pour créer un groupe de sous-réseaux

1. Identifiez les sous-réseaux privés pour votre base de données dans le VPC.

   a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.

   b. Choisissez VPC Dashboard (Tableau de bord du VPC), puis Subnets (Sous-réseaux).

   c. Notez les ID des sous-réseaux nommés tutorial-dual-stack-subnet-private1-us-west-2a et tutorial-dual-stack-subnet-private2-us-west-2b.

      Vous aurez besoin des ID de sous-réseau lorsque vous créerez votre groupe de sous-réseau de base de données.

2. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.

   Assurez-vous de vous connecter à la console Amazon RDS et non à la console Amazon VPC.

3. Dans le panneau de navigation, choisissez Subnet groups (Groupes de sous-réseaux).

4. Choisissez Create DB Subnet Group (Créer groupe de sous-réseaux de base de données).

5. Sur la page Create DB subnet group (Créer groupe de sous-réseaux de base de données), définissez ces valeurs dans Subnet group details (Détails de groupe de sous-réseaux) :

   - Nom: **`tutorial-dual-stack-db-subnet-group`**
   - Description: **`Tutorial Dual-Stack DB Subnet Group`**
   - VPC : tutorial-dual-stack-vpc (identifiant vpc)

6. Dans la section Add subnets (Ajouter des sous-réseaux), choisissez des valeurs pour les options Availability Zones (Zones de disponibilité) et Subnets (Sous-réseaux).

   Pour ce tutoriel, choisissez us-east-2a et us-east-2b pour les Availability Zones (Zones de disponibilité). Pour Subnets (Sous-réseaux), choisissez les sous-réseaux privés que vous avez identifiés à l'étape précédente.

7. Sélectionnez Créer.

Votre nouveau groupe de sous-réseaux DB apparaît dans la liste des groupes de sous-réseaux sur la console RDS. Vous pouvez choisir le groupe de sous-réseau de base de données pour afficher

ses détails. Il s'agit notamment des protocoles d'adressage pris en charge, de tous les sous-réseaux associés au groupe et du type de réseau pris en charge par le groupe de sous-réseaux de base de données.

Créer une instance Amazon EC2 en mode double pile

Pour créer une instance Amazon EC2, suivez les instructions de la section Lancer une instance à l'aide du nouvel assistant de lancement d'instance du guide de l'utilisateur Amazon EC2.

Sur la page Configure Instance Details (Configurer les détails d'instance), spécifiez les valeurs suivantes et conservez les valeurs par défaut des autres paramètres :

- Réseau – Choisissez un VPC existant avec des sous-réseaux publics et privés, tels que tutorial-dual-stack-vpc (vpc-*identifier*), créés dans Créer un VPC avec des sous-réseaux publics et privés.
- Sous-réseau — Choisissez un sous-réseau public existant, tel que subnet- ***identifier \| tutorial-dual-stack-subnet -public1-us-east-2a \| us-east-2a*** créé dans. Créer un groupe de sécurité VPC pour une instance publique Amazon EC2
- Auto-assign Public IP (Attribuer automatiquement l'adresse IP publique) : choisissez Enable (Activer).
- Auto-assign IPv6 IP (Affectation automatique de l'IPv6) : choisissez Enable (Activer).
- Firewall (security groups) [Pare-feu (groupes de sécurité)] : choisissez Select an existing security group (Sélectionnez un groupe de sécurité existant).
- Common security groups (Groupes de sécurité communs) : choisissez un groupe de sécurité existant, tel que le `tutorial-securitygroup` créé dans Créer un groupe de sécurité VPC pour une instance publique Amazon EC2. Assurez-vous que le groupe de sécurité que vous choisissez inclut des règles entrantes pour l'accès SSH (Secure Shell) et HTTP.

Création d'un(e) instance de base de données en mode double pile

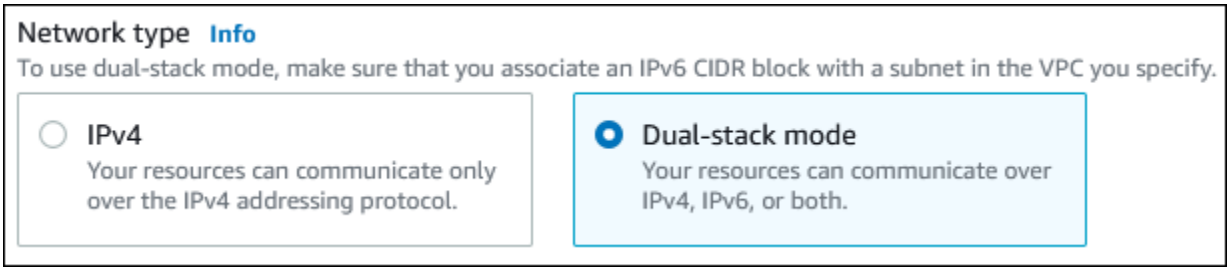
Dans cette étape, vous créez un(e) instance de base de données qui fonctionne en mode double pile.

Pour créer une instance de base de données

1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à l'adresse https://console.aws.amazon.com/rds/.
2. L'exemple utilise la région USA Est (Ohio).

3. Dans le panneau de navigation, choisissez Databases (Bases de données).
4. Choisissez Create database (Créer une base de données).
5. Sur la page Create database (Créer une base de données), vérifiez que l'option Standard create (Création standard) est activée, puis choisissez le type de moteur de base de données MySQL.
6. Dans la section Connectivity (Connectivité), définissez les valeurs suivantes :

   - Network type (Type de réseau) – choisissez Dual-stack mode (Mode double pile)

   - Virtual private cloud (VPC) (Cloud privé virtuel (VPC)) – choisissez un VPC existant avec des sous-réseaux publics et privés, tel que tutorial-dual-stack-vpc (vpc-*identifier*) créé dans Créer un VPC avec des sous-réseaux publics et privés.

     Le VPC doit avoir des sous-réseaux dans des zones de disponibilité différentes.
   - DB Subnet group (Groupe de sous-réseau de la base de données) : choisissez un groupe de sous-réseau de base de données pour le VPC, tel que tutorial-dual-stack-db-subnet-group créé dans Création d'un groupe de sous-réseaux DB.
   - Public access (Accès public) : choisissez No (Non).
   - VPC security group (firewall) [Groupe de sécurité VPC (pare-feu)] : sélectionnez Choose existing (Choisir l'existant).
   - Existing VPC security groups (Groupes de sécurité VPC existants) – choisissez un groupe de sécurité VPC existant qui est configuré pour un accès privé, tel que tutorial-dual-stack-db-securitygroup créé dans Créer un groupe de sécurité VPC pour une instance de base de données privé(e).

     Supprimez les autres groupes de sécurité, tels que le groupe de sécurité par défaut, en cliquant sur le signe X qui lui est associé.
   - Availability Zone (Zone de disponibilité) : choisissez us-west-2a.

     Pour éviter le trafic inter-zones, assurez-vous que l'instance de base de données et l'instance EC2 se trouvent dans la même zone de disponibilité.

7. Pour les sections restantes, spécifiez vos paramètres d'instance de base de données. Pour obtenir des informations sur chaque paramètre, consultez Paramètres des instances de base de données.

Se connecter à votre instance Amazon EC2 et à votre instance de base de données

Une fois votre instance Amazon EC2 et votre instance de base de données créées en mode double pile, vous pouvez vous connecter à chacune d'elles à l'aide du protocole IPv6. Pour vous connecter à une instance Amazon EC2 à l'aide du protocole IPv6, suivez les instructions de la section Connexion à votre instance Linux dans le guide de l'utilisateur Amazon EC2.

Pour vous connecter à votre instance de base de données RDS for MySQL depuis l'instance Amazon EC2, suivez les instructions dans Se connecter à une instance de base de données MySQL.

Suppression du VPC

Après avoir créé le VPC et d'autres ressources pour ce didacticiel, vous pouvez les supprimer si elles ne sont plus nécessaires.

Si vous avez ajouté des ressources dans le VPC que vous avez créé pour ce tutoriel, vous devrez peut-être les supprimer avant de pouvoir supprimer le VPC. Les instances Amazon EC2 ou les instances de bases de données sont des exemples de ressources. Pour plus d'informations, consultez Supprimer votre VPC dans le Guide de l'utilisateur Amazon VPC.

Pour supprimer un VPC et les ressources associées

1. Supprimez le groupe de sous-réseaux de base de données :
    a. Ouvrez la console Amazon RDS à l'adresse https://console.aws.amazon.com/rds/.
    b. Dans le panneau de navigation, choisissez Subnet groups (Groupes de sous-réseaux).
    c. Sélectionnez le groupe de sous-réseaux de base de données à supprimer, par exemple tutorial-db-subnet-group.
    d. Choisissez Supprimer, puis Supprimer dans la fenêtre de confirmation.
2. Notez l'ID du VPC :
    a. Ouvrez la console Amazon VPC à l’adresse https://console.aws.amazon.com/vpc/.
    b. Choisissez Tableau de bord du VPC, puis VPC.
    c. Dans la liste, identifiez le VPC que vous avez créé, tel que tutorial-dual-stack-vpc.

d. Notez la valeur VPC ID (ID de VPC) du VPC que vous avez créé. Il vous servira dans les étapes suivantes.

3. Supprimez les groupes de sécurité :

   a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
   b. Choisissez Tableau de bord du VPC, puis Groupes de sécurité.
   c. Sélectionnez le groupe de sécurité pour l'instance de base de données Amazon RDS, par exemple tutorial-dual-stack-db-securitygroup.
   d. Pour Actions, choisissez Delete security groups (Supprimer des groupes de sécurité), puis Delete (Supprimer) sur la page de confirmation.
   e. Sur la page Security Groups (Groupes de sécurité), sélectionnez le groupe de sécurité pour l'instance Amazon EC2, par exemple tutorial-dual-stack-securitygroup.
   f. Pour Actions, choisissez Delete security groups (Supprimer des groupes de sécurité), puis Delete (Supprimer) sur la page de confirmation.

4. Supprimez la passerelle NAT :

   a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
   b. Choisissez Tableau de bord du VPC, puis Passerelles NAT.
   c. Sélectionnez la passerelle NAT du VPC que vous avez créé. Utilisez l'ID de VPC pour identifier la passerelle NAT correcte.
   d. Pour Actions, choisissez Delete NAT gateway (Supprimer la Passerelle NAT).
   e. Sur la page de confirmation, entrez **`delete`** et choisissez Supprimer.

5. Supprimer le VPC

   a. Ouvrez la console Amazon VPC à l'adresse https://console.aws.amazon.com/vpc/.
   b. Choisissez Tableau de bord du VPC, puis VPC.
   c. Sélectionnez le VPC que vous voulez supprimer, tel que tutorial-dual-stack-vpc.
   d. Pour Actions, choisissez Supprimer le VPC.

      La page de confirmation affiche les autres ressources associées au VPC qui seront également supprimées, y compris les sous-réseaux qui lui sont associés.

   e. Sur la page de confirmation, entrez **`delete`** et choisissez Supprimer.

6. Libérez les adresses IP élastiques :

   a. Ouvrez la console Amazon EC2 à l'adresse https://console.aws.amazon.com/ec2/.

b. Choisissez Tableau de bord EC2, puis Adresses IP Elastic.

c. Sélectionnez l'adresse IP élastique à libérer.

d. Pour Actions, choisissez Release Elastic IP addresses (Libérer les adresses IP élastiques).

e. Sur la page de confirmation, sélectionnez Libérer.

Déplacement vers un VPC d'une instance de base de données n'appartenant pas à un VPC

Certaines instances de base de données héritées de la plateforme EC2-Classic ne sont pas dans un VPC. Si votre instance de base de données n'est pas dans un VPC, vous pouvez utiliser AWS Management Console pour déplacer facilement votre instance de base de données vers un VPC. Avant de déplacer une instance de base de données, qui n'est pas dans un VPC, vers un VPC, vous devez créer le VPC.

EC2-Classic a été retiré le 15 août 2022. Si vous n'avez pas migré d'EC2-Classic vers un VPC, nous vous recommandons de le faire dès que possible. Pour plus d'informations, consultez Migrer d'EC2-Classic vers un VPC dans le Guide de l'utilisateur Amazon EC2 et le blog EC2-Classic Networking is Retiring – Here's How to Prepare.

Important

Si vous êtes un nouveau client Amazon RDS, si vous n'avez jamais créé d'instance de base de données auparavant ou si vous créez une instance de base de données dans une région AWS que vous n'avez pas encore utilisée, il est fort probable que vous soyez sur la plateforme EC2-VPC et que vous ayez un VPC par défaut. Pour plus d'informations sur l'utilisation des instances de base de données dans un VPC, consultez Utilisation d'un(e) instance de base de données dans un VPC.

Suivez les étapes ci-après pour créer un VPC pour votre instance de base de données.

- Étape 1 : Création d'un VPC
- Étape 2 : créer un groupe de sous-réseaux de base de données
- Étape 3 : créer un groupe de sécurité VPC

Une fois que vous avez créé le VPC, procédez comme suit pour déplacer votre instance de base de données vers le VPC.

- Mettre à jour le VPC pour une instance de base de données

Nous vous recommandons vivement de créer une sauvegarde de votre instance de base de données immédiatement avant la migration. Cela garantit que vous serez en mesure de restaurer les données en cas d'échec de la migration. Pour plus d'informations, consultez Sauvegarde, restauration et exportation de données.

Voici quelques limitations pour le déplacement de votre instance de base de données vers le VPC.

- Classes d'instance de base de données de génération précédente – Les classes d'instance de base de données de génération précédente peuvent ne pas être prises en charge sur la plateforme de VPC. Lorsque vous déplacez une instance de base de données vers un VPC, choisissez une classe d'instance de base de données db.m3 ou db.r3. Après avoir déplacé l'instance de base de données vers un VPC, vous pouvez mettre à l'échelle l'instance de base de données pour utiliser une classe d'instance de base de données ultérieure. Pour obtenir la liste complète des classes d'instance prises en charge par les VPC, veuillez consulter Types d'instance Amazon RDS.
- Multi-AZ – Le déplacement vers un VPC d'une instance de base de données Multi-AZ ne se trouvant pas dans un VPC n'est pas pris en charge actuellement. Pour déplacer votre instance de base de données vers un VPC, modifiez d'abord l'instance de base de données de sorte qu'il s'agisse d'un déploiement mono-AZ. Définissez le paramètre Déploiement multi-AZ sur Non. Après avoir déplacé l'instance de base de données vers un VPC, modifiez-la à nouveau de manière à en faire un déploiement multi-AZ. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.
- Réplicas en lecture – Le déplacement vers un VPC d'une instance de base de données avec des réplicas en lecture ne se trouvant pas dans un VPC n'est pas pris en charge actuellement. Pour déplacer votre instance de base de données vers un VPC, supprimez d'abord tous ses réplicas de lecture. Après avoir déplacé l'instance de base de données vers un VPC, recréez les réplicas en lecture. Pour plus d'informations, consultez Utilisation des réplicas en lecture d'instance de base de données.
- Groupes d'options – Si vous déplacez votre instance de base de données vers un VPC et que l'instance de base de données utilise un groupe d'options personnalisé, modifiez le groupe d'options associé à votre instance de base de données. Les groupes d'options sont propres à la plateforme, et le déplacement vers un VPC est une modification de plateforme. Pour utiliser un groupe d'options personnalisé dans ce cas, attribuez le groupe d'options VPC par défaut à l'instance de base de données, assignez à l'instance de base de données un groupe d'options utilisé par d'autres instances de base de données dans le VPC vers lequel vous effectuez le déplacement, ou créez un groupe d'options et assignez-le à l'instance de base de données. Pour plus d'informations, consultez Utilisation de groupes d'options.

Alternatives pour le déplacement vers un VPC d'une instance de base de données ne se trouvant pas dans un VPC avec un temps d'arrêt minimal

Les alternatives suivantes permettent de déplacer vers un VPC une instance de base de données ne se trouvant pas dans un VPC avec un temps d'arrêt minimal. Ces alternatives entraînent une perturbation minimale de l'instance de base de données source et lui permettent de servir le trafic utilisateur pendant la migration. Toutefois, le temps nécessaire à la migration vers un VPC varie en fonction de la taille de la base de données et des caractéristiques de la charge de travail active.

- AWS Database Migration Service (AWS DMS) – AWS DMS permet la migration dynamique des données tout en maintenant l'instance de base de données source pleinement opérationnelle, mais réplique uniquement un ensemble limité d'instructions DDL. AWS DMS ne propage pas les éléments tels que les index, les utilisateurs, les privilèges, les procédures stockées et d'autres modifications de base de données qui ne sont pas directement liées aux données de table. En outre, AWS DMS n'utilise pas automatiquement les instantanés RDS pour la création de l'instance de base de données initiale, ce qui peut augmenter le temps de migration. Pour plus d'informations, consultez AWS Database Migration Service.
- Restauration d'instantanés de base de données ou restauration ponctuelle – Vous pouvez déplacer vers un VPC une instance de base de données en restaurant un instantané de l'instance de base de données ou en restaurant une instance de base de données à un moment donné. Pour plus d'informations, consultez Restauration vers une instance de base de données et Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Quotas et contraintes pour Amazon RDS

Vous trouverez ci-après une description des quotas de ressources et des contraintes d'attribution de noms pour Amazon RDS.

Rubriques

- Quotas dans Amazon RDS
- Contraintes d'affectation de noms dans Amazon RDS
- Nombre maximal de connexions à une base de données
- Limites de taille des fichiers dans Amazon RDS

Quotas dans Amazon RDS

Chaque AWS compte dispose de quotas, pour chaque AWS région, sur le nombre de ressources Amazon RDS qui peuvent être créées. Une fois qu'un quota de ressource a été atteint, les appels supplémentaires pour créer cette ressource échouent avec une exception.

Le tableau suivant répertorie les ressources et leurs quotas par AWS région.

Nom	Par défaut	Ajusta	Description
Autorisations par groupe de sécurité de base de données	Chaque Région prise en charge : 20	Non	Nombre d'autorisations de groupe de sécurité par groupe de sécurité de base de données
Points de terminaison personnalisés par cluster de bases de données	Chaque Région prise en charge : 5	Oui	Le nombre maximum de points de terminaison personnalisés que vous pouvez créer par cluster de base de données Aurora dans ce compte dans la région actuelle. Cette valeur reflète le plus grand nombre de points de terminaison

Nom	Par défaut	Ajusta	Description
			personnalisés dans un cluster de base de données du compte. Les autres clusters de base de données du compte peuvent avoir un nombre inférieur de points de terminaison personnalisés.
Versions de moteur personnalisées	Chaque Région prise en charge : 40	Oui	Nombre maximal de versions de moteur personnalisées autorisées sur ce compte dans la région actuelle
Groupes de paramètres de cluster DB	Chaque Région prise en charge : 50	Oui	Le nombre maximum de groupes de paramètres de cluster de base de données
Clusters de bases de données	Chaque Région prise en charge : 40	Oui	Le nombre maximum de clusters Aurora autorisés sur ce compte dans la région actuelle
Instances de base de données	ap-south-1 : 20 Chacune des autres régions prises en charge : 40	Oui	Le nombre maximum d'instances de base de données autorisées dans ce compte dans la région actuelle

Nom	Par défaut	Ajusta	Description
groupes de partitions de base de données	Chaque Région prise en charge : 5	Oui	Le nombre maximal de groupes de partitions de base de données pour la base de données Aurora Limitless dans ce compte dans la région actuelle
Groupes de sous-réseaux DB	ap-south-1 : 20 Chacune des autres régions prises en charge : 50	Oui	Nombre maximal de groupes de sous-réseaux de base de données
Taille du corps de requête HTTP de l'API de données	Toutes les Régions prises en charge : 4 mégaoctets	Non	Taille maximale autorisée pour le corps de la demande HTTP.
Nombre maximal de paires cluster-secret simultanées de l'API de données	Chaque Région prise en charge : 30	Non	Nombre maximal de paires uniques de clusters de base de données Aurora Serverless v1 et de secrets dans les demandes d'API de données simultanées pour ce compte dans la AWS région actuelle.

Nom	Par défaut	Ajusta	Description
Nombre maximal de requêtes simultanées de l'API de données	Chaque Région prise en charge : 500	Non	Nombre maximal de demandes d'API de données adressées à un cluster de base de données Aurora Serverless v1 qui utilisent le même secret et peuvent être traitées en même temps. Les demandes supplémentaires sont mises en file d'attente et traitées à mesure que les demandes en cours de traitement sont terminées.
Taille maximale du jeu de résultats d'API de données	Chaque Région prise en charge : 1 mégaoctet	Non	Taille maximale du jeu de résultats de base de données pouvant être renvoyé par l'API de données.
Taille maximale de l'API de données de la chaîne de réponse JSON	Toutes les régions prises en charge : 10 mégaoctets	Non	Taille maximale de la chaîne de réponse JSON simplifiée renvoyée par l'API de données RDS.

Nom	Par défaut	Ajusta	Description
Demandes d'API de données par seconde	Chaque Région prise en charge : 1 000 par seconde	Non	Le nombre maximal de demandes à l'API de données par seconde autorisé pour ce compte dans la AWS région actuelle. Ce quota s'applique uniquemen t aux clusters Amazon Aurora Serverless v1.
Abonnements aux événements	Chaque Région prise en charge : 20	Oui	Le nombre maximum d'abonnements à des événements
Rôles IAM par cluster de bases de données	Chaque Région prise en charge : 5	Oui	Le nombre maximum de rôles IAM associés à un cluster de base de données
Rôles IAM par instance de base de données	Chaque Région prise en charge : 5	Oui	Le nombre maximum de rôles IAM associés à une instance de base de données
Intégrations	Chaque Région prise en charge : 100	Non	Le nombre maximum d'intégrations autorisées dans ce compte dans la région actuelle AWS
Instantané de cluster de bases de données manuel	Chaque Région prise en charge : 100	Oui	Le nombre maximum d'instantanés manuels du cluster de base de données

Nom	Par défaut	Ajusta	Description
Instantanés d'instance de base de données manuels	Chaque Région prise en charge : 100	Oui	Le nombre maximum d'instantanés manuels de l'instance de base de données
Groupes d'options	Chaque Région prise en charge : 20	Oui	Le nombre maximum de groupes d'options
Groupes de paramètres	ap-south-1 : 20 Chacune des autres régions prises en charge : 50	Oui	Le nombre maximum de groupes de paramètres
Proxys	Chaque Région prise en charge : 20	Oui	Le nombre maximum de proxys autorisés sur ce compte dans la région actuelle AWS
Réplicas en lecture par principale	Chaque région prise en charge : 15	Oui	Le nombre maximum de réplicas en lecture par instance de base de données principale. Ce quota ne peut pas être ajusté pour Amazon Aurora.
Instances de base de données réservées	Chaque Région prise en charge : 40	Oui	Le nombre maximum d'instances de base de données réservées autorisées dans ce compte dans la AWS région actuelle

Nom	Par défaut	Ajusta	Description
Groupes de sécurité	ap-south-1 : 20 Chacune des autres régions prises en charge : 25	Oui	Le nombre maximum de groupes de sécurité de base de données
Sous-réseaux par groupe de sous-réseaux de base de données	Chaque Région prise en charge : 20	Non	Le nombre maximum de sous-réseaux par groupe de sous-réseaux de base de données
Stockage total pour toutes les instances de base de données	Chaque Région prise en charge : 100 000 gigaoctets	Oui	Le stockage total maximal (en Go) sur les volumes EBS pour toutes les instances de base de données Amazon RDS additionnées. Ce quota ne s'applique pas à Amazon Aurora, dont le volume de cluster maximal est de 128 TiB pour chaque cluster de base de données.

Note

Par défaut, vous pouvez avoir jusqu'à 40 instances de bases de données. Les instances de base de données RDS, les instances de base de données Aurora, les instances Amazon Neptune et les instances Amazon DocumentDB sont concernées par ce quota.
Les limitations suivantes s'appliquent aux instances de bases de données Amazon RDS :

- 10 instances de chaque édition SQL Server (Enterprise, Standard, Web et Express) sous le modèle « license-included (licence incluse) »
- 10 instances pour Oracle sous le modèle « license-included (licence incluse) »

- 40 pour Db2 dans le cadre du modèle de bring-your-own-license licence « » (BYOL)
- 40 instances pour MySQL, MariaDB ou PostgreSQL
- 40 pour Oracle dans le cadre du modèle de licence bring-your-own-license « » (BYOL)

Si votre application nécessite plus d'instances de base de données, vous pouvez demander des instances de base de données supplémentaires en ouvrant la console Service Quotas. Dans le volet de navigation, choisissez Services AWS . Choisissez Amazon Relational Database Service (Amazon RDS), choisissez un quota et suivez les instructions pour demander une augmentation de quota. Pour de plus amples informations, veuillez consulter Demande d'augmentation de quota dans le Guide de l'utilisateur Service Quotas.
Pour RDS for Oracle, vous pouvez créer jusqu'à 15 répliques de lecture par instance de base de données source dans chaque région, mais nous recommandons de limiter le nombre de répliques à 5 afin de minimiser le délai de réplication.
Les sauvegardes gérées par AWS Backup sont considérées comme des instantanés de de base de données manuels, mais ne sont pas prises en compte dans le quota de snapshots de manuel. Pour plus d'informations à ce sujet AWS Backup, consultez le guide du AWS Backup développeur.

Notez que le quota par défaut pour les sauvegardes automatisées entre régions est de 20 pour un Compte AWS, et le quota par défaut pour le nombre de demandes de copie d'instantanés simultanées est de 20 pour chaque région pour un Compte AWS. RDS ne peut pas ajuster ce quota pour les éléments suivants : Régions AWS

- Afrique (Le Cap)
- Asie-Pacifique (Hong Kong)
- Asie-Pacifique (Hyderabad)
- Asie-Pacifique (Melbourne)
- Canada-Ouest (Calgary)
- Europe (Zurich)

Si vous utilisez une opération d'API RDS et dépassez le quota par défaut pour le nombre d'appels par seconde, l'API Amazon RDS émet une erreur similaire à la suivante.

ClientError: Une erreur s'est produite (ThrottlingException) lors de l'appel de l'*API_name*opération : Débit dépassé.

Réduisez ici le nombre d'appels par seconde. Le quota est destiné à couvrir la plupart des cas d'utilisation. Si des quotas plus élevés sont nécessaires, vous pouvez demander une augmentation de quota en utilisant l'une des options suivantes :

- Depuis la console, ouvrez la console Service Quotas.
- À partir de AWS CLI, utilisez la request-service-quota-increase AWS CLI commande.

Pour plus d'informations, consultez le Guide de l'utilisateur Service Quotas.

Contraintes d'affectation de noms dans Amazon RDS

Les contraintes de dénomination dans Amazon RDS () sont les suivantes :

- Identifiant du base de données :
  - Doit contenir entre 1 et 63 caractères alphanumériques ou traits d'union.
  - Le premier caractère doit être une lettre.
  - Il ne peut pas se terminer par un trait d'union ou contenir deux traits d'union consécutifs.
  - Doit être unique pour toutes les instances de base de données par AWS compte et par AWS région.
- Nom initial de la base de données :
  - Les contraintes relatives aux noms de bases de données diffèrent pour chaque moteur de base de données. Pour plus d'informations, consultez les paramètres disponibles lors de la création de chaque instance de base de données.
  - SQL Server — Créez vos bases de données après avoir créé votre instance de base de données.
- Nom d'utilisateur principal : les contraintes relatives au nom d'utilisateur principal diffèrent pour chaque moteur de base de données. Pour plus d'informations, consultez les paramètres disponibles lors de la création du d'instances de base de données.
- Mot de passe principal :
  - Le mot de passe de l'utilisateur principal de la base de données peut contenir tout caractère ASCII imprimable à l'exception de `/`, `'`, `"`, `@`, ou d'un espace.

    Pour Oracle, `&` est une limite de caractères supplémentaire.

  - Le mot de passe peut contenir le nombre suivant de caractères ASCII imprimables en fonction du moteur de base de données :
    - DB2 : 8—255
    - MariaDB et MySQL : entre 8 et 41
    - Oracle : entre 8 et 30
    - SQL Server et PostgreSQL : entre 8 et 128
- Groupe de paramètres de base de données :
  - Ils doivent contenir entre 1 et 255 caractères alphanumériques.
  - Le premier caractère doit être une lettre.
  - Les traits d'union sont autorisés, mais le nom ne peut pas se terminer par un trait d'union ni contenir deux traits d'union consécutifs.
- Groupe de sous-réseaux de base de données :
  - Il doivent contenir entre 1 et 255 caractères.
  - Les caractères alphanumériques, les espaces, les traits d'union, les traits de soulignement et les points sont autorisés.

Nombre maximal de connexions à une base de données

Le nombre maximal de connexions simultanées à une base de données varie selon le type de moteur de base de données et l'allocation de mémoire pour la classe d'instance de base de données. Le nombre maximal de connexions est généralement défini dans le groupe de paramètres associé à l'instance base de données. L'exception est Microsoft SQL Server, pour lequel il est défini dans les propriétés du serveur de l'instance de base de données dans SQL Server Management Studio (SSMS).

Les connexions de base de données consomment de la mémoire La définition d'une valeur trop élevée pour l'un de ces paramètres peut entraîner une condition de mémoire insuffisante ayant pour effet qu'une instance de base de données passe à l'état incompatible-parameters. Pour de plus amples informations, veuillez consulter Diagnostic et résolution d'un état de paramètres incompatibles pour une limite de mémoire.

Si vos applications ouvrent et ferment régulièrement des connexions, ou si elles ont ouvert un grand nombre de connexions de longue durée, nous vous recommandons d'utiliser Proxy Amazon RDS. RDS Proxy est un proxy de base de données entièrement géré et hautement disponible qui utilise

le regroupement de connexions pour partager les connexions de base de données de manière sécurisée et efficace. Pour en savoir plus sur RDS Proxy, consultez Proxy Amazon RDS .

> **Note**
>
> Pour Oracle, vous définissez le nombre maximal de processus utilisateur, et de sessions utilisateur et système.
> Pour DB2, vous ne pouvez pas définir le nombre maximum de connexions. La limite est de 64 000.

Le tableau suivant présente des informations sur le nombre maximal de connexions à la base de données pour les différents moteurs de base de données.

Moteur de base de données	Paramètre	Valeurs autorisées	Valeur par défaut	Description
MariaDB	`max_connections`	1–100000	• Pour MariaDB 10.5 et versions supérieures, la valeur par défaut est : AU MOINS ({DBInstanceClassMemory/25165760} ,12000) La formule est effectivement équivalente à MB/25. Si le calcul de la valeur par défaut aboutit à une valeur supérieure à 12 000, Amazon RDS définit la limite à 12 000. • Pour la version 10.4 de MariaDB :	Nombre de connexions client simultanées autorisées

Moteur de base de données	Paramètre	Valeurs autorisées	Valeur par défaut	Description
			{DBInstanceClassMemory/12582880} La formule est effectivement équivalente à MB/12.	
MySQL	`max_connections`	1–100000	{DBInstanceClassMemory/12582880} La formule est effectivement équivalente à MB/12.	Nombre de connexions client simultanées autorisées
Oracle	`processes`	80–20000	AU MOINS ({DBInstanceClassMemory/9868951}, 20000)	Processus utilisateur
Oracle	`sessions`	100–65535	Ne s'applique pas	Sessions utilisateur et système
PostgreSQL	`max_connections`	6—262143	AU MOINS ({DBInstanceClassMemory/9531392}, 5000)	Nombre maximal de connexions simultanées
SQL Server	`user connections`	0–32767	0 (illimité)	Nombre maximum de connexions simultanées. Pour plus d'informations, voir Configuration des connexions utilisateur (option de configuration du serveur).

`DBInstanceClassMemory` est en octets. Pour plus de détails sur le calcul de cette valeur, veuillez consulter Spécification des paramètres de base de données. En raison de la mémoire réservée au système d'exploitation et aux processus de gestion RDS, cette taille de mémoire est inférieure à la valeur en gibioctets (Gio) indiquée dans Spécifications matérielles pour les classes d'instance de base de données .

Par exemple, certaines classes d'instance de base de données disposent de 8 Gio de mémoire, soit 8 589 934 592 octets. Pour une instance de base de données MySQL s'exécutant sur une classe d'instance de base de données avec 8 Gio de mémoire, telle que db.m7g.large, l'équation qui utilise la mémoire totale serait `8589934592/12582880=683`. Or, la variable `DBInstanceClassMemory` soustrait automatiquement les quantités réservées au système d'exploitation et aux processus RDS qui gèrent l'instance de base de données. Le reste de la soustraction est ensuite divisé par 12 582 880, Ce calcul donne une valeur approximative de 630 pour `max_connections` au lieu de 683. Cette valeur varie en fonction de la classe d'instance de base de données et du moteur de base de données.

Lorsqu'une instance de base de données MariaDB ou MySQL s'exécute sur une classe d'instance de base de données de petite taille, comme db.t3.micro ou db.t3.small, la quantité totale de mémoire disponible est faible. Pour ces classes d'instance de base de données, RDS réserve une part importante de la mémoire disponible, ce qui affecte la valeur de `max_connections`. Par exemple, le nombre maximal de connexions par défaut pour une instance de base de données MySQL s'exécutant sur une classe d'instance de base de données db.t3.micro est d'environ 60. Vous pouvez déterminer la valeur de `max_connections` pour votre instance de base de données MariaDB ou MySQL en vous y connectant et en exécutant la commande SQL suivante :

```
SHOW GLOBAL VARIABLES LIKE 'max_connections';
```

Limites de taille des fichiers dans Amazon RDS

Des limites de taille de fichier s'appliquent à certaines instances de base de données Amazon RDS. Pour de plus amples informations, veuillez consulter les limites spécifiques aux moteurs suivantes :

- Limites de taille des fichiers MariaDB dans Amazon RDS
- Limites de taille des fichiers MySQL dans Amazon RDS
- Limites de taille des fichiers Oracle dans Amazon RDS

Dépannage d'Amazon RDS

Utilisez les sections suivantes pour résoudre les problèmes que vous rencontrez avec les instances de base de données dans Amazon RDS et Amazon Aurora.

Rubriques

- Impossible de se connecter à l'instance de base de données Amazon RDS
- Problèmes de sécurité Amazon RDS
- Résolution des problèmes liés à l'état de réseau incompatible
- Réinitialisation du mot de passe du propriétaire de l'instance de base de données
- Panne ou redémarrage d'une instance de base de donnéesAmazon RDS
- Modifications de paramètre de base de données Amazon RDS n'entrant pas en vigueur
- Manque d'espace de stockage de l'instance de base de données Amazon RDS
- Amazon RDS : nombre insuffisant d'instances de base de données disponibles
- Problèmes liés à la mémoire libérable dans Amazon RDS
- Problèmes MySQL et MariaDB
- Impossible de définir la période de rétention des sauvegardes sur 0

Pour de plus amples informations sur le débogage des problèmes à l'aide de l'API Amazon RDS, veuillez consulter Applications de dépannage sur Amazon RDS.

Impossible de se connecter à l'instance de base de données Amazon RDS

Voici des causes courantes empêchant la connexion à une instance de base de données :

- Règles entrantes – Les règles d'accès appliquées par votre pare-feu local et les adresses IP autorisées à accéder à votre instance de base de données peuvent ne pas correspondre. Le problème est probablement lié aux règles entrantes de votre groupe de sécurité.

  Par défaut, les instances de base de données n'autorisent pas l'accès. L'accès est accordé via un groupe de sécurité associé au VPC qui autorise le trafic entrant et sortant de l'instance de base de données. Si nécessaire, ajoutez au groupe de sécurité des règles entrantes et sortantes pour votre

situation. Vous pouvez indiquer une adresse IP, une plage d'adresses IP ou un autre groupe de sécurité VPC.

> **Note**
>
> Lorsque vous ajoutez une nouvelle règle entrante, vous pouvez choisir Mon adresse IP pour Source afin d'autoriser l'accès à l'instance de base de données à partir de l'adresse IP détectée dans votre navigateur.

Pour de plus amples informations sur la configuration des groupes de sécurité, veuillez consulter Créer un groupe de sécurité qui autorise l'accès à votre instance de base de données dans votre VPC.

> **Note**
>
> Les connexions client à partir d'adresses IP dans la plage 169.254.0.0/16 ne sont pas autorisées. Il s'agit d'une plage d'adresses IP privées automatiques (APIPA, Automatic Private IP Addressing Range), qui est utilisée pour l'adressage de liens locaux.

- Accessibilité publique – Pour vous connecter à votre instance de base de données depuis l'extérieur du VPC, par exemple en utilisant une application cliente, une adresse IP publique doit lui être attribuée.

  Pour rendre l'instance accessible au public, modifiez-la et choisissez Oui sous Public accessibility (Accessibilité publique). Pour de plus amples informations, veuillez consulter Masquer un(e) instance de base de données dans un VPC depuis Internet.
- Port : le port que vous avez spécifié lors de la création de l'instance de base de données ne peut pas être utilisé pour envoyer ou recevoir des communications en raison des restrictions de votre pare-feu local. Pour déterminer si votre réseau autorise l'utilisation du port spécifié pour les communications entrantes et sortantes, vérifiez auprès de votre administrateur réseau.
- Disponibilité – Pour une instance de base de données récemment créée, celle-ci a un état `creating` (création) jusqu'à ce qu'elle soit prête à l'emploi. Lorsque l'état devient `available` (disponible), vous pouvez vous connecter à l'instance de base de données. Selon la taille de votre instance de base de données, vous devez parfois patienter une vingtaine de minutes avant qu'elle ne soit disponible.

- Passerelle Internet – Pour qu'une instance de base de données soit publiquement accessible, les sous-réseaux de son groupe de sous-réseaux de base de données doivent avoir une passerelle Internet.

  Pour configurer une passerelle Internet pour un sous-réseau

  1. Connectez-vous à la console Amazon RDS AWS Management Console et ouvrez-la à https://console.aws.amazon.com/rds/l'adresse.
  2. Dans le panneau de navigation, sélectionnez Bases de données, puis sélectionnez le nom de l'instance de base de données.
  3. Dans l'onglet Connectivity & security (Connectivité et sécurité), notez les valeurs de l'ID du VPC sous VPC et de l'ID du sous-réseau sous Sous-réseaux (subnets).
  4. Ouvrez la console Amazon VPC à l’adresse https://console.aws.amazon.com/vpc/.
  5. Dans le panneau de navigation, choisissez Passerelles Internet. Vérifiez qu'il existe une passerelle Internet attachée à votre VPC. Sinon, choisissez Créer une passerelle Internet pour créer une passerelle Internet. Sélectionnez la passerelle Internet, puis choisissez Attacher au VPC et suivez les instructions pour l'attacher à votre VPC.
  6. Dans le panneau de navigation, sélectionnez Sous-réseaux, puis sélectionnez votre sous-réseau.
  7. Dans l'onglet Table de routage, vérifiez qu'il existe une route avec `0.0.0.0/0` comme destination et la passerelle Internet pour votre VPC comme cible.

     Si vous vous connectez à votre instance à l'aide de son IPv6 adresse, vérifiez qu'il existe un itinéraire pour tout le IPv6 trafic (`::/0`) qui pointe vers la passerelle Internet. Sinon, procédez comme suit :

     a. Choisissez l'ID de la table de routage (rtb-xxxxxxxx) pour accéder à cette dernière.
     b. Dans l'onglet Routes, choisissez Edit routes (Modifier les routes). Choisissez Add route (Ajouter une route) et utilisez `0.0.0.0/0` comme destination et la passerelle Internet comme cible.

        Pour IPv6, choisissez Ajouter un itinéraire, utilisez `::/0` comme destination et la passerelle Internet comme cible.
     c. Choisissez Save routes (Enregistrer les acheminements).

En outre, si vous essayez de vous connecter au IPv6 point de terminaison, assurez-vous que la plage d' IPv6 adresses client est autorisée à se connecter à l'instance de base de données.

Pour de plus amples informations, veuillez consulter Utilisation d'un(e) instance de base de données dans un VPC.

Pour les problèmes de connexion spécifiques au moteur, consultez les rubriques suivantes :

- Résolution des problèmes de connexion à votre instance de base de données de SQL serveur
- Résolution des problèmes de connexion à votre instance de base de données Oracle
- Résolution des problèmes de connexion à votre instance RDS for PostgreSQL
- Maximum de connexions MySQL et MariaDB

Test d'une connexion à une instance de base de données

Vous pouvez tester votre connexion à une instance de base de données à l'aide des outils courants Linux ou Microsoft Windows.

Depuis un terminal Linux ou Unix, vous pouvez tester la connexion en saisissant les informations suivantes. Remplacez *DB-instance-endpoint* par le point de terminaison et *port* par le port de votre instance de base de données.

```
nc -zv DB-instance-endpoint port
```

Par exemple, le code suivant illustre un exemple de commande et la valeur renvoyée.

```
nc -zv postgresql1.c6c8mn7fake0.us-west-2.rds.amazonaws.com 8299

  Connection to postgresql1.c6c8mn7fake0.us-west-2.rds.amazonaws.com 8299 port [tcp/
vvr-data] succeeded!
```

Les utilisateurs Windows peuvent utiliser Telnet pour tester la connexion à une instance de base de données. Les actions Telnet ne sont prises en charge que pour le test de la connexion. Si l'opération aboutit, l'action ne retourne aucun message. Si une connexion n'aboutit pas, vous recevez un message d'erreur similaire au message suivant.

```
C:\>telnet sg-postgresql1.c6c8mntfake0.us-west-2.rds.amazonaws.com 819

 Connecting To sg-postgresql1.c6c8mntfake0.us-west-2.rds.amazonaws.com...Could not
open
 connection to the host, on port 819: Connect failed
```

Si les actions Telnet aboutissent, votre groupe de sécurité est correctement configuré.

> **Note**
>
> Amazon RDS n'accepte pas le trafic ICMP (Internet Control Message Protocol), y compris ping.

Dépannage des problèmes d'authentification de connexion

Dans certains cas, vous pouvez vous connecter à votre instance de base de données, mais vous obtenez des erreurs d'authentification. Dans ce cas, il se peut que vous vouliez réinitialiser le mot de passe utilisateur maître de l'instance de base de données. Vous pouvez modifier l'instance RDS pour ce faire.

Pour de plus amples informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Problèmes de sécurité Amazon RDS

Pour éviter les problèmes de sécurité, n'utilisez jamais votre adresse e-mail d'utilisateur Compte AWS root ni le mot de passe d'un compte utilisateur. La meilleure pratique consiste à utiliser votre utilisateur root pour créer des utilisateurs et les attribuer à des comptes utilisateur de base de données. Vous pouvez également utiliser votre utilisateur root pour créer d'autres comptes utilisateur, si nécessaire.

Pour plus d'informations sur la création d'utilisateurs, consultez Création d'un utilisateur IAM dans votre Compte AWS. Pour plus d'informations sur la création d'utilisateurs dans AWS IAM Identity Center, voir Gérer les identités dans IAM Identity Center.

Message d'erreur « Échec de l'extraction des attributs du compte. Certaines fonctions de la console sont peut être dégradées. »

Plusieurs raisons peuvent expliquer cette erreur. Cela peut être dû au fait que votre compte ne dispose pas de certaines autorisations ou que votre compte n'a pas été correctement configuré. Si votre compte est nouveau, vous n'avez peut-être pas attendu qu'il soit prêt. S'il s'agit d'un compte existant, il est possible que vos stratégies d'accès ne contiennent pas certaines autorisations permettant d'exécuter certaines actions, comme la création d'une instance de base de données. Pour résoudre le problème, votre administrateur doit fournir les rôles nécessaires à votre compte. Pour de plus amples informations, veuillez consulter la documentation IAM.

Résolution des problèmes liés à l'état de réseau incompatible

L'état de réseau incompatible signifie que la base de données est peut-être toujours accessible au niveau de la base de données, mais que vous ne pouvez pas la modifier ni la redémarrer.

Causes

L'état de réseau incompatible de votre instance de base de données peut être le résultat de l'une des actions suivantes :

- Modification de la classe d'instances de base de données.
- Modification de l'instance de base de données pour utiliser un déploiement de cluster de bases de données multi-AZ.
- Remplacement d'un hôte suite à un événement de maintenance.
- Lancement d'une instance de base de données de remplacement.
- Restauration à partir d'une sauvegarde d'instantané.
- Démarrage d'une instance de base de données qui a été arrêtée.

Résolution

Utiliser start-db-instance la commande

Pour corriger une base de données figurant dans l'état de réseau incompatible, suivez ces instructions :

1. Ouvrez le https://console.aws.amazon.com/rds/et choisissez Databases dans le volet de navigation.
2. Choisissez l'instance de base de données dont l'état de réseau est incompatible et notez l'identifiant de l'instance de base de données, l'ID VPC et le sous-réseau dans l'onglet Connectivité IDs et sécurité.
3. Utilisez le AWS CLI pour exécuter la `start-db-instance` commande. Spécifiez la valeur `--db-instance-identifier`.

> **Note**
>
> L'exécution de cette commande lorsque votre base de données est en mode d'incompatibilité peut entraîner des temps d'arrêt.
> La commande `start-db-instance` ne résout pas ce problème pour les instances de base de données RDS for SQL Server.

Le statut de votre base de données passe à Disponible si la commande s'exécute correctement.

Si votre base de données redémarre, l'instance de base de données peut exécuter la dernière opération exécutée sur l'instance avant son passage à l'état de réseau incompatible. Cela peut ramener l'instance à l'état de réseau incompatible.

Si la commande `start-db-instance` échoue ou que l'instance revient à l'état de réseau incompatible, ouvrez la page Bases de données dans la console RDS et sélectionnez la base de données. Accédez à la section Journaux et événements. La section Événements récents affiche les étapes de résolution supplémentaires à suivre. Les messages sont classés comme suit :

- VÉRIFICATION DES RESSOURCES INTERNES : il se peut que des problèmes soient liés à vos ressources internes.
- VÉRIFICATION DNS : vérifiez les noms d'hôte et la résolution DNS pour le VPC dans la console VPC.
- VÉRIFICATION ENI : l'interface réseau Elastic (ENI) pour votre base de données n'existe peut-être pas.
- VÉRIFICATION DE LA PASSERELLE : la passerelle Internet de votre base de données accessible au public n'est pas attachée au VPC.
- VÉRIFICATION IP : il n'y a pas d'adresses IP libres dans vos sous-réseaux.

- VÉRIFICATION DES GROUPES DE SÉCURITÉ : aucun groupe de sécurité n'est associé à votre base de données ou les groupes de sécurité sont non valides.
- VÉRIFICATION DES SOUS-RÉSEAUX : il n'y a aucun sous-réseau valide dans votre groupe de sous-réseaux de base de données ou il y a des problèmes avec votre sous-réseau.
- VÉRIFICATION DU VPC : le VPC associé à votre base de données est non valide.

Effectuer une point-in-time restauration

Il est recommandé de disposer d'une sauvegarde (instantanée ou logique) au cas où votre base de données passerait à l'état de réseau incompatible. Consultez Présentation des sauvegardes. Si vous avez activé les sauvegardes automatiques, arrêtez temporairement toute écriture dans la base de données et effectuez une point-in-time restauration.

> **Note**
>
> Une fois qu'une instance est passée à l'état de réseau incompatible, il est possible que l'instance de base de données ne soit pas accessible pour effectuer une sauvegarde logique.

Si vous n'avez pas activé les sauvegardes automatiques, créez une nouvelle instance de base de données. Migrez ensuite les données à l'aide d'AWS Database Migration Service (AWS DMS) ou en utilisant un outil de sauvegarde et de restauration.

Si cela ne résout pas le problème, contactez Support pour obtenir de l'aide supplémentaire.

Réinitialisation du mot de passe du propriétaire de l'instance de base de données

Si l'accès à votre instance de base de données est verrouillé, vous pouvez vous connecter en tant qu'utilisateur principal. Ensuite, vous pouvez réinitialiser les informations d'identification pour d'autres utilisateurs ou rôles administratifs. Si vous ne parvenez pas à vous connecter en tant qu'utilisateur principal, le propriétaire du AWS compte peut réinitialiser le mot de passe de l'utilisateur principal. Pour de plus amples informations sur les comptes ou rôles administratifs que vous devrez peut-être réinitialiser, veuillez consulter Privilèges du compte utilisateur principal.

Vous pouvez modifier le mot de passe de l'instance de base de données à l'aide de la console Amazon RDS modify-db-instance, de la AWS CLI commande ou de l'opération DBInstanceModify

API. Pour plus d'informations sur la modification d'une instance de base de données, veuillez consulter Modification d'une RDS instance de base de données Amazon.

Panne ou redémarrage d'une instance de base de donnéesAmazon RDS

Une instance de base de données peut connaître une panne au redémarrage. Cela peut également se produire quand l'instance de base de données est placée dans un état qui empêche d'y accéder ou quand la base de données est redémarrée. Un redémarrage peut se produire lorsque vous redémarrez manuellement votre instance de base de données. Un redémarrage peut également se produire quand vous modifiez un paramètre de l'instance de base de données qui nécessite un redémarrage avant que la modification ne puisse prendre effet.

Un redémarrage de l'instance de base de données se produit lorsque vous démarrez un paramètre qui nécessite un redémarrage ou quand vous provoquez manuellement un redémarrage. Un redémarrage peut se produire immédiatement si vous modifiez un paramètre et demandez que la modification prenne effet immédiatement. Cela peut également se produire pendant la fenêtre de maintenance de l'instance de base de données.

Un redémarrage d'instance de base de données se produit immédiatement quand l'une des conditions suivantes est vraie :

- Vous remplacez la période de rétention des sauvegardes pour une instance de base de données de 0 par une valeur différente de zéro, ou d'une valeur différente de 0 par 0. Vous définissez ensuite Ajouter un rôle (Appliquer immédiatement) sur `true`.
- Vous pouvez modifier la classe d'instance de base de données et Appliquer immédiatement est défini sur la valeur `true` (vrai).
- Vous remplacez le type de stockage Magnetic (Standard) [Magnétique (Standard)] par General Purpose (SSD) [Usage général (SSD)]) ou Provisioned IOPS (SSD) [IOPS dimensionné (SSD)], ou le type Provisioned IOPS (SSD) [IOPS dimensionné (SSD)] ou General Purpose (SSD) [Usage général (SSD)] par Magnetic (Standard) [Magnétique (Standard)].

Un redémarrage d'instance de base de données se produit pendant la fenêtre de maintenance quand l'une des conditions suivantes est vraie :

- Vous remplacez la période de rétention des sauvegardes pour une instance de base de données de 0 par une valeur différente de zéro, ou d'une valeur différente de zéro par zéro, et Appliquer immédiatement est défini sur la valeur `false` (faux).
- Vous pouvez modifier la classe d'instance de base de données et Appliquer immédiatement est défini sur la valeur `false` (vrai).

Lorsque vous modifiez un paramètre statique d'un groupe de paramètres de base de données, la modification prend effet seulement après le redémarrage de l'instance de base de données associée au groupe de paramètres. La modification nécessite un redémarrage manuel. L'instance de base de données n'est pas redémarrée automatiquement pendant la fenêtre de maintenance.

Pour afficher un tableau qui illustre les actions d'une instance de base de données et l'effet de l'application de la valeur Apply Immediately (Appliquer immédiatement), consultez Modification d'une RDS instance de base de données Amazon.

Modifications de paramètre de base de données Amazon RDS n'entrant pas en vigueur

Dans certains cas, vous pouvez modifier un paramètre dans un groupe de paramètres de base de données, mais vous ne voyez pas que les modifications prennent effet. Vous devrez alors probablement redémarrer l'instance de base de données associée au groupe de paramètres de base de données. Lorsque vous modifiez un paramètre dynamique, la modification prend effet immédiatement. Lorsque vous modifiez un paramètre statique, la modification ne prend effet que lorsque vous redémarrez l'instance de base de données associée au groupe de paramètres.

Vous pouvez redémarrer une instance de base de données en utilisant la console RDS. Vous pouvez également appeler explicitement l'opération d'API `RebootDBInstance`. Vous pouvez redémarrer sans basculement si l'instance de base de données se trouve dans un déploiement multi-AZ. Les critères pour redémarrer l'instance de base de données associée après un changement de paramètre statique contribuent à atténuer le risque d'erreur de configuration d'un paramètre affectant un appel d'API. Par exemple, appeler `ModifyDBInstance` pour changer la classe de l'instance de base de données. Pour de plus amples informations, veuillez consulter Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).

Manque d'espace de stockage de l'instance de base de données Amazon RDS

Si votre instance de base de données ne dispose plus d'un espace de stockage suffisant, il se peut qu'elle ne soit plus disponible. Nous vous recommandons vivement de surveiller en permanence la `FreeStorageSpace` métrique publiée dans CloudWatch pour vous assurer que votre instance de base de données dispose de suffisamment d'espace de stockage disponible.

Si votre instance de base de données ne dispose plus d'un stockage suffisant, son état devient `storage-full`. Par exemple, l'appel de l'opération d'API `DescribeDBInstances` pour une instance de base de données qui a consommé la totalité de son stockage produit l'effet suivant.

```
aws rds describe-db-instances --db-instance-identifier mydbinstance

DBINSTANCE  mydbinstance  2009-12-22T23:06:11.915Z  db.m5.large  mysql8.0  50  sa
storage-full  mydbinstance.clla4j4jgyph.us-east-1.rds.amazonaws.com  3306
us-east-1b  3
 SECGROUP  default  active
 PARAMGRP  default.mysql8.0  in-sync
```

Pour vous remettre de ce scénario, ajoutez de l'espace de stockage à votre instance à l'aide de l'opération `ModifyDBInstance` API ou de la AWS CLI commande suivante.

Dans Linux, macOS, ou Unix:

```
aws rds modify-db-instance \
    --db-instance-identifier mydbinstance \
    --allocated-storage 60 \
    --apply-immediately
```

Dans Windows:

```
aws rds modify-db-instance ^
    --db-instance-identifier mydbinstance ^
    --allocated-storage 60 ^
    --apply-immediately
```

```
DBINSTANCE  mydbinstance  2009-12-22T23:06:11.915Z  db.m5.large  mysql8.0  50  sa
storage-full  mydbinstance.clla4j4jgyph.us-east-1.rds.amazonaws.com  3306
```

```
us-east-1b  3  60
 SECGROUP  default  active
 PARAMGRP  default.mysql8.0  in-sync
```

Désormais, lorsque vous décrivez votre instance de base de données, vous constatez que son état est `modifying` (modification), ce qui signifie que le stockage est en cours de mise à l'échelle.

```
aws rds describe-db-instances --db-instance-identifier mydbinstance
```

```
DBINSTANCE  mydbinstance  2009-12-22T23:06:11.915Z  db.m5.large  mysql8.0  50  sa
modifying  mydbinstance.clla4j4jgyph.us-east-1.rds.amazonaws.com
3306  us-east-1b  3  60
 SECGROUP  default  active
 PARAMGRP  default.mysql8.0  in-sync
```

Une fois la mise à l'échelle du stockage terminée, l'état de votre instance de base de données devient (disponible `available`.

```
aws rds describe-db-instances --db-instance-identifier mydbinstance
```

```
DBINSTANCE  mydbinstance  2009-12-22T23:06:11.915Z  db.m5.large  mysql8.0  60  sa
available  mydbinstance.clla4j4jgyph.us-east-1.rds.amazonaws.com  3306
us-east-1b  3
 SECGROUP  default  active
 PARAMGRP  default.mysql8.0  in-sync
```

L'opération `DescribeEvents` vous permet de recevoir des notifications lorsque vous n'avez plus d'espace de stockage disponible. Par exemple, dans ce scénario, si vous appelez `DescribeEvents` après ces opérations, vous observez le résultat suivant.

```
aws rds describe-events --source-type db-instance --source-identifier mydbinstance
```

```
2009-12-22T23:44:14.374Z  mydbinstance  Allocated storage has been exhausted db-
instance
2009-12-23T00:14:02.737Z  mydbinstance  Applying modification to allocated storage db-
instance
2009-12-23T00:31:54.764Z  mydbinstance  Finished applying modification to allocated
 storage
```

Amazon RDS : nombre insuffisant d'instances de base de données disponibles

L'erreur `InsufficientDBInstanceCapacity` peut être renvoyée lorsque vous essayez de créer, de démarrer ou de modifier une instance de base de données. Elle peut également être renvoyée lorsque vous essayez de restaurer une instance de base de données à partir d'un instantané de base de données. Lorsque cette erreur est renvoyée, la cause la plus fréquente est que la classe d'instance de base de données spécifique n'est pas disponible dans la zone de disponibilité demandée. Vous pouvez essayer une des actions suivantes pour résoudre le problème :

- Renouveler la demande avec une classe d'instance de base de données différente.
- Renouveler la demande avec une zone de disponibilité différente.
- Renouveler la demande sans spécifier de zone de disponibilité explicite.

Pour plus d'informations sur la résolution des problèmes de capacité d'instance pour Amazon EC2, consultez la section Capacité d'instance insuffisante dans le guide de EC2 l'utilisateur Amazon.

Pour savoir comment modifier une instance de base de données, consultez Modification d'une RDS instance de base de données Amazon.

Problèmes liés à la mémoire libérable dans Amazon RDS

La mémoire libérable est la quantité totale de mémoire vive (RAM) sur une instance de base de données qui peut être mise à la disposition du moteur de base de données. Il s'agit de la somme de la mémoire libre du système d'exploitation et des mémoires tampon/cache de page disponibles. Le moteur de base de données utilise la plus grande partie de la mémoire sur l'hôte, mais les processus du système d'exploitation utilisent également une partie de la mémoire vive. La mémoire actuellement allouée au moteur de base de données ou utilisée par les processus du système d'exploitation n'est pas incluse dans la mémoire libérable. Lorsque le moteur de base de données manque de mémoire, l'instance de base de données peut utiliser l'espace temporaire normalement utilisé pour la mise en mémoire tampon et la mise en cache. Comme mentionné précédemment, cet espace temporaire est inclus dans la mémoire libérable.

Vous utilisez la `FreeableMemory` métrique dans Amazon CloudWatch pour surveiller la mémoire disponible. Pour de plus amples informations, veuillez consulter Outils de surveillance pour Amazon RDS.

Si votre instance de base de données manque constamment de mémoire libérable ou utilise l'espace d'échange, envisagez d'augmenter la taille de la classe d'instance de base de données. Pour de plus amples informations, veuillez consulter Classes d'instances de base de données .

Vous pouvez également modifier les paramètres de mémoire. Par exemple, sur RDS for MySQL, vous pouvez ajuster la taille du paramètre `innodb_buffer_pool_size`. Ce paramètre est défini par défaut sur 75 % de la mémoire physique. Pour obtenir des conseils de dépannage de MySQL, consultez Comment résoudre les problèmes liés à un manque de mémoire libérable dans une base de données Amazon RDS for MySQL ?

Problèmes MySQL et MariaDB

Vous pouvez diagnostiquer et corriger les problèmes des instances de base de données MySQL et MariaDB.

Rubriques

- Maximum de connexions MySQL et MariaDB
- Diagnostic et résolution d'un état de paramètres incompatibles pour une limite de mémoire
- Diagnostic et résolution du retard entre réplicas en lecture
- Diagnostic et résolution d'une défaillance de la réplication en lecture MySQL ou MariaDB
- La création de déclencheurs avec la journalisation binaire activée requiert le privilège SUPER
- Diagnostic et résolution des défaillances de point-in-time restauration
- Erreur d'arrêt de réplication
- La création de réplica en lecture échoue ou la réplication s'arrête avec l'erreur irrécupérable 1236
- La réplication de la réplique en lecture ne parvient pas à initialiser la structure des métadonnées

Maximum de connexions MySQL et MariaDB

Le nombre maximum de connexions autorisées à une instance de bases de données RDS for MySQL ou RDS pour MariaDB est basé sur la quantité de mémoire disponible pour la classe de l'instance de bases de données. Une classe d'instance de base de données avec plus de mémoire disponible entraîne un plus grand nombre de connexions disponibles. Pour plus d'informations sur les classes d'instance de base de données, consultez Classes d'instances de base de données .

La limite de connexions pour une instance de base de données est définie par défaut au nombre maximum pour la classe de l'instance de base de données. Vous pouvez limiter le nombre de

connexions simultanées à toutes les valeurs jusqu'au nombre maximum de connexions autorisées. Utilisez le paramètre `max_connections` dans le groupe de paramètres pour l'instance de base de données. Pour de plus amples informations, veuillez consulter Nombre maximal de connexions à une base de données et Groupes de paramètres pour (Amazon RDS).

Vous pouvez récupérer le nombre maximum de connexions autorisées pour une instance de base de données MySQL ou MariaDB en exécutant la requête suivante.

```
SELECT @@max_connections;
```

Vous pouvez récupérer le nombre maximum de connexions actives pour une instance de base de données MySQL ou MariaDB en exécutant la requête suivante.

```
SHOW STATUS WHERE `variable_name` = 'Threads_connected';
```

Diagnostic et résolution d'un état de paramètres incompatibles pour une limite de mémoire

Une instance de base de données MariaDB ou MySQL peut être placée dans un statut incompatible-parameters en raison d'une limite de mémoire lorsque les conditions suivantes sont remplies :

- L'instance de base de données est redémarrée au moins trois fois en une heure ou au moins cinq fois en une journée quand le statut de l'instance de base de données est Disponible.
- Une tentative de redémarrage de l'instance de base de données échoue car une action de maintenance ou un processus de surveillance n'a pas pu redémarrer l'instance de base de données.
- L'utilisation potentielle de la mémoire de l'instance de base de données dépasse à hauteur de 1,2 fois la mémoire allouée à sa classe d'instance de base de données.

Lorsqu'une instance de base de données est redémarrée pour la troisième fois en une heure ou pour la cinquième fois en une journée, l'utilisation de la mémoire est vérifiée. Cette vérification effectue un calcul d'utilisation potentielle de la mémoire de l'instance de base de données. La valeur qui en découle correspond à la somme des valeurs suivantes :

- Valeur 1 – Somme des paramètres suivants :
  - `innodb_additional_mem_pool_size`
  - `innodb_buffer_pool_size`

Vous pouvez modifier la valeur de`innodb_buffer_pool_size`. Cependant, la valeur ne correspondra pas toujours à ce que vous avez saisi. Cette incompatibilité se produit pour plusieurs raisons. Tout d'abord, si l'instance de base de données est une micro instance de base de données, nous annulons la valeur par défaut et la définissons sur 256 Mo. Pour de plus amples informations, veuillez consulter Remplacer innodb_buffer_pool_size.

Ensuite, nous nous assurons que 500 Mo de mémoire sont réservés sur l'instance de base de données pour le gestionnaire d'hôte, le moteur, le système d'exploitation et le noyau.

Enfin, nous optimisons `innodb_buffer_pool_size` en le divisant en unités. Le responsable hôte arrondit au multiple inférieur le plus proche de ces unités. Les unités sont calculées en multipliant `innodb_buffer_pool_chunk_size` par`innodb_buffer_pool_instances`. Pour plus d'informations, consultez Configuration de la taille du pool de mémoire tampon InnoDB dans la documentation MySQL.

La valeur par défaut `innodb_buffer_pool_instances` est 8, sauf si `innodb_buffer_pool_size` elle est inférieure à 1 Go. Si la valeur `innodb_buffer_pool_size` est inférieure à 1 Go, la valeur par défaut `innodb_buffer_pool_instances` est 1. La valeur par défaut `innodb_buffer_pool_chunk_size` est de 128 Mo.

  - `innodb_log_buffer_size`
  - `key_buffer_size`
  - `query_cache_size` (MySQL version 5.7 uniquement)
  - `tmp_table_size`
- Valeur 2 – Paramètre `max_connections` multiplié par la somme des paramètres suivants :
  - `binlog_cache_size`
  - `join_buffer_size`
  - `read_buffer_size`
  - `read_rnd_buffer_size`
  - `sort_buffer_size`
  - `thread_stack`
- Valeur 3 – Si le paramètre `performance_schema` est activé, multipliez le paramètre `max_connections` par `429498`.

  Si le `performance_schema` paramètre est désactivé, cette valeur est nulle.

Ainsi, la valeur renvoyée par le calcul est la suivante :

```
Value 1 + Value 2 + Value 3
```

Lorsque cette valeur dépasse à hauteur de 1,2 fois la mémoire allouée à la classe d'instance de base de données utilisée par l'instance de base de données, cette dernière est placée dans un état incompatible-parameters . Pour plus d'informations sur la mémoire allouée aux classes d'instances de bases de données, consultez Spécifications matérielles pour les classes d'instance de base de données .

Le calcul multiplie la valeur du paramètre `max_connections` par la somme de plusieurs paramètres. Si le paramètre `max_connections` est défini sur une valeur élevée, la vérification peut renvoyer une valeur excessivement élevée pour l'utilisation potentielle de la mémoire de l'instance de base de données. Dans ce cas, pensez à diminuer la valeur du paramètre `max_connections`.

Pour résoudre le problème, procédez comme suit :

1. Ajustez les paramètres de mémoire du groupe de paramètres de base de données associé à l'instance de base de données. Procédez de telle sorte que l'utilisation potentielle de la mémoire soit inférieure à hauteur de 1,2 fois la mémoire allouée à sa classe d'instance de base de données.

   Pour plus d'informations sur la définition des paramètres, consultez Modification des paramètres d'un groupe de paramètres de base de données dans Amazon RDS (Aurora).
2. Redémarrez l'instance de base de données.

   Pour plus d'informations sur la définition des paramètres, consultez Démarrage d'une RDS instance de base de données Amazon précédemment arrêtée.

Diagnostic et résolution du retard entre réplicas en lecture

Après que vous avez créé un réplica en lecture MySQL ou MariaDB et que le réplica en lecture est disponible, Amazon RDS réplique d'abord les modifications apportées à l'instance de base de données source à partir du moment où l'opération de création du réplica en lecture a été initiée. Durant cette phase, la durée du retard de réplication pour le réplica en lecture est supérieure à 0. Vous pouvez surveiller ce délai dans Amazon en CloudWatch consultant la `ReplicaLag`métrique Amazon RDS.

La métrique `ReplicaLag` contient la valeur du champ `Seconds_Behind_Master` de la commande MariaDB ou MySQL `SHOW REPLICA STATUS`. Pour plus d'informations, consultez Instruction SHOW REPLICA STATUS dans la documentation sur MySQL.

Lorsque la métrique `ReplicaLag` atteint 0, le réplica a rattrapé l'instance de bases de données source. Si la métrique `ReplicaLag` retourne -1, la réplication n'est probablement pas active. Pour résoudre une erreur de réplication, consultez Diagnostic et résolution d'une défaillance de la réplication en lecture MySQL ou MariaDB. Une valeur de -1 pour `ReplicaLag` peut également signifier que la valeur `Seconds_Behind_Master` ne peut pas être déterminée ou qu'elle est `NULL`.

> **Note**
>
> Les versions précédentes de MariaDB utilisaient `SHOW SLAVE STATUS` à la place de `SHOW REPLICA STATUS`. Si vous utilisez une version de MariaDB inférieure à 10.5, utilisez. `SHOW SLAVE STATUS`

La métrique `ReplicaLag` retourne -1 pendant une panne réseau ou lorsqu'un correctif est appliqué pendant la fenêtre de maintenance. Dans ce cas, attendez que la connexion réseau soit restaurée ou que la fenêtre de maintenance finisse avant de vérifier à nouveau la métrique `ReplicaLag` .

La technologie de réplication en lecture MySQL et MariaDB est asynchrone. Vous pouvez vous attendre à des augmentations occasionnelles de la métrique `BinLogDiskUsage` sur l'instance de base de données source, et de la métrique `ReplicaLag` sur le réplica en lecture. Prenez l'exemple d'une situation dans laquelle un volume élevé d'opérations d'écriture sur l'instance de base de données source se produit en parallèle. Au même moment, les opérations d'écriture sur le réplica en lecture sont sérialisées à l'aide d'un seul thread d'I/O. Une telle situation peut entraîner un décalage entre l'instance source et le réplica en lecture.

Pour de plus amples informations sur les réplicas en lecture et MySQL, veuillez consulter Détails d'implémentation de réplication dans la documentation MySQL. Pour de plus amples informations sur les réplicas en lecture et MariaDB, veuillez consulter Présentation de la réplication dans la documentation MariaDB.

Vous pouvez réduire le retard entre les mises à jour d'une instance de base de données source et les mises à jour suivantes du réplica en lecture en procédant comme suit :

- Définissez la classe d'instance de base de données du réplica en lecture de telle sorte que sa taille de stockage soit comparable à celle de l'instance de base de données source.

- Veillez à ce que les paramètres des groupes de paramètres de base de données utilisés par l'instance de base de données source et le réplica en lecture soient compatibles. Pour obtenir plus d'informations et un exemple, reportez-vous à la présentation du paramètre `max_allowed_packet` dans la section suivante.
- Désactivez le cache de requête. Pour les tables modifiées fréquemment, l'utilisation du cache de requête peut augmenter le retard, parce que le cache est verrouillé et souvent actualisé. Si tel est le cas, il se peut que vous constatiez un retard de réplica inférieur si vous désactivez le cache de requête. Vous pouvez désactiver le cache de requête en définissant le paramètre `query_cache_type parameter` avec la valeur 0 dans le groupe de paramètres DB de l'instance de base de données. Pour de plus amples informations sur le cache de requête, veuillez consulter Configuration du pare-feu Windows.
- Préparez le groupe de tampons sur le réplica en lecture pour InnoDB pour MySQL ou MariaDB. Supposons par exemple que vous disposez d'un ensemble réduit de tables mises à jour fréquemment et que vous utilisez le schéma de table InnoDB ou XtraDB. Dans ce cas, videz ces tables sur le réplica en lecture. Le moteur de base de données analyse alors les lignes des tables du disque et les met en cache dans le groupe de tampons. Cette approche peut réduire le retard de réplica. Voici un exemple.

  Dans Linux, macOS, ou Unix:

```
PROMPT> mysqldump \
    -h <endpoint> \
    --port=<port> \
    -u=<username> \
    -p <password> \
    database_name table1 table2 > /dev/null
```

  Dans Windows:

```
PROMPT> mysqldump ^
    -h <endpoint> ^
    --port=<port> ^
    -u=<username> ^
    -p <password> ^
    database_name table1 table2 > /dev/null
```

Diagnostic et résolution d'une défaillance de la réplication en lecture MySQL ou MariaDB

Amazon RDS surveille l'état de réplication de vos réplicas lus. RDS met à jour le champ Replication State (État de réplication) de l'instance du réplica en lecture sur `Error` si la réplication s'arrête pour une raison quelconque. Vous pouvez passer en revue les détails de l'erreur associée et déclenchée par les moteurs MySQL ou MariaDB, en consultant le champ Erreur de réplication. Des événements indiquant l'état du réplica en lecture sont également générés, y compris RDS-EVENT-0045, RDS-EVENT-0046 et RDS-EVENT-0057. Pour plus d'informations sur les événements et l'abonnement aux événements, consultez Utilisation des notifications d'RDSévénements Amazon. Si un message d'erreur MySQL est renvoyé, veuillez consulter l'erreur dans la documentation sur les messages d'erreur MySQL. Si un message d'erreur MariaDB est renvoyé, consultez l'erreur dans la documentation sur les messages d'erreur MariaDB.

Voici d'autres situations courantes susceptibles d'entraîner des erreurs de réplication :

- La valeur du paramètre `max_allowed_packet` d'un réplica en lecture est inférieure au paramètre `max_allowed_packet` de l'instance de base de données source.

  Le paramètre `max_allowed_packet` est un paramètre personnalisé que vous pouvez définir dans un groupe de paramètres de base de données. Le paramètre `max_allowed_packet` est utilisé pour spécifier la taille maximale du langage de manipulation de données (DML) qui peut être exécuté sur la base de données. Dans certains cas, la valeur `max_allowed_packet` de l'instance de base de données source peut être supérieure à la valeur `max_allowed_packet` du réplica en lecture. Dans ces cas, le processus de réplication peut lancer une erreur et arrêter la réplication. L'erreur la plus courante est `packet bigger than 'max_allowed_packet' bytes`. Vous pouvez corriger cette erreur en indiquant à la source et au réplica en lecture d'utiliser des groupes de paramètres de base de données avec les mêmes valeurs du paramètre `max_allowed_packet`.
- Écriture sur les tables d'un réplica en lecture. Si vous créez des index sur un réplica en lecture, le paramètre `read_only` doit être défini sur 0 pour créer les index. Si vous écrivez dans des tables sur le réplica en lecture, cela peut interrompre la réplication.
- Utilisation d'un moteur de stockage non transactionnel tel que MyISAM. Les réplicas en lecture nécessitent un moteur de stockage transactionnel. La réplication n'est prise en charge que pour les moteurs de stockage suivants : InnoDB pour MySQL ou MariaDB.

  Pour convertir une table MyISAM en InnoDB, exécutez la commande suivante :

```
alter table <schema>.<table_name> engine=innodb;
```

- Utilisation de requêtes non déterministes non sécurisées telles que `SYSDATE()`. Pour de plus amples informations, veuillez consulter Determination of Safe and Unsafe Statements in Binary Logging dans la documentation MySQL.

Les étapes suivantes peuvent vous aider à résoudre votre erreur de réplication :

- Si vous rencontrez une erreur logique et que vous pouvez l'ignorer en toute sécurité, suivez la procédure décrite dans Ignorer l'erreur de réplication actuelle pour RDS for My SQL. Votre instance de base de données MySQL ou MariaDB doit exécuter une version incluant la procédure `mysql_rds_skip_repl_error`. Pour de plus amples informations, veuillez consulter mysql.rds_skip_repl_error.
- Si vous rencontrez un problème de position du journal binaire (binlog), vous pouvez modifier la position de rediffusion de la réplique à l'aide de la commande ou.
- Il se peut que vous rencontriez un problème de performance temporaire en raison d'une charge DML élevée. Si tel est le cas, vous pouvez définir le paramètre `innodb_flush_log_at_trx_commit` sur 2 dans le groupe de paramètres de base de données pour le réplica en lecture. Cette action peut aider le réplica en lecture à se rattraper, même si l'atomicité, la cohérence, l'isolation et la durabilité s'en trouvent temporairement réduites.
- Vous pouvez supprimer le réplica en lecture et créer une instance à l'aide du même identifiant d'instance de base de données. Dans ce cas, le point de terminaison reste le même que celui de votre ancien réplica en lecture.

Si une erreur de réplication est corrigée, le champ Replication State (Statut de réplication) prend la valeur replicating (réplication en cours). Pour plus d'informations, consultez Résolution d'un problème lié à SQL la réplique My read.

La création de déclencheurs avec la journalisation binaire activée requiert le privilège SUPER

Lors de la création de déclencheurs dans une instance de bases de données RDS for MySQL ou RDS pour MariaDB, vous pouvez recevoir l'erreur suivante.

```
"You do not have the SUPER privilege and binary logging is enabled"
```

L'utilisation des déclencheurs lorsque la journalisation binaire est activée nécessite le privilège SUPER, qui est limité pour les instances de bases de données RDS for MySQL et RDS pour MariaDB. Vous pouvez créer des déclencheurs lorsque la journalisation binaire est activée sans le privilège SUPER en définissant le paramètre `log_bin_trust_function_creators` avec la valeur true. Pour définir le paramètre `log_bin_trust_function_creators` sur true, créez un groupe de paramètres DB ou modifiez un groupe de paramètres DB existant.

Vous pouvez créer un nouveau groupe de paramètres de bases de données afin que vous puissiez créer des déclencheurs dans votre instance de bases de données RDS for MySQL ou RDS for MariaDB avec la journalisation binaire activée. Pour cela, utilisez les commandes de l'interface de ligne de commande suivantes. Pour modifier un groupe de paramètres existant, commencez par l'étape 2.

Pour créer un groupe de paramètres et autoriser les déclencheurs avec la journalisation binaire activée à l'aide de l'interface de ligne de commande (CLI)

1. Créez un groupe de paramètres.

   Dans Linux, macOS, ou Unix:

   ```
   aws rds create-db-parameter-group \
       --db-parameter-group-name allow-triggers \
       --db-parameter-group-family mysql8.0 \
       --description "parameter group allowing triggers"
   ```

   Dans Windows:

   ```
   aws rds create-db-parameter-group ^
       --db-parameter-group-name allow-triggers ^
       --db-parameter-group-family mysql8.0 ^
       --description "parameter group allowing triggers"
   ```

2. Modifiez le groupe de paramètres DB pour autoriser les déclencheurs.

   Dans Linux, macOS, ou Unix:

   ```
   aws rds modify-db-parameter-group \
       --db-parameter-group-name allow-triggers \
       --parameters "ParameterName=log_bin_trust_function_creators,
    ParameterValue=true, ApplyMethod=pending-reboot"
   ```

Dans Windows:

```
aws rds modify-db-parameter-group ^
    --db-parameter-group-name allow-triggers ^
    --parameters "ParameterName=log_bin_trust_function_creators,
 ParameterValue=true, ApplyMethod=pending-reboot"
```

3. Modifiez votre instance de base de données pour utiliser le nouveau groupe de paramètres DB.

   Dans Linux, macOS, ou Unix:

   ```
   aws rds modify-db-instance \
       --db-instance-identifier mydbinstance \
       --db-parameter-group-name allow-triggers \
       --apply-immediately
   ```

   Dans Windows:

   ```
   aws rds modify-db-instance ^
       --db-instance-identifier mydbinstance ^
       --db-parameter-group-name allow-triggers ^
       --apply-immediately
   ```

4. Pour que les modifications deviennent effectives, redémarrez manuellement l'instance de base de données.

   ```
   aws rds reboot-db-instance --db-instance-identifier mydbinstance
   ```

Diagnostic et résolution des défaillances de point-in-time restauration

Restauration d'une instance de base de données qui inclut des tables temporaires

Lorsque vous tentez de point-in-time restaurer (PITR) votre instance de base de données MySQL ou MariaDB, vous pouvez rencontrer l'erreur suivante.

```
Database instance could not be restored because there has been incompatible database
 activity for restore
functionality. Common examples of incompatible activity include using temporary tables,
 in-memory tables,
```

```
or using MyISAM tables. In this case, use of Temporary table was detected.
```

Cette restauration s'appuie à la fois sur les instantanés de sauvegarde et les journaux binaires de MySQL ou MariaDB pour restaurer votre instance de base de données à une date spécifique. Les informations des tables temporaires peuvent ne pas être fiables dans les journaux binaires et provoquer une erreur de restauration à un instant dans le passé. Si vous utilisez des tables temporaires dans votre instance de base de données MySQL ou MariaDB, vous pouvez réduire le risque de défaillance du PITR en effectuant des sauvegardes plus fréquentes. Une telle défaillance est plus à même de se produire entre la création d'une table temporaire et l'instantané de sauvegarde suivant.

Restauration d'une instance de base de données qui inclut des tables en mémoire

Il se peut que vous rencontriez un problème lors de la restauration d'une base de données qui comporte des tables en mémoire. Les tables en mémoire sont vidées lors d'un redémarrage. En conséquence, vos tables en mémoire peuvent être vides après un redémarrage. Lorsque vous utilisez les tables en mémoire, nous vous recommandons de concevoir l'architecture de votre solution de façon à gérer les tables vides en cas de redémarrage. Si vous utilisez des tables en mémoire avec des instances de base de données répliquées, vous devrez peut-être recréer les réplicas en lecture après un redémarrage. Cela peut s'avérer nécessaire si un réplica en lecture redémarre et ne peut pas restaurer les données à partir d'une table en mémoire vide.

Pour plus d'informations sur les sauvegardes et les restaurations à un instant dans le passé, consultez Présentation des sauvegardes et Restauration d'une instance de base de données à une heure spécifiée pour Amazon RDS.

Erreur d'arrêt de réplication

Lorsque vous appelez la commande `mysql.rds_skip_repl_error`, un message d'erreur peut s'afficher pour indiquer que la réplication a rencontré une erreur ou est désactivée.

Ce message d'erreur s'affiche car la réplication a été arrêtée et ne peut pas être redémarrée.

Si vous avez besoin d'ignorer un grand nombre d'erreurs, le retard de réplication peut augmenter et dépasser la période de rétention par défaut pour les fichiers journaux binaires. Dans ce cas, il se peut que vous rencontriez une erreur fatale car les fichiers journaux binaires ont été purgés avant d'être rejoués sur la réplique. Cette purge entraîne l'arrêt de la réplication et vous ne pouvez plus appeler la commande `mysql.rds_skip_repl_error` pour ignorer les erreurs de réplication.

Vous pouvez atténuer ce problème en augmentant le nombre d'heures pendant lequel les fichiers journaux binaires sont conservés sur votre source de réplication. Une fois que vous avez augmenté le temps de rétention de journaux binaires, vous pouvez redémarrer la réplication et appeler la commande `mysql.rds_skip_repl_error` en fonction des besoins.

Pour définir le temps de rétention du journal binaire, utilisez la procédure mysql.rds_set_configuration. Spécifiez un paramètre de configuration des heures de rétention des journaux binaires, ainsi que le nombre d'heures pendant lequel conserver les fichiers journaux binaires sur le cluster DB, 720 heures au plus (30 jours). L'exemple suivant définit la période de rétention des fichiers journaux binaires à 48 heures.

```
CALL mysql.rds_set_configuration('binlog retention hours', 48);
```

La création de réplica en lecture échoue ou la réplication s'arrête avec l'erreur irrécupérable 1236

Après avoir modifié les valeurs de paramètre par défaut pour une instance de base de données MySQL ou MariaDB, vous pouvez rencontrer un des problèmes suivants :

- Vous ne pouvez pas créer un réplica en lecture pour l'instance de base de données.
- La réplication échoue avec `fatal error 1236`.

Certaines valeurs de paramètres par défaut pour les instances de base de données MySQL et MariaDB aident à s'assurer que la base de données est conforme à ACID et que les réplicas en lecture sont sûrs en cas d'incident. Pour cela, les paramètres s'assurent que chaque validation est entièrement synchronisée en écrivant la transaction dans le journal binaire avant qu'elle ne soit validée. La modification de ces paramètres à partir de leurs valeurs par défaut pour améliorer les performances peut entraîner l'échec de la réplication quand une transaction n'a pas été écrite dans le journal binaire.

Pour résoudre ce problème, définissez les valeurs de paramètres suivantes :

- `sync_binlog = 1`
- `innodb_support_xa = 1`
- `innodb_flush_log_at_trx_commit = 1`

La réplication de la réplique en lecture ne parvient pas à initialiser la structure des métadonnées

Lorsque vous avez tenté de démarrer la réplication, le message d'erreur suivant s'est affiché :

```
Read Replica Replication Error - SQLError: 13124, reason: Replica failed to initialize
 applier metadata structure from the repository
```

Cette erreur se produit en cas de problème lié à la structure des métadonnées de la réplique. Pour corriger la structure des métadonnées, vous devez créer une nouvelle réplique.

Pour éviter que cela ne se reproduise à l'avenir, effectuez l'une des actions suivantes :

- Si possible, désactivez le multithreading sur vos répliques. À partir de MySQL 8.0.27, le multi-threading est activé par défaut.
- Si vous devez utiliser le multithreading sur vos répliques, nous vous recommandons d'utiliser la réplication basée sur le GTID. Pour de plus amples informations, veuillez consulter Utilisation de GTID la réplication basée.

Impossible de définir la période de rétention des sauvegardes sur 0

Plusieurs raisons peuvent vous obliger à définir la période de rétention des sauvegardes sur 0. Par exemple, vous pouvez immédiatement désactiver les sauvegardes automatiques en définissant la période de rétention sur 0.

Dans certains cas, vous pouvez définir la valeur sur 0 et recevoir un message indiquant que la période de rétention doit être comprise entre 1 et 35. Vérifiez alors que vous n'avez pas configuré un réplica en lecture pour l'instance. En effet, les réplicas en lecture requièrent des sauvegardes pour la gestion des journaux des réplicas en lecture, ce qui ne vous permet pas de définir la période de rétention sur 0.

Référence d'API Amazon RDS

Outre la AWS Management Console et l'AWS Command Line Interface (AWS CLI), Amazon RDS fournit également une API. Vous pouvez utiliser l'API pour automatiser les tâches de gestion de vos instances de base de données et d'autres objets dans Amazon RDS.

- Pour obtenir la liste alphabétique des opérations d'API, consultez Actions.
- Pour obtenir la liste alphabétique des types de données, consultez Types de données.
- Pour consulter la liste des paramètres de requête courants, reportez-vous à la page Paramètres courants.
- Pour la description des codes d'erreur, veuillez consulter la page Erreurs courantes.

Pour plus d'informations sur l'AWS CLI, consultez Référence de l'AWS Command Line Interface pour Amazon RDS.

Rubriques

- Utilisation de l'API Query
- Applications de dépannage sur Amazon RDS

Utilisation de l'API Query

Les sections suivantes abordent brièvement l'authentification de la demande et les paramètres utilisés avec l'API Query.

Pour obtenir des informations générales sur le fonctionnement de l'API Query, veuillez consulter Demandes de requête dans le Amazon EC2 API Reference.

Paramètres Query (Requête)

Ces demandes basées sur Query HTTP sont des demandes HTTP qui utilisent le verbe HTTP GET ou POST et un paramètre Query appelé `Action`.

Chaque demande Query doit inclure certains paramètres communs pour gérer l'authentification et la sélection d'une action.

Certaines actions demandent des listes de paramètres. Ces listes sont spécifiées en utilisant la notation `param.`*n*. Les valeurs de *n* sont des nombres entiers à partir de 1.

Pour plus d'informations sur les régions et les points de terminaison Amazon RDS, consultez Amazon Relational Database Service (RDS) dans la section Régions et points de terminaison de la Référence générale d'Amazon Web Services.

Authentification de demande Query

Vous pouvez uniquement envoyer des demandes Query via HTTPS, et vous devez inclure une signature dans chaque demande Query. Vous devez utiliser le processus AWS Signature Version 4 ou 2. Pour de plus amples informations, veuillez consulter Processus de signature Signature Version 4 et Processus de signature Signature Version 2.

Applications de dépannage sur Amazon RDS

Amazon RDS fournit des erreurs spécifiques et descriptives pour vous aider à résoudre vos problèmes tout en interagissant avec l'API Amazon RDS.

Rubriques

- Récupération d'erreurs
- Conseils pour le dépannage

Pour de plus amples informations sur le dépannage des instances de base de données Amazon RDS, veuillez consulter Dépannage d'Amazon RDS.

Récupération d'erreurs

Généralement, vous souhaitez que votre application vérifie si une demande a généré une erreur avant de passer du temps à traiter les résultats. Le moyen le plus simple de déterminer si une erreur s'est produite est de rechercher un nœud `Error` dans la réponse de l'API Amazon RDS.

La syntaxe XPath fournit une méthode simple pour rechercher la présence d'un nœud `Error`. Elle fournit également un moyen relativement simple de récupérer le code et le message d'erreur. L'extrait de code suivant utilise Perl et le module XML::XPath pour déterminer si une erreur s'est produite lors d'une demande. Si une erreur s'est produite, le code imprime le premier code et message d'erreur dans la réponse.

```
use XML::XPath;
    my $xp = XML::XPath->new(xml =>$response);
    if ( $xp->find("//Error") )
```

```
    {print "There was an error processing your request:\n", " Error code: ",
    $xp->findvalue("//Error[1]/Code"), "\n", " ",
    $xp->findvalue("//Error[1]/Message"), "\n\n"; }
```

Conseils pour le dépannage

Nous vous conseillons d'utiliser les processus suivants pour diagnostiquer et résoudre les problèmes avec l'API Amazon RDS :

- Vérifiez qu'Amazon RDS fonctionne normalement dans la région AWS que vous ciblez en consultant la page http://status.aws.amazon.com.
- Vérifiez la structure de votre demande.

  Chaque opération Amazon RDS possède une page de référence dans la référence de l'API Amazon RDS. Revérifiez que vous utilisez les paramètres correctement. Pour des idées sur les éventuels problèmes, observez les exemples de demandes ou de scénarios utilisateur pour voir s'ils effectuent des opérations similaires.
- Consultez AWS re:Post.

  Amazon RDS possède une communauté de développement où vous pouvez chercher des solutions aux problèmes rencontrés par d'autres. Pour consulter les rubriques, accédez à AWS re:Post.

Historique du document

Version de l'API actuelle : 2014-10-31

Le tableau ci-après décrit les modifications importantes dans chaque édition du Guide de l'utilisateur Amazon RDS après mai 2018. Pour recevoir les notifications des mises à jour de cette documentation, abonnez-vous à un flux RSS.

> **Note**
>
> Vous pouvez filtrer les nouvelles fonctions de Amazon RDS sur la page Nouveautés en matière de base de données. Pour Produits, choisissez Amazon RDS. Ensuite, effectuez une recherche à l'aide de mots clés tels que **`RDS Proxy`** ou **`Oracle 2023`**.

Modification	Description	Date
Amazon RDS prend en charge MariaDB 11.4.8, 10.11.14 et 10.6.23	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 11.4.8, 10.11.14 et 10.6.2 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	14 août 2025
Politique relative aux rôles liés à un service personnalisé Amazon RDS mise à jour	La politique RDSCustomServiceRolePolicy gérée par Amazon a été mise à jour pour se conformer au comportement CopySnapshot d'autorisation d'EBS. La mise à jour `ec2:CopySnapshot` supprime une instruction et ajoute deux nouvelles instructions pour les autorisations de capture d'écran source	7 août 2025

	et de destination. Pour plus d'informations, consultez Mises à jour des politiques AWS gérées pour Amazon RDS.	
Amazon RDS prend en charge MySQL 8.4.6 et 8.0.43	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL 8.4.6 et 8.0.43. MySQL 8.4.6 inclut l'ajout du plugin de base de données rds_security_audit, un plugin RDS MySQL interne qui collecte les métriques liées à la sécurité. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	1er août 2025
RDS pour Oracle prend en charge les classes d'instance de base de données db.m6in et db.r6in	Vous pouvez utiliser les classes d'instance de base de données db.m6in et db.r6in avec RDS pour Oracle avec le modèle BYOL (Bring Your Own License). Pour plus d'informations, consultez RDS pour les classes d'instance de base de données Oracle.	29 juillet 2025

Les intégrations sans ETL de RDS pour PostgreSQL et RDS pour Oracle sont désormais disponibles	Vous pouvez désormais créer des intégrations zéro ETL avec RDS pour PostgreSQL et RDS pour Oracle comme source. Pour plus d'informations, consultez la section Intégrations Amazon RDS Zero-ETL.	23 juillet 2025
Amazon RDS pour DB2 prend en charge l'autorisation de groupe à l'aide de Kerberos	Vous pouvez désormais effectuer une autorisation de groupe à l'aide de Kerberos avec votre Active Directory autogéré. Pour plus d'informations, consultez Configuration de l'authentification Kerberos pour RDS pour les instances de base de données DB2.	21 juillet 2025
Les intégrations sans ETL avec Amazon SageMaker AI sont généralement disponibles	Les intégrations sans ETL rendent les données transactionnelles disponibles dans un Amazon SageMaker lakehouse quelques secondes après leur écriture sur une instance de base de données RDS pour MySQL. Pour plus d'informations, consultez Travailler avec les intégrations Amazon RDS Zero-ETL.	30 juin 2025

Amazon RDS prend en charge MySQL 9.3 dans l'environnement Database Preview	MySQL 9.3 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 9.3 dans l'environnement Database Preview.	16 juin 2025
Amazon RDS pour Db2 prend en charge les répliques interrégionales en mode veille	Vous pouvez désormais créer des répliques RDS pour Db2 entre régions en mode veille pour la reprise après sinistre. Pour plus d'informations, consultez la section Utilisation des répliques pour RDS pour DB2.	11 juin 2025
Amazon RDS prend en charge MariaDB 10.11.13 et 11.4.7	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB versions 10.11.13 et 11.4.7. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	4 juin 2025

End-of-life informations pour Performance Insights	AWS a annoncé la end-of-life date de Performance Insights : le 30 novembre 2025. Après cette date, Amazon RDS ne prendra plus en charge l'expérience de la console Performance Insights, les périodes de rétention flexibles (1 à 24 mois) et les tarifs associés. Nous vous recommandons de mettre à niveau vos instances de base de données vers Database Insights avant cette date. Pour plus d'informations, consultez la section Surveillance de la charge de base de données avec Performance Insights sur Amazon RDS.	30 mai 2025
Support étendu Amazon RDS version 5.7.44-RDS.20250508 pour RDS pour MySQL	La version 5.7.44-RDS.20250508 de RDS Extended Support est désormais disponible pour RDS for MySQL. Pour plus d'informations, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	27 mai 2025

RDS Custom pour Oracle prend en charge les classes d'instance db.m7i et db.r7i	Vous pouvez désormais utiliser les classes d'instance db.m7i et db.r7i pour RDS Custom for Oracle. Pour plus d'informations, consultez Prise en charge des classes d'instances de base de données pour RDS Custom for Oracle.	22 mai 2025
Affichage des dates de support pour les versions majeures du moteur open source	Vous pouvez désormais consulter les informations sur les dates de début et de fin des versions majeures des moteurs open source dans le support standard RDS et le support étendu RDS à l'aide de l'API AWS CLI et de l'API RDS. Pour plus d'informations, consultez la section Affichage des dates de support pour les versions du moteur dans Amazon RDS Extended Support.	20 mai 2025
Amazon RDS prend en charge MariaDB 10.5.29 et 10.6.22	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB versions 10.5.29 et 10.6.22. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	20 mai 2025

Amazon RDS prend en charge MySQL 8.0.42	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL 8.0.42. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	12 mai 2025
Amazon RDS prend en charge MySQL 8.4.5	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL 8.4.5. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	1er mai 2025
Amazon RDS prend en charge la réplication de sauvegarde pour l'Asie-Pacifique (Hong Kong), l'Asie-Pacifique (Melbourne), l'Asie-Pacifique (Malaisie), le Canada Ouest (Calgary) et l'Europe (Zurich)	La réplication de sauvegarde est désormais disponible pour les bases de données des régions Asie-Pacifique (Hong Kong), Asie-Pacifique (Melbourne), Asie-Pacifique (Malaisie), Canada Ouest (Calgary) et Europe (Zurich). Pour plus d'informations sur les régions disponibles, consultez la section Réplication de sauvegardes automatisées vers une autre Région AWS.	30 avril 2025

Amazon RDS prend en charge la gestion des mots de passe des utilisateurs principaux dans l'architecture mutualisée Oracle.	Vous pouvez les utiliser AWS Secrets Manager pour gérer les mots de passe des utilisateurs principaux pour les bases de données de conteneurs RDS for Oracle (CDBs) et les bases de données enfichables (). PDBs Les configurations à locataire unique et à locataires multiples de l'architecture CDB prennent toutes deux en charge les mots de passe gérés. Pour plus d'informations, voir Comptes utilisateurs et privilèges dans une base de données CDB.	24 avril 2025
Amazon RDS for Oracle prend en charge les classes d'instance m6id et r6id pour BYOL	Vous pouvez désormais créer des instances RDS pour Oracle DB à l'aide des classes d'instance m6id et r6id sur le modèle BYOL. Les éditions Enterprise et Standard 2 sont toutes deux prises en charge. Pour plus d'informations, voir Édition prise en charge, classe d'instance et combinaisons de licences dans RDS pour Oracle.	9 avril 2025

Amazon RDS prend en charge MariaDB 11.8 dans l'environnement Database Preview	MariaDB 11.8 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez la version 11.8 de MariaDB dans l'environnement Database Preview.	2 avril 2025
Mise à jour des autorisations de rôle lié à un service IAM	La `AmazonRDSCustomServiceRolePolicy` politique accorde désormais des autorisations supplémentaires pour gérer les EC2 paires de clés et les intégrer à Amazon SQS. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	25 mars 2025
Amazon RDS prend en charge MySQL 9.2 dans l'environnement Database Preview	MySQL 9.2 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 9.2 dans l'environnement Database Preview.	24 mars 2025

Mise à jour de la politique existante	Amazon RDS a ajouté de nouvelles autorisations à la politique gérée `AmazonRDSCustomInstanceProfileRolePolicy` afin de permettre l'utilisation de secrets gérés RDS Custom sur une instance personnalisée RDS. Pour plus d'informations, consultez la politique AWS gérée : AmazonRDSCustomInstanceProfileRolePolicy.	20 mars 2025
Support étendu Amazon RDS version 5.7.44-RDS.20250213 pour RDS pour MySQL	La version 5.7.44-RDS.20250213 de RDS Extended Support est désormais disponible pour RDS for MySQL. Pour plus d'informations, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	12 mars 2025
Mise à jour des autorisations de rôle lié à un service IAM	La `AmazonRDSCustomServiceRolePolicy` politique accorde désormais des autorisations supplémentaires pour répertorier et restaurer les secrets de Secrets Manager. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	6 mars 2025

Amazon RDS pour DB2 prend en charge les classes d'instance db.m7i et db.r7i	Vous pouvez désormais utiliser les classes d'instance db.m7i et db.r7i pour RDS pour Db2. Pour en savoir plus, consultez la section Classes d'instances de base de données.	27 février 2025
Amazon RDS pour Oracle prend en charge les ensembles SPBs de correctifs spatiaux ()	RDS for Oracle publie désormais des versions du moteur Spatial Patch Bundles (SPB) pour Oracle Database 19c. Une version du moteur SPB contient des correctifs Release Update (RU) ainsi que des correctifs spécifiques à Oracle Spatial. Généralement, un SPB est publié 2 à 3 semaines après le RU correspondant. Vous pouvez mettre à niveau votre instance de base de données d'un RU standard vers un SPB sur la même version du moteur de base de données ou une version supérieure. Pour plus d'informations, consultez la section Mises à niveau des versions mineures d'Oracle.	25 février 2025

Amazon RDS prend en charge MariaDB 11.4.5, 10.11.11, 10.6.21 et 10.5.28	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 11.4.5, 10.11.11, 10.6.21 et 10.5.28 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	24 février 2025
Amazon RDS prend en charge CloudWatch Database Insights	Vous pouvez désormais utiliser Database Insights pour surveiller les bases de données Amazon RDS. Pour plus d'informations, consultez la section Surveillance des bases de données Amazon RDS avec CloudWatch Database Insights.	20 février 2025
Amazon RDS prend en charge MySQL 8.4.4 et MySQL 8.0.41	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL 8.4.4 et 8.0.41. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	19 février 2025
Support étendu Amazon RDS version 5.7.44-RDS.20250103 pour RDS pour MySQL	La version 5.7.44-RDS.20250103 de RDS Extended Support est désormais disponible pour RDS for MySQL. Pour plus d'informations, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	13 février 2025

Amazon RDS prend en charge MariaDB 11.7 dans l'environnement Database Preview	MariaDB 11.7 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez la version 11.7 de MariaDB dans l'environnement Database Preview.	9 janvier 2025
Amazon RDS prend en charge MariaDB 11.4.4, 10.11.10, 10.6.20 et 10.5.27	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 11.4.4, 10.11.10, 10.6.20 et 10.5.27 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	20 décembre 2024
Amazon RDS pour DB2 prend en charge plusieurs bases de données	Vous pouvez ajouter jusqu'à 50 bases de données DB2 à une instance de base de données RDS pour DB2. Pour plus d'informations, consultez Plusieurs bases de données sur une instance de base de données Amazon RDS pour DB2.	20 décembre 2024

Amazon RDS prend en charge MySQL 9.1 dans l'environnement Database Preview	MySQL 9.1 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 9.1 dans l'environnement Database Preview.	19 décembre 2024
Amazon RDS prend en charge les classes d'instance db.m8g et db.r8g	Vous pouvez désormais utiliser les classes d'instance db.m8g et db.r8g pour RDS pour MySQL, RDS pour PostgreSQL et RDS pour MariaDB. Pour en savoir plus, consultez la section Classes d'instances de base de données.	21 novembre 2024
Amazon RDS prend en charge MySQL 8.4	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.4. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	21 novembre 2024

Amazon Relational Database Service prend en charge EC2 la migration automatique des bases de données	Vous pouvez utiliser la console RDS pour migrer une EC2 base de données vers Amazon RDS. Amazon RDS utilise AWS Database Migration Service (AWS DMS) pour migrer votre base de EC2 données source. A WS DMS vous permet de migrer des bases de données relationnelles dans votre AWS Cloud. Pour plus d'informations, consultez Migration automatique des EC2 bases de données vers Amazon Relational Database Service à l'aide de. AWS Database Migration Service	20 novembre 2024
Les blue/green déploiements Amazon RDS prennent en charge le dimensionnement du stockage	Vous pouvez désormais utiliser les blue/green déploiements Amazon RDS pour ajuster les paramètres de stockage dans un environnement écologique afin d'optimiser l'allocation des ressources. Pour plus d'informations, consultez la section Spécification des modifications lors de la création d'un blue/green déploiement.	20 novembre 2024

Les blue/green déploiements Amazon RDS ajoutent l'initialisation du stockage	Amazon RDS prend désormais en charge l'initialisation du stockage dans des environnements écologiques pour les blue/green déploiements, améliorant ainsi les performances des volumes dès la première utilisation sans affecter la disponibilité. Pour plus d'informations, consultez la section Chargement différé et initialisation du stockage pour les blue/green déploiements.	20 novembre 2024
Amazon RDS pour Oracle prend en charge les classes d'instance db.m7i et db.r7i pour BYOL	Vous pouvez utiliser les classes d'instance db.m7i et db.r7i pour RDS for Oracle sur le modèle de licence BYOL. Toutes les éditions d'Oracle Database sont prises en charge. Pour plus d'informations, consultez les sections Types de classes d'instances de base de données et RDS pris en charge pour les classes d'instance de base de données Oracle.	18 novembre 2024

Amazon RDS Proxy prend en charge RDS pour PostgreSQL 17	Vous pouvez désormais créer des proxys à l'aide du proxy RDS pour les instances de base de données RDS pour PostgreSQL 17. Pour plus d'informations, consultez la section Utilisation du proxy Amazon RDS.	15 novembre 2024
Amazon RDS prend en charge MySQL 8.0.40	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.40. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	13 novembre 2024
Amazon RDS prend en charge la réplication de sauvegarde pour l'Afrique (Le Cap) et l'Asie-Pacifique (Hyderabad)	La réplication de sauvegarde est désormais disponible pour les bases de données des régions Afrique (Le Cap) et Asie-Pacifique (Hyderabad). Pour plus d'informations sur les régions disponibles, consultez la section Réplication de sauvegardes automatisées vers une autre Région AWS.	1er novembre 2024

Amazon RDS prend en charge les classes d'instance db.m7i et db.r7i	Vous pouvez désormais utiliser les classes d'instance db.m7i et db.r7i pour RDS pour MySQL, RDS pour PostgreSQL et RDS pour MariaDB. Pour en savoir plus, consultez la section Classes d'instances de base de données.	29 octobre 2024
Amazon RDS pour Oracle prend en charge la version 24.1.v1 d'Oracle APEX	Vous pouvez utiliser Oracle APEX 24.1.v1 avec Oracle Database 19c ou version ultérieure. Pour plus d'informations, consultez Oracle Application Express.	22 octobre 2024
Les clusters de base de données multi-AZ prennent en charge l'authentification de base de données IAM	Vous pouvez désormais vous authentifier auprès de votre cluster de base de données multi-AZ à l'aide de l'authentification de base de données AWS Identity and Access Management (IAM). Pour plus d'informations, consultez la section Paramètres de création de clusters de bases de données multi-AZ.	17 octobre 2024
Amazon RDS prend en charge MariaDB 11.4	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 11.4. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	15 octobre 2024

Amazon RDS prend en charge Console-to-Code	Vous pouvez désormais l'utiliser Console-to-Code pour générer du code à partir d'actions que vous effectuez dans la console RDS. Le code généré peut vous aider à écrire du code pour automatiser votre utilisation d'autres AWS services. Pour plus d'informations, consultez Utiliser Console-to-Code pour générer du code pour les actions de votre console Amazon RDS.	3 octobre 2024
Amazon RDS pour Oracle prend en charge l'agent de gestion Oracle 13.5.0.0.v2	RDS prend en charge l'agent de gestion Oracle 13.5.0.0.v2, qui nécessite Oracle Management Service (OMS) 13.5.0.23. Pour plus d'informations, consultez la section Exigences relatives à l'agent de gestion.	25 septembre 2024

Les intégrations sans ETL avec Amazon Redshift sont généralement disponibles	Les intégrations sans ETL rendent les données transacti onnelles disponibles dans Amazon Redshift quelques secondes après leur écriture sur une instance de base de données RDS pour MySQL. La fonctionnalité est désormais disponible pour tous. Pour plus d'informa tions, consultez Utilisation des intégrations Amazon RDS Zero-ETL avec Amazon Redshift.	12 septembre 2024
Amazon RDS prend en charge MariaDB 10.11.9, 10.6.19 et 10.5.26	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 10.11.9, 10.6.19 et 10.5.26 de MariaDB. Pour plus d'informa tions, consultez la section Exigences relatives à l'agent de gestion.	4 septembre 2024
Amazon RDS for Oracle prend en charge les options OEM, OEMAGENT et OLS pour l'architecture CDB	Vous pouvez désormais utiliser Oracle Enterprise Manager et Oracle Label Security with RDS pour les instances Oracle CDB. Pour plus d'informations, consultez Oracle Enterprise Manager et Oracle Label Security.	4 septembre 2024

RDS Custom pour SQL Server est disponible dans d'autres régions	RDS Custom for SQL Server est désormais disponible dans les régions USA Ouest (Californie du Nord), Asie-Pacifique (Osaka) et Europe (Paris). Pour plus d'informations, consultez Régions prises en charge et moteurs de base de données pour RDS Custom.	29 août 2024
Support étendu Amazon RDS version 5.7.44-RDS.20240808 pour RDS pour MySQL	La version 5.7.44-RDS.20240808 de RDS Extended Support est désormais disponible pour RDS for MySQL. Pour plus d'informations, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	29 août 2024
Amazon RDS est disponible dans la région Asie-Pacifique (Malaisie)	Amazon RDS est désormais disponible dans la région Asie-Pacifique (Malaisie). Pour plus d'informations, consultez Régions et zones de disponibilité.	22 août 2024
Amazon RDS prend en charge MySQL 8.0.39	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.39. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	12 août 2024

Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSPreviewServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDSPreview` service. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	7 août 2024
Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSBetaServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDSBeta` service. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	7 août 2024
Amazon RDS prend en charge MySQL 8.4 dans l'environnement Database Preview	MySQL 8.4 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 8.4 dans l'environnement Database Preview.	1er août 2024

Mise à jour des autorisations de rôle lié à un service IAM	La `AmazonRDSCustomServiceRolePolicy` politique accorde désormais des autorisations supplémentaires pour communiquer avec les services Amazon RDS via un autre service Région AWS et pour copier EC2 des images. Pour plus d'informations, consultez les mises à jour des politiques AWS gérées par Amazon RDS.	18 juillet 2024
Amazon RDS prend en charge MariaDB 11.4 dans l'environnement Database Preview	MariaDB 11.4 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez la version 11.4 de MariaDB dans l'environnement Database Preview.	18 juillet 2024
AWS Le pilote ODBC pour MySQL est généralement disponible	Le pilote ODBC Amazon Web Services (AWS) pour MySQL est un pilote client conçu pour la haute disponibilité de RDS pour MySQL. Pour plus d'informations, consultez Connexion à RDS pour MySQL avec le pilote ODBC Amazon Web Services (AWS) pour MySQL.	18 juillet 2024

Mise à jour de la politique existante	Amazon RDS a supprimé `sns:Publish` l'autorisation `AmazonRDSServiceRolePolicy` du rôle lié au `AWSServiceRoleForRDS` service. Pour plus d'informations, consultez la politique AWS gérée : Amazon RDSService RolePolicy.	2 juillet 2024
AWS Marketplace offre privée pour Db2	AWS Marketplace prend désormais en charge les offres privées pour une licence Db2 jusqu'à Amazon RDS AWS Marketplace pour Db2. Pour plus d'informations, consultez la section Obtention d'une offre privée.	1 juillet 2024
Exporter les données instantanées d'un cluster de bases de données multi-AZ vers Amazon S3	Vous pouvez désormais exporter des données instantanées de clusters de bases de données multi-AZ vers Amazon S3. Pour plus d'informations, veuillez consulter Exportation de données d'instantanés de bases de données vers Amazon S3.	27 juin 2024

Amazon RDS for Oracle prend en charge les classes d'instances préconfigurées optimisées pour la mémoire r6i	Les classes d'instance de base de données Oracle db.r6i sont optimisées pour les charges de travail qui nécessitent de la mémoire, du stockage et par vCPU supplémentaires. I/O Par exemple, le multithreading est activé dans db.r6i.8xlarge.tpc 2.mem4x et fournit 4 fois plus de mémoire que db.r6i.8x large. Pour plus d'informations, consultez Classes d'instances RDS pour Oracle.	21 juin 2024
Support étendu Amazon RDS version 5.7.44-RDS.20240529 pour RDS pour MySQL	La version 5.7.44-RD S.20240529 de RDS Extended Support est désormais disponible pour RDS for MySQL. Pour plus d'informa tions, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	20 juin 2024
Amazon RDS prend en charge MySQL 8.0.37	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.37. Pour plus d'informa tions, consultez MySQL sur les versions Amazon RDS.	18 juin 2024

Amazon RDS prend en charge MariaDB 10.11.8, 10.6.18, 10.5.25 et 10.4.34	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 10.11.8, 10.6.18, 10.5.25 et 10.4.34 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	14 juin 2024
Amazon RDS met fin à la prise en charge des classes d'instances de base de données db.m4, db.r4 et db.t2	Pour les moteurs de base de données RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL, vous ne pouvez plus créer d'instances de base de données utilisant les classes d'instance db.m4, db.r4 et db.t2. RDS met automatiquement à niveau les instances de base de données existantes qui utilisent ces classes vers une génération plus récente. Pour en savoir plus, consultez la section Classes d'instances de base de données.	4 juin 2024

Les clusters de bases de données multi-AZ sont disponibles en supplément Régions AWS	Vous pouvez créer des clusters de bases de données multi-AZ dans plusieurs Régions AWS versions. Pour un tableau répertoriant toutes les régions prises en charge, consultez Régions prises en charge et moteurs de base de données pour les clusters de bases de données multi-AZ dans Amazon RDS.	29 mai 2024
AWS Pilote Python généralement disponible	Le pilote Python Amazon Web Services (AWS) est conçu comme un wrapper Python avancé. Ce wrapper complète et étend les fonctionnalités du pilote open source Psycopg. Pour plus d'informations, consultez la section Connexion aux instances de base de données avec les AWS pilotes.	23 mai 2024
Le proxy RDS est disponible dans un plus grand nombre de régions	RDS Proxy est désormais disponible dans les régions Asie-Pacifique (Hyderabad), Asie-Pacifique (Melbourne), Moyen-Orient (Émirats arabes unis), Israël (Tel Aviv), Canada Ouest (Calgary) et Europe (Zurich). Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	21 mai 2024

Licence DB2 via AWS Marketplace	Une fois la licence DB2 activée AWS Marketplace, vous pouvez désormais payer un tarif horaire pour vous abonner à des licences DB2 pour Amazon RDS pour DB2. Pour plus d'informations, consultez les options de licence Amazon RDS pour DB2.	21 mai 2024
Amazon RDS prend en charge un accès détaillé pour Performance Insights	Vous pouvez désormais autoriser ou refuser l'accès à des dimensions individuelles dans Performance Insights. Cet accès détaillé peut être utilisé pour `GetResourceMetrics` `DescribeDimensionKeys` , et `GetDimensionKeyDetails` pour des actions. Pour plus d'informations, consultez la section Octroi d'un accès détaillé à Performance Insights.	21 mai 2024
Versions de support étendu d'Amazon RDS pour RDS pour MySQL	Vous pouvez consulter toutes les versions de RDS Extended Support pour les versions de RDS pour MySQL. Pour plus d'informations, consultez les versions Amazon RDS Extended Support pour RDS for MySQL.	16 mai 2024

Amazon RDS prend en charge MySQL 8.3 dans l'environnement Database Preview	MySQL 8.3 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 8.3 dans l'environnement Database Preview.	30 avril 2024
Amazon RDS pour DB2 prend en charge les fuseaux horaires	RDS pour Db2 prend désormais en charge la définition de fuseaux horaires locaux pour les nouvelles instances de base de données RDS for Db2. Pour plus d'informations, consultez Fuseaux horaires locaux pour Amazon RDS pour les instances de base de données DB2.	25 avril 2024
Mise à jour des autorisations de rôle lié à un service IAM	La `AmazonRDSCustomServiceRolePolicy` politique accorde désormais des autorisations supplémentaires pour associer un rôle de service en tant que profil d'instance à une instance personnalisée RDS. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	19 avril 2024

Amazon RDS for Oracle prend en charge le passage à Oracle Data Guard dans tous les domaines Régions AWS	Vous pouvez désormais utiliser le passage à Oracle Data Guard dans toutes les régions prises en charge. Pour plus d'informations, voir Présentation du passage à Oracle Data Guard.	16 avril 2024
RDS Custom pour Oracle prend en charge Oracle Standard Edition 2	Vous pouvez désormais créer des instances de base de données à l'aide de l'édition Standard 2 sur Oracle Database 12c version 1 (12.1), 12c version 2 (12.2), 18c et 19c. Vous pouvez créer les deux CDBs et non-CDBs. Pour plus d'informations, consultez la section Support relatif à l'édition et aux licences pour RDS Custom pour Oracle.	11 avril 2024
Amazon RDS pour Oracle prend en charge la version 23.2.v1 d'Oracle APEX	Vous pouvez utiliser Oracle APEX 23.2.v1 avec Oracle Database 19c ou version ultérieure. Pour plus d'informations, consultez Oracle Application Express.	11 avril 2024

Mise à jour des autorisations de rôle liées au service RDS Custom	`AmazonRDSCustomServiceRolePolicy` Il accorde désormais des autorisations supplémentaires pour permettre à RDS Custom for SQL Server d'obtenir des informations sur le type d' EC2 instance et de modifier le type d'instance hôte de base de données. Pour plus d'informations, voir Mises à jour des politiques AWS gérées.	8 avril 2024
Amazon RDS Custom pour Oracle prend en charge la classe d'instance de base de données db.x2iezn	Vous pouvez désormais utiliser la classe d'instance db.x2iezn pour les instances de base de données RDS Custom for Oracle. Pour plus d'informations, consultez Prise en charge des classes d'instances de base de données pour RDS Custom for Oracle.	26 mars 2024
Amazon RDS prend en charge les classes d'instance db.c6gd pour les clusters de bases de données multi-AZ	Vous pouvez désormais utiliser les classes d'instance db.c6gd pour les déploiements de clusters de bases de données multi-AZ. Pour plus d'informations, consultez la section Disponibilité des classes d'instance pour les clusters de base de données multi-AZ.	21 mars 2024

Support étendu Amazon RDS	La création ou la restauration d'une base de données RDS pour MySQL 5.7 ou RDS pour PostgreSQL 11 inscrit désormais automatiquement cette base de données dans Amazon RDS Extended Support afin que vos applications existantes continuent de fonctionner telles quelles. Vous pouvez vous désinscrire du support étendu RDS pour éviter des frais après la date de fin du support standard RDS pour votre moteur de base de données. Pour plus d'informations, consultez Utilisation du support étendu Amazon RDS.	21 mars 2024
Intégration de RDS pour DB2 avec AWS License Manager	RDS pour Db2 est désormais intégré à. AWS License Manager Si vous utilisez le modèle BYOL (Bring your own license), l' AWS License Manager intégration permet de surveiller l'utilisation de vos licences DB2 au sein de votre organisation. Pour plus d'informations, consultez la section Intégration avec AWS License Manager.	20 mars 2024

Rotation des certificats CA pour les clusters de bases de données multi-AZ	Vous pouvez désormais alterner les certificats CA pour vos clusters de bases de données multi-AZ. Envisagez d'utiliser l'un des nouveaux certificats CA rds-ca-rsa 2048-g1, rds-ca-rsa 4096-g1 ou 384-g1. rds-ca-ecc Pour plus d'informations, consultez Rotation de votre SSL/TLS certificat.	6 mars 2024
Amazon RDS prend en charge le stockage io2 Block Express	Vous pouvez désormais créer des instances de base de données RDS qui utilisent le type de stockage io2 Block Express. Pour plus d'informations, consultez io2 Block Express storage.	6 mars 2024
RDS Custom pour SQL Server prend en charge les classes d'instance de base de données db.r5b et db.x2iedn	Vous pouvez désormais utiliser les classes d'instance db.r5b et db.x2iedn pour les instances de base de données RDS Custom for SQL Server. Pour plus d'informations, consultez la section Prise en charge des classes d'instance de base de données pour RDS Custom pour SQL Server.	4 mars 2024

RDS Custom for Oracle est disponible dans la région Moyen-Orient (EAU)	Vous pouvez créer RDS Custom pour les instances de base de données Oracle dans la région du Moyen-Orient (Émirats arabes unis). Pour un tableau répertoriant toutes les régions prises en charge Régions AWS, voir Régions prises en charge et moteurs de base de données pour RDS Custom for Oracle.	4 mars 2024
Nouvelle politique AWS gérée	Amazon RDS a ajouté une nouvelle politique gérée nommée `AmazonRDSCustomInstanceProfileRolePolicy` pour permettre à RDS Custom d'effectuer des actions d'automatisation et des tâches de gestion de base de données via un profil d' EC2 instance. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	27 février 2024
Amazon RDS prend en charge MariaDB 10.11.7, 10.6.17, 10.5.24 et 10.4.33	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 10.11.7, 10.6.17, 10.5.24 et 10.4.33 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	26 février 2024

Les clusters de bases de données multi-AZ Amazon RDS prennent en charge le volume de stockage Amazon EBS gp3	Les clusters de base de données multi-AZ prennent désormais en charge les volumes EBS basés sur des SSD gp3. Pour plus d'informations, consultez la section Stockage GP3.	26 février 2024
Assistance Amazon RDS pour la AWS Secrets Manager région d'Israël (Tel Aviv)	Amazon RDS prend en charge Secrets Manager dans la région d'Israël (Tel Aviv). Pour plus d'informations, consultez Gestion des mots de passe avec Amazon RDS et AWS Secrets Manager.	21 février 2024
Amazon RDS pour DB2 prend en charge la journalisation des audits	RDS pour Db2 prend désormais en charge la journalisation des audits au niveau de la base de données. Lorsque vous activez la journalisation d'audit pour une base de données RDS pour DB2, Amazon RDS enregistre l'activité de la base de données et stocke les journaux d'audit dans Amazon S3. Pour plus d'informations, consultez la section Journalisation des audits DB2.	15 février 2024

Support étendu Amazon RDS	Amazon RDS active désormais automatiquement le support étendu Amazon RDS lorsque les versions majeures du moteur RDS pour MySQL et RDS pour PostgreSQL dans vos instances de base de données et vos clusters de bases de données multi-AZ atteignent la date de fin de support standard RDS. Pour plus d'informations, consultez Utilisation du support étendu Amazon RDS.	15 février 2024
Amazon RDS prend en charge MySQL 8.0.36	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.36. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	12 février 2024
Amazon RDS prend en charge le classement EBCDIC pour RDS pour DB2	Vous pouvez désormais créer des bases de données DB2 qui utilisent des séquences de classement EBCDIC pour trier le contenu des bases de données. Pour plus d'informations, consultez le classement EBCDIC pour les bases de données DB2 sur Amazon RDS.	29 janvier 2024

Mise à jour du certificat CA par défaut	Le certificat CA par défaut est défini sur `rds-ca-rsa2048-g1` . Pour plus d'informations, consultez Utilisation SSL/TLS pour chiffrer une connexion à une instance de base de données.	26 janvier 2024
Amazon RDS pour PostgreSQL prend en charge deux nouvelles caisses pour PL/Rust, croaring-rs et num-bigint	Vous pouvez utiliser deux nouvelles caisses dans Amazon RDS for PostgreSQL. Pour plus d'informations, consultez la section Utilisation de caisses avec PL/Rust.	24 janvier 2024
Amazon RDS pour PostgreSQL prend en charge la version 1.3 du protocole TLS	Vous pouvez utiliser la version 1.3 de Transport Layer Security (TLS) dans RDS pour PostgreSQL. Pour plus d'informations, consultez la section Utilisation de SSL avec une instance de base de données PostgreSQL.	24 janvier 2024
RDS Custom pour SQL Server prend en charge Microsoft SQL Server 2022	Vous pouvez désormais créer RDS Custom pour les instances de base de données SQL Server qui utilisent SQL Server 2022. Pour plus d'informations, consultez la section Utilisation de RDS Custom pour SQL Server.	22 janvier 2024

Mise à jour des autorisations de politique AWS gérées	Le `AmazonRDSServiceRolePolicy` rôle `AWSServiceRoleForRDS` lié au service a une nouvelle déclaration. IDs Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	19 janvier 2024
RDS Custom for Oracle prend en charge la région Europe (Paris)	Vous pouvez créer RDS Custom pour les instances de base de données Oracle dans la région Europe (Paris). Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	18 janvier 2024
Amazon RDS for MySQL prend en charge la réplication multi-sources	Vous pouvez désormais utiliser la réplication multi-source sur RDS pour les instances de base de données MySQL. Pour plus d'informations, consultez Configuration de la réplication multi-source sur RDS pour MySQL.	16 janvier 2024
Amazon RDS prend en charge MySQL 8.2 dans l'environnement Database Preview	MySQL 8.2 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 8.2 dans l'environnement Database Preview.	11 janvier 2024

RDS Proxy est disponible dans la région Europe (Espagne)	RDS Proxy est désormais disponible dans la région Europe (Espagne). Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	8 janvier 2024
Amazon RDS est disponible dans la région Ouest du Canada (Calgary)	Amazon RDS est désormais disponible dans la région de l'Ouest du Canada (Calgary) . Pour plus d'informations, consultez Régions et zones de disponibilité.	20 décembre 2023
Amazon RDS pour DB2 prend en charge 5 000 utilisateurs locaux	Vous pouvez désormais ajouter jusqu'à 5 000 utilisateurs locaux à une liste d'autorisation. Pour plus d'informations, consultez rdsadmin.add_user.	20 décembre 2023
Amazon RDS permet de consulter les recommandations et d'y répondre	Les recommandations d'Amazon RDS incluent désormais des recommandations proactives basées sur des seuils et des recommandations réactives basées sur l'apprentissage automatique pour RDS pour PostgreSQL. Pour plus d'informations, consultez les recommandations d'Amazon RDS.	19 décembre 2023

Amazon RDS prend en charge MariaDB 10.11.6, 10.6.16, 10.5.23 et 10.4.32	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 10.11.6, 10.6.16, 10.5.23 et 10.4.32 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	12 décembre 2023
Amazon RDS introduit des intégrations sans ETL avec Amazon Redshift (version préliminaire)	Les intégrations Zero-ETL fournissent une solution entièrement gérée permettant de rendre les données transactionnelles disponibles dans Amazon Redshift quelques secondes après leur écriture sur une instance de base de données RDS pour MySQL. Pour plus d'informations, consultez Utilisation des intégrations Amazon RDS Zero-ETL avec Amazon Redshift (version préliminaire).	28 novembre 2023
Amazon RDS prend en charge les moteurs de IBM Db2 base de données	Vous pouvez désormais exécuter des moteurs IBM Db2 de base de données dans Amazon RDS. Pour plus d'informations, consultez Amazon RDS pour DB2.	27 novembre 2023

RDS pour PostgreSQL prend en charge les mises à niveau des versions majeures de PostgreSQL 16.1 et les mises à niveau des versions mineures vers les versions 15.5, 14.10, 13.13, 12.17 et 11.22	Avec RDS pour PostgreSQL, vous pouvez désormais mettre à niveau le moteur de base de données vers la version majeure 16.1 et les mises à niveau mineures vers les versions 15.5, 14.10, 13.13, 12.17 et 11.22. Pour plus d'informations, consultez la section Mise à niveau du moteur de base de données PostgreSQL pour Amazon RDS.	17 novembre 2023
RDS Custom pour Oracle prend en charge les groupes d'options	Vous pouvez créer ou modifier un groupe d'options et l'associer à une instance de base de données RDS Custom pour Oracle. L'`Timezone`option est désormais prise en charge. Pour plus d'informations, consultez la section Utilisation des groupes d'options dans RDS Custom pour Oracle.	17 novembre 2023

Amazon RDS for MySQL prend en charge le plug-in Group Replication	Vous pouvez désormais configurer un cluster actif-actif avec des instances de base de données RDS pour MySQL version 8.0.35 ou supérieure en utilisant le plugin Group Replication développé et maintenu par la communauté MySQL. Pour plus d'informations, consultez Configuration de clusters actifs-actifs pour RDS for MySQL.	17 novembre 2023
Amazon RDS Proxy prend en charge RDS pour PostgreSQL 16.1	Vous pouvez désormais créer des proxys à l'aide du proxy RDS pour les instances de base de données RDS pour PostgreSQL 16.1. Pour plus d'informations, consultez la section Utilisation du proxy Amazon RDS.	17 novembre 2023
RDS Custom pour SQL Server prend en charge l'édition Microsoft SQL Server 2019 Developer	Vous pouvez créer des instances de base de données RDS Custom pour SQL Server qui utilisent l'édition SQL Server 2019 Developer. Pour plus d'informations, consultez Modèle Bring Your Own Media avec RDS Custom for SQL Server.	16 novembre 2023

Mises à niveau mineures des clusters de bases de données multi-AZ avec un temps d'arrêt minimal	Lorsque vous effectuez une mise à niveau de version mineure d'un cluster de base de données multi-AZ, Amazon RDS met désormais à niveau les instances de base de données du lecteur avant l'instance du rédacteur , réduisant ainsi considéra blement les temps d'arrêt. Vous pouvez encore réduire les temps d'arrêt à une seconde ou moins en utilisant le proxy RDS. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'un cluster de bases de données multi-AZ.	16 novembre 2023
RDS pour SQL Server prend en charge Microsoft SQL Server 2022	Vous pouvez désormais créer des instances de base de données RDS qui utilisent SQL Server 2022. Pour de plus amples informations, veuillez consulter Versions de Microsoft SQL Server sur Amazon RDS.	15 novembre 2023

RDS for MySQL prend en charge la mise à niveau des instantanés de la version 5.7 à la version 8.0	Vous pouvez désormais mettre à niveau la version du moteur d'un instantané RDS pour MySQL de la version 5.7 à la version 8.0. Vous pouvez le faire en utilisant le AWS Management Console ou le `ModifyDBSnapshot` fonctionnement de l'API RDS ou AWS CLI. Pour plus d'informations, consultez Mise à niveau d'une version du moteur de capture instantanée de base de données MySQL.	15 novembre 2023
RDS Custom pour SQL Server prend en charge la restauration instantanée de 1 000 bases de données	Vous pouvez désormais rendre jusqu'à 1 000 bases de données éligibles à une sauvegarde complète et à une restauration instantanée sur votre instance de base de données RDS Custom for SQL Server. Pour plus d'informations, consultez Restaurer une instance RDS Custom pour SQL Server à un moment donné.	15 novembre 2023

RDS Custom pour SQL Server prend en charge l'utilisation d'une clé principale de service	RDS Custom pour SQL Server prend désormais en charge l'utilisation d'une clé principale de service (SMK). Un SMK vous permet de chiffrer des objets tels que des informations d'identification et d'utiliser les fonctionnalités de SQL Server telles que le TDE et le chiffrement des colonnes. Pour plus d'informations, consultez la section Utilisation d'une clé principale de service avec RDS Custom pour SQL Server.	13 novembre 2023
Amazon RDS prend en charge MySQL 8.1 dans l'environnement de prévisualisation de base de données	MySQL 8.1 est désormais disponible dans l'environnement Database Preview dans l'est des États-Unis (Ohio) Région AWS. Pour plus d'informations, consultez MySQL version 8.1 dans l'environnement de prévisualisation de base de données.	10 novembre 2023
RDS prend en charge MySQL 8.0.35 et MySQL 5.7.44	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.35 et 5.7.44. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	9 novembre 2023

RDS Proxy prend en charge les clusters de base de données multi-AZ	RDS Proxy prend désormais en charge la connexion aux clusters de base de données multi-AZ. Pour plus d'informations, consultez Utilisation des points de terminaison du proxy Amazon RDS.	9 novembre 2023
RDS Custom pour Oracle est disponible dans AWS GovCloud (US) Regions	Amazon RDS est désormais disponible dans les AWS GovCloud (US) Regions. Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	9 novembre 2023
L'option Écritures optimisées pour Amazon RDS prend en charge la classe d'instances de base de données db.m5	L'option Écritures optimisées pour Amazon RDS prend désormais en charge la classe d'instances de base de données db.m5. Pour plus d'informations, consultez Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB et Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MySQL.	9 novembre 2023

Amazon RDS for Oracle prend en charge la configuration à locataires multiples de l'architecture CDB	Grâce à la fonctionnalité à locataires multiples RDS for Oracle, RDS fournit une architecture multilocataire Oracle entièrement gérée et une expérience pour vos bases de données Oracle. Vous pouvez utiliser RDS APIs pour créer plusieurs bases de données PDBs, appelées bases de données mutualisées, dans une CDB. RDS propose la configuration à locataires multiples de l'architecture CDB comme alternative à la configuration à locataire unique existante. Pour plus d'informations, consultez Configuration à locataires multiples de l'architecture CDB.	8 novembre 2023

Amazon RDS exporte les métriques Performance Insights vers Amazon CloudWatch	Performance Insights vous permet d'exporter les tableaux de bord de métriques préconfigurés ou personnalisés vers Amazon. CloudWatch Les tableaux de bord des métriques exportés peuvent être consultés dans la CloudWatch console. Vous pouvez également exporter un widget métrique Performance Insights sélectionné et consulter les données des métriques dans la CloudWatch console. Pour plus d'informations, consultez Exporter les métriques Performance Insights vers CloudWatch.	8 novembre 2023
Amazon RDS Custom for Oracle vous permet de mettre à niveau le système d'exploitation sur une instance de base de données	Vous pouvez désormais mettre à niveau la base de données ou le système d'exploitation (SE) d'une instance de base de données RDS Custom for Oracle à l'aide de la commande d'interface de ligne de commande `modify-db-instance`. Pour plus d'informations, consultez Mise à niveau d'une instance de base de données pour Amazon RDS Custom for Oracle.	7 novembre 2023

RDS Proxy prend en charge Extended Protocol for RDS for PostgreSQL	Vous pouvez désormais exécuter des protocoles de requête étendus sur une instance de base de données RDS for PostgreSQL. Pour plus d'informations, consultez la section Utilisation du proxy Amazon RDS.	6 novembre 2023
Support de RDS pour PostgreSQL pour les déploiements RDS Blue/Green	Vous pouvez désormais créer un blue/green déploiement à partir d'une instance de base de données RDS pour PostgreSQL. Pour plus d'informations, consultez Utilisation des Blue/Green déploiements Amazon RDS pour les mises à jour de bases de données.	26 octobre 2023
Mise à jour des politiques AWS gérées	Les politiques gérées par `AmazonRDSPerformanceInsightsReadOnly` et `AmazonRDSPerformanceInsightsFullAccess` incluent désormais `Sid` (ID d'instruction) comme identifiant dans l'instruction de la politique. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	23 octobre 2023

RDS Custom for Oracle prend en charge la région Europe (Milan)	Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	23 octobre 2023
Activer l'option Écritures optimisées pour RDS sur les bases de données existantes	Vous pouvez désormais activer l'option Écritures optimisées pour RDS sur une instance de base de données existante même si elle a été créée avec une version de moteur, une classe d'instanc es de base de données ou une configuration de système de fichiers qui ne prend pas en charge cette fonctionn alité. Pour plus d'informa tions, consultez Activation de l'option Écritures optimisée s pour RDS sur une base de données existante pour RDS for MySQL et Activation de l'option Écritures optimisée s pour RDS sur une base de données existante pour RDS for MariaDB.	19 octobre 2023

Amazon RDS prend en charge l'utilisation de volumes dédiés aux journaux (DLV).	Vous pouvez désormais utiliser un volume de journal dédié (DLV) avec RDS pour MariaDB, RDS pour MySQL et RDS pour PostgreSQL. DLVs sont idéaux pour les bases de données ayant un stockage alloué important, des exigences élevées en matière d'E/S par seconde (IOPS) ou des charges de travail sensibles à la latence. Pour plus d'informations, consultez Utilisation d'un volume dédié aux journaux (DLV).	17 octobre 2023
Amazon RDS for PostegreSQL, MySQL et MariaDB prennent en charge les nouvelles classes d'instances de base de données	Vous pouvez créer des instances de base de données Amazon RDS exécutant PostgreSQL, MySQL et MariaDB qui utilisent les classes d'instances de base de données db.m6.in, db.m6idn, db.r6.in et db.r6.idn. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	12 octobre 2023

Amazon RDS for PostgreSQL prend en charge pgactive	L'extension pgactive est disponible dans Amazon RDS for PostgreSQL. Pour en savoir plus, consultez Utilisation des extensions PostgreSQL avec Amazon RDS for PostgreSQL.	9 octobre 2023
RDS Custom for Oracle est disponible dans la région Asie-Pacifique (Jakarta)	Vous pouvez créer RDS Custom pour les instances de base de données Oracle dans la région Asie-Pacifique (Jakarta). Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	5 octobre 2023
RDS Custom for SQL Server prend en charge les nouveaux classements au niveau du serveur	RDS Custom for SQL Server prend désormais en charge un large éventail de classements de serveur, aussi bien dans l'encodage traditionnel que dans l'encodage UTF-8, pour les paramètres régionaux SQL_Latin, Japonais, Allemand et Arabe. Pour en savoir plus, consultez Prise en charge des classements et des caractères pour les instances de base de données RDS Custom for SQL Server.	26 septembre 2023

Mise à jour des autorisations de politique AWS gérées	Le rôle `AWSServiceRoleForRDSCustom` lié au service dispose `AmazonRDSCustomServiceRolePolicy` de nouvelles autorisations qui permettent à RDS Custom de créer, de modifier et de supprimer EventBridge des règles gérées. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	20 septembre 2023
Amazon RDS publie les contre-métriques Performance Insights sur Amazon CloudWatch	La fonction mathématique des métriques DB_PERF_INSIGHTS de la CloudWatch console vous permet d'interroger Amazon RDS pour obtenir les indicateurs de compteur Performance Insights. Pour plus d'informations, consultez Création d' CloudWatch alarmes pour surveiller Amazon RDS.	20 septembre 2023
Performance Insights prend en charge les statistiques au niveau de la synthèse pour SQL Server	Lorsque vous utilisez Performance Insights, vous pouvez afficher les statistiques SQL au niveau de l'instruction et de la synthèse pour Amazon RDS for SQL Server. Pour en savoir plus, consultez Analyse des requêtes en cours d'exécution dans SQL Server.	18 septembre 2023

Amazon RDS for PostgreSQL, MySQL et MariaDB prennent en charge les types de classe d'instances de base de données db.m6.id et db.r6.id	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL, MySQL ou MariaDB qui utilisent les types de classe d'instances de base de données db.m6.id et db.r6.id. Ces types offrent un NVMe stockage SSD local. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	11 septembre 2023
Prise en charge de la mise à niveau de version majeure pour les clusters de bases de données multi-AZ RDS for PostgreSQL	Vous pouvez désormais effectuer des mises à niveau de version majeure de vos clusters de bases de données multi-AZ RDS for PostgreSQ L. Pour plus d'informations, consultez Mise à niveau de la version du moteur d'un cluster de bases de données multi-AZ.	7 septembre 2023

Amazon RDS prend en charge MariaDB 10.11.5, 10.6.15, 10.5.22 et 10.4.31	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 10.11.5, 10.6.15, 10.5.22 et 10.4.31. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	7 septembre 2023
Support étendu Amazon RDS	Amazon RDS annonce la possibilité de continuer à exécuter les versions majeures de moteur RDS for MySQL et RDS for PostgreSQL dans vos instances de base de données après la date de fin du support standard RDS. Pour plus d'informations, consultez Utilisation du support étendu Amazon RDS.	1er septembre 2023
RDS Custom prend en charge le démarrage et l'arrêt d'une instance de base de données RDS Custom for SQL Server	RDS Custom prend maintenant en charge le démarrage et l'arrêt d'une instance de base de données RDS Custom for SQL Server. Pour plus d'informations, consultez Démarrage et arrêt d'une instance de base de données RDS Custom for SQL Server.	31 août 2023

L'option Écritures optimisées pour Amazon RDS prend en charge la classe d'instances de base de données db.r5	L'option Écritures optimisées pour Amazon RDS prend désormais en charge la classe d'instances de base de données db.r5. Pour plus d'informations, consultez Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB et Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MySQL.	31 août 2023
Amazon RDS for Oracle prend en charge la mise à niveau automatique des fichiers de fuseau horaire pour CDBs	Avec l'option `TIMEZONE_FILE_AUTOUPGRADE`, vous pouvez mettre à niveau le fichier de fuseau horaire actuel vers la dernière version sur votre base de données de conteneur (CDB) RDS for Oracle. Pour plus d'informations, consultez Mise à niveau automatique du fichier Oracle sur le fuseau horaire.	29 août 2023

L'option Écritures optimisées pour Amazon RDS prend en charge les classes d'instances de base de données db.m6g et db.m6i	L'option Écritures optimisées pour Amazon RDS prend désormais en charge les classes d'instances de base de données db.m6g et db.m6i. Pour plus d'informations, consultez Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB et Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MySQL.	28 août 2023
Amazon RDS prend en charge MariaDB 10.11	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.11. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	21 août 2023

Mise à jour des autorisations de politique AWS gérées	La politique `AmazonRDSCustomServiceRolePolicy` du rôle lié à un service `AWSServiceRoleForRDSCustom` dispose de nouvelles autorisations qui permettent à RDS Custom de créer des interfaces réseau. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	18 août 2023
Mise à jour des autorisations de politique AWS gérées	La politique gérée `AmazonRDSFullAccess` dispose de nouvelles autorisations qui vous permettent de générer, d'afficher et de supprimer le rapport d'analyse des performances pendant une période donnée. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	17 août 2023

Mise à jour des autorisations de politique AWS gérées	L'ajout de nouvelles autorisations à la politique gérée `AmazonRDSPerformanceInsightsReadOnly` et l'ajout d'une nouvelle politique gérée `AmazonRDSPerformanceInsightsFullAccess` vous permet de générer un rapport d'analyse de la charge de base de données pour une période donnée. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	16 août 2023
Amazon RDS prend en charge l'analyse des performances pendant une période donnée	L'analyse des performances vous permet de créer et d'examiner des rapports d'analyse des performances pour une période spécifique. Ce rapport fournit les informations identifiées et des recommandations pour résoudre les problèmes de performances. Pour plus d'informations, consultez Analyse de la charge de la base de données pour une période donnée (langue française non garantie).	16 août 2023

Amazon RDS Custom for Oracle prend en charge les classes d'instances de base de données db.r5b et db.x2iedn	Vous pouvez désormais utiliser les classes d'instances db.r5b et db.x2iedn pour les instances de base de données RDS Custom for Oracle. Pour plus d'informations, consultez Prise en charge des classes d'instances de base de données pour RDS Custom for Oracle.	16 août 2023
Amazon RDS Custom for Oracle prend en charge les classes d'instances de base de données db.m6i, db.r6i et db.t3	Vous pouvez désormais utiliser les classes d'instances db.m6i, db.r6i et db.t3 pour les instances de base de données RDS Custom for Oracle. Pour plus d'informations, consultez Prise en charge des classes d'instances de base de données pour RDS Custom for Oracle.	15 août 2023
Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 16 bêta 3 dans l'environnement en préversion de base de données	PostgreSQL version 16 Beta 3 est désormais disponible dans l'environnement de prévisualisation de base de données dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez Utilisation de l'environnement de prévisualisation de base de données.	11 août 2023

Amazon RDS prend en charge MySQL 8.0.34 et 5.7.43	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.34 et 5.7.43. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	9 août 2023
RDS for SQL Server prend en charge l'affichage des métriques du système d'exploitation pour le réplica de secours	Vous pouvez désormais examiner les métriques du système d'exploitation pour le réplica de secours pour RDS for SQL Server. Pour plus d'informations, consultez Affichage des métriques du système d'exploitation dans la console RDS.	3 août 2023
RDS pour Oracle prend en charge Oracle Data Guard pour CDBs	RDS pour Oracle prend en charge les répliques de lecture Data Guard pour les bases de données conteneurs Oracle Database 19c et 21c (). CDBs Vous pouvez créer, gérer et promouvoir des répliques de lecture dans un CDB, comme dans un autre CDB, en utilisant le RDS existant. APIs Pour plus d'informations, consultez Réplicas de lecture multilocataires.	1er août 2023

Amazon RDS est disponible dans la région Israël (Tel Aviv)	Amazon RDS est désormais disponible dans la région Israël (Tel Aviv). Pour plus d'informations, consultez Régions et zones de disponibilité.	1er août 2023
Amazon RDS prend en charge Oracle APEX version 23.1.v1	Vous pouvez utiliser Oracle APEX 23.1.v1 avec Oracle Database 19c ou version ultérieure. Pour plus d'informations, consultez Oracle Application Express.	26 juillet 2023
Amazon RDS Custom for Oracle prend en charge un SID Oracle autre que celui par défaut	Lorsque vous créez une instance de base de données RDS Custom for Oracle à l'aide d'Oracle Database 19c, vous pouvez spécifier un identifiant système Oracle autre que celui par défaut (SID Oracle). Cette valeur est également le nom de la CDB. Pour plus d'informations, consultez Considérations relatives à l'architecture multilocataire.	21 juillet 2023

RDS for SQL Server prend en charge Active Directory autogéré	Vous pouvez désormais utiliser Active Directory autogéré pour joindre directement vos instances de base de données RDS for SQL Server à vos domaines Microsoft Active Directory (AD). Les domaines AD autogéré peuvent être sur site ou dans le cloud. Pour plus d'informations, consultez Utilisation d'Active Directory autogéré.	7 juillet 2023
Prise en charge de la réplication logique PostgreSQL pour les clusters de bases de données multi-AZ	Vous pouvez désormais utiliser la réplication logique PostgreSQL avec votre cluster de bases de données multi-AZ pour répliquer et synchroniser des tables individuelles plutôt que l'ensemble d'une instance de base de données. Pour plus d'informations, consultez Configuration de la réplication logique PostgreSQL avec des clusters de bases de données multi-AZ pour Amazon RDS.	6 juillet 2023

Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 16 bêta 2 dans l'environnement en préversion de base de données	PostgreSQL version 16 Beta 2 est désormais disponible dans l'environnement de prévisualisation de base de données dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez Utilisation de l'environnement de prévisualisation de base de données.	6 juillet 2023
Mise à jour des autorisations de politique AWS gérées	La politique `AmazonRDSCustomServiceRolePolicy` du rôle lié à un service `AWSServiceRoleForRDSCustom` dispose de nouvelles autorisations qui permettent à RDS Custom for Oracle d'utiliser des instantanés. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	23 juin 2023
RDS prend en charge MariaDB 10.6.14, 10.5.21 et 10.4.30	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 10.6.14, 10.5.21 ou 10.4.30. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	22 juin 2023

RDS prend en charge MySQL 8.0.33 et 5.7.42	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.33 et 5.7.42. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	15 juin 2023
RDS prend en charge MariaDB 10.6.13, 10.5.20, 10.4.29 et 10.3.39	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 10.6.13, 10.5.20, 10.4.29 ou 10.3.39. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	15 juin 2023
RDS for Oracle prend en charge les espaces de table transportables	Vous pouvez migrer des données depuis une base de données Oracle sur site vers une instance de base de données RDS for Oracle en utilisant des espaces de table transportables. Cette technique ne nécessite pas de licence supplémentaire et constitue la technique de migration qui offre le moins de temps d'arrêt. Pour plus d'informations, consultez Migration à l'aide des espaces de table transportables Oracle.	15 juin 2023

Amazon RDS prend en charge RDS Proxy avec RDS for MariaDB version 10.6	Vous pouvez désormais créer un proxy RDS avec une base de données RDS for MariaDB version 10.6. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	15 juin 2023
RDS Custom for SQL Server prend en charge le modèle Bring Your Own Media (BYOM)	Vous pouvez désormais créer une version de moteur personnalisée (CEV) à l'aide de votre propre support SQL Server. Pour plus d'informations, consultez Modèle Bring Your Own Media avec RDS Custom for SQL Server.	8 juin 2023

RDS for Oracle peut convertir une base de données non-CDB Oracle Database 19c en CDB	Si votre instance de base de données exécute Oracle Database 19c avec une RU d'avril 2021 ou ultérieure, vous pouvez convertir une base de données non-CDB en CDB (base de données de conteneurs). Après avoir converti l'architecture, vous pouvez mettre à niveau votre CDB 19c vers une CDB 21c. Cette étape est nécessaire car vous ne pouvez pas mettre à niveau votre base de données et convertir l'architecture à l'aide d'une seule commande. Pour plus d'informations, consultez Conversion d'une base de données non-CDB RDS for Oracle en CDB.	31 mai 2023
Clusters de bases de données multi-AZ disponibles dans les régions de Chine	Les clusters de bases de données multi-AZ sont désormais disponibles en Régions AWS Chine (Pékin) et en Chine (Ningxia). Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour les clusters de base de données multi-AZ dans Amazon RDS.	30 mai 2023

Prise en charge par la fonctionnalité Lectures optimisées pour Amazon RDS des clusters de bases de données multi-AZ	La fonctionnalité Lectures optimisées pour Amazon RDS prend désormais en charge les clusters de bases de données multi-AZ. Pour plus d'informations, consultez Amélioration des performances des requêtes pour RDS for MySQL avec Amazon RDS Optimized Reads et Amélioration des performances des requêtes pour RDS for PostgreSQL avec Lectures optimisées pour Amazon RDS.	30 mai 2023
RDS Custom for Oracle prend en charge la région Asie-Pacifique (Jakarta)	Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	29 mai 2023
Créer un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ RDS for PostgreSQL comme source	Vous pouvez à présent créer un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ RDS for PostgreSQL comme source. Auparavant, seul RDS for MySQL était pris en charge. Pour plus d'informations, consultez Création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ.	24 mai 2023

Amazon RDS fournit des CloudWatch statistiques et des statistiques combinées sur les performances dans le tableau de bord Performance Insights.	Amazon RDS fournit désormais une vue consolidée des CloudWatch statistiques et indicateurs de performance dans le tableau de bord Performance Insights. Pour plus d'informations, consultez la section Affichage des indicateurs combinés avec le tableau de bord Performance Insights.	24 mai 2023
Lectures optimisées pour Amazon RDS disponibles dans les régions de Chine	Lectures optimisées pour Amazon RDS est désormais disponible dans les Régions AWS de Chine (Beijing) et de Chine (Ningxia). Pour plus d'informations, consultez Amélioration des performances des requêtes pour RDS for MariaDB avec Amazon RDS Optimized Reads et Amélioration des performances des requêtes pour RDS for MySQL avec Amazon RDS Optimized Reads.	24 avril 2023
Support Amazon RDS pour les AWS Secrets Manager régions de Chine	Amazon RDS prend en charge Secrets Manager dans les régions de Chine (Beijing) et Chine (Ningxia). Pour plus d'informations, consultez Gestion des mots de passe avec Amazon RDS et AWS Secrets Manager.	20 avril 2023

RDS Custom for Oracle prend en charge la réutilisation de l'AMI IDs pour les nouveaux CEVs	Lorsque vous créez une version de moteur personnalisée (CEV), RDS Custom for Oracle utilise par défaut la Amazon Machine Image (AMI) la plus récente disponible. Vous pouvez désormais spécifier un ID AMI qui a été utilisé dans une CEV précédente. Pour plus d'informations, consultez Création d'une CEV.	19 avril 2023
Amazon RDS prend en charge la publication d'événements avec des balises pour ses abonnés d'une rubrique	Les notifications d'événements Amazon RDS envoyées à Amazon Simple Notification Service (Amazon SNS) ou EventBridge Amazon contiennent désormais des balises d'événement dans le corps du message. Ces balises fournissent des données sur la ressource affectée par l'événement de service. Pour plus d'informations, consultez Amazon RDS event notification tags and attributes (Balises et attributs de notification d'événement Amazon RDS).	17 avril 2023

Acheter des instances réservées pour un cluster de bases de données multi-AZ	Vous pouvez désormais acheter des instances de base de données réservées pour un cluster de bases de données multi-AZ. Pour plus d'informations, consultez Instances de base de données réservées pour un cluster de bases de données multi-AZ.	12 avril 2023
Amazon RDS prend en charge les classes d'instance db.m7g et db.r7g	Vous pouvez désormais utiliser les classes d'instance db.m7g et db.r7g pour les instances de base de données RDS for MySQL, RDS for MariaDB et RDS for PostgreSQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	12 avril 2023
Mettre à jour vers des autorisations du rôle lié à un service pour Amazon RDS Custom	La politique `AmazonRDSCustomServiceRolePolicy` accorde désormais des autorisations supplémentaires pour permettre à RDS Custom for SQL Server d'utiliser Amazon SQS et de créer des instantanés. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	6 avril 2023

Procéder à la migration vers un cluster de bases de données multi-AZ RDS for MySQL à l'aide d'un réplica en lecture	Vous pouvez désormais utiliser un réplica en lecture pour procéder à la migration d'un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ RDS for MySQL vers un déploiement de cluster de bases de données multi-AZ RDS for MySQL avec un temps d'arrêt réduit. Pour plus d'informations, consultez Migration vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture.	6 avril 2023
Créer un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ	Vous pouvez désormais créer un réplica en lecture d'une instance de base de données à partir d'un cluster de bases de données multi-AZ afin de dimensionner au-delà de la capacité de calcul du cluster source. Pour plus d'informations, consultez Création d'un réplica en lecture d'une instance de base de données avec un cluster de bases de données multi-AZ.	6 avril 2023

Amazon RDS Custom for SQL Server prend en charge le déploiement multi-AZ	Vous pouvez créer un déploiement multi-AZ avec RDS Custom for SQL Server. Pour plus d'informations, consultez Managing a Multi-AZ deployment for RDS Custom for SQL Server (Gestion d'un déploiement multi-AZ pour RDS Custom for SQL Server).	6 avril 2023
Mise à jour des autorisations de politique AWS gérées	Les `AmazonRDSReadOnlyAccess` politiques `AmazonRDSFullAccess` et accords accordent désormais des autorisations supplémentaires pour permettre l'affichage des résultats d'Amazon DevOps Guru dans la console RDS. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	30 mars 2023
Amazon RDS prend en charge Oracle APEX version 22.2.v1	Vous pouvez utiliser Oracle APEX 22.2.v1 avec toutes les versions prises en charge d'Oracle Database. Pour plus d'informations, consultez Oracle Application Express.	30 mars 2023

Amazon DevOps Guru disponible pour RDS pour PostgreSQL	RDS pour PostgreSQL vous alerte en cas de récentes anomalies détectées par Amazon Guru. DevOps La page de détails de la base de données de la console vous alerte en cas de situation actuelle et d'anomalies survenues au cours des dernières 24 heures. DevOpsGuru publie des informations proactives contenant des recommand ations pour vous aider à résoudre les problèmes liés à vos bases de données RDS pour PostgreSQL avant qu'ils ne se produisent. Pour plus d'informations, consultez Comment fonctionne DevOps Guru for RDS.	30 mars 2023
RDS Custom prend en charge le volume de stockage Amazon EBS gp3	RDS Custom for Oracle et RDS Custom for SQL Server prennent tous deux en charge les volumes EBS basés sur des SSD io1, gp2 et gp3. Pour plus d'informations, consultez Exigences générales pour RDS Custom for Oracle et Exigences générales pour RDS Custom for SQL Server.	29 mars 2023

Mise à jour des autorisations de politique AWS gérées	Les `AmazonRDSReadOnlyAccess` politiques `AmazonRDSFullAccess` et accords accordent désormais des autorisations supplémentaires à Amazon CloudWatch. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	16 mars 2023
RDS Proxy est disponible dans les régions de Chine	RDS Proxy est désormais disponible dans les régions de Chine (Beijing) et de Chine (Ningxia). Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	15 mars 2023
RDS Proxy est disponible dans la région Asie-Pacifique (Jakarta)	RDS Proxy est désormais disponible dans la région Asie-Pacifique (Jakarta). Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	8 mars 2023

Écritures optimisées pour Amazon RDS améliore les performances des transactions d'écriture pour RDS for MariaDB	Vous pouvez améliorer les performances des transactions d'écriture pour les instances de base de données RDS for MariaDB avec Écritures optimisées pour Amazon RDS. Pour plus d'informations, consultez Improving write performance with Amazon RDS Optimized Writes for MariaDB (Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MariaDB).	7 mars 2023
Amazon RDS pour PostgreSQL versions 15.2	Les nouvelles fonctionnalités d'Amazon RDS pour PostgreSQL 15.2 incluent la commande SQL standard « MERGE » pour les requêtes SQL conditionnelles, des améliorations des performances pour le tri en mémoire et sur disque, et la prise en charge de la validation en deux phases et du filtrage des lignes et des colonnes pour la réplication logique.	27 février 2023
RDS Custom pour Oracle est disponible dans les régions Canada (Centre) et Amérique du Sud (São Paulo)	Pour un tableau répertoriant toutes les régions prises en charge Régions AWS, voir Régions prises en charge et moteurs de base de données pour RDS Custom for Oracle.	22 février 2023

Amazon RDS prend en charge les sauvegardes automatisées entre régions pour RDS pour MariaDB et RDS pour MySQL	Vous pouvez maintenant répliquer des instantanés de base de données et des journaux de transactions entre Régions AWS pour les instances de base de données RDS pour MariaDB et RDS pour MySQL. Pour plus d'informations, consultez Réplication des sauvegardes automatiques dans une autre Région AWS.	22 février 2023
Amazon RDS pour Oracle prend en charge les préavis de mises à niveau automatiques de version mineure	RDS vous informe à l'avance de la date à laquelle une nouvelle version mineure du moteur RDS pour Oracle sera disponible. RDS commence à planifier les mises à niveau automatiques de version mineure de vos instances de base de données RDS pour Oracle à la date de disponibilité. Pour plus d'informations, consultez Avant de planifier la mise à niveau automatique d'une version mineure.	21 février 2023

Amazon RDS pour SQL Server prend en charge les flux d'activité de base de données	Vous pouvez désormais surveiller une instance de base de données SQL Server à l'aide des flux d'activité de base de données. Une instance de base de données SQL Server comporte l'audit de serveur, qui est géré par Amazon RDS. Vous pouvez définir les politiques d'enregistrement des événements de serveur dans la spécification d'audit de serveur. Vous pouvez créer une spécification d'audit de base de données et définir les politiques d'enregistrement des événements de base de données. Le flux d'activité est collecté et transmis à Amazon Kinesis. Dans Kinesis, vous pouvez surveiller le flux d'activité pour une analyse plus approfondie. Pour plus d'informations, consultez Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données.	15 février 2023

RDS prend en charge MySQL 8.0.32 et 5.7.41	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.32 et 5.7.41. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	7 février 2023
Amazon RDS for Oracle prend en charge de nouvelles suites de chiffrement pour SSL	Si vous exécutez Oracle Database 19c ou 21c, vous pouvez spécifier six nouvelles suites de chiffrement dans l'option SSL pour RDS for Oracle. Ces suites prennent en charge FIPS et sont conformes à la norme FedRAMP. Pour plus d'informations, consultez Oracle Secure Sockets Layer.	3 février 2023
Amazon RDS for Oracle prend en charge de nouvelles suites de chiffrement pour Oracle Enterprise Manager	Vous pouvez utiliser quatre nouvelles suites de chiffrement conformes à la norme FedRAMP pour l'option OEM. Pour plus d'informations, consultez Oracle Management Agent pour Enterprise Manager Cloud Control.	3 février 2023
RDS pour Oracle prend en charge les flux d'activité de base de données dans les régions Asie-Pacifique (Hyderabad), Europe (Espagne) et Moyen-Orient (EAU)	Pour plus d'informations, consultez Régions prises en charge et moteurs de base de données pour les flux d'activité des bases de données dans Amazon RDS.	27 janvier 2023

Procéder à la migration vers un cluster de bases de données multi-AZ RDS for PostgreSQL à l'aide d'un réplica en lecture	L'utilisation d'un réplica en lecture vous permet de procéder à la migration d'un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ RDS for PostgreSQL vers un déploiement de cluster de bases de données multi-AZ RDS for PostgreSQL avec un temps d'arrêt réduit. Pour plus d'informations, consultez Migration vers un cluster de bases de données multi-AZ à l'aide d'un réplica en lecture.	23 janvier 2023
Amazon RDS est disponible dans la région Asie-Pacifique (Melbourne)	Amazon RDS est désormais disponible dans la région Asie-Pacifique (Melbourne). Pour plus d'informations, consultez Régions et zones de disponibilité.	23 janvier 2023
RDS pour MariaDB prend en charge le renforcement des connexions SSL/TLS	RDS pour MariaDB prend désormais en charge le SSL/TLS renforcement des connexions en définissant le paramètre sur. `require_secure_transport` `ON` Pour plus d'informations, consultez Exiger toutes SSL/TLS les connexions à une instance de base de données MariaDB.	19 janvier 2023

Amazon RDS Optimized Reads améliore les performances des requêtes pour RDS for MariaDB	Vous pouvez accélérer le traitement des requêtes pour les instances de base de données RDS for MariaDB avec Amazon RDS Optimized Reads. Pour plus d'informations, consultez Improving query performance for RDS for MariaDB with Amazon RDS Optimized Reads (Amélioration des performances des requêtes pour RDS for MariaDB avec Amazon RDS Optimized Reads).	11 janvier 2023
Restaurer un instantané de cluster de bases de données multi-AZ dans une instance de base de données	Vous pouvez désormais restaurer un instantané de cluster de bases de données multi-AZ dans un déploiement mono-AZ ou un déploiement d'instance de base de données multi-AZ. Pour plus d'informations, consultez Restoring from a Multi-AZ DB cluster snapshot to a DB instance (Restauration d'un instantané de cluster de bases de données multi-AZ dans une instance de base de données).	10 janvier 2023

Spécification de l'autorité de certification (CA) lors de création d'une instance de base de données	Vous pouvez désormais spécifier l'autorité de certification à utiliser pour le certificat de serveur d'une instance de base de données lors de la création de l'instance de base de données. Pour plus d'informations, consultez Autorités de certification.	5 janvier 2023
RDS Custom for SQL Server prend en charge les versions de moteur personnalisées	Une version de moteur personnalisée (CEV) pour RDS Custom for SQL Server est une Amazon Machine Image (AMI) avec Microsoft SQL Server préinstallé. Vous choisissez une AMI Amazon EC2 Windows à utiliser comme image de base et vous pouvez installer d'autres logiciels sur le système d'exploitation (OS). Vous pouvez personnaliser la configuration du système d'exploitation et de SQL Server pour répondre aux besoins de votre entreprise. Pour plus d'informations, consultez Working with custom engine versions for RDS Custom for SQL Server (Utilisation des versions de moteur personnalisées pour RDS Custom for SQL Server).	28 décembre 2022

Utilisez les Blue/Green déploiements Amazon RDS disponibles en supplément Régions AWS	La fonctionnalité Blue/Green Déploiements est désormais disponible dans les régions de Chine (Pékin) et de Chine (Ningxia). Pour plus d'informations, consultez Utilisation des Blue/Green déploiements Amazon RDS pour les mises à jour de bases de données.	22 décembre 2022
Mise à jour des autorisations de rôle lié à un service IAM	La RDSService RolePolicy politique d'Amazon accorde désormais des autorisations supplémentaires à AWS Secrets Manager. Pour plus d'informations, consultez Mises à jour Amazon RDS des politiques gérées par AWS.	22 décembre 2022
Amazon RDS prend en charge le renommage d'un cluster de bases de données multi-AZ	Vous pouvez désormais renommer un cluster de bases de données multi-AZ. Pour plus d'informations, consultez Renaming a Multi-AZ DB cluster (Renommage d'un cluster de bases de données multi-AZ).	22 décembre 2022

Amazon RDS s'intègre à la gestion des mots AWS Secrets Manager de passe	Amazon RDS peut gérer le mot de passe d'utilisateur principal pour une instance de base de données ou un cluster de bases de données multi-AZ dans Secrets Manager. Pour plus d'informations, consultez Gestion des mots de passe avec Amazon RDS et AWS Secrets Manager.	22 décembre 2022
Amazon RDS Optimized Writes prend en charge les classes d'instance de base de données db.r6g et db.r6gd	Amazon RDS Optimized Writes prend en charge les classes d'instance de base de données db.r6g et db.r6gd. Pour plus d'informations, consultez Improving write performance with Amazon RDS Optimized Writes (Amélioration des performances d'écriture avec Amazon RDS Optimized Writes).	22 décembre 2022
Amazon RDS Custom pour Oracle prend en charge les nouvelles Régions AWS	Vous pouvez créer des instances de base de données RDS Custom for Oracle dans les régions Asie-Pacifique (Séoul) et Asie-Pacifique (Osaka). Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour RDS Custom for Oracle.	21 décembre 2022

Amazon RDS on AWS Outposts prend en charge les répliques en lecture	Vous pouvez désormais créer un réplica en lecture à partir d'une instance de base de données RDS sur Outposts MySQL ou PostgreSQL. Pour plus d'informations, consultez Création de réplicas en lecture pour Amazon RDS sur AWS Outposts.	19 décembre 2022
RDS Custom for Oracle prend en charge la modification de la classe d'instances de base de données	Vous pouvez désormais modifier la classe d'instanc es de votre instance de base de données RDS Custom for Oracle. Pour plus d'informa tions, consultez Gestion d'une instance de base de données Amazon RDS Custom for Oracle.	16 décembre 2022
RDS for MySQL et RDS for PostgreSQL prennent en charge les classes d'instance de base de données db.x2iedn	Vous pouvez désormais utiliser les classes d'instance de base de données db.x2iedn pour les instances de base de données RDS for MySQL et RDS for PostgreSQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	14 décembre 2022

Amazon RDS Optimized Writes prend en charge les classes d'instance de base de données db.x2iedn	Amazon RDS Optimized Writes prend désormais en charge les classes d'instance de base de données db.x2iedn. Pour plus d'informations, consultez Improving write performance with Amazon RDS Optimized Writes (Amélioration des performances d'écriture avec Amazon RDS Optimized Writes).	14 décembre 2022
Amazon RDS prend en charge la copie de groupes d'options de base de données lors de la copie d'instantanés de base de données	Vous pouvez désormais copier un groupe d'options dans Comptes AWS le cadre d'une demande de copie instantanée sur les bases de données RDS pour Oracle. Pour plus d'informations, consultez Considérations relatives au groupe d'options.	13 décembre 2022
Amazon RDS prend en charge RDS Proxy avec RDS for PostgreSQL version 14	Vous pouvez désormais créer un proxy RDS avec une base de données RDS for PostgreSQL version 14. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	13 décembre 2022

Amazon RDS for Oracle prend en charge les classes d'instance db.x2idn, db.x2iedn et db.x2iezn	Vous pouvez désormais utiliser les classes d'instance db.x2idn, db.x2iedn et db.x2iezn pour les instances de base de données Amazon RDS for Oracle. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données et Classes d'instances RDS for Oracle prises en charge.	12 décembre 2022
Les instances de base de données RDS for PostgreSQL prennent en charge le kit Trusted Language Extensions pour PostgreSQL	Trusted Language Extensions pour PostgreSQL est un kit de développement open source qui vous permet de créer des extensions PostgreSQL à hautes performances et de les exécuter en toute sécurité sur votre instance de base de données RDS for PostgreSQL. Pour plus d'informations, consultez Working with Trusted Language Extensions for PostgreSQL (Utilisation de Trusted Language Extensions pour PostgreSQL).	30 novembre 2022

Utiliser les Blue/Green déploiements Amazon RDS pour les mises à jour des bases de données	Vous pouvez apporter des modifications à une instance de base de données dans un environnement intermédiaire et tester les modifications sans affecter votre instance de base de données de production. Lorsque vous êtes prêt, vous pouvez promouvoir l'environnement intermédiaire comme nouvel environnement de production, avec un temps d'arrêt minimal. Pour plus d'informations, consultez Utilisation des Blue/Green déploiements Amazon RDS pour les mises à jour de bases de données.	27 novembre 2022
Amazon RDS Optimized Writes améliore les performances des transactions d'écriture pour RDS for MySQL	Vous pouvez améliorer les performances des transactions d'écriture pour les instances de base de données RDS for MySQL avec Amazon RDS Optimized Writes. Pour plus d'informations, consultez Improving write performance with Amazon RDS Optimized Writes for MySQL (Amélioration des performances d'écriture avec Écritures optimisées pour Amazon RDS for MySQL).	27 novembre 2022

Amazon RDS Optimized Reads améliore les performances des requêtes pour RDS for MySQL	Vous pouvez accélérer le traitement des requêtes pour les instances de base de données RDS for MySQL avec Amazon RDS Optimized Reads. Pour plus d'informations, consultez Improving query performance with Amazon RDS Optimized Reads (Amélioration des performances des requêtes avec Amazon RDS Optimized Reads).	27 novembre 2022
Amazon RDS est disponible dans la région Asie-Pacifique (Hyderabad)	Amazon RDS est désormais disponible dans la région Asie-Pacifique (Hyderabad) Pour plus d'informations, consultez Régions et zones de disponibilité.	22 novembre 2022
RDS prend en charge MariaDB 10.6.11, 10.5.18, 10.4.27 et 10.3.37	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions 10.6.11, 10.5.18, 10.4.27 et 10.3.37. de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	18 novembre 2022

RDS Custom for Oracle prend en charge la définition de paramètres d'installation autres que ceux par défaut dans une version de moteur personnalisée (CEV)	Lorsque vous créez une version CEV, vous pouvez définir des valeurs autres que celles par défaut pour la base Oracle, le répertoire de base de base de données Oracle, l'ID et le nom d'utilisateur UNIX, ainsi que l'ID et le nom de groupe UNIX. De cette façon, vous avez plus de contrôle sur l'installation des bases de données sur votre instance de base de données RDS Custom for Oracle. Pour plus d'informations, consultez Préparation du manifeste CEV.	18 novembre 2022
Amazon RDS prend en charge Oracle APEX version 21.1.v1	Vous pouvez utiliser Oracle APEX 22.1.v1 avec toutes les versions prises en charge d'Oracle Database. Pour plus d'informations, consultez Oracle Application Express.	18 novembre 2022

RDS for SQL Server prend en charge les réplicas en lecture entre régions	Vous pouvez désormais créer un réplica en lecture entre régions pour améliorer les capacités de reprise après sinistre, réduire la latence de lecture des applications et décharger les charges de travail de lecture de l'instance de base de données principal e. Pour plus d'informations, consultez la section Création d'un réplica lu dans un autre Région AWS.	16 novembre 2022
Amazon RDS est disponibl e dans la région Europe (Espagne)	Amazon RDS est désormais disponible dans la région Europe (Espagne). Pour plus d'informations, consultez Régions et zones de disponibi lité.	16 novembre 2022
RDS for SQL Server prend en charge les serveurs liés pour la base de données Oracle	Vous pouvez désormais créer un serveur lié pour accéder à des bases de données Oracle externes afin de lire des données et d'exécuter des commandes SQL. Pour plus d'informations, consultez Linked Servers with Oracle OLEDB with RDS for SQL Server (Serveurs liés avec Oracle OLEDB avec RDS for SQL Server).	15 novembre 2022

RDS Custom for Oracle prend en charge Oracle Multitenant	Vous pouvez créer une instance de base de données RDS Custom for Oracle en tant que base de données de conteneur (CDB). Après sa création, la CDB contient la racine CDB, le conteneur initial de PDB et une PDB. Vous pouvez en ajouter d'autres PDBs manuellement à l'aide d'Oracle SQL. Pour plus d'informations, consultez Présentation de l'architecture Amazon RDS Custom for Oracle.	15 novembre 2022
Amazon RDS for Oracle prend en charge l'intégration Amazon EFS	Si vous ajoutez l'option `EFS_INTEGRATION` dans votre groupe d'options, vous pouvez transférer des fichiers entre votre instance de base de données RDS for Oracle et un système de fichiers Amazon EFS. Pour plus d'informations, consultez Amazon EFS.	15 novembre 2022
RDS prend en charge MySQL 8.0.31 et 5.7.40	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.31 et 5.7.40. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	10 novembre 2022

Amazon RDS est disponible dans la région Europe (Zurich)	Amazon RDS est désormais disponible dans la région Europe (Zurich). Pour plus d'informations, consultez Régions et zones de disponibilité.	9 novembre 2022
L'accès aux sauvegardes des journaux de transactions est désormais disponible pour RDS for SQL Server	Vous pouvez désormais afficher et copier les sauvegardes des journaux de transactions de base de données vers un compartiment Amazon S3. Pour plus d'informations, consultez Accès aux sauvegardes des journaux de transactions.	7 novembre 2022
Clusters de bases de données multi-AZ pris en charge en supplément Régions AWS	Les clusters de base de données multi-AZ sont désormais disponibles en supplément Régions AWS. Pour plus d'informations, consultez Régions et moteurs de base de données pris en charge pour les clusters de base de données multi-AZ dans Amazon RDS.	4 novembre 2022

Amazon RDS prend en charge le stockage gp3	Vous pouvez désormais créer des instances de base de données Amazon RDS qui utilisent des volumes de stockage SSD à usage général (gp3) Amazon EBS, ce qui vous permet de personnaliser les performances de stockage indépendamment de la capacité de stockage. Pour plus d'informations, consultez Stockage SSD à usage général.	4 novembre 2022
Amazon RDS prend en charge un nouvel événement pour les mises à jour du système d'exploitation	Amazon RDS prend désormais en charge un nouvel événement d'instance de base de données, RDS-EVENT-0230, dans la catégorie d'événement Application de correctifs de sécurité. Ce nouvel événement vous avertit lorsqu'une mise à jour du système d'exploitation est disponible pour votre instance de base de données. Pour plus d'informations, consultez Surveillance des événements Amazon RDS et Utilisation des mises à jour du système d'exploitation.	28 octobre 2022

Amazon RDS for Oracle prend en charge les classes d'instance à mémoire optimisée r5b préconfigurées	Les classes d'instance de base de données Oracle db.r5b sont optimisées pour les charges de travail qui nécessitent de la mémoire, du stockage et par vCPU supplémentaires. I/O Par exemple, le multithreading est activé dans db.r5b.4x large.tpc2.mem2x et fournit deux fois plus de mémoire que db.r5b.4xlarge. Pour plus d'informations, consultez Classes d'instances RDS pour Oracle.	27 octobre 2022
Amazon RDS prend en charge 15 réplicas en lecture pour RDS pour les instances de base de données MariaDB, MySQL et PostgreSQL	Vous pouvez désormais créer jusqu'à 15 réplicas en lecture pour RDS pour les instances de base de données MariaDB, MySQL et PostgreSQL. Pour plus d'informations, consultez Utilisation des réplicas en lecture.	20 octobre 2022
Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 15 RC 3 dans l'environnement de prévisualisation de base de données	PostgreSQL version 15 Beta 3 est désormais disponible dans l'environnement de prévisualisation de base de données dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez Utilisation de l'environnement de prévisualisation de base de données.	18 octobre 2022

Amazon RDS prend en charge la configuration automatique de la connectivité entre une base de données RDS et une instance EC2	Vous pouvez utiliser le AWS Management Console pour configurer la connectivité entre une instance de base de données RDS existante ou un cluster de base de données multi-AZ et une EC2 instance. Pour plus d'informations, consultez Connexion automatique d'une EC2 instance à une base de données RDS.	14 octobre 2022
AWS Le pilote JDBC pour PostgreSQL est généralement disponible	Le pilote AWS JDBC pour PostgreSQL est un pilote client conçu pour RDS pour PostgreSQL. Le pilote JDBC AWS pour PostgreSQL est désormais généralement disponible. Pour plus d'informations, consultez Connexion avec le pilote AWS JDBC pour PostgreSQL.	6 octobre 2022
Amazon RDS for Oracle prend en charge Oracle APEX version 21.2.v1	Oracle APEX 21.2 inclut le correctif 33420059. Pour plus d'informations, consultez les exigences relatives à la version d'Oracle APEX.	3 octobre 2022

RDS prend en charge MySQL 5.7.39	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.39. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	29 septembre 2022
RDS prend en charge MariaDB version 10.6.10	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.6.10. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	29 septembre 2022
RDS Proxy prend en charge RDS for SQL Server	Vous pouvez désormais créer un RDS Proxy pour une instance de base de données RDS qui exécute Microsoft SQL Server version 2014 ou ultérieure. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	19 septembre 2022
RDS prend en charge MariaDB versions 10.5.17, 10.4.26 et 10.3.36	Vous pouvez désormais créer des instances de bases de données Amazon RDS en exécutant MariaDB versions 10.5.17, 10.4.26 et 10.3.36. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	15 septembre 2022

Amazon RDS for Oracle prend en charge le stockage local d'instance pour les données temporaires.	Vous pouvez désormais lancer Amazon RDS for Oracle sur les types d'instances EC2 Amazon db.r5d et db.m5d avec le tablespace temporaire et le Database Smart Flash Cache (le cache flash) configurés pour utiliser un magasin d'instances. En stockant des fichiers de données temporaires localement, vous pouvez bénéficier de latences réduites de lecture et d'écriture par rapport aux stockages standard basés sur Amazon EBS. Pour plus d'informations, consultez Stockage de données Oracle temporaires dans le stockage d'instances.	14 septembre 2022
Performance Insights affiche les 25 principales requêtes SQL	Dans le tableau de bord Performance Insights, l'onglet SQL maximum présente les 25 requêtes SQL qui contribuent le plus à la charge de la base de données. Pour plus d'informations, consultez Présentation de l'onglet SQL maximum.	13 septembre 2022

RDS prend en charge MySQL 8.0.30	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.30. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	9 septembre 2022
Amazon RDS est disponible dans la région Moyen-Orient (EAU)	Amazon RDS est désormais disponible dans la région du Moyen-Orient (UAE). Pour plus d'informations, consultez Régions et zones de disponibilité.	30 août 2022
Amazon RDS for SQL Server prend en charge les abonnements SSRS Email	Vous pouvez désormais utiliser l'extension Email de SQL Server Reporting Services (SSRS) pour envoyer des rapports aux utilisateurs et vous abonner à des rapports sur le serveur de rapports. Pour de plus amples informations, consultez Support for SQL Server Reporting Services in RDS for SQL Server (Prise en charge de SQL Server Reporting Services dans RDS for SQL Server).	17 août 2022

RDS for Oracle prend en charge les sauvegardes de réplicas en lecture	Vous pouvez activer les sauvegardes automatiques et créer des instantanés manuels des réplicas de RDS for Oracle. Pour obtenir plus d'informations, consultez la rubrique Working with RDS for Oracle replica backups (Utilisation de RDS pour les sauvegardes de réplicas Oracle).	23 août 2022
RDS for Oracle prend en charge la commutation d'Oracle Data Guard	Une commutation est une inversion des rôles entre une base de données principale et un réplica Oracle monté ou ouvert. Lors d'une commutation, la base de données principale d'origine passe à un rôle secondaire, tandis que la base de données secondaire d'origine passe au rôle principal. Pour obtenir plus d'informations, consultez la section Performing an Oracle Data Guard switchover (Exécution d'une commutation d'Oracle Data Guard).	23 août 2022

Amazon RDS prend en charge la configuration automatique de la connectivité avec une instance EC2	Lorsque vous créez une instance de base de données ou un cluster de base de données multi-AZ, vous pouvez utiliser le AWS Management Console pour configurer la connectivité entre une instance Amazon Elastic Compute Cloud et la nouvelle instance de base de données ou le nouveau cluster de base de données. Pour plus d'informations, consultez Configurer la connectivité réseau automatique avec une EC2 instance pour une nouvelle instance de base de données et Configurer la connectivité réseau automatique avec une EC2 instance pour un nouveau cluster de base de données.	22 août 2022

RDS Custom for Oracle prend en charge la promotion des réplicas d'Oracle	Si vous utilisez RDS Custom for Oracle, vous pouvez promouvoir vos réplicas Oracle gérés à l'aide de la commande CLI `promote-read-replica` . Vous pouvez également supprimer votre instance de base de données principale, ce qui amène RDS Custom for Oracle à promouvoir vos réplicas Oracle gérés en instances autonomes. Pour obtenir plus d'informations, consultez la rubrique Working with Oracle replicas for RDS Custom for Oracle (Utilisation des réplicas Oracle pour RDS Custom for Oracle).	5 août 2022
RDS for MySQL prend en charge le renforcement des connexions SSL/TLS	RDS for MySQL prend désormais en charge le renforcement des SSL/TLS connexions en définissant le `require_secure_transport` paramètre sur. `ON` Pour plus d'informations, consultez Exiger une SSL/TLS connexion à une instance de base de données MySQL.	1er août 2022

Amazon RDS a rendu obsolète la prise en charge d'Oracle Database 12c Version 1 (12.1.0.2).	La prise en charge de la version 12.1.0.2 est rendue obsolète pour les modèles de licence BYOL et LI. Le 1er août 2022, RDS for Oracle commence les mises à niveau automatiques des instances de base de données 12c Version 1 (12.1.0.2) et des instantanés 12.1.0.2 restaurés vers Oracle Database 19c. Pour plus d'informations, consultez la chronologie de fin de prise en charge sur AWS re:Post.	1er août 2022
RDS Proxy prend en charge RDS for MariaDB	Vous pouvez maintenant créer un proxy RDS pour une instance de base de données RDS qui exécute MariaDB version 10.2, 10.3, 10.4, ou 10.5. La prise en charge de MariaDB est incluse dans la famille des moteurs MySQL. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	26 juillet 2022

RDS for MariaDB prend en charge les classes d'instance de base de données db.r5b	Vous pouvez maintenant créer des instances de base de données RDS for MariaDB qui utilisent les classes d'instanc e de base de données db.r5b. Pour plus d'informations, consultez Supported DB engines for DB instance classes (Moteurs de base de données pris en charge pour les classes d'instances de base de données).	25 juillet 2022
RDS for Oracle prend en charge la modification des flux d'activité des bases de données	Si vous utilisez RDS for Oracle, vous pouvez modifier l'état de la politique d'audit d'un flux d'activité de base de données en verrouillant (par défaut) ou en déverroui llant. Au lieu d'arrêter un flux d'activité, vous pouvez déverrouiller son état de politique, personnaliser votre politique d'audit, puis verrouill er à nouveau l'état de politique . Pour obtenir plus d'informa tions, consultez la rubrique Modifying a database activity stream (Modification d'un flux d'activité de base de données).	22 juillet 2022

Performance Insights prend en charge la région Asie-Pacifique (Jakarta)	Auparavant, vous ne pouviez pas utiliser Performance Insights dans la région Asie-Pacifique (Jakarta). Cette restriction a été supprimée . Pour plus d'informations, consultez Régions prises en charge et moteurs de base de données pour Performance Insights dans Amazon RDS.	21 juillet 2022
Microsoft SQL Server 2012 a atteint la fin de sa prise en charge sur Amazon RDS	Microsoft SQL Server 2012 a atteint la fin de sa prise en charge, ce qui coïncide avec le plan de Microsoft de mettre fin à la prise en charge étendue pour cette version le 12 juillet 2022. Toutes les instances existantes de Microsoft SQL Server 2012 doivent être automatiquement mises à niveau vers la dernière version mineure de Microsoft SQL Server 2014 à compter du 1er juin 2022. Pour de plus amples informations, consultez Microsoft SQL Server 2012 support on Amazon RDS (Prise en charge de Microsoft SQL Server 2012 sur Amazon RDS).	12 juillet 2022

RDS prend en charge MariaDB 10.6.8, 10.5.16, 10.4.25, 10.3.35 et 10.2.44.	Vous pouvez désormais créer des instances de base de données Amazon RDS en exécutant les versions 10.6.8, 10.5.16, 10.4.25, 10.3.35 et 10.2.44. Pour obtenir plus d'informations, consultez Supported MariaDB versions on Amazon RDS (Versions de MariaDB prises en charge sur Amazon RDS).	8 juillet 2022
RDS Performance Insights prend en charge des périodes de conservation supplémentaires	Auparavant, Performance Insights ne proposait que deux périodes de conservation : 7 jours (par défaut) ou 2 ans (731 jours). Désormais, si vous avez besoin de conserver vos données de performance pendant plus de 7 jours, vous pouvez spécifier de 1 à 24 mois. Pour obtenir plus d'informations, consultez la section Pricing and data retention for Performance Insights (Tarification et conservation des données pour Performance Insights).	1et juillet 2022

RDS Custom prend en charge les régions Asie-Pacifique (Mumbai) et Europe (Londres)	Vous pouvez créer des instances de base de données RDS Custom pour Oracle et RDS Custom pour SQL Server dans deux nouvelles instances Régions AWS : Asie-Pacifique (Mumbai) et Europe (Londres) . Pour plus d'informations, consultez Prise en charge de Région AWS pour RDS Custom for Oracle et Prise en charge de Région AWS pour RDS Custom for SQL Server.	21 juin 2022
RDS Custom for Oracle prend en charge Oracle Database 18c et 12c Release 2 (12.2)	Vous pouvez désormais créer une CEV pour RDS Custom for Oracle à l'aide des fichiers d'installation d'Oracle Database 18c et 12c Release 2 (12.2). Vous pouvez les utiliser CEVs pour créer une instance de base de données personnal isée RDS pour Oracle. Pour plus d'informations, veuillez consulter la section Utilisation des versions personnalisées du moteur pour Amazon RDS Custom for Oracle.	21 juin 2022

Les clusters de bases de données Multi-AZ prennent en charge les classes d'instance de base de données db.m5d et db.r5d	Vous pouvez désormais créer des clusters de bases de données Multi-AZ qui utilisent les classes d'instance de base de données db.m5d et db.r5d. Pour plus d'informations, consultez Déploiements de clusters de base de données Multi-AZ et Types de classes d'instance de base de données.	21 juin 2022
Clusters de bases de données multi-AZ disponibles en supplément Régions AWS	Vous pouvez désormais créer des clusters de bases de données Multi-AZ dans les régions suivantes : Europe (Francfort) et Europe (Stockholm). Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données Multi-AZ.	21 juin 2022
RDS for Microsoft SQL Server prend en charge la migration des bases de données qui utilisent Transparent Data Encryption (TDE)	RDS for SQL Server prend désormais en charge la migration des bases de données Microsoft SQL Server avec TDE activé, à l'aide de la sauvegarde et de la restauration natives. Pour plus d'informations, consultez Prise en charge de Transparent Data Encryption dans SQL Server.	14 juin 2022

Amazon RDS prend en charge la publication d'événements dans des rubriques Amazon SNS chiffrées	Amazon RDS peut désormais publier des événements dans des rubriques Amazon Simple Notification Service (Amazon SNS) où le chiffrement côté serveur (SSE) est activé, afin de renforcer la protection des événements contenant des données sensibles. Pour plus d'informations, consultez Abonnement à la notification d'évènement Amazon RDS.	1 juin 2022
RDS prend en charge MySQL 5.7.38	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.38. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	31 mai 2022
RDS for PostgreSQL prend en charge les réplicas en lecture en cascade	Vous pouvez désormais utiliser des réplicas en lecture en cascade avec RDS for PostgreSQL version 14.1 et versions ultérieures. Pour plus d'informations, consultez Utilisation de réplicas en lecture PostgreSQL dans Amazon RDS.	4 mai 2022

Amazon RDS on AWS Outposts prend en charge les opérations de mise à l'échelle, de stockage et de mise à l'échelle automatique.	Vous pouvez désormais modifier la taille de stockage des instances de bases de données sur votre Outpost et utiliser la mise à l'échelle automatique du stockage. Pour plus d'informations, consultez Prise en charge d'Amazon RDS sur AWS Outposts pour les fonctions Amazon RDS.	2 mai 2022
Clusters de bases de données multi-AZ disponibles en supplément Régions AWS	Vous pouvez désormais créer des clusters de bases de données Multi-AZ dans les régions suivantes : Asie-Pacifique (Singapour) et Asie-Pacifique (Sydney). Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données Multi-AZ.	29 avril 2022
Amazon RDS prend en charge le mode double pile	Les instances de bases de données peuvent désormais fonctionner en mode double pile. En mode double pile, les ressources peuvent communiquer avec l'instance de base de données IPv4 via ou IPv6 les deux. Pour plus d'informations, consultez la section Amazon RDS IP addressing (Adressage IP Amazon RDS).	29 avril 2022

Amazon RDS publie des statistiques d'utilisation sur Amazon CloudWatch	L'espace de `AWS/Usage` noms d'Amazon CloudWatch inclut les mesures d'utilisation au niveau du compte pour vos quotas de service Amazon RDS. Pour plus d'informations, consultez les statistiques CloudWatch d'utilisation d'Amazon pour Amazon RDS.	28 avril 2022
Amazon RDS for MySQL prend en charge les classes d'instance de base de données db.m6i et db.r6i	Vous pouvez désormais utiliser les classes d'instance de base de données db.m6i et db.r6i pour les instances de base de données Amazon RDS exécutant MySQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	28 avril 2022
Amazon RDS for PostgreSQL prend en charge les classes d'instance de base de données db.m6i et db.r6i	Vous pouvez désormais utiliser les classes d'instance de base de données db.m6i et db.r6i pour les instances de base de données Amazon RDS exécutant PostgreSQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	27 avril 2022

Amazon RDS for MariaDB prend en charge les classes d'instance de base de données db.m6i et db.r6i	Vous pouvez désormais utiliser les classes d'instance de base de données db.m6i et db.r6i pour les instances de base de données Amazon RDS exécutant MariaDB. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	26 avril 2022
Amazon RDS on AWS Outposts prend en charge les déploiements multi-AZ	Vous pouvez désormais créer une instance de base de données de secours sur un Outpost différent. Pour plus d'informations, consultez Amazon RDS sur la AWS Outposts prise en charge des fonctionnalités Amazon RDS.	19 avril 2022
Amazon RDS for Oracle prend en charge les classes d'instance db.m6i et db.r6i.	Si vous exécutez Oracle Database 19c, vous pouvez utiliser les classes d'instance db.m6i et db.r6i. Les classes db.m6i sont des classes d'instance polyvalentes adaptées à un large éventail de charges de travail. Pour plus d'informations, consultez Classes d'instances RDS pour Oracle.	8 avril 2022

Amazon RDS for SQL Server prend en charge la réplication des tâches de l'agent SQL Server	Lorsque vous activez cette fonction, les tâches de l'agent SQL Server créées, modifiées ou supprimées sur l'hôte principal sont automatiquement synchronisées sur l'hôte secondaire dans une configuration Multi-AZ. Pour obtenir plus d'informations, consultez la section Using SQL Server Agent (Utilisation de l'agent SQL Server).	7 avril 2022
Amazon RDS prend en charge RDS Proxy avec RDS for PostgreSQL version 13	Vous pouvez désormais créer un proxy RDS avec une base de données RDS for PostgreSQL version 13. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	4 avril 2022
Amazon RDS prévoit de rendre obsolète Oracle Database version 12c	Oracle Database version 12c sera bientôt obsolète. Oracle Corporation ne fournira plus de correctifs pour les versions 12c d'Oracle Database après ces end-of-support dates. Amazon RDS prévoit de commencer à mettre automatiquement à niveau les instances de bases de données Oracle Database version 12c vers Oracle Database version 19c.	22 mars 2022

Notes de mise à jour de Amazon RDS for PostgreSQL	Il existe désormais un guide séparé pour les notes de mise à jour de Amazon RDS for PostgreSQL. Pour plus d'informations, consultez la section Amazon RDS for PostgreSQL Release Notes (Notes de mise à jour de Amazon RDS for PostgreSQL).	22 mars 2022
Notes de mise à jour de Amazon RDS for Oracle	Il existe désormais un guide distinct pour les notes de mise à jour de Amazon RDS for Oracle. Pour plus d'informations, consultez la section Amazon RDS for Oracle Release Notes (Notes de mise à jour de Amazon RDS for Oracle).	22 mars 2022
Clusters de bases de données multi-AZ disponibles en supplément Régions AWS	Vous pouvez désormais créer des clusters de bases de données Multi-AZ dans les régions suivantes : USA Est (Ohio) et Asie-Pacifique (Tokyo). Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données Multi-AZ.	15 mars 2022

Amazon RDS for PostgreSQL versions 14.2, 13.6, 12.10, 11.15 et 10.20	RDS for PostgreSQL prend désormais en charge les versions 14.2, 13.6, 12.10, 11.15 et 10.20. Les versions 14.2 et 13.6 ajoutent la prise en charge de deux nouveaux encapsuleurs de données étrangers. L'extension `mysql_fdw` permet à PostgreSQL de travailler avec des données stockées dans les bases de données MySQL, MariaDB et Aurora MySQL. L'extension `tds_fdw` permet à PostgreSQL de travailler avec des données stockées dans les bases de données SQL Server. Pour de plus amples informations, veuillez consulter Versions de bases de données PostgreSQL prises en charge.	12 mars 2022
RDS prend en charge MySQL 5.7.37	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.37. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	11 mars 2022

Amazon RDS for SQL Server prend en charge de nouvelles classes d'instance de base de données	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant Microsoft SQL Server qui utilisent les classes d'instance de base de données db.m6i et db.r6i. Pour de plus amples informations, veuillez consulter Prise en charge des classes d'instances de bases de données pour Microsoft SQL Server.	9 mars 2022
Amazon RDS for Oracle prend en charge Oracle Database 21c	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle Database 21c (21.0.0.0). Il s'agit de la première version de Oracle Database qui ne prend en charge que l'architecture multilocation (CDB). Pour plus d'informations, consultez Oracle Database 21c avec Amazon RDS.	7 mars 2022
RDS prend en charge MariaDB 10.6.7, 10.5.15, 10.4.24, 10.3.34 et 10.2.43	Vous pouvez désormais créer des instances de base de données Amazon RDS en exécutant les versions 10.6.7, 10.5.15, 10.4.24, 10.3.34 et 10.2.43 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	3 mars 2022

AWS Le pilote JDBC pour MySQL est généralement disponible	Le pilote AWS JDBC pour MySQL est un pilote client conçu pour RDS for MySQL. Le pilote AWS JDBC pour MySQL est désormais disponible pour tous. Pour en savoir plus, veuillez consulter la section Connexion avec le pilote JDBC pour MySQL d'Amazon Web Services.	2 mars 2022
Clusters de bases de données Multi-AZ généralement disponibles	Un déploiement de cluster de base de données multi-AZ est un mode de déploiement à haute disponibilité d'Amazon RDS qui compte deux instances de base de données de secours accessibles en lecture. Les clusters de bases de données Multi-AZ sont désormais généralement disponibles. Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données Multi-AZ.	1er mars 2022
RDS prend en charge MySQL 8.0.28	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.28. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	28 février 2022

Amazon RDS for Oracle prend en charge de nouveaux paramètres pour le chiffrement réseau natif (NNE)	Pour contrôler si les clients peuvent se connecter à l'aide de méthodes de chiffrement et de total de contrôle non sécurisées, définissez `SQLNET.ALLOW_WEAK_CRYPTO_CLIENTS` et `SQLNET.ALLOW_WEAK_CRYPTO` dans l'option NNE. Des exemples de méthodes non sécurisées incluent LE DES, le 3DES et RC4. MD5 Pour plus d'informations, voir Paramètres de l'option NNE.	25 février 2022
Amazon RDS for SQL Server prend en charge les groupes de disponibilité Always On (toujours actifs) pour Microsoft SQL Server 2017 Standard Edition	Lorsque vous créez une instance de base de données en utilisant la configuration Multi-AZ sur SQL Server 2017 Standard Edition 14.00.3401.7 et les versions ultérieures, RDS utilise automatiquement les groupes de disponibilité. Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Microsoft SQL Server.	18 février 2022
Amazon Aurora prend en charge Database Activity Streams (Flux d'activités de base de données) dans la Région Asie-Pacifique (Jakarta)	Pour plus d'informations, consultez la section Support Régions AWS pour les flux d'activité des bases de données.	16 février 2022

Prise en charge d'Amazon RDS Custom for Oracle pour Oracle Database 12.1	Vous pouvez désormais créer des versions de moteur personnalisées pour RDS Custom for Oracle qui utilisent Oracle Database 12.1 Enterprise Edition. Pour plus d'informations, veuillez consulter la section Utilisation des versions personnalisées du moteur pour Amazon RDS Custom for Oracle.	4 février 2022
Amazon RDS for MariaDB prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.6. Pour plus d'informations, veuillez consulter la section Prise en charge de MariaDB 10.6 sur Amazon RDS.	3 février 2022

Performance Insights prend en charge la capture de plans pour les requêtes Oracle	La console Performance Insights prend en charge une nouvelle dimension de plan pour les principaux éléments SQL. Lorsque vous effectuez une tranche par plan, vous pouvez voir quels sont les plans utilisés par vos principal es requêtes Oracle. Si une requête utilise plusieurs plans, vous pouvez comparer les plans côte à côte dans la console et déterminer quel plan est le plus efficace. Vous pouvez également effectuer une réduction du niveau pour voir quelles étapes d'un plan présentent le coût le plus élevé. Pour de plus amples informations, veuillez consulter Analyse des plans d'exécution d'Oracle à l'aide du tableau de bord de Performance Insights.	27 janvier 2022

Performance Insights soutient de nouvelles APIs	Performance Insights prend en charge les éléments suivants APIs : `GetResourceMetadata` `ListAvailableResourceDimensions` , et`ListAvailableResourceMetrics` . Pour plus d'informations, veuillez consulter la section Récupération de métriques avec l'API Performance Insights de ce manuel ainsi que la Référence d'API Amazon RDS Performance Insights.	12 janvier 2022
RDS Proxy prend en charge les événements	Le proxy RDS génère désormais des événements auxquels vous pouvez vous abonner et consulter dans CloudWatch Events ou configurer pour les envoyer à Amazon EventBridge. Pour en savoir plus, veuillez consulter la section Utilisation des événements RDS Proxy.	11 janvier 2022
Amazon RDS for SQL Server prend en charge le mode SSAS multidimensionnel	RDS for SQL Server prend en charge l'exécution de SQL Server Analysis Services (SSAS) en mode tabulaire ou multidimensionnel. Pour plus d'informations, veuillez Prise en charge de SQL Server Analysis Services dans SQL Server.	7 janvier 2022

Proxy RDS disponible en supplément Régions AWS	RDS Proxy est désormais disponible dans les Régions suivantes : Afrique (Le Cap), Asie-Pacifique (Hong Kong), Asie-Pacifique (Osaka), Europe (Milan), Europe (Paris), Europe (Stockholm), Moyen-Orient (Bahreïn) et Amérique du Sud (Sao Paulo). Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	5 janvier 2022
RDS prend en charge MySQL 8.0.27	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.27. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	21 décembre 2021
Amazon RDS est disponible dans la Région Asie-Pacifique (Jakarta)	Amazon RDS est désormais disponible dans la Région Asie-Pacifique (Jakarta). Pour plus d’informations, consultez Régions et zones de disponibilité.	13 décembre 2021

Amazon RDS prend en charge MariaDB 10.5.13, 10.4.22, 10.3.32 et 10.2.41	Vous pouvez désormais créer des instances de base de données Amazon RDS en exécutant les versions 10.5.13, 10.4.22, 10.3.32 et 10.2.41 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	8 décembre 2021
Amazon RDS Custom for SQL Server	Amazon RDS Custom est un service de base de données géré destiné aux applications héritées, personnalisées et empaqueté es nécessitant un accès à l'environnement de base de données et au système d'exploitation sous-jacents. Avec Amazon RDS Custom, vous bénéficiez de l'automat isation d'Amazon RDS et de la flexibilité d'Amazon. EC2 Pour plus d'informations, consultez Utilisation d'Amazon RDS Custom.	1 décembre 2021

Clusters de bases de données multi-AZ (version préliminaire)	Vous pouvez désormais créer des clusters de bases de données multi-AZ pour RDS for MySQL et RDS for PostgreSQL. Un déploiement de cluster de base de données multi-AZ est un mode de déploiement à haute disponibilité d'Amazon RDS qui compte deux instances de base de données de secours accessibles en lecture. Les clusters de bases de données multi-AZ sont disponibles en version préliminaire. Pour de plus amples informations, veuillez consulter Déploiements de clusters de bases de données multi-AZ (version de prévisualisation).	23 novembre 2021
Amazon RDS prend en charge RDS Proxy avec RDS for PostgreSQL version 12	Vous pouvez désormais créer un proxy RDS avec une base de données RDS for PostgreSQL version 12. Pour plus d'informations sur RDS Proxy, consultez Utilisation d'Amazon RDS Proxy.	22 novembre 2021

Amazon RDS on AWS Outposts prend en charge les sauvegardes locales	Vous pouvez stocker des sauvegardes automatisées et des instantanés manuels dans votre Outpost Région AWS ou localement sur votre Outpost. Pour plus d'informations, consultez Amazon RDS sur la AWS Outposts prise en charge des fonctionnalités Amazon RDS.	22 novembre 2021
Support d'Amazon RDS pour les comptes multiples AWS KMS keys	Vous pouvez utiliser une clé KMS d'un autre AWS compte pour le chiffrement lorsque vous exportez des instantanés de base de données vers Amazon S3. Pour plus d'informations, veuillez consulter Exportation de données d'instantanés de bases de données vers Amazon S3.	3 novembre 2021
Amazon RDS on AWS Outposts prend en charge la publication des journaux du moteur de base de données dans Logs CloudWatch	RDS on Outposts prend désormais en charge la publication des journaux du moteur de base de données dans Logs CloudWatch . Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	2 novembre 2021

Amazon RDS Custom for Oracle	Amazon RDS Custom est un service de base de données géré destiné aux applications héritées, personnalisées et empaquetées nécessitant un accès à l'environnement de base de données et au système d'exploitation sous-jacents. Avec Amazon RDS Custom, vous bénéficiez de l'automatisation d'Amazon RDS et de la flexibilité d'Amazon. EC2 Pour plus d'informations, consultez Utilisation d'Amazon RDS Custom.	26 octobre 2021
Prise en charge de la réplication retardée pour RDS for MySQL version 8.0	À partir de RDS for MySQL version 8.0.26, vous pouvez configurer la réplication retardée pour les instances de base de données RDS for MySQL version 8.0. Pour plus d'informations, consultez Configuration de la réplication retardée avec MySQL.	25 octobre 2021
Prise en charge de MySQL 8.0.26	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.26. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	25 octobre 2021

Prise en charge de la réplication basée sur GTID pour RDS for MySQL version 8.0	À partir de RDS for MySQL version 8.0.26, vous pouvez configurer la réplication basée sur GTID pour les instances de base de données RDS for MySQL version 8.0. Pour plus d'informations, consultez Utilisation de la réplication GTID pour RDS for MySQL.	25 octobre 2021
Amazon RDS prend en charge RDS Proxy avec RDS for MySQL 8.0	Vous pouvez désormais créer un proxy RDS pour une instance de base de données RDS for MySQL 8.0. Pour plus d'informations, consultez la section Utilisation d'Amazon RDS Proxy.	21 octobre 2021
Amazon RDS on AWS Outposts prend en charge des versions supplémentaires de RDS pour MySQL	RDS sur Outposts prend désormais en charge les versions 8.0.23 et 8.0.25 de RDS for MySQL. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	20 octobre 2021
Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 14 RC 1 dans l'environnement de prévisualisation de base de données	PostgreSQL version 14 RC 1 est désormais disponible dans l'environnement de prévisualisation de base de données dans l'est des États-Unis (Ohio). Région AWS Pour plus d'informations, consultez Utilisation de l'environnement de prévisualisation de base de données.	19 octobre 2021

Amazon RDS prend également en charge Performance Insights Régions AWS	Performance Insights est disponible dans les régions suivantes : Moyen-Orient (Bahreïn), Afrique (Le Cap), Europe (Milan) et Asie-Paci fique (Osaka). Pour plus d'informations, consultez Régions prises en charge et moteurs de base de données pour Performance Insights dans Amazon RDS.	5 octobre 2021
Performance Insights prend en charge les statistiques au niveau récapitulatif pour Oracle	Lorsque vous utilisez Performance Insights, vous pouvez afficher les statistiq ues SQL au niveau de l'instruc tion et du récapitulatif pour Amazon RDS for Oracle. Pour de plus amples informations, consultez Analyzing running queries in Oracle (Analyse des requêtes en cours d'exécution dans Oracle).	4 octobre 2021
Amazon RDS on AWS Outposts prend en charge des versions supplémentaires de RDS pour PostgreSQL	RDS sur Outposts prend désormais en charge RDS for PostgreSQL versions 12.8 et 13.4. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	1er octobre 2021

Amazon RDS prend en charge Oracle APEX version 21.1.v1	Vous pouvez utiliser Oracle APEX 21.1.v1 avec toutes les versions prises en charge d'Oracle Database. Pour plus d'informations, consultez Oracle Application Express.	24 septembre 2021
Amazon RDS for Oracle prend en charge le chiffrement côté client pour NNE	Lorsque vous configurez NNE, vous pouvez éviter de forcer le chiffrement côté serveur. Par exemple, vous pouvez ne pas forcer toutes les communications client à utiliser le chiffrement car le serveur en a besoin. Dans ce cas, vous pouvez forcer le chiffrement côté client à l'aide des options `SQLNET.*CLIENT`. Pour plus d'informations, consultez Chiffrement du réseau natif Oracle.	24 septembre 2021
Amazon RDS for MySQL et RDS for PostgreSQL prennent en charge de nouvelles classes d'instance de base de données	Vous pouvez désormais utiliser les classes d'instance db.r5b, db.t4g et db.x2g pour créer des instances de base de données Amazon RDS exécutant MySQL ou PostgreSQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	15 septembre 2021

Amazon RDS for Microsoft SQL Server prend en charge Java Database Connectivity (JDBC) avec Microsoft Distributed Transaction Coordinator (MSDTC)	Les transactions JDBC XA sont désormais prises en charge avec MSDTC pour SQL Server 2017 versions 14.00.3223.3 et ultérieures, et pour SQL Server 2019. Pour plus d'informations, consultez Support for Microsoft Distributed Transaction Coordinator in RDS for SQL Server.	7 septembre 2021
Amazon RDS prend en charge MariaDB 10.5.12, 10.4.21, 10.3.31 et 10.2.40	Vous pouvez désormais créer des instances de bases de données Amazon RDS en exécutant les versions 10.5.12, 10.4.21, 10.3.31 et 10.2.40 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	2 septembre 2021
Amazon RDS a mis fin à la prise en charge d'Oracle Database 18c	Vous pouvez créer des instances de base de données uniquement pour Oracle Database 12c et Oracle Database 19c. Si vous avez des instantanés Oracle Database 18c, mettez-les à niveau vers une version ultérieure. Pour plus d'informations, consultez Mise à niveau d'un instantané Oracle DB.	17 août 2021

Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 14 bêta 2 dans l'environnement en préversion de base de données	Pour plus d'informations sur la version PostgreSQL 14 bêta 1, consultez Notes de mise à jour de PostgreSQL 14 bêta 1. Pour plus d'informations sur la version PostgreSQL 14 bêta 2, consultez Notes de mise à jour de PostgreSQL 14 bêta 2. Pour plus d'informations sur l'environnement en préversion de base de données, consultez Utilisation de l'environnement de prévisualisation de base de données.	9 août 2021
Amazon RDS prend en charge RDS Proxy dans un VPC partagé	Vous pouvez désormais créer un RDS Proxy dans un VPC partagé. Pour plus d'informations sur RDS Proxy, consultez « Gestion des connexions avec le proxy Amazon RDS » dans le Guide de l'utilisateur Amazon RDS ou le Guide de l'utilisateur Aurora.	6 août 2021
Amazon RDS prend en charge MariaDB version 10.2.39	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.2.39. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	4 août 2021

Amazon RDS for Oracle ajoute l'option `TIMEZONE_FILE_AUTOUPGRADE`	Avec cette option, vous pouvez mettre à niveau le fichier sur le fuseau horaire actuel vers la dernière version de votre instance de base de données Oracle. Pour plus d'informations, consultez Mise à niveau automatique du fichier Oracle sur le fuseau horaire.	30 juillet 2021
Amazon RDS étend la prise en charge des sauvegardes automatisées entre régions	Vous pouvez maintenant répliquer des instantanés de bases de données et des journaux de transactions entre davantage de Région s AWS. Pour plus d'informa tions, consultez la section Réplication de sauvegardes automatisées vers une autre AWS région.	19 juillet 2021
Prise en charge de MySQL 5.7.34	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.34. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	8 juillet 2021

Amazon RDS on AWS Outposts prend en charge des versions supplémentaires de RDS pour PostgreSQL	RDS sur Outposts prend désormais en charge RDS for PostgreSQL versions 12.7 et 13.3. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	8 juillet 2021
Amazon RDS for PostgreSQL prend en charge oracle_fdw	Vous pouvez utiliser l'extension oracle_fdw afin de fournir un encapsuleur de données externes pour accéder aux bases de données Oracle. Pour plus d'informations, consultez Accès aux données externes à l'aide de l'extension oracle_fdw.	8 juillet 2021

Amazon RDS prend en charge Oracle Management Agent (OMA) version 13.5	Vous pouvez utiliser Oracle Management Agent (OMA) version 13.5 avec Oracle Enterprise Manager (OEM) Cloud Control 13c versions 5 et ultérieures. Amazon RDS for Oracle installe OMA, qui communique ensuite avec votre Oracle Management Service (OMS) pour fournir des informations de surveillance. Si vous exécutez OMS 13.5, vous pouvez gérer vos bases de données en installant OMA 13.5. Pour plus d'informations, consultez Oracle Management Agent pour Enterprise Manager Cloud Control.	7 Juillet 2021

Amazon RDS for Oracle prend en charge le téléchargement de journaux à partir d'Amazon S3	Si les journaux de reprise archivés ne figurent pas sur votre instance mais sont protégés par votre période de rétention des sauvegardes, vous pouvez utiliser `rdsadmin.rdsadmin_archive_log_download` pour les télécharger à partir d'Amazon S3. RDS for Oracle enregistre les journaux dans le répertoire `/rdsdbdata/log/arch` sur votre instance de base de données. Pour plus d'informations, consultez Téléchargement des journaux de reprise archivés à partir d'Amazon S3.	2 juillet 2021
Amazon RDS prend en charge MariaDB versions 10.4.18. et 10.5.9	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB versions 10.4.18 et 10.5.9. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	30 juin 2021

Amazon RDS for Oracle prend en charge les flux d'activité de base de données	Vous pouvez désormais surveiller une instance de base de données Oracle à l'aide des flux d'activité de base de données. Une base de données Oracle écrit des enregistrements dans le journal d'activité d'audit unifié. Lorsque vous démarrez un flux d'activité de base de données sur une instance de base de données Oracle, Amazon Kinesis diffuse toutes les activités correspondant aux politiques d'audit d'Oracle Database. Pour plus d'informations, consultez Surveillance d'Amazon RDS à l'aide des flux d'activité de base de données.	23 Juin 2021
Amazon RDS for Oracle introduit des classes d'instances à mémoire optimisée	Les nouvelles classes d'instance de base de données Oracle sont optimisées pour les charges de travail qui nécessitent de la mémoire, du stockage et I/O par vCPU supplémentaires. Pour plus d'informations, consultez Classes d'instances RDS pour Oracle.	23 Juin 2021

Prise en charge de MySQL 8.0.25	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.25. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	18 juin 2021
Amazon RDS on AWS Outposts prend en charge des versions supplémentaires de RDS pour PostgreSQL	RDS on Outposts prend désormais en charge les versions 12.5, 12.6, 13.1 et 13.2 de RDS for PostgreSQL. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	28 mai 2021
Amazon RDS prend en charge MariaDB versions 10.2.37 et 10.3.28	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 10.2.37 et 10.3.28 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	27 mai 2021

Amazon RDS for Oracle prend en charge la base de données de conteneur (CDB) multi-locataires	Une architecture multi-locataires permet à une base de données Oracle d'être une base de données de conteneur (CDB). Dans Oracle Database 19c, votre CDB peut inclure une seule base de données enfichable (PDB). L'expérience utilisateur avec une PDB est essentiellement identique à l'expérience utilisateur avec une base de données non-CDB. Pour plus d'informations, voir Architecture RDS for Oracle.	25 mai 2021
Amazon RDS on AWS Outposts prend en charge Amazon RDS pour SQL Server	RDS sur Outposts prend désormais en charge Amazon RDS for SQL Server. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	11 mai 2021

Amazon RDS étend la prise en charge des sauvegardes automatisées entre régions	Vous pouvez désormais configurer les instances de base de données Amazon RDS exécutant Microsoft SQL Server pour répliquer les instantanés de base de données et les journaux de transactions dans une autre région. AWS Pour plus d'informations, consultez la section Réplication de sauvegardes automatisées vers une autre AWS région.	7 mai 2021
Amazon RDS prend en charge les sauvegardes automatisées entre régions pour les instances de base de données chiffrées	Vous pouvez maintenant répliquer des instantanés de base de données et des journaux de transactions vers une autre région AWS pour les instances de base de données Amazon RDS chiffrées exécutant Oracle ou PostgreSQL. Pour plus d'informations, consultez la section Réplication de sauvegardes automatisées vers une autre AWS région.	3 mai 2021
Amazon RDS on AWS Outposts prend en charge la surveillance d'Amazon CloudWatch	RDS on Outposts prend désormais en charge la surveillance d'Amazon CloudWatch . Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	21 avril 2021

RDS pour PostgreSQL prend en charge les fonctions AWS Lambda	Vous pouvez désormais invoquer des fonctions AWS Lambda pour vos instances de base de données RDS pour PostgreSQL. Pour en savoir plus, consultez Appel d'une fonction AWS Lambda à partir d'une instance de base de données RDS for PostgreSQL.	13 avril 2021
RDS for SQL Server prend en charge les événements étendus	Vous pouvez utiliser les événements étendus SQL Server pour capturer des informations de débogage et de dépannage. Pour plus d'informations, consultez Utilisation des événements étendus avec Amazon RDS pour Microsoft SQL Server.	8 avril 2021
Prise en charge de MySQL 8.0.23, 5.7.33 et 5.6.51	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 8.0.23, 5.7.33 et 5.6.51. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	31 mars 2021

Échec de la restauration automatique pour la mise à niveau Amazon RDS for MySQL	En cas d'échec d'une mise à niveau d'instance de base de données de MySQL version 5.7 vers MySQL version 8.0, Amazon RDS annule automatiquement les modifications effectuées pour la mise à niveau. Après la restauration, l'instance de base de données MySQL exécute MySQL version 5.7. Pour plus d'informations, consultez Restauration après l'échec d'une mise à niveau de MySQL 5.7 vers 8.0.	18 mars 2021
Amazon RDS prend en charge les réplicas en lecture entre régions dans les régions opt-in	Vous pouvez maintenant répliquer des instances de base de données vers des régions opt-in. Pour plus d'informations, consultez la section Création d'une réplique lue dans une autre AWS région.	18 mars 2021

Amazon RDS prévoit de rendre obsolète Oracle Database version 18c	Oracle Database 18c (18.0.0.0) sera bientôt obsolète. Oracle Corporation ne fournira plus de correctifs pour Oracle Database 18c après end-of-support cette date. Le 1er juillet 2021, Amazon RDS prévoit de débuter la mise à niveau automatique des instances Oracle Database 18c vers Oracle Database 19c. Avant le début des mises à niveau automatiques, nous vous recommandons vivement de mettre à niveau manuellement vos instances Oracle Database 18c existantes vers Oracle Database 19c. Pour plus d'informations, consultez Préparation en vue de la mise à niveau automatique d'Oracle Database 18c.	11 mars 2021
Amazon RDS a mis fin au support d'Oracle Database 11g	Vous pouvez uniquement créer des instances de base de données pour Oracle Database 12c version 1 (12.1.0.2) et versions ultérieures. Si vous avez des instantanés Oracle Database 11g, mettez-les à niveau vers une version ultérieure. Pour plus d'informations, consultez Mise à niveau d'un instantané Oracle DB.	11 mars 2021

Amazon RDS prend en charge les sauvegardes continues des instances de base de données dans AWS Backup	Vous pouvez désormais créer des sauvegardes automatisées AWS Backup et restaurer des instances de base de données à partir de ces sauvegardes à une heure spécifiée. Pour plus d'informations, consultez la section Utilisation AWS Backup pour gérer les sauvegardes automatisées.	10 mars 2021
Amazon RDS prend en charge Oracle Management Agent (OMA) version 13.4	Vous pouvez utiliser Oracle Management Agent (OMA) version 13.4 avec Oracle Enterprise Manager (OEM) Cloud Control 13c version 4 mise à jour 9. Amazon RDS for Oracle installe OMA, qui communique ensuite avec votre Oracle Management Service (OMS) pour fournir des informations de surveillance. Si vous exécutez OMS 13.4, vous pouvez gérer vos bases de données en installant OMA 13.4. Pour plus d'informations, consultez Oracle Management Agent pour Enterprise Manager Cloud Control.	10 mars 2021

Améliorations du point de terminaison proxy RDS	Vous pouvez créer d'autres points de terminaison associés à chaque proxy RDS. La création d'un point de terminaison dans un autre VPC permet un accès entre VPC pour le proxy. Les proxies pour les clusters Aurora MySQL peuvent également avoir des points de terminaison en lecture seule. Ces points de terminaison du lecteur se connectent aux instances de base de données de lecteurs dans les clusters et peuvent améliorer l'évolutivité et la disponibilité de la lecture pour les applications exigeantes en requêtes. Pour de plus amples informations sur RDS Proxy, veuillez consulter « Gestion des connexions avec le proxy Amazon RDS » dans le Guide de l'utilisateur Amazon RDS ou le Guide de l'utilisateur Aurora.	8 mars 2021

Amazon RDS étend la prise en charge des sauvegardes automatisées entre régions	Vous pouvez désormais configurer les instances de base de données Amazon RDS exécutant PostgreSQL pour répliquer les instantan és de base de données et les journaux de transactions dans une autre région. AWS Pour plus d'informations, consultez la section Réplication de sauvegardes automatisées vers une autre AWS région.	8 mars 2021
Filtres de réplication pour Amazon RDS for MariaDB et MySQL pris en charge dans les régions Chine (Pékin) et Chine (Ningxia)	Le filtrage de réplication est désormais pris en charge dans les régions Chine (Pékin) et Chine (Ningxia). Pour plus d'informations, consultez la section Configuration des filtres de réplication avec MariaDB et Configuration des filtres de réplication avec MySQL.	5 mars 2021
Amazon RDS prend en charge la copie d'instantanés de base de données entre régions dans les régions opt-in	Vous pouvez désormais copier des instantanés de base de données vers et depuis les régions optionnelles. AWS Pour plus d'informations, voir Copier des instantanés d'une AWS région à l'autre.	4 mars 2021

Amazon RDS for SQL Server prend maintenant en charge les groupes de disponibilité toujours actifs pour Standard Edition	Lorsque vous créez une instance de base de données à l'aide de la configuration multi-AZ sur SQL Server 2019 pour le moteur de base de données Standard Edition, RDS utilise automatiquement les groupes de disponibilité. Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Microsoft SQL Server.	23 février 2021
Amazon RDS for Oracle introduit des procédures liées aux conseillers	Le package `rdsadmin_util` inclut les procédures `advisor_task_set_parameter`, `advisor_task_drop` et `dbms_stats_init`. Vous pouvez utiliser ces procédures pour modifier, arrêter et réactiver des tâches de conseiller telles que `AUTO_STATS_ADVISOR_TASK`. Pour plus d'informations, consultez Définition des paramètres pour les tâches de conseiller.	23 février 2021

Amazon RDS fournit des raisons de basculement pour les instances de base de données multi-AZ	Vous pouvez désormais afficher des explications plus détaillées lorsqu'une instance de base de données multi-AZ bascule vers un réplica de secours. Pour plus d'informations, voir la section Processus de basculement pour Amazon RDS.	18 février 2021
Amazon RDS étend la prise en charge de l'exportation d'instantanés vers Amazon S3	Vous pouvez désormais exporter les données d'instantané de base de données vers Amazon S3 dans Chine. Pour de plus amples informations, veuillez consulter Exportation de données d'instantanés de bases de données vers Amazon S3.	17 février 2021

Filtres de réplication pour Amazon RDS for MariaDB et MySQL	Vous pouvez configurer des filtres de réplication pour les instances MySQL et MariaDB. Les filtres de réplication spécifient quelles bases de données et quelles tables sont répliquées dans un réplica en lecture. Vous pouvez créer des listes de bases de données et de tables à inclure ou à exclure pour chaque réplica en lecture. Pour plus d'informations, consultez la section Configuration des filtres de réplication avec MariaDB et Configuration des filtres de réplication avec MySQL.	12 février 2021
RDS pour Oracle prend en charge Oracle APEX 20.2v1	Vous pouvez utiliser Oracle APEX 20.2.v1 avec toutes les versions prises en charge d'Oracle Database. Pour plus d'informations, consultez Oracle Application Express.	2 février 2021

Amazon RDS for SQL Server prend en charge le stockage local d'instance pour la base de données tempdb	Vous pouvez désormais lancer Amazon RDS for SQL Server sur les types d'instance EC2 Amazon db.r5d et db.m5d avec la base de données tempdb configurée pour utiliser un magasin d'instances. En plaçant les fichiers de données tempdb et les fichiers journaux en local, vous pouvez bénéficier de latences réduites de lecture et d'écriture par rapport aux stockages standard basés sur Amazon EBS. Pour plus d'informations, consultez la section Prise en charge du stockage d'instance pour la base de données tempdb sur Amazon RDS pour SQL Server.	27 janvier 2021
Amazon RDS for PostgreSQL prend en charge pg_partman et pg_cron	Amazon RDS for PostgreSQL prend désormais en charge les extensions pg_partman et pg_cron. Pour plus d'informations sur l'extension pg_partman, consultez Gestion des partitions PostgreSQL avec l'extension pg_partman. Pour plus d'informations sur l'extension pg_cron, consultez Planification de la maintenance avec l'extension PostgreSQL pg_cron.	12 janvier 2021

Amazon RDS prend en charge la publication du journal de l'agent de gestion Oracle sur Amazon CloudWatch Logs	Le journal d'Oracle Management Agent comprend emctl.log, emdctlj.log, gcagent.log, gcagent_errors.log, emagent.nohup et secure.log. Amazon RDS publie chacun de ces journaux sous la forme d'un flux de CloudWatch journaux distinct. Pour plus d'informations, consultez Publier des journaux Oracle sur Amazon CloudWatch Logs.	28 décembre 2020
Amazon RDS on AWS Outposts prend en charge des versions de base de données supplémentaires	RDS sur Outposts prend désormais en charge des versions supplémentaires de MySQL et PostgreSQL. Pour plus d'informations, consultez le support d'Amazon RDS on AWS Outposts pour les fonctionnalités d'Amazon RDS.	23 décembre 2020
Amazon RDS on AWS Outposts soutient Co IPs	RDS on Outposts prend désormais en charge les adresses IP (Co) détenues par les clients. IPs IPsCo-fournissez une connectivité locale ou externe aux ressources de vos sous-réseaux Outpost via votre réseau local. Pour plus d'informations, consultez Adresses IP clients pour RDS sur Outposts.	22 décembre 2020

Amazon RDS for Oracle planifie la mise à niveau des instances BYOL 11g vers 19c	Le 4 janvier 2021, nous prévoyons de commencer à mettre automatiq uement à niveau toutes les éditions des instances d'Oracle Database 11g sur le modèle Bring Your Own License (BYOL) vers Oracle Database 19c. La dernière mise à jour (RU) Oracle sera installée sur toutes les instances d'Oracle Database 11g, y compris sur les instances réservées . Pour plus d'informations, consultez Préparation de la mise à niveau automatique d'Oracle 11g BYOL.	11 décembre 2020
Amazon RDS prend en charge la réplication de sauvegardes automatisées vers une autre région AWS	Vous pouvez désormais configurer vos instances de base de données Amazon RDS pour répliquer les instantanés et les journaux de transactions AWS dans la région de destination de votre choix. Pour plus d'informa tions, consultez la section Réplication de sauvegardes automatisées vers une autre AWS région.	4 décembre 2020

Amazon RDS for Oracle et Microsoft SQL Server prend en charge une nouvelle classe d'instance de base de données	Vous pouvez désormais utiliser la classe d'instance db.r5b pour créer des instances de base de données Amazon RDS qui exécutent Oracle ou SQL Server. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour les classes d'instances de base de données.	4 décembre 2020
Prise en charge de MariaDB 10.2.32	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.2.32. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	25 novembre 2020
Amazon RDS for SQL Server prend en charge la suite Microsoft Business Intelligence sur SQL Server 2019	Vous pouvez désormais exécuter SQL Server Analysis Services, SQL Server Integration Services et SQL Server Reporting Services sur des instances de base de données à l'aide de la dernière version majeure. Pour plus d'informations, consultez Options pour le moteur de base de données Microsoft SQL Server.	24 novembre 2020

Amazon RDS for PostgreSQL version 13 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 13 dans l'environnement en préversion de base de données. Pour plus d'informations, consultez Versions de PostgreSQL 13.	24 novembre 2020
Amazon RDS Performance Insights introduit de nouvelles dimensions	Vous pouvez regrouper la charge de base de données en fonction des groupes de dimensions pour la base de données (PostgreSQL, MySQL et MariaDB), l'application (PostgreSQL) et le type de séance (PostgreSQL). Amazon RDS prend également en charge les dimensions db.name (PostgreSQL, MySQL et MariaDB), db.application.name (PostgreSQL) et db.session_type.name (PostgreSQL). Pour plus d'informations, consultez Tableau des principaux éléments de charge.	24 novembre 2020
Amazon RDS for MariaDB prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.5. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	23 novembre 2020

Prise en charge de MySQL 5.6.49	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MySQL version 5.6.49. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	20 novembre 2020
Prise en charge de MySQL 5.5.62	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MySQL version 5.5.62. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	20 novembre 2020
Performance Insights prend en charge l'analyse des statistiques pour l'exécution des requêtes PostgreSQL	Il est désormais possible d'analyser les statistiques des requêtes en cours d'exécution à l'aide de Performance Insights pour les instances de base de données PostgreSQL. Pour plus d'informations, consultez Statistiques pour PostgreSQL.	18 novembre 2020

Amazon RDS étend la prise en charge de la scalabilité automatique	Vous pouvez désormais activer la scalabilité automatique du stockage lors de la création d'un réplica en lecture, de la restauration d'une instance de base de données à une heure spécifiée ou de la restauration d'une instance de base de données MySQL à partir d'une sauvegarde Amazon S3. Pour plus d'informations, consultez Gestion automatique de la capacité avec la scalabilité automatique du stockage Amazon RDS.	18 novembre 2020
Amazon RDS for SQL Server prend en charge Database Mail	Avec Database Mail, vous pouvez envoyer des e-mails à partir de votre instance de base de données Amazon RDS for SQL Server. Après avoir spécifié les destinataires de l'e-mail, vous pouvez ajouter des fichiers ou des résultats de requête au message que vous envoyez. Pour plus d'informations, consultez Utilisation de Database Mail sur Amazon RDS pour SQL Server.	4 novembre 2020

Prise en charge de MySQL 8.0.21	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.21. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	22 octobre 2020
Amazon RDS étend la prise en charge de l'exportation d'instantanés vers Amazon S3	Vous pouvez désormais exporter les données des instantanés de base de données vers Amazon S3 dans toutes les AWS régions commerciales. Pour plus d'informations, veuillez consulter Exportation de données d'instantanés de bases de données vers Amazon S3.	22 octobre 2020
Amazon RDS for PostgreSQL prend en charge les mises à niveau de réplica en lecture	Avec Amazon RDS for PostgreSQL, lorsque vous effectuez une mise à niveau de version majeure de l'instance de base de données principale, les réplicas en lecture sont également automatiquement mis à niveau. Pour de plus amples informations, veuillez consulter Mise à niveau du moteur de base de données PostgreSQL.	15 octobre 2020

Amazon RDS for MariaDB, MySQL et PostgreSQL prennent en charge les classes d'instance de base de données Graviton2	Vous pouvez désormais utiliser les classes d'instance de base de données Graviton2 db.m6g.x et db.r6g.x pour créer des instances de base de données Amazon RDS exécutant MariaDB, MySQL ou PostgreSQL. Pour plus d'informations, consultez Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	15 octobre 2020
Amazon RDS for SQL Server prend en charge les mises à niveau vers SQL Server 2019	Vous pouvez mettre à niveau vos instances de base de données SQL Server vers SQL Server 2019. Pour de plus amples informations, veuillez consulter Mise à niveau du moteur de base de données Microsoft SQL Server.	6 octobre 2020

Amazon RDS for Oracle prend en charge la spécification du jeu de caractères national	Le jeu de caractères national, également appelé jeu de caractères NCHAR, est utilisé dans les types de données NCHAR, NVARCHAR2 et NCL0B. Lorsque vous créez une base de données, vous pouvez la spécifier AL16 UTF16 (par défaut) ou UTF8 sous forme de jeu de caractères NCHAR. Pour de plus amples informations, veuillez consulter Jeux de caractères Oracle pris en charge dans Amazon RDS.	2 octobre 2020
Prise en charge de MySQL 5.7.31	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.31. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	1er octobre 2020

Amazon RDS for PostgreSQL prend en charge l'exportation de données vers Amazon S3	Vous pouvez interroger des données à partir d'une instance de base de données PostgreSQL et les exporter directement vers des fichiers stockés dans un compartiment Amazon S3. Pour de plus amples informations, veuillez consulter la documentation relative à l'exportation de données à partir d'une instance de base de données RDS for PostgreSQL vers Amazon S3.	24 septembre 2020
Amazon RDS pour MySQL 8.0 prend en charge Percona XtraBackup	Vous pouvez désormais utiliser Percona XtraBackup pour restaurer une sauvegarde dans une instance de base de données Amazon RDS for MySQL 8.0. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données MySQL.	17 septembre 2020
Amazon RDS for SQL Server prend en charge la sauvegarde et la restauration natives sur les instances de base de données avec les réplicas en lecture	Vous pouvez restaurer une sauvegarde native SQL Server sur une instance de base de données avec réplicas en lecture configurés. Pour de plus amples informations, veuillez consulter Importation et exportation de bases de données SQL Server.	16 septembre 2020

Amazon RDS for SQL Server prend en charge des fuseaux horaires supplémentaires	Vous pouvez faire correspondre votre fuseau horaire d'instance de base de données avec le fuseau horaire choisi. Pour de plus amples informations, veuillez consultez Fuseau horaire local pour les instances de bases de données Microsoft SQL Server.	11 septembre 2020
Amazon RDS for PostgreSQL version 13 bêta 3 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 13 bêta 3 dans l'environnement en préversion de base de données Pour plus d'informations, consultez Versions de PostgreSQL 13.	9 septembre 2020

Amazon RDS for SQL Server prend en charge l'indicateur de trace 692	Vous pouvez désormais utiliser l'indicateur de trace 692 comme paramètre de démarrage à l'aide de groupes de paramètres de base de données. L'activation de cet indicateur de trace désactive les insertions rapides lors du chargement en masse de données dans un segment de mémoire ou un index cluster. Pour de plus amples informations, veuillez consulter la section relative à la désactivation des insertions rapides pendant le chargement en masse.	27 août 2020
Amazon RDS for SQL Server prend en charge Microsoft SQL Server 2019	Vous pouvez désormais créer des instances de base de données RDS qui utilisent SQL Server 2019. Pour de plus amples informations, veuillez consulter Versions de Microsoft SQL Server sur Amazon RDS.	26 août 2020

RDS for Oracle prend en charge la base de données de réplica montée	Lorsque vous créez ou modifiez un réplica Oracle, vous pouvez le placer en mode monté. Étant donné que la base de données de réplica n'accepte pas les connexion s utilisateur, elle ne peut pas servir de charge de travail en lecture seule. Le réplica monté supprime les fichiers de journalisation archivés après leur application. L'utilisation principale des réplicas montés est la reprise après sinistre inter-région. Pour de plus amples informations, veuillez consulter Généralités sur les réplicas Oracle.	13 août 2020
RDS for Oracle prévoit la mise à niveau des instances 11g LI SE1	Le 1er novembre 2020, nous prévoyons de commencer à mettre à niveau automatiq uement les instances Oracle Database 11g SE1 License Included (LI) vers Oracle Database 19c pour Amazon RDS for Oracle. La dernière mise à jour (RU) Oracle sera installée sur toutes les instances 11g, y compris les instances réservées. Pour plus d'informations, voir Préparation à la mise à niveau automatique d'Oracle Database 11g SE1.	31 juillet 2020

Amazon RDS prend en charge les nouvelles classes d'instance de base de données Graviton2 dans la version préliminaire pour PostgreSQL et MySQL	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL ou MySQL qui utilisent les classes d'instance de base de données db.m6g.x et db.r6g.x. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	30 juillet 2020
RDS pour Oracle prend en charge Oracle APEX 20.1v1	Vous pouvez utiliser Oracle APEX 20.1v1 avec toutes les versions prises en charge d'Oracle Database. Pour de plus amples informations, veuillez consulter Oracle Application Express.	28 juillet 2020
Prise en charge de MySQL 8.0.20	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.20. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	23 juillet 2020

Amazon RDS for MariaDB et MySQL prennent en charge les nouvelles classes d'instance de base de données	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB et MySQL qui utilisent les classes d'instance de base de données db.m5.16xlarge, db.m5.8xlarge, db.r5.16xlarge et db.r5.8xlarge. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	23 juillet 2020
RDS for SQL Server prend en charge la désactivation des anciennes versions de TLS et les chiffrements	Vous pouvez activer et désactiver certains protocoles de sécurité et chiffrements. Pour de plus amples informations, veuillez consulter Configuration des protocoles et des chiffrements de sécurité.	21 juillet 2020
RDS prend en charge Oracle Spatial sur SE2	Vous pouvez utiliser Oracle Spatial dans Standard Edition 2 (SE2) pour toutes les versions de 12.2, 18c et 19c. Pour de plus amples informations, veuillez consulter Oracle Spatial.	9 juillet 2020

Amazon RDS prend en charge AWS PrivateLink	Amazon RDS prend désormais en charge la création de points de terminaison Amazon VPC pour les appels d'API Amazon RDS afin de maintenir le trafic entre les applications et Amazon RDS sur le réseau. AWS Pour plus d'informations, consultez Amazon RDS et points de terminaison de VPC d'interface (AWS PrivateLink).	9 juillet 2020
Amazon RDS for PostgreSQL versions 9.4.x a atteint la fin de sa période de prise en charge.	Amazon RDS for PostgreSQL ne prend plus en charge les versions 9.4.x. Pour connaître les versions prises en charge, veuillez consulter Versions de base de données PostgreSQL prises en charge.	8 juillet 2020
Prise en charge de MariaDB 10.3.23 et 10.4.13	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 10.3.23 et 10.4.13. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	6 juillet 2020
Amazon RDS sur AWS Outposts	Vous pouvez créer des instances de base de données Amazon RDS sur AWS Outposts. Pour plus d'informations, consultez Utilisation d'Amazon RDS sur AWS Outposts.	6 juillet 2020

Amazon RDS for Oracle crée automatiquement les fichiers d'inventaire	Pour ouvrir des demandes de service pour des clients BYOL, Oracle Support demande des fichiers d'inventaire générés par Opatch. Amazon RDS for Oracle crée automatiquement des fichiers d'inventaire toutes les heures dans le répertoire BDUMP Pour de plus amples informations, veuillez consulter Accès aux fichiers Opatch.	6 juillet 2020
Prise en charge de MySQL 5.7.30 et 5.6.48	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL versions 5.7.30 et 5.6.48. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	25 juin 2020

Amazon RDS for Oracle prend en charge ADRCI	L'utilitaire ADRCI (Automatic Diagnostic Repository Command Interpreter) est un outil de ligne de commande Oracle qui vous permet de gérer les données de diagnostic. En utilisant les fonctions du package Amazon RDS `rdsadmin_adrci_util` , vous pouvez répertorier et compiler les problèmes et les incidents, et afficher les fichiers de suivi. Pour de plus amples informations, veuillez consulter Tâches de diagnostic communes DBA pour les instances de bases de données Oracle.	17 juin 2020
Prise en charge de MySQL 8.0.19	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.19. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	2 juin 2020

MySQL 8.0 prend en charge les noms de table en minuscules	Vous pouvez désormais définir le paramètre `lower_case_table_names` sur 1 pour les instances de base de données Amazon RDS exécutant MySQL version 8.0.19 et versions 8.0 supérieures. Pour de plus amples informations, veuillez consulter Exceptions des paramètres MySQL pour les instances de base de données Amazon RDS.	2 juin 2020
Amazon RDS for Microsoft SQL Server prend en charge SQL Server Integration Services (SSIS)	SSIS est une plateforme d'intégration de données et d'applications de flux de travail. Vous pouvez activer SSIS sur des instances de base de données existantes ou nouvelles. Il est installé sur la même instance de base de données que votre moteur de base de données. Pour de plus amples informations, veuillez consulter Prise en charge de SQL Server Integration Services dans SQL Server.	19 mai 2020

Amazon RDS for Microsoft SQL Server prend en charge SQL Server Reporting Services (SSRS)	SSRS est une application basée sur un serveur et utilisée pour la génération et la distribution de rapports. Vous pouvez activer SSRS sur des instances de base de données existantes ou nouvelles. Il est installé sur la même instance de base de données que votre moteur de base de données. Pour de plus amples informati ons, veuillez consulter Support for SQL Server Reporting Services in SQL Server.	15 mai 2020
Amazon RDS for Microsoft SQL Server prend en charge l'intégration S3 sur les instances Multi-AZ	Vous pouvez désormais utiliser Amazon S3 avec des fonctionnalités SQL Server telles que l'insertion en bloc sur des instances de base de données Multi-AZ. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS pour SQL Server avec Amazon S3.	15 mai 2020
Amazon RDS for Oracle prend en charge la purge de la corbeille	La procédure `rdsadmin.rdsadmin_util.purge_dba_recyclebin` purge la corbeille. Pour de plus amples informations, veuillez consulter Purging the Recycle Bin.	13 mai 2020

Amazon RDS for Oracle améliore la facilité de gestion d'Automatic Workload Repository (AWR)	Les procédures `rdsadmin.rdsadmin_diagnostic_util` génèrent des rapports AWR et extraient des données AWR dans des fichiers de vidage. Pour de plus amples informations, veuillez consulter Generating Performance Reports with Automatic Workload Repository (AWR).	13 mai 2020
Amazon RDS for Microsoft SQL Server prend en charge Microsoft Distributed Transaction Coordinator (MSDTC)	Amazon RDS for SQL Server prend en charge les transactions distribuées entre les hôtes. Pour de plus amples informations, veuillez consulter Support for Microsoft Distributed Transaction Coordinator in SQL Server.	4 mai 2020
Amazon RDS for Microsoft SQL Server prend en charge les nouvelles versions	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant les versions SQL Server CU19 14.00.3281.6, 2016 SP2 CU11 13.00.5598.27, 2014 SP3 CU4 12.00.6329.1 et 2012 GDR 11.0.7493.4 pour toutes les éditions. SP4 Pour de plus amples informations, veuillez consulter Versions de Microsoft SQL Server sur Amazon RDS.	28 avril 2020

Amazon RDS disponible dans la région Région Europe (Milan)	Amazon RDS est désormais disponible dans la région Région Europe (Milan). Pour de plus amples informations, veuillez consulter Régions et zones de disponibilité.	28 avril 2020
Amazon RDS prend en charge les Local Zones	Vous pouvez désormais lancer des instances de base de données dans un sous-réseau de zone locale. Pour plus d'informations, veuillez consulter Régions, zones de disponibilité et Local Zones.	23 avril 2020
Amazon RDS disponible dans la région Région Afrique (Le Cap)	Amazon RDS est désormais disponible dans la région Région Afrique (Le Cap). Pour de plus amples informations, veuillez consulter Régions et zones de disponibilité.	22 avril 2020
Amazon RDS for Microsoft SQL Server prend en charge SQL Server Analysis Services (SSAS)	SSAS est un outil de traitement analytique en ligne (OLAP) et d'exploration de données installé dans SQL Server. Vous pouvez activer SSAS sur des instances de base de données existantes ou nouvelles. Il est installé sur la même instance de base de données que votre moteur de base de données. Pour plus d'informations, consultez Prise en charge de SQL Server Analysis Services dans SQL Server.	17 avril 2020

Proxy Amazon RDS for PostgreSQL	Le proxy Amazon RDS est désormais disponible pour PostgreSQL. Vous pouvez utiliser le proxy RDS pour réduire la surcharge de connexions sur votre instance de base de données ainsi que le risque d'erreurs liées à un trop grand nombre de connexions. Le proxy RDS est actuellement disponibl e en version préliminaire pour PostgreSQL. Pour plus d'informations, veuillez consulter Gestion des connexions avec le proxy Amazon RDS (version préliminaire).	8 avril 2020
Amazon RDS for Oracle prend en charge Oracle APEX version 19.2.v1	Amazon RDS for Oracle prend désormais en charge Oracle Application Express (APEX) version 19.2.v1. Pour de plus amples informations, veuillez consulter Oracle Application Express.	8 avril 2020
Amazon RDS for MariaDB prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent MariaDB version 10.4. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	6 avril 2020

Amazon RDS Performance Insights est disponible pour Amazon RDS for MariaDB 10.4	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for MariaDB version 10.4. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	6 avril 2020
Amazon RDS for PostgreSQL versions 9.3.x a atteint la fin de sa période de prise en charge	Amazon RDS for PostgreSQL ne prend plus en charge les versions 9.3.x. Pour connaître les versions prises en charge, veuillez consulter Versions de base de données PostgreSQL prises en charge.	3 avril 2020
Amazon RDS for Microsoft SQL Server prend en charge les réplicas en lecture	Vous pouvez désormais créer des réplicas en lecture pour les instances de base de données SQL Server. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture.	3 avril 2020
Amazon RDS for Microsoft SQL Server prend en charge les sauvegardes dans plusieurs fichiers	Vous pouvez désormais sauvegarder des bases de données dans plusieurs fichiers à l'aide de la fonction de sauvegarde et de restauration natives de SQL Server. Pour de plus amples informations, veuillez consulter Sauvegarde d'une base de données.	2 avril 2020

Intégration d'Amazon RDS for Oracle avec AWS License Manager	Amazon RDS for Oracle est désormais intégré AWS License Managerà. Si vous utilisez le modèle Bring Your Own License, AWS License Manager l'intégration facilite le suivi de l'utilisation de vos licences Oracle au sein de votre organisation. Pour plus d'informations, consultez la section Intégration avec AWS License Manager.	23 mars 2020
Prise en charge de 64 Tio sur les instances db.r5 dans Amazon RDS for MariaDB et MySQL	Vous pouvez désormais créer des instances de base de données Amazon RDS for MariaDB et MySQL qui utilisent la classe d'instanc e de base de données db.r5 avec une capacité maximale de stockage de 64 Tio. Pour de plus amples informati ons, veuillez consulter Autres facteurs ayant un impact sur les performances de stockage.	18 mars 2020
Prise en charge de MySQL 8.0.17	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.17. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	10 mars 2020

Amazon RDS Performance Insights est disponible pour Amazon RDS for MySQL 8.0	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for MySQL version 8.0.17 et versions 8.0 ultérieures. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	10 mars 2020
Prise en charge de MySQL 5.6.46	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.6.46. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	28 février 2020
Amazon RDS Performance Insights est disponible pour Amazon RDS for MariaDB 10.3	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for MariaDB version 10.3.13 et versions 10.3 ultérieures. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	26 février 2020
Prise en charge de MySQL 5.7.28	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.28. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	20 février 2020

Prise en charge de MariaDB 10.3.20	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MariaDB version 10.3.20. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	20 février 2020
Amazon RDS for Microsoft SQL Server prend en charge une nouvelle classe d'instance de base de données	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant SQL Server qui utilisent la classe d'instance de base de données db.z1d. Pour de plus amples informations, veuillez consulter Prise en charge des classes d'instances de bases de données pour Microsoft SQL Server.	19 février 2020
Prise en charge des domaines Active Directory entre comptes et entre VPC dans Amazon RDS for SQL Server	Amazon RDS pour Microsoft SQL Server prend désormais en charge l'association d'instances de base de données à des domaines Active Directory détenus par différents comptes VPCs et. Pour de plus amples informations, veuillez consulter Utilisation de l'authentification Windows avec une instance de base de données Microsoft SQL Server.	13 février 2020

Option OLAP d'Oracle	Amazon RDS for Oracle prend désormais en charge l'option OLAP (Online Analytical Processing) pour les instances de base de données Oracle. Vous pouvez utiliser Oracle OLAP pour analyser de grandes quantités de données en créant des objets et des cubes dimensionnels conformément à la norme OLAP. Pour de plus amples informations, veuillez consulter Option OLAP d'Oracle.	13 février 2020
Prise en charge de FIPS 140-2 pour Oracle	Amazon RDS pour Oracle prend en charge la publicati on 140-2 (FIPS 140-2) de la Federal Information Processin g Standard pour les connexion s. SSL/TLS Pour de plus amples informations, veuillez consulter Prise en charge de FIPS.	11 février 2020

Amazon RDS for PostgreSQL prend en charge de nouvelles classes d'instance de base de données	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL qui utilisent les classes d'instance de base de données db.m5.16xlarge, db.m5.8xlarge, db.r5.16xlarge et db.r5.8xlarge. Pour de plus amples informations, veuillez consulter Moteurs de base de données pris en charge pour toutes les classes d'instances de base de données disponibles.	11 février 2020
Performance Insights prend en charge l'analyse statistique de l'exécution des requêtes MariaDB et MySQL	Il est désormais possible d'analyser les statistiques des requêtes en cours d'exécution à l'aide de Performance Insights pour les instances de base de données MariaDB et MySQL. Pour de plus amples informations, veuillez consulter Analyse des statistiques pour les requêtes en cours d'exécution.	4 février 2020

Prise en charge de l'exportation des données d'instantanés de bases de données vers Amazon S3 for MariaDB, MySQL et PostgreSQL	Amazon RDS prend en charge l'exportation des données d'instantanés de bases de données vers Amazon S3 for MariaDB, MySQL et PostgreSQL. Pour de plus amples informations, veuillez consulter Exportation de données d'instantanés de bases de données vers Amazon S3.	23 janvier 2020
Amazon RDS for MySQL prend en charge l'authentification Kerberos	Vous pouvez désormais utiliser l'authentification Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à vos instances de base de données Amazon RDS for MySQL. Pour de plus amples informations, veuillez consulter Utilisation de l'authentification Kerberos pour MySQL.	21 janvier 2020

Amazon RDS Performance Insights prend en charge l'affichage de texte SQL supplémentaire pour Amazon RDS for Microsoft SQL Server	Amazon RDS Performance Insights prend désormais en charge l'affichage de texte SQL supplémentaire dans le tableau de bord Performance Insights pour les instances de base de données Amazon RDS for Microsoft SQL Server. Pour de plus amples informations, veuillez consulter Affichage de texte SQL supplémentaire sur le tableau de bord de Performance Insights.	17 décembre 2019

Proxy Amazon RDS	Vous pouvez réduire la surcharge de gestion des connexions sur votre cluster et réduire le risque d'erreurs liées au « nombre de connexions trop élevé » à l'aide du proxy Amazon RDS. Vous associez chaque proxy à une instance de base de données RDS ou un cluster de base de données Aurora. Ensuite, vous utilisez le point de terminaison du proxy dans la chaîne de connexion de votre application. Le proxy Amazon RDS est actuellement en version préliminaire publique. Il prend en charge le moteur de base de données RDS for MySQL. Pour de plus amples informations, veuillez consulter Gestion des connexions avec le proxy Amazon RDS (version préliminaire).	3 décembre 2019

Amazon RDS activé AWS Outposts (version préliminaire)	Avec Amazon RDS activé AWS Outposts, vous pouvez créer des bases de données relationnelles AWS gérées dans vos centres de données locaux. RDS sur Outposts vous permet d'exécuter des bases de données RDS sur AWS Outposts. Pour plus d'informations, consultez Amazon RDS sur AWS Outposts (version préliminaire).	3 décembre 2019
Amazon RDS for Oracle prend en charge les réplicas en lecture entre régions	Amazon RDS for Oracle prend désormais en charge les réplicas en lecture entre régions avec Active Data Guard. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture et Utilisation des réplicas en lecture Oracle.	26 novembre 2019
Performance Insights prend en charge l'analyse statistique de l'exécution de requêtes Oracle	Il est désormais possible d'analyser les statistiques des requêtes en cours d'exécution à l'aide de Performance Insights pour les instances de bases de données Oracle. Pour de plus amples informations, veuillez consulter Analyse des statistiques pour les requêtes en cours d'exécution.	25 novembre 2019

Amazon RDS pour Microsoft SQL Server prend en charge la publication de journaux CloudWatch dans Logs	Vous pouvez configurer votre instance de base de données Amazon RDS for SQL Server pour publier les événement s du journal directement sur CloudWatch Amazon Logs. Pour plus d'informations, consultez la section Publicati on des journaux SQL Server sur Amazon CloudWatch Logs.	25 novembre 2019
Amazon RDS for Microsoft SQL Server prend en charge de nouvelles classes d'instanc es de bases de données	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant SQL Server qui utilisent les classes d'instanc es de bases de données db.x1e et db.x1. Pour de plus amples informations, veuillez consulter Prise en charge des classes d'instances de bases de données pour Microsoft SQL Server.	25 novembre 2019
Amazon RDS for Microsoft SQL Server prend en charge les restaurations différentielles et de journaux	Vous pouvez restaurer des sauvegardes différentielles et des journaux à l'aide de la sauvegarde et de la restaurat ion natives SQL Server. Pour de plus amples informations, veuillez consulter Utilisation de la sauvegarde et de la restauration natives.	25 novembre 2019

Fonction multi-AZ prise en charge sur Amazon RDS for Microsoft SQL Server dans les nouvelles régions	La fonction multi-AZ sur SQL Server est désormais disponible dans les régions : Chine, Moyen-Orient (Bahreïn) et Europe (Stockholm). Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Microsoft SQL Server.	22 novembre 2019
Amazon RDS for Microsoft SQL Server prend désormais en charge l'insertion en bloc et l'intégration S3	Vous pouvez transférer des fichiers entre une instance de base de données SQL Server et un compartiment Amazon S3. Ensuite, vous pouvez utiliser Amazon S3 avec des fonctionnalités SQL Server, telles que l'insertion en bloc. Pour de plus amples informations, veuillez consulter Intégration d'une instance de base de données Amazon RDS pour SQL Server avec Amazon S3.	21 novembre 2019

Compteurs Performance Insights pour Amazon RDS for Microsoft SQL Server	Vous pouvez désormais ajouter des compteurs de performances à vos graphiques Performance Insights pour les instances de bases de données Microsoft SQL Server. Pour plus d'informations, veuillez consulter Performance Insights counters for Amazon RDS for Microsoft SQL Server (Compteurs Performance Insights pour Amazon RDS for Microsoft SQL Server).	12 novembre 2019
Amazon RDS for Microsoft SQL Server prend en charge de nouvelles tailles de classes d'instances de bases de données	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant SQL Server qui utilisent les tailles d'instances 8xlarge et 16xlarge pour les classes d'instances de bases de données db.m5 et db.r5. Les tailles d'instance « small » à 2xlarge sont désormais disponibles pour la classe d'instance db.t3. Pour de plus amples informations, veuillez consulter Prise en charge des classes d'instances de bases de données pour Microsoft SQL Server.	11 novembre 2019

Prise en charge pour les mises à niveau d'instantanés PostgreSQL	Si vous avez des instantanés de bases de données Guides existants de vos instances de bases de données Amazon RDS PostgreSQL, vous pouvez désormais les mettre à niveau vers une version ultérieure du moteur de base de données PostgreSQL. Pour de plus amples informations, veuillez consulter Mise à niveau d'un instantané de base de données PostgreSQL.	7 novembre 2019
Amazon RDS for Oracle prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle Database 19c (19.0). Pour de plus amples informations, veuillez consulter Oracle Database 19c avec Amazon RDS.	7 novembre 2019
Amazon RDS for PostgreSQL version 12.0 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 12.0 dans l'environnement en préversion de base de données. Pour de plus amples informations, veuillez consulter PostgreSQL version 12.0 dans l'environnement en préversion de base de données.	1er novembre 2019

Amazon RDS for PostgreSQL prend en charge l'authentification Kerberos	Vous pouvez désormais utiliser l'authentification Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de base de données Amazon RDS exécutant PostgreSQL. Pour de plus amples informations, veuillez consulter Utilisation de l'authentification Kerberos avec Amazon RDS pour PostgreSQL.	28 octobre 2019
Tâches de base de données OEM Management Agent pour les instances de base de données Oracle	Les instances de base de données Amazon RDS for Oracle prennent désormais en charge les procédures d'appel de certaines commandes EMCTL sur Management Agent. Pour de plus amples informations, veuillez consulter Tâches de base de données de l'agent OEM.	24 octobre 2019

Amazon RDS for PostgreSQL prend en charge les bases de données transportables PostgreSQL	Les bases de données transportables PostgreSQL offrent une méthode extrêmement rapide de migration d'une base de données RDS PostgreSQL entre deux instances de base de données. Pour de plus amples informations, veuillez consulter Transport de bases de données PostgreSQL entre des instances de base de données.	8 octobre 2019
Amazon RDS for Oracle prend en charge l'authentification Kerberos	Vous pouvez désormais utiliser l'authentification Kerberos pour authentifier les utilisateurs lorsqu'ils se connectent à votre instance de base de données Amazon RDS qui exécute Oracle. Pour de plus amples informations, veuillez consulter Utilisation de l'authentification Kerberos avec Amazon RDS for Oracle.	30 septembre 2019

Amazon RDS for PostgreSQL version 12 bêta 3 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 12 bêta 3 dans l'environnement en préversion de base de données. Pour plus d'informations, consultez PostgreSQL version 12 bêta 3 sur Amazon RDS dans l'environnement en préversion de base de données.	28 août 2019
Prise en charge de MySQL 8.0.16	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.16. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	19 août 2019
Amazon RDS for Oracle prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle Database 18c (18.0). Pour plus d'informations, consultez Oracle Database 18c avec Amazon RDS.	15 août 2019

Management Agent pour OEM 13c Version 3	Les instances de base de données Amazon RDS for Oracle prennent désormais en charge Management Agent pour le contrôle de cloud Oracle Enterprise Manager (OEM) 13c Version 3. Pour de plus amples informations, veuillez consulter Oracle Management Agent pour Enterprise Manager Cloud Control.	7 août 2019
Amazon RDS for PostgreSQL version 12 bêta 2 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 12 bêta 2 dans l'environnement en préversion de base de données Pour plus d'informations, consultez PostgreSQL version 12 bêta 2 sur Amazon RDS dans l'environnement en préversion de base de données.	6 août 2019
Amazon RDS prend en charge les classements de serveur pour SQL Server	Amazon RDS for SQL Server prend en charge une sélection de classements pour les nouvelles instances de base de données. Pour plus d'informations, consultez Classements et jeux de caractères pour Microsoft SQL Server.	29 juillet 2019

Amazon RDS for Oracle prend en charge Oracle APEX version 19.1.v1	Amazon RDS for Oracle prend désormais en charge Oracle Application Express (APEX) version 19.1.v1. Pour de plus amples informations, veuillez consulter Oracle Application Express.	28 juin 2019
Amazon RDS for PostgreSQL version 13 bêta 1 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 13 bêta 1 dans l'environnement en préversion de base de données Pour plus d'informations, consultez Versions de PostgreSQL 13.	22 juin 2019
Scalabilité automatique du stockage Amazon RDS	Le dimensionnement automatique du stockage pour les instances de base de données Amazon RDS permet à Amazon RDS d'étendre automatiquement le stockage associé à une instance de base de données afin de réduire les risques de problèmes. out-of-space Pour plus d'informations sur la scalabilité automatique du stockage, consultez Utilisation du stockage pour les instances de base de données Amazon RDS.	20 juin 2019

Amazon RDS for Oracle prend en charge les classes d'instance de base de données db.z1d	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle qui utilisent les classes d'instances de base de données db.z1d. Pour plus d'informations, consultez Classe d'instance de base de données.	13 juin 2019
Amazon RDS Performance Insights prend en charge l'affichage de texte SQL supplémentaire pour Amazon RDS for Oracle	Amazon RDS Performance Insights prend désormais en charge l'affichage de texte SQL supplémentaire dans le tableau de bord Performance Insights pour les instances de base de données Amazon RDS for Oracle. Pour de plus amples informations, veuillez consulter Affichage de texte SQL supplémentaire sur le tableau de bord de Performance Insights.	10 juin 2019
Amazon RDS ajoute la prise en charge de restaurations natives des bases de données allant jusqu'à 16 To	Vous pouvez dorénavant effectuer la restauration native des bases allant jusqu'à 16 To depuis SQL Server vers Amazon RDS. Pour plus d'informations, consultez Amazon RDS for SQL Server : limitations et recommandations.	4 juin 2019

Amazon RDS ajoute la prise en charge de Microsoft SQL Server Audit	Amazon RDS for Microsoft SQL Server vous permet d'auditer des événement s au niveau du serveur et de la base de données à l'aide de SQL Server Audit et de consulter les résultats sur votre instance de base de données ou d'envoyer les fichiers journaux d'audit directement à Amazon S3. Pour de plus amples informati ons, veuillez consulter SQL Server Audit.	23 mai 2019
Améliorations des recommand ations Amazon RDS	Amazon RDS fournit maintenant des recommand ations automatisées pour les ressources de base de données. Par exemple, Amazon RDS fournit maintenant des recommand ations pour les paramètres de base de données. Pour de plus amples informations, veuillez consulter Utilisati ons des recommandations Amazon RDS.	22 mai 2019

Prise en charge de davantage de bases de données par instance de base de données pour Amazon RDS for SQL Server	Vous pouvez créer jusqu'à 30 bases de données sur chacune de vos instances de base de données exécutant Microsoft SQL Server. Pour de plus amples informations, veuillez consultez Limites pour les instances de bases de données for Microsoft SQL Server.	21 mai 2019
Prise en charge de 64 Tio et de 80k d'IOPS de stockage pour Amazon RDS for MariaDB, MySQL et PostgreSQL	Vous pouvez désormais créer des instances de bases de données Amazon RDS for MySQL, MariaDB et PostgreSQL avec jusqu'à 64 Tio de stockage et 80 000 IOPS provisionnés. Pour plus d'informations, consultez Stockage d'instance de base de données.	20 mai 2019
Amazon RDS for MySQL prend en charge les vérifications préalables aux mises à niveau	Lorsque vous mettez à niveau une instance de base de données de MySQL 5.7 vers MySQL 8.0, Amazon RDS recherche les incompatibilités en premier lieu. Pour de plus amples informations, veuillez consulter Vérifications préalables aux mises à niveau de MySQL 5.7 vers 8.0.	17 mai 2019

Prise en charge du plugin de validation de mot de passe MySQL	Vous pouvez désormais utiliser le plugin MySQL `validate_password` pour une sécurité améliorée des instances de base de données Amazon RDS for MySQL. Pour de plus amples informations, veuillez consulter Utilisation du plug-in de validation de mot de passe.	16 mai 2019
Compteurs Performance Insights pour Amazon RDS for Oracle	Vous pouvez désormais ajouter des compteurs de performances à vos graphiques Performance Insights pour les instances de base de données Oracle. Pour de plus amples informations, veuillez consulter Compteurs Performance Insights Counters pour Amazon RDS pour Oracle.	8 mai 2019
Prise en charge de la facturation par seconde	Amazon RDS est désormais facturé par tranches d'une seconde dans toutes les AWS régions, à l'exception AWS GovCloud des États-Unis pour les instances à la demande. Pour de plus amples informations, veuillez consulter Facturation des instances de base de données pour Amazon RDS.	25 avril 2019

Prise en charge de l'importation de données de Amazon S3 pour Amazon RDS for PostgreSQL	Vous pouvez désormais importer les données des fichiers Amazon S3 dans une table d'une instance de base de données RDS PostgreSQL. Pour de plus amples informations, veuillez consulter Importation de données Amazon S3 dans une instance de base de données RDS PostgreSQL.	24 avril 2019
Prise en charge de la restauration des sauvegardes 5.7 depuis Amazon S3	Vous pouvez désormais créer une sauvegarde de votre base de données MySQL version 5.7, la stocker sur Amazon S3, puis restaurer le fichier de sauvegarde sur une nouvelle instance de base de données Amazon RDS qui exécute MySQL. Pour de plus amples informations, veuillez consulter Restauration d'une sauvegarde dans une instance de base de données MySQL.	17 avril 2019

Prise en charge de plusieurs mises à niveau de version majeures pour Amazon RDS for PostgreSQL	Avec Amazon RDS for PostgreSQL, vous pouvez désormais choisir parmi plusieurs versions majeures lorsque vous mettez à niveau le moteur de base de données. Cette fonctionn alité vous permet de passer à une nouvelle version majeure lorsque vous mettez à niveau certaines versions de moteur PostgreSQL. Pour de plus amples informations, veuillez consulter Mise à niveau du moteur de base de données PostgreSQL.	16 avril 2019
Prise en charge de 64 Tio de stockage pour Amazon RDS for Oracle	Vous pouvez désormais créer des instances de bases de données Amazon RDS for Oracle avec jusqu'à 64 Tio de stockage et 80 000 IOPS provisionnés. Pour plus d'informations, consultez Stockage d'instance de base de données.	4 avril 2019
Prise en charge de MySQL 8.0.15	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0.15. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	3 avril 2019

Prise en charge de MariaDB 10.3.13	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent la version 10.3.13 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	3 avril 2019
Microsoft SQL Server 2008 R2 a atteint la fin de sa prise en charge sur Amazon RDS	Microsoft SQL Server 2008 R2 a atteint la fin de sa prise en charge, ce qui coïncide avec le plan de Microsoft de mettre fin à la prise en charge étendue de cette version le 9 juillet 2019. Tout instantané de Microsoft SQL Server 2008 R2 existant doit être mis à niveau automatiquement vers la dernière version mineure de Microsoft SQL Server 2012 à compter du 1er juin 2019. Pour de plus amples informations, veuillez consulter Prise en charge de Microsoft SQL Server 2008 R2 sur Amazon RDS.	2 avril 2019

Groupes de disponibilité Always On pris en charge dans Microsoft SQL Server 2017	Vous pouvez désormais utiliser les groupes de disponibilité Always On dans SQL Server 2017 Enterprise Edition 14.00.3049.1 ou une version ultérieure. Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Microsoft SQL Server.	29 mars 2019
Afficher les métriques de volume	Vous pouvez désormais afficher les métriques pour les volumes Amazon Elastic Block Store (Amazon EBS), qui sont les appareils physiques utilisés pour le stockage dans les bases de données et les journaux. Pour de plus amples informations, veuillez consulter Affichage de la surveillance améliorée.	20 mars 2019
Prise en charge de MySQL 5.7.25	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 5.7.25. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	19 mars 2019

Amazon RDS for Oracle prend en charge les tâches RMAN DBA	Amazon RDS for Oracle prend maintenant en charge les tâches Oracle Recovery Manager (RMAN) DBA, dont les sauvegardes RMAN. Pour de plus amples informations, veuillez consulter Tâches courantes DBA Recovery Manager (RMAN) pour les instances de base de données Oracle.	14 mars 2019
Amazon RDS for PostgreSQL prend en charge la version 11.1	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL version 11.1. Pour de plus amples informations, veuillez consulter PostgreSQL Version 11.1 sur Amazon RDS.	12 mars 2019
La restauration de plusieurs fichiers est disponible dans Amazon RDS for SQL Server	Vous pouvez désormais restaurer à partir de plusieurs fichiers avec Amazon RDS for SQL Server. Pour de plus amples informations, veuillez consulter Restauration d'une base de données.	11 mars 2019
MariaDB 10.2.21	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent la version 10.2.21 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	11 mars 2019

Amazon RDS for Oracle prend en charge les réplicas en lecture	Amazon RDS for Oracle prend désormais en charge les réplicas en lecture avec Active Data Guard. Pour de plus amples informations, veuillez consulter Utilisation des réplicas en lecture et Utilisation des réplicas en lecture Oracle.	11 mars 2019
Amazon RDS Performance Insights est disponible pour Amazon RDS for MariaDB	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for MariaDB. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	11 mars 2019
MySQL 8.0.13 et 5.7.24	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 8.0.13 et 5.7.24 de MySQL. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	8 mars 2019
Amazon RDS Performance Insights est disponible pour Amazon RDS for SQL Server	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for SQL Server. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	4 mars 2019

Amazon RDS for Oracle prend en charge l'intégration Amazon S3	Vous pouvez désormais transférer des fichiers entre une instance de base de données Amazon RDS for Oracle et un compartiment Amazon S3. Pour de plus amples informations, veuillez consulter Intégration de Amazon RDS pour Oracle et Amazon S3.	26 février 2019
Amazon RDS for MySQL et Amazon RDS for MariaDB prennent en charge les classes d'instances de bases de données db.t3	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant MySQL ou MariaDB qui utilisent les classes d'instances de bases de données db.t3. Pour plus d'informations, consultez Classe d'instance de base de données.	20 février 2019
Amazon RDS for MySQL et Amazon RDS for MariaDB prennent en charge les classes d'instances de bases de données db.r5	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant MySQL ou MariaDB qui utilisent les classes d'instances de bases de données db.r5. Pour plus d'informations, consultez Classe d'instance de base de données.	20 février 2019

Compteurs Performance Insights pour RDS for MySQL et PostgreSQL	Vous pouvez désormais ajouter des compteurs de performances à vos graphiques Performance Insights pour les instances de base de données MySQL et PostgreSQL. Pour de plus amples informations, veuillez consulter Composants du tableau de bord de Performance Insights.	19 février 2019
Amazon RDS for PostgreSQL prend désormais en charge l'ajustement du paramètre d'autovacuum adaptatif	L'ajustement du paramètre d'autovacuum adaptatif avec Amazon RDS for PostgreSQL permet d'éviter le renvoi à la ligne de l'ID de transaction en ajustant les valeurs du paramètre d'autovacuum automatiquement. Pour de plus amples informations, veuillez consulter Réduction de la probabilité de renvoi à la ligne de l'ID de transaction.	12 février 2019
Amazon RDS for Oracle prend en charge Oracle APEX versions 18.1.v1 et 18.2.v1	Amazon RDS for Oracle prend désormais en charge Oracle Application Express (APEX) versions 18.1.v1. et 18.2.v1. Pour de plus amples informations, veuillez consulter Oracle Application Express.	11 février 2019

Amazon RDS Performance Insights prend en charge l'affichage de texte SQL supplémentaire pour RDS for MySQL	Amazon RDS Performance Insights prend désormais en charge l'affichage de texte SQL supplémentaire dans le tableau de bord Performance Insights pour les instances de base de données MySQL. Pour de plus amples informations, veuillez consulter Affichage de texte SQL supplémentaire sur le tableau de bord de Performance Insights.	6 février 2019
Amazon RDS for PostgreSQL prend en charge les classes d'instance de base de données db.t3	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL qui utilisent les classes d'instances de base de données db.t3. Pour plus d'informations, consultez Classe d'instance de base de données.	25 janvier 2019
Amazon RDS for Oracle prend en charge les classes d'instance de base de données db.t3	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle qui utilisent les classes d'instances de base de données db.t3. Pour plus d'informations, consultez Classe d'instance de base de données.	25 janvier 2019

Amazon RDS Performance Insights prend en charge l'affichage de texte SQL supplémentaire pour Amazon RDS PostgreSQL	Amazon RDS Performance Insights prend désormais en charge l'affichage de texte SQL supplémentaire dans le tableau de bord Performance Insights pour les instances de base de données Amazon RDS PostgreSQL. Pour de plus amples informations, veuillez consulter Affichage de texte SQL supplémentaire sur le tableau de bord de Performance Insights.	24 janvier 2019
Amazon RDS for Oracle prend en charge une nouvelle version de SQLT	Amazon RDS for Oracle prend désormais en charge SQLT version 12.2.180725. Pour de plus amples informations, veuillez consulter Oracle SQLT.	22 janvier 2019
Amazon RDS for PostgreSQL prend en charge les classes d'instance de base de données db.r5	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL qui utilisent les classes d'instances de base de données db.r5. Pour plus d'informations, consultez Classe d'instance de base de données.	19 décembre 2018

Amazon RDS for PostgreSQL prend désormais en charge la gestion restreinte des mots de passe	Amazon RDS pour PostgreSQL permet de limiter les utilisateurs autorisés à gérer les mots de passe et les modifications de leur date d'expiration via le paramètre `rds.restrict_password_commands` et le rôle `rds_password`. Pour plus d'informations, consultez Restriction de la gestion des mots de passe.	19 décembre 2018
Amazon RDS for PostgreSQL prend en charge le téléchargement des journaux de base de données vers Amazon Logs CloudWatch	Amazon RDS for PostgreSQL prend en charge le téléchargement des journaux de base de données vers Logs. CloudWatch Pour plus d'informations, consultez la section Publication de journaux PostgreSQL dans des journaux. CloudWatch	10 décembre 2018
Amazon RDS for Oracle prend en charge les classes d'instance de base de données db.r5	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle qui utilisent les classes d'instances de base de données db.r5. Pour plus d'informations, consultez Classe d'instance de base de données.	20 novembre 2018

Rétention des sauvegardes lors de la suppression d'une instance de base de données	Amazon RDS conserve des sauvegardes automatiques lorsque vous supprimez une instance de base de données. Pour plus d'informations, consultez la page Utilisation des sauvegardes.	15 novembre 2018
Amazon RDS for PostgreSQL prend en charge les classes d'instance de base de données db.m5	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant PostgreSQL qui utilisent les classes d'instances de base de données db.m5. Pour plus d'informations, consultez Classe d'instance de base de données.	15 novembre 2018
Amazon RDS for Oracle prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent la version 12.2 d'Oracle.	13 novembre 2018
Amazon RDS for SQL Server prend en charge la fonction AlwaysOn	Amazon RDS for SQL Server prend en charge les groupes de disponibilité Always On. Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Microsoft SQL Server.	8 novembre 2018

Amazon RDS for PostgreSQL prend en charge l'accès au réseau sortant avec des serveurs DNS personnalisés	Amazon RDS for PostgreSQL prend en charge l'accès au réseau sortant avec des serveurs DNS personnalisés. Pour plus d'informations, consultez Utilisation d'un serveur DNS personnalisé pour l'accès au réseau sortant.	8 novembre 2018
Amazon RDS for MariaDB, MySQL et PostgreSQL prend en charge 32 Tio de stockage	Vous pouvez désormais créer des instances de bases de données Amazon RDS for MySQL, MariaDB et PostgreSQL avec jusqu'à 32 Tio de stockage. Pour plus d'informations, consultez Stockage d'instance de base de données.	7 novembre 2018
Amazon RDS for Oracle prend en charge les types de données étendus	Vous pouvez désormais activer les types de données étendus sur les instances de base de données Amazon RDS exécutant Oracle. Avec les types de données étendus, la taille maximale est de 32 767 octets pour les types de données VARCHAR2N VARCHAR2, et RAW. Pour plus d'informations, consultez Utilisation des types de données étendus.	6 novembre 2018

Amazon RDS for Oracle prend en charge les classes d'instance de base de données db.m5	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant Oracle qui utilisent les classes d'instances de base de données db.m5. Pour plus d'informations, consultez Classe d'instance de base de données.	2 novembre 2018
Migration d'Amazon RDS for Oracle depuis SE SE1 SE2 ou vers EE	Vous pouvez désormais migrer de n'importe quelle édition Oracle Database Standard Edition (SE SE1, ouSE2) vers Oracle Database Enterprise Edition (EE). Pour plus d'informations, consultez Migration d'une édition Oracle à une autre.	31 octobre 2018
Amazon RDS peut désormais arrêter les instances multi-AZ	Amazon RDS peut désormais arrêter une instance de base de données faisant partie d'un déploiement multi-AZ. Auparavant, la fonction arrêter une instance était limitée pour les instances multi-AZ. Pour plus d'informations, consultez Arrêt temporaire d'une instance de base de données Amazon RDS.	29 octobre 2018

Amazon RDS Performance Insights est disponible pour Amazon RDS for Oracle	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for Oracle. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	29 octobre 2018
Amazon RDS for PostgreSQL prend en charge PostgreSQL version 11 dans l'environnement en préversion de base de données	Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 11 dans l'environnement en préversion de base de données. Pour plus d'informations, consultez PostgreSQL version 11 sur Amazon RDS dans l'environnement en préversion de base de données.	25 octobre 2018
MySQL prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de base de données Amazon RDS exécutant MySQL version 8.0. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	23 octobre 2018
MariaDB prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent la version 10.3 de MariaDB. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	23 octobre 2018

Amazon RDS for Oracle prend en charge Oracle JVM	Amazon RDS for Oracle prend désormais en charge l'option Oracle Java Virtual Machine (JVM). Pour plus d'informations, consultez Oracle Java Virtual Machine.	16 octobre 2018
Groupe de paramètres personnalisés pour la restauration et la restauration à un instant dans le passé	Vous pouvez désormais spécifier un groupe de paramètres personnalisés lorsque vous restaurez un instantané ou procédez à une opération de restauration à un instant dans le passé. Pour plus d'informations, consultez Restauration à partir d'un instantané de base de données et Restauration d'une instance de base de données à une date spécifiée.	15 octobre 2018
Amazon RDS for Oracle prend en charge 32 Tio de stockage	Vous pouvez désormais créer des instances de base de données Oracle RDS avec un maximum de 32 Tio de stockage. Pour plus d'informations, consultez Stockage d'instance de base de données.	15 octobre 2018

Amazon RDS for MySQL prend en charge GTIDs	Amazon RDS for MySQL prend désormais en charge les identificateurs de transacti on globaux GTIDs (), qui sont uniques dans toutes les instances de base de données et dans une configuration de réplication. Pour plus d'informa tions, consultez Utilisation de la réplication GTID pour RDS for MySQL.	10 octobre 2018
MySQL 5.7.23, 5.6.41 et 5.5.61	Vous pouvez désormais créer des instances de base de données Amazon RDS qui exécutent les versions MySQL 5.7.23, 5.6.41 et 5.5.61. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	8 octobre 2018
Amazon RDS for Oracle prend en charge une nouvelle version de SQLT	Amazon RDS for Oracle prend désormais en charge SQLT version 12.2.180331. Pour de plus amples informations, veuillez consulter Oracle SQLT.	4 octobre 2018
Amazon RDS for PostgreSQ L prend désormais en charge l'authentification IAM	Amazon RDS for PostgreSQ L prend désormais en charge l'authentification IAM. Pour plus d'informations, consultez Authentification de base de données IAM pour MySQL et PostgreSQL.	27 septembre 2018

Vous pouvez activer la protection contre la suppression pour vos instances de base de données Amazon RDS	Lorsque vous activez la protection contre la suppression pour une instance de base de données, la base de données ne peut être supprimée par aucun utilisateur. Pour plus d'informations, consultez Suppression d'une instance de base de données.	26 septembre 2018
Amazon RDS for MySQL et Amazon RDS for MariaDB prennent désormais en charge les classes d'instances de bases de données db.m5	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant MySQL ou MariaDB qui utilisent les classes d'instances de bases de données db.m5. Pour plus d'informations, consultez Classe d'instance de base de données.	18 septembre 2018
Amazon RDS prend désormais en charge les mises à niveau à SQL Server 2017	Vous pouvez mettre à jour une instance de base de données existante vers SQL Server 2017 depuis n'importe quelle version, sauf SQL Server 2008. Pour mettre à niveau SQL Server 2008, mettez-vous d'abord à niveau avec une autre version. Pour plus d'informations, consultez Mise à niveau du moteur de base de données Microsoft SQL Server.	11 septembre 2018

Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 11 bêta 3 dans l'environnement en préversion de base de données	Dans cette édition, la taille de segment Write-Ahead Log (WAL) (wal_segment_size) est désormais définie à 64Mo. Pour plus d'informations sur la version PostgreSQL 11 Beta 3, consultez PostgreSQL 11 Beta 3 Released (Publication de PostgreSQL 11 Beta 3). Pour plus d'informations sur l'environnement en préversion de base de données, consultez Working with the Database Preview Environment (Travailler avec l'environnement en préversion de base de données).	7 septembre 2018
Guide de l'utilisateur Amazon Aurora	Le Amazon Aurora Guide de l'utilisateur présente tous les concepts Amazon Aurora et fournit des instructions sur l'utilisation des différentes fonctions, à la fois avec la console et via l'interface de ligne de commande. Le Amazon RDS Guide de l'utilisateur prend désormais en charge les moteurs de base de données non-Aurora.	31 août 2018

Amazon RDS Performance Insights est disponible pour RDS for MySQL	Amazon RDS Performance Insights est désormais disponible pour RDS for MySQL. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	28 août 2018
Édition compatible avec Aurora PostgreSQL prend désormais en charge Aurora Auto Scaling	L'Auto Scaling des réplicas Aurora est désormais disponible pour Édition compatible avec Aurora PostgreSQL. Pour plus d'informations, consultez Amazon Aurora auto scaling with Aurora replicas.	16 août 2018
Aurora Serverless pour Aurora MySQL	Aurora Serverless est une configuration à scalabilité automatique et à la demande pour Amazon Aurora. Pour plus d'informations, veuillez consulter Utilisation de Amazon Aurora Serverless.	9 août 2018
MySQL 5.7.22 et 5.6.40	Vous pouvez désormais créer des instances de bases de données Amazon RDS en exécutant les versions 5.7.22 et 5.6.40 de MySQL. Pour plus d'informations, consultez MySQL sur les versions Amazon RDS.	6 août 2018

Aurora est désormais disponible dans la région Chine (Ningxia)	Aurora MySQL et Aurora PostgreSQL sont désormais disponibles dans la région Chine (Ningxia). Pour plus d'informations, consultez Disponibilité de Amazon Aurora MySQL et Disponibilité de Amazon Aurora PostgreSQL.	6 août 2018
Amazon RDS for MySQL prend en charge la réplication retardée	Amazon RDS for MySQL prend désormais en charge la réplication retardée comme politique pour la reprise après sinistre. Pour plus d'informations, consultez Configuration de la réplication retardée avec MySQL.	6 août 2018
Amazon RDS Performance Insights est disponible pour Aurora MySQL	Amazon RDS Performance Insights est désormais disponible pour Aurora MySQL. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	6 août 2018
Intégration d'Amazon RDS Performance Insights à Amazon CloudWatch	Amazon RDS Performance Insights publie automatiquement des statistiques sur Amazon CloudWatch. Pour plus d'informations, consultez les métriques Performance Insights publiées sur CloudWatch.	6 août 2018

Recommandations concernant Amazon RDS	Amazon RDS fournit maintenant des recommandations automatisées pour les ressources de base de données. Pour de plus amples informations, veuillez consulter Utilisations des recommandations Amazon RDS.	25 juillet 2018
Copies instantanées incrémentielles entre les régions AWS	Amazon RDS prend en charge les copies instantanées incrémentielles entre AWS les régions, tant pour les instances chiffrées que non chiffrées. Pour plus d'informations, voir Copier des instantanés d'une AWS région à l'autre.	24 juillet 2018
Amazon RDS Performance Insights est disponible pour Amazon RDS for PostgreSQL	Amazon RDS Performance Insights est désormais disponible pour Amazon RDS for PostgreSQL. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	18 juillet 2018
Amazon RDS for Oracle prend en charge Oracle APEX version 5.1.4.v1	Amazon RDS for Oracle prend désormais en charge Oracle Application Express (APEX) version 5.1.4.v1. Pour de plus amples informations, veuillez consulter Oracle Application Express.	10 juillet 2018

Amazon RDS for Oracle prend en charge la publication de journaux sur Amazon CloudWatch Logs	Amazon RDS for Oracle prend désormais en charge la publication de données d'alerte, d'audit, de trace et de journal d'écoute dans un groupe CloudWatch de journaux dans Logs. Pour plus d'informations, consultez Publier des journaux Oracle sur Amazon CloudWatch Logs.	9 juillet 2018
MariaDB 10.2.15, 10.1.34 et 10.0.35	Vous pouvez désormais créer des instances de bases de données Amazon RDS en exécutant les versions 10.2.15, 10.1.34 et 10.0.35 de MySQL. Pour plus d'informations, consultez MariaDB sur les versions Amazon RDS.	5 juillet 2018
Aurora PostgreSQL 1.2 est disponible et compatible avec PostgreSQL 9.6.8	Aurora PostgreSQL 1.2 est désormais disponible et compatible avec PostgreSQL 9.6.8. Pour plus d'informations, consultez la Version 1.2.	27 juin 2018
Les réplicas en lecture pour Amazon RDS PostgreSQL prennent en charge les déploiements multi-AZ	Les réplicas en lecture RDS dans Amazon RDS PostgreSQL prennent désormais en charge plusieurs zones de disponibilité. Pour plus d'informations, consultez Utilisation des réplicas en lecture PostgreSQL.	25 juin 2018

Performance Insights est disponible pour Aurora PostgreSQL	Performance Insights est généralement disponible pour Aurora PostgreSQL, avec la prise en charge de la conservation étendue des données de performance. Pour de plus amples informations, veuillez consulter Utilisation d'Amazon RDS Performance Insights.	21 juin 2018
Aurora PostgreSQL est disponible dans la Région USA Ouest (Californie du Nord)	Aurora PostgreSQL est désormais disponible dans la région USA Ouest (Californie du Nord). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora PostgreSQL.	11 juin 2018
Amazon RDS for Oracle prend désormais en charge la configuration de l'UC	Amazon RDS for Oracle prend en charge la configuration du nombre de cœurs d'UC et du nombre de threads pour chaque cœur du processus d'une classe d'instance de base de données. Pour plus d'informations, consultez Configuration du processeur de la classe d'instance de base de données.	5 juin 2018

Mises à jour antérieures

Le tableau ci-après décrit les modifications importantes apportées dans chaque version du Guide de l'utilisateur Amazon RDS avant juin 2018.

Modification	Description	Date de modification
Amazon RDS for PostgreSQL prend désormais en charge PostgreSQL version 11 bêta 1 dans l'environnement en préversion de base de données	PostgreSQL version 11 Bêta 1 contient plusieurs améliorations qui sont décrites dans PostgreSQL 11 Bêta 1 Released. Pour plus d'informations sur l'environnement en préversion de base de données, consultez Utilisation de l'environnement de prévisualisation de base de données.	31 mai 2018
Amazon RDS for Oracle prend désormais en charge TLS versions 1.0 et 1.2	Amazon RDS for Oracle prend en charge le protocole TLS (Transport Layer Security) version 1.0 et 1.2. Pour plus d'informations, consultez TLSversions pour l'SSLoption Oracle.	30 mai 2018
Aurora MySQL prend en charge la publication de journaux sur Amazon CloudWatch Logs	Aurora MySQL prend désormais en charge la publication de données générales, lentes, d'audit et de journaux d'erreurs dans un groupe de CloudWatch journaux dans Logs. Pour plus d'informations, consultez Publier Aurora MySQL dans des CloudWatch journaux.	23 mai 2018
Environnement en préversion de base de données pour Amazon RDS PostgreSQL	Vous pouvez désormais lancer une nouvelle instance de Amazon RDS PostgreSQL en mode préversion. Pour plus d'informations sur l'environnement en préversion de base de données, consulte, Utilisation de l'environnement de prévisualisation de base de données.	22 mai 2018
Les instances de bases de données Amazon RDS for Oracle prennent	Les instances de base de données Oracle prennent désormais en charge les classes d'instances de base de données db.x1e et db.x1. Pour plus d'informations, consultez Classes d'instances de base de données et	22 mai 2018

Modification	Description	Date de modification
en charge les nouvelles classes d'instance de base de données	RDS pour les classes d'instance de base de données Oracle.	
Amazon RDS PostgreSQL prend désormais en charge postgres_fdw sur un réplica en lecture.	Vous pouvez désormais utiliser postgres_fdw pour vous connecter à un serveur distant à partir d'un réplica en lecture. Pour plus d'informations, consultez Utilisation de l'extension postgres_fdw pour accéder à des données externes.	17 mai 2018
Amazon RDS for Oracle prend désormais en charge la définition des paramètres sqlnet.ora	Vous pouvez désormais définir les paramètres sqlnet.ora avec Amazon RDS for Oracle. Pour plus d'informations, consultez Modification des propriétés de connexion à l'aide des paramètres sqlnet.ora.	10 mai 2018
Aurora PostgreSQL est disponible dans la région Asie-Pacifique (Séoul).	Aurora PostgreSQL est désormais disponible dans la région Asie-Pacifique (Séoul). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora PostgreSQL.	9 mai 2018
Aurora MySQL prend en charge le retour sur trace	Aurora MySQL permet désormais d'effectuer un retour sur trace d'un cluster de base de données à une heure spécifique, sans restaurer les données à partir d'une sauvegarde. Pour plus d'informations, consultez Retour en arrière d'un cluster de base de données Amazon Aurora.	9 mai 2018

Modification	Description	Date de modification
Aurora MySQL prend en charge la migration et la réplication chiffrées depuis une base de données MySQL externe	Aurora MySQL prend désormais en charge la migration et la réplication chiffrées depuis une base de données MySQL externe. Pour plus d'informations, consultez Migrer des données depuis une base de données MySQL externe vers un cluster de base de données Amazon Aurora MySQL et la Réplication entre Aurora et MySQL ou entre Aurora et un autre cluster de base de données Aurora.	25 avril 2018
Support de l'édition compatible avec Aurora PostgreSQL pour le protocole. Copy-on-Write	Vous pouvez désormais cloner des bases de données dans un cluster de bases de données Aurora PostgreSQL. Pour plus d'informations, consultez Clonage de base de données dans un cluster de base de données Aurora.	10 avril 2018
MariaDB 10.2.12, 10.1.31 et 10.0.34	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 10.2.12, 10.1.31 et 10.0.34 de MariaDB. Pour plus d'informations, consultez Versions de MariaDB sur Amazon RDS.	21 mars 2018
Prise en charge d'Aurora PostgreSQL pour les nouvelles régions	Aurora PostgreSQL est désormais disponible dans les régions UE(Londres) et Asie Pacifique (Singapour). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora PostgreSQL.	13 mars 2018
MySQL 5.7.21, 5.6.39 et 5.5.59	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 5.7.21, 5.6.39 et 5.5.59. de MySQL. Pour plus d'informations, consultez Versions de MySQL sur Amazon RDS.	9 mars 2018

Modification	Description	Date de modification
Amazon RDS for Oracle prend désormais en charge les services de données Oracle REST	Amazon RDS for Oracle prend en charge Oracle REST Data Services dans le cadre de l'option APEX. Pour plus d'informations, consultez Oracle Application Express (APEX).	9 mars 2018
L'édition compatible avec Amazon Aurora MySQL est disponible dans une nouvelle région AWS	Aurora MySQL est désormais disponible dans la région Asie-Pacifique (Singapour). Pour obtenir la liste complète des AWS régions pour Aurora MySQL, consultez Disponibilité pour Amazon Aurora MySQL.	6 mars 2018
Les instances de base de données Amazon RDS s'exécutant sur Microsoft SQL Server prennent en charge la capture de données modifiées (CDC)	Les instances de base de données exécutant Amazon RDS for Microsoft SQL Server prennent à présent en charge la capture de données modifiées (CDC). Pour plus d'informations, consultez Modifier le support de capture de données pour les instances de base de données Microsoft SQL Server.	6 février 2018
Aurora MySQL prend en charge une nouvelle version majeure	Vous pouvez désormais créer des clusters de bases de données Aurora MySQL en exécutant la version 5.7 de MySQL. Pour plus d'informations, consultez Amazon Aurora MySQL Database Engine Updates 2018-02-06 Mises à jour de moteur de base de données Amazon Aurora MySQL de 2018-02-06à.	6 février 2018

Modification	Description	Date de modification
Publier les journaux MySQL et MariaDB sur Amazon Logs CloudWatch	Vous pouvez désormais publier les données des journaux MySQL et MariaDB dans Logs. CloudWatch Pour plus d'informations, consultez Publication de journaux MySQL sur Amazon CloudWatch Logs et Publier des logs MariaDB sur Amazon Logs CloudWatch .	17 janvier 2018
Prise en charge Multi-AZ pour les réplicas en lecture	Vous pouvez désormais créer un réplica en lecture en tant qu'instance de base de données Multi-AZ. Amazon RDS crée une instance de secours de votre réplica dans une autre zone de disponibilité pour la prise en charge du basculement pour le réplica. La création de votre réplica en lecture en tant qu'instance de base de données multi-AZ est indépendante du fait que la base de données source soit ou non une instance de base de données multi-AZ. Pour plus d'informations, consultez Utilisation des réplicas en lecture d'instance de base de données.	11 janvier 2018
Amazon RDS for MariaDB prend en charge une nouvelle version majeure	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent la version 10.2 de MariaDB. Pour plus d'informations, consultez Prise en charge de MariaDB 10.2 sur Amazon RDS.	3 janvier 2018
Amazon Aurora Édition compatible avec PostgreSQL disponible dans une nouvelle région AWS	Aurora PostgreSQL est maintenant disponible dans la région EU (Paris). Pour obtenir la liste complète des AWS régions pour Aurora PostgreSQL, consultez Disponibilité pour Amazon Aurora PostgreSQL.	22 décembre 2017

Modification	Description	Date de modification
Aurora PostgreSQL prend en charge de nouveaux types d'instance	Aurora PostgreSQL prend maintenant en charge de nouveaux types d'instance. Pour obtenir la liste complète des types d'instance, consultez Choisir la classe d'instance de base de données.	20 décembre 2017
L'édition compatible avec Amazon Aurora MySQL est disponible dans une nouvelle région AWS	Aurora MySQL est maintenant disponible dans la région EU (Paris). Pour obtenir la liste complète des AWS régions pour Aurora MySQL, consultez Disponibilité pour Amazon Aurora MySQL.	18 décembre 2017
Aurora MySQL prend en charge les jointures par hachage	Cette fonctionnalité peut améliorer les performances de requêtes lorsque vous devez joindre une grande quantité de données au moyen d'une équijointure. Pour plus d'informations, consultez Traitement des raccords de hachage dans Aurora MySQL.	11 décembre 2017
Aurora MySQL prend en charge les fonctions natives pour appeler les fonctions AWS Lambda	Vous pouvez appeler les fonctions natives `lambda_sync` et `lambda_async` lorsque vous utilisez Aurora MySQL. Pour plus d'informations, consultez Invocation d'une fonction lambda à partir d'un cluster de base de données Amazon Aurora MySQL.	11 décembre 2017
Ajout de l'éligibilité HIPAA à Aurora PostgreSQL	Aurora PostgreSQL prend désormais en charge les applications conformes à la loi HIPAA. Pour plus d'informations, consultez Utilisation de Amazon Aurora PostgreSQL.	6 décembre 2017

Modification	Description	Date de modification
AWS Régions supplémentaires disponibles pour Amazon Aurora avec compatibilité avec PostgreSQL	Amazon Aurora compatible avec PostgreSQL est désormais disponible dans quatre nouvelles régions. AWS Pour plus d'informations, consultez Disponibilité pour Amazon Aurora PostgreSQL.	22 novembre 2017
Modification du stockage pour les instances de bases de données Amazon RDS exécutant Microsoft SQL Serve	Vous pouvez désormais modifier le stockage de vos instances de bases de données Amazon RDS exécutant SQL Server. Pour plus d'informations, consultez Modification d'une RDS instance de base de données Amazon.	21 novembre 2017
Amazon RDS prend en charge 16 Tio de stockage pour les moteurs basés sur Linux	Vous pouvez désormais créer des instances de bases de données MySQL, MariaDB, PostgreSQL et Oracle RDS avec un maximum de 16 Tio de stockage. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS.	21 novembre 2017
Amazon RDS prend en charge une augmentation rapide du stockage	Vous pouvez désormais ajouter en quelques minutes du stockage aux instances de bases de données MySQL, MariaDB, PostgreSQL et Oracle RDS. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS.	21 novembre 2017
Amazon RDS prend en charge MariaDB versions 10.1.26 et 10.0.32	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 10.1.26 et 10.0.32 de MariaDB. Pour plus d'informations, consultez Versions de MariaDB sur Amazon RDS.	20 novembre 2017

Modification	Description	Date de modification
Amazon RDS for Microsoft SQL Server prend désormais en charge de nouvelles classes d'instances de bases de données	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant SQL Server qui utilisent les classes d'instances de bases de données db.r4 et db.m4.16xlarge. Pour plus d'informations, consultez prise en charge des classes d'instance de base de données pour Microsoft SQL Server.	20 novembre 2017
Amazon RDS for MySQL et MariaDB prend désormais en charge de nouvelles classes d'instances de bases de données	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant MySQL et MariaDB qui utilisent les classes d'instances de bases de données db.r4, db.m4.16xlarge, db.t2.xlarge et db.t2.2xlarge. Pour plus d'informations, consultez Classes d'instances de base de données .	20 novembre 2017
SQL Server 2017	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant Microsoft SQL Server 2017. Vous pouvez également créer des instances de base de données exécutant SQL Server 2016 SP1 CU5. Pour de plus amples informations, veuillez consulter Amazon RDS for Microsoft SQL Server.	17 novembre 2017
Restauration des sauvegardes MySQL d'Amazon S3	Vous pouvez désormais créer une sauvegarde de votre base de données sur site, la stocker sur Amazon S3, puis restaurer le fichier de sauvegarde sur une nouvelle instance de base de données Amazon RDS qui exécute MySQL. Pour plus d'informations, consultez Restauration d'une sauvegarde dans une instance de base de données Amazon RDS for MySQL.	17 novembre 2017

Modification	Description	Date de modification
Auto Scaling avec réplicas Aurora	Amazon Aurora MySQL prend désormais en charge l'Auto Scaling Aurora. L'Auto Scaling Aurora ajuste dynamiquement le nombre de réplicas Aurora sur la base des réductions et augmentations de connectivité ou de charge de travail. Pour plus d'informations, consultez Amazon Aurora Auto Scaling with Aurora replicas.	17 novembre 2017
Prise en charge de l'édition par défaut d'Oracle	Les instances de bases de données Amazon RDS for Oracle prennent maintenant en charge la configuration de l'édition par défaut de l'instance de base de données. Pour plus d'informations, consultez Définition de l'édition par défaut d'une instance de base de données.	3 novembre 2017
Validation des fichiers d'instance de base de données Oracle	Les instances de bases de données Amazon RDS for Oracle prend maintenant en charge la validation des fichiers d'instance de base de données avec l'utilitaire de validation logique Oracle Recovery Manager (RMAN). Pour plus d'informations, consultez Validation des fichiers de base de données dans RDS Oracle .	3 novembre 2017
Management Agent pour OEM 13c	Les instances de bases de données Amazon RDS for Oracle prennent désormais en charge Management Agent pour Oracle Enterprise Manager (OEM) Cloud Control 13c. Pour plus d'informations, consultez Oracle Management Agent pour Enterprise Manager Cloud Control.	1 novembre 2017
Reconfiguration du stockage pour les instantanés Microsoft SQL Server	Vous pouvez désormais reconfigurer le stockage lorsque vous restaurez un instantané sur une instance de base de données Amazon RDS exécutant Microsoft SQL Server. Pour plus d'informations, consultez Restauration vers une instance de base de données.	26 octobre 2017

Modification	Description	Date de modification
Lecture anticipée asynchrone des clés pour Édition compatible avec Aurora MySQL	La lecture anticipée asynchrone des clés (AKP) améliore les performances des jointures d'index non mises en cache en récupérant au préalable les clés en mémoire avant qu'elles ne soient nécessaires. Pour plus d'informations, consultez Utilisation de la pré-extraction de clé asynchrone dans Amazon Aurora.	26 octobre 2017
MySQL 5.7.19, 5.6.37 et 5.5.57	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 5.7.19, 5.6.37 et 5.5.57 de MySQL. Pour plus d'informations, consultez Versions de MySQL sur Amazon RDS.	25 octobre 2017
Disponibilité générale d'Amazon Aurora avec compatibilité PostgreSQL	Amazon Aurora avec compatibilité PostgreSQL permet de configurer, de gérer et de dimensionner de façon simple et économique vos déploiements PostgreSQL nouveaux et existants, vous permettant ainsi de vous concentrer sur votre activité et vos applications. Pour plus d'informations, consultez Utilisation de Amazon Aurora PostgreSQL.	24 octobre 2017
Les instances de bases de données Amazon RDS for Oracle prennent en charge les nouvelles classes d'instance de base de données	Les instances de base de données Amazon RDS for Oracle prennent désormais en charge les classes d'instances à mémoire optimisée de nouvelle génération (db.r4). Les instances de base de données Amazon RDS for Oracle prennent désormais également en charge les nouvelles classes d'instance suivantes de la génération actuelle : db.m4.16xlarge, db.t2.xlarge et db.t2.2xlarge. Pour plus d'informations, consultez Classes d'instances de base de données et RDS pour les classes d'instance de base de données Oracle.	23 octobre 2017

Modification	Description	Date de modification
Nouvelle fonction	Vos instances réservées nouvelles et existantes peuvent maintenant couvrir plusieurs tailles dans la même classe d'instance de base de données. Des instances réservées de taille flexible sont disponibles pour les instances de base de données ayant la même AWS région, le même moteur de base de données et la même famille d'instances, et pour toutes les configurations AZ. Les instances réservées de taille flexible sont disponibles pour les moteurs de base de données suivants : Amazon Aurora, MariaDB, MySQL, Oracle (Réutilisez vos licences), PostgreSQL. Pour plus d'informations, consultez Instances de base de données réservées de taille flexible.	11 octobre 2017
Nouvelle fonction	Vous pouvez désormais utiliser l'option Oracle SQLT pour régler une instruction SQL afin d'obtenir des performances optimales. Pour plus d'informations, consultez Oracle SQLT.	22 septembre 2017
Nouvelle fonction	Si vous avez des instantanés de base de données Guides de vos instances de bases de données Amazon RDS for Oracle, vous pouvez désormais les mettre à niveau vers une version ultérieure du moteur de base de données Oracle. Pour plus d'informations, consultez Mise à niveau d'un instantané de base de données Oracle.	20 septembre 2017
Nouvelle fonction	Vous pouvez désormais utiliser Oracle Spatial pour stocker, récupérer, mettre à jour et effectuer des requêtes sur les données spatiales dans vos instances de bases de données Amazon RDS exécutant Oracle. Pour plus d'informations, consultez Oracle Spatial.	15 septembre 2017

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais utiliser Oracle Locator pour prendre en charge les applications basées sur des services sans fil et sur Internet, ainsi que les solutions GIS basées sur le partenariat avec vos instances de bases de données Amazon RDS exécutant Oracle. Pour plus d'informations, consultez Oracle Locator.	15 septembre 2017
Nouvelle fonction	Vous pouvez désormais utiliser Oracle Multimedia pour stocker, gérer et récupérer des images, des fichiers audio et vidéo, ainsi que d'autres données multimédias hétérogènes dans vos instances de bases de données Amazon RDS exécutant Oracle.	15 septembre 2017
Nouvelle fonctionnalité	Vous pouvez désormais exporter les journaux d'audit de vos clusters de bases de données Amazon Aurora MySQL vers Amazon CloudWatch Logs. Pour plus d'informations, consultez la section Publication des journaux Aurora MySQL sur Amazon CloudWatch Logs.	14 septembre 2017
Nouvelle fonction	Amazon RDS prend désormais en charge plusieurs versions d'Oracle Application Express (APEX) pour vos instances de bases de données exécutant Oracle. Pour plus d'informations, consultez Oracle Application Express (APEX).	13 septembre 2017

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais utiliser Amazon Aurora pour migrer un instantané de base de données chiffré ou non chiffré, ou une instance de base de données MySQL vers un cluster de bases de données Aurora MySQL chiffré. Pour plus d'informations, consultez Migration d'un instantané RDS for MySQL vers Aurora et Migration de données d'une instance de base de données MySQL vers un cluster de base de données Amazon Aurora MySQL en utilisant un réplica en lecture Aurora.	5 septembre 2017
Nouvelle fonction	Vous pouvez utiliser les bases de données Amazon RDS for Microsoft SQL Server afin de développer des applications conformes à la loi HIPAA. Pour plus d'informations, consultez Prise en charge du programme de conformité pour les instances de bases de données Microsoft SQL Server.	31 août 2017
Nouvelle fonction	Vous pouvez maintenant utiliser les bases de données Amazon RDS for MariaDB afin de développer des applications conformes à la loi HIPAA. Pour plus d'informations, consultez Amazon RDS for MariaDB.	31 août 2017
Nouvelle fonction	Vous pouvez désormais créer des instances de bases de données Amazon RDS exécutant Microsoft SQL Server avec un stockage alloué pouvant atteindre 16 Tio et des IOPS allouées pour stocker des plages de 1:1–50:1. Pour plus d'informations, consultez Stockage d'instance de base de données Amazon RDS.	22 août 2017

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais utiliser les déploiements Multi-AZ pour les instances de bases de données exécutant Microsoft SQL Server dans la région UE (Francfort). Pour plus d'informations, consultez Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.	3 août 2017
Nouvelle fonction	Vous pouvez désormais créer des instances de bases de données Amazon RDS qui exécutent les versions 10.1.23 et 10.0.31 de MariaDB. Pour plus d'informations, consultez Versions de MariaDB sur Amazon RDS.	17 juillet 2017
Nouvelle fonctionnalité	Amazon RDS prend désormais en charge Microsoft SQL Server Enterprise Edition avec le modèle de licence incluse dans toutes les AWS régions. Pour de plus amples informations, veuillez consulter Gestion des licences Microsoft SQL Server sur Amazon RDS.	13 juillet 2017
Nouvelle fonction	Amazon RDS for Oracle prend désormais en charge les grandes pages du noyau Linux pour obtenir une base de données plus évolutive. L'utilisation de grandes pages se traduit par des tables de page réduites et une réduction du temps UC de gestion de la mémoire, ce qui augmente les performances des instances de bases de données volumineuses. Vous pouvez utiliser les grandes pages avec vos instances de bases de données Amazon RDS qui exécutent toutes les éditions des versions 12.1.0.2 et 11.2.0.4 d'Oracle. Pour plus d'informations, consultez Activation de HugePages pour une instance RDS for Oracle.	7 juillet 2017

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour la prise en charge du chiffrement au repos (EAR) pour les classes d'instance de base de données db.t2.small et db.t2.medium pour tous les moteurs de base de données autres qu'Aurora. Pour plus d'informations, consultez Disponibilité du chiffrement Amazon RDS.	27 juin 2017
Nouvelle fonction	Mise à jour pour prendre en charge Amazon Aurora dans la région Europe (Francfort). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora MySQL.	16 juin 2017
Nouvelle fonctionnalité	Vous pouvez désormais spécifier un groupe d'options lorsque vous copiez un instantané de base de données entre plusieurs AWS régions. Pour de plus amples informations, veuillez consulter Considérations relatives aux groupes d'options.	12 juin 2017
Nouvelle fonctionnalité	Vous pouvez désormais copier des instantanés de base de données créés à partir d'instances de base de données spécialisées dans différentes AWS régions. Vous pouvez copier les instantanés à partir d'instances de bases de données qui utilisent Oracle TDE, Microsoft SQL Server TDE et Multi-AZ avec mise en miroir de Microsoft SQL Server. Pour plus d'informations, consultez Copie d'un instantané de base de données.	12 juin 2017
Nouvelle fonction	Amazon Aurora vous permet désormais de copier rapidement et économiquement toutes vos bases de données dans un cluster de bases de données Amazon Aurora. Pour plus d'informations, consultez Clonage de base de données dans un cluster de base de données Aurora.	12 juin 2017

Modification	Description	Date de modification
Nouvelle fonctionnalité	Amazon RDS prend désormais en charge Microsoft SQL Server 2016. SP1 CU2 Pour de plus amples informations, veuillez consulter Amazon RDS for Microsoft SQL Server.	7 juin 2017
Version préliminaire	Version préliminaire publique d'Amazon Aurora avec compatibilité PostgreSQL. Pour plus d'informations, consultez Utilisation de Amazon Aurora PostgreSQL.	19 avril 2017
Nouvelle fonction	Amazon Aurora vous permet désormais d'exécuter une opération ALTER TABLE tbl_name ADD COLUMN col_name column_definition pratiquement instantanément. L'opération s'effectue sans nécessiter la copie de la table et sans impact matériel sur les autres instructions DML. Pour plus d'informations, consultez Modification des tableaux dans Amazon Aurora à l'aide de Fast DDL.	5 avril 2017
Nouvelle fonction	Nous avons ajouté une nouvelle commande de surveillance, SHOW VOLUME STATUS, qui permet d'afficher le nombre de nœuds et de disques dans un volume. Pour plus d'informations, consultez Affichage du statut de volume pour un cluster de base de données Aurora.	5 avril 2017
Nouvelle fonctionnalité	Vous pouvez désormais utiliser votre propre logique personnalisée dans les fonctions de vérification de mot de passe personnalisé pour Oracle sur Amazon RDS. Pour plus d'informations, consultez Création de fonctions personnalisées pour vérifier les mots de passe.	21 mars 2017

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais accéder à vos fichiers de journalisation Redo en ligne et archivés sur vos instances de bases de données Oracle sur Amazon RDS. Pour plus d'informations, consultez Accès aux journaux de reprise en ligne et archivés.	21 mars 2017
Nouvelle fonction	Vous pouvez désormais copier des instantanés de cluster de bases de données chiffrés et non chiffrés entre les comptes d'une même région. Pour plus d'informations, consultez Copying a DB Cluster Snapshot Across Accounts (Copier une instantané de cluster de base de données entre plusieurs comptes).	7 mars 2017
Nouvelle fonction	Vous pouvez désormais partager des instantanés de cluster de bases de données chiffrés et non chiffrés entre les comptes d'une même région. Pour plus d'informations, consultez Sharing a DB Cluster Snapshot (Partager un instantané de cluster de base de données).	7 mars 2017
Nouvelle fonction	Vous pouvez désormais répliquer des clusters de base de données Amazon Aurora MySQL chiffrés pour créer des réplicas Aurora entre régions. Pour plus d'informations, consultez la section Réplication de clusters de bases de données Aurora MySQL entre AWS régions.	7 mars 2017
Nouvelle fonction	Vous pouvez désormais demander à ce que toutes les connexions de votre instance de base de données exécutant Microsoft SQL Server utilisent SSL (Secure Sockets Layer). Pour plus d'informations, consultez Utilisation SSL avec une instance de base de données Microsoft SQL Server.	27 février 2017

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais définir votre fuseau horaire local sur l'un 15 fuseaux horaires supplémentaires. Pour plus d'informations, consultez Fuseaux horaires pris en charge.	27 février 2017
Nouvelle fonction	Vous pouvez désormais utiliser la procédure Amazon RDS `msdb.dbo.rds_shrink_tempdbfile` pour réduire la base de données tempdb sur vos instances de base de données exécutant Microsoft SQL Server. Pour plus d'informations, consultez Réduction de la base de données tempdb.	17 février 2017
Nouvelle fonction	Vous pouvez désormais compresser vos fichiers de sauvegarde lorsque vous exportez votre base de données Microsoft SQL Server Enterprise and Standard Edition d'une instance de base de données Amazon RDS vers Amazon S3. Pour plus d'informations, consultez Compression des fichiers de sauvegarde.	17 février 2017
Nouvelle fonction	Amazon RDS prend désormais en charge des serveurs DNS personnalisés pour résoudre des noms DNS utilisés dans l'accès réseau sortant sur vos instances de bases de données Oracle. Pour plus d'informations, consultez Configuration d'un DNS serveur personnalisé.	26 janvier 2017
Nouvelle fonction	Amazon RDS prend désormais en charge la création d'un réplica en lecture chiffré dans une autre région. Pour plus d'informations, voir Création d'un réplica en lecture dans une autre Région AWS et Créer DBInstance ReadReplica.	23 janvier 2017

Modification	Description	Date de modification
Nouvelle fonction	Amazon RDS prend désormais en charge la mise à niveau d'un instantané de base de données MySQL de MySQL version 5.1 vers MySQL 5.5.	20 janvier 2017
Nouvelle fonction	Amazon RDS prend désormais en charge la copie d'un instantané de base de données chiffré vers une autre région pour les moteurs de base de données MariaDB, MySQL, Oracle, PostgreSQL et Microsoft SQL Server. Pour plus d'informations, consultez Copie d'un instantané de base de données et Copier DBSnapshot.	20 décembre 2016
Nouvelle fonction	Amazon Aurora MySQL prend désormais en charge l'indexation spatiale. L'indexation spatiale améliore les performances des requêtes sur des jeux de données volumineux pour les requêtes qui utilisent des données spatiales. Pour plus d'informations, consultez Amazon Aurora MySQL et données spatiales.	14 décembre 2016
Nouvelle fonctionnalité	Amazon RDS prend désormais en charge l'accès réseau sortant sur vos instances de bases de données exécutant Oracle. Vous pouvez utiliser utl_http, utl_tcp et utl_smtp pour vous connecter au réseau à partir de votre instance de base de données. Pour plus d'informations, consultez Configuration de l'accès UTL_HTTP à l'aide de certificats et d'un portefeuille Oracle.	5 décembre 2016
Nouvelle fonction	Amazon RDS a annulé la prise en charge de la version 5.1 de MySQL. Cependant, vous pouvez restaurer des instantanés MySQL 5.1 sur une instance MySQL 5.5. Pour plus d'informations, consultez Moteurs de stockage pris en charge pour RDS for MySQL.	15 novembre 2016

Modification	Description	Date de modification
Nouvelle fonctionnalité	Amazon RDS prend désormais en charge Microsoft SQL Server 2016 RTM CU2. Pour de plus amples informations, veuillez consulter Amazon RDS for Microsoft SQL Server.	4 novembre 2016
Nouvelle fonction	Amazon RDS prend désormais en charge les mises à niveau de version majeure pour les instances de bases de données exécutant Oracle. Vous pouvez désormais mettre à niveau vos instances de bases de données de 11g vers 12c. Pour plus d'informations, consultez Mise à niveau du moteur de base de données RDS for Oracle.	2 novembre 2016
Nouvelle fonction	Vous pouvez désormais créer des instances de base de données exécutant Microsoft SQL Server 2014 Enterprise Edition. Amazon RDS prend désormais en charge SQL Server 2014 SP2 pour toutes les éditions et toutes les régions. Pour de plus amples informations, veuillez consulter Amazon RDS for Microsoft SQL Server.	25 octobre 2016
Nouvelle fonctionnalité	Amazon Aurora MySQL s'intègre désormais à d'autres AWS services : vous pouvez charger du texte ou des données XML dans une table à partir d'un compartiment Amazon S3, ou appeler une AWS Lambda fonction à partir du code de base de données. Pour plus d'informations, consultez la section Intégration d'Aurora MySQL à d'autres AWS services.	18 octobre 2016

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez maintenant accéder à la base de données tempdb sur vos instances de bases de données Amazon RDS exécutant Microsoft SQL Server. Vous pouvez accéder à la base de données tempdb à l'aide de Transact-SQL via Microsoft SQL Server Management Studio (SSMS) ou via toute autre application cliente SQL standard. Pour plus d'informations, consultez Accès à la base de données tempdb sur les instances de base de données Microsoft SQL Server sur Amazon RDS.	29 septembre 2016
Nouvelle fonction	Vous pouvez désormais utiliser le package UTL_MAIL avec vos instances de bases de données Amazon RDS qui exécutent Oracle. Pour plus d'informations, consultez Oracle UTL _ MAIL.	20 septembre 2016
Nouvelles fonctions	Vous pouvez maintenant définir le fuseau horaire de vos nouvelles instances de bases de données Microsoft SQL Server sur un fuseau horaire local, correspondant à celui de vos applications. Pour plus d'informations, consultez Fuseau horaire local pour les instances de base de données Microsoft SQL Server.	19 septembre 2016
Nouvelle fonctionnalité	Vous pouvez désormais utiliser l'option Oracle Label Security pour contrôler l'accès aux lignes individuelles des tables dans vos instances de bases de données Amazon RDS exécutant Oracle Database 12c. Avec Oracle Label Security, vous pouvez garantir la conformité réglementaire à un modèle d'administration basé sur une politique et vous assurer que l'accès aux données sensibles est limité aux utilisateurs disposant du niveau d'autorisation approprié. Pour plus d'informations, consultez Oracle Label Security.	8 septembre 2016

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais vous connecter à un cluster de bases de données Amazon Aurora à l'aide du point de terminaison de lecteur, qui équilibre la charge des connexions entre les réplicas Aurora qui sont disponibles dans le cluster de bases de données. À mesure que les clients demandent de nouvelles connexions au point de terminaison de lecteur, Aurora répartit les demandes de connexion entre les réplicas Aurora dans le cluster de bases de données. Cette fonctionnalité peut aider à équilibrer votre charge de travail entre les différents réplicas Aurora de votre cluster de base de données. Pour plus d'informations, consultez Points de terminaison Amazon Aurora.	8 septembre 2016
Nouvelle fonction	Vous pouvez désormais prendre en charge le contrôle du cloud Oracle Enterprise Manager sur vos instances de bases de données Amazon RDS qui exécutent Oracle. Vous pouvez activer l'agent de gestion sur vos instances de bases de données et partager les données avec votre Oracle Management Service (OMS). Pour plus d'informations, consultez Oracle Management Agent pour Enterprise Manager Cloud Control.	1 septembre 2016
Nouvelle fonction	Cette version ajoute la prise en charge permettant d'obtenir un ARN pour une ressource. Pour plus d'informations, consultez Obtenir un compte existant ARN pour Amazon RDS.	23 août 2016
Nouvelle fonction	Vous pouvez maintenant affecter jusqu'à 50 balises pour chacune des ressources Amazon RDS afin de gérer vos ressources et suivre les coûts. Pour plus d'informations, consultez Marquage d' Amazon RDS.	19 août 2016

Modification	Description	Date de modification
Nouvelle fonction	Amazon RDS prend désormais en charge le modèle « Licence incluse » pour Oracle Standard Edition Two. Pour plus d'informations, consultez Création d'une instance de base de données Amazon RDS. Vous pouvez désormais modifier le modèle de licence de vos instances de bases de données Amazon RDS exécutant Microsoft SQL Server et Oracle. Pour plus d'informations, consultez Gestion des licences Microsoft SQL Server sur Amazon RDS et RDSpour les options de licence Oracle.	5 août 2016
Nouvelle fonctionnalité	Amazon RDS prend désormais en charge les sauvegarde et restauration natives pour les bases de données Microsoft SQL Server à l'aide de fichiers de sauvegarde complète (fichiers .bak). Vous pouvez désormais facilement migrer des bases de données SQL Server vers Amazon RDS, et importer et exporter des bases de données dans un seul fichier facilement transportable, en utilisant Amazon S3 pour le stockage et le chiffrement. AWS KMS Pour de plus amples informations, veuillez consulter Importation et exportation de bases de données SQL Server à l'aide de la sauvegarde et de la restauration natives.	27 juillet 2016

Modification	Description	Date de modification
Nouvelle fonction	Vous pouvez désormais copier les fichiers source à partir d'une base de données MySQL dans un compartiment Amazon Simple Storage Service (Amazon S3), puis restaurer un cluster DB Amazon Aurora à partir de ces fichiers. Cette option peut être considérablement plus rapide que la migration des données à l'aide de `mysqldump` . Pour plus d'informations, consultez Migration des données d'une base de données MySQL externe vers un cluster de bases de données Aurora MySQL.	20 juillet 2016
Nouvelle fonctionnalité	Vous pouvez désormais restaurer un instantané de cluster de base de données Amazon Aurora non chiffré pour créer un cluster de base de données Amazon Aurora chiffré en incluant une clé de chiffrement AWS Key Management Service (AWS KMS) lors de l'opération de restauration. Pour de plus amples informations, veuillez consulter Chiffrer des ressources Amazon RDS.	30 juin 2016
Nouvelle fonctionnalité	Vous pouvez employer l'utilitaire Oracle Repository Creation Utility (RCU) pour créer un référentiel sur Amazon RDS for Oracle. Pour plus d'informations, consultez Utilisation de l'utilitaire Oracle Repository Creation Utility sur RDS for Oracle .	17 juin 2016
Nouvelle fonction	Ajout de la prise en charge des réplicas en lecture PostgreSQL entre régions. Pour plus d'informations, consultez Création d'un réplica en lecture dans une autre Région AWS.	16 juin 2016

Modification	Description	Date de modification
Nouvelle fonctionnalité	Vous pouvez désormais utiliser le AWS Management Console pour ajouter facilement le Multi-AZ avec mise en miroir à une instance de base de données Microsoft SQL Server. Pour de plus amples informations, veuillez consulter Ajout de Multi-AZ à une instance de base de données Microsoft SQL Server.	9 juin 2016
Nouvelle fonctionnalité	Vous pouvez désormais utiliser les déploiements Multi-AZ à l'aide de la mise en miroir SQL Server dans les Régions supplémentaires suivantes : Asie-Pacifique (Sydney), Asie-Pacifique (Tokyo) et Amérique du Sud (Sao Paulo). Pour de plus amples informations, veuillez consulter Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.	9 juin 2016
Nouvelle fonction	Mis à jour pour prendre en charge la version 10.1 de MariaDB. Pour plus d'informations, consultez Amazon RDS for MariaDB.	1 juin 2016
Nouvelle fonction	Mise à jour pour prendre en charge les clusters de bases de données Amazon Aurora entre régions qui sont des réplicas en lecture. Pour plus d'informations, consultez Réplication de clusters de base de données Aurora MySQL entre régions AWS.	1 juin 2016
Nouvelle fonction	La surveillance améliorée est disponible pour les instances de bases de données Oracle. Pour plus d’informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée et Modification d'une RDS instance de base de données Amazon.	27 mai 2016

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge le partage Guide d'instantanés pour les instantanés de cluster de base de données Amazon Aurora. Pour plus d'informations, consultez Sharing a DB Cluster Snapshot (Partager un instantané de cluster de base de données).	18 mai 2016
Nouvelle fonction	Vous pouvez désormais utiliser le plugin d'audit MariaDB pour vous connecter à l'activité de base de données sur les instances de bases de données MariaDB et MySQL. Pour plus d'informations, consultez Options pour le moteur de base de données MariaDB et Options pour les instances My SQL DB.	27 avril 2016
Nouvelle fonction	Les mises à niveau de version majeure sur place sont désormais disponibles pour la mise à niveau de la version 5.6 vers la version 5.7 de MySQL. Pour plus d'informations, consultez Mises à niveau du moteur de base de données RDS pour MySQL.	26 avril 2016
Nouvelle fonction	La supervision améliorée est disponible pour les instances de bases de données Microsoft SQL Server. Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.	22 avril 2016
Nouvelle fonctionnalité	Mis à jour pour fournir un affichage Amazon Aurora Clusters dans la console Amazon RDS. Pour plus d'informations, consultez Affichage d'un cluster de base de données Aurora.	1 avril 2016
Nouvelle fonction	Mise à jour pour prendre en charge SQL Server Multi-AZ avec la mise en miroir dans la région Asie-Pacifique (Séoul). Pour plus d'informations, consultez Déploiements multi-AZ pour Amazon pour RDS Microsoft Server SQL.	31 mars 2016

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge Amazon Aurora Multi-AZ avec la mise en miroir dans la région Asie-Pacifique (Séoul). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora MySQL.	31 mars 2016
Nouvelle fonction	Les instances de bases de données PostgreSQL peuvent exiger des connexions pour utiliser SSL. Pour plus d'informations, consultez Utilisation de SSL avec une instance de base de données PostgreSQL.	25 mars 2016
Nouvelle fonction	La supervision améliorée est disponible pour les instances de bases de données PostgreSQL. Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.	25 mars 2016
Nouvelle fonction	Les instances de bases de données Microsoft SQL Server peuvent désormais utiliser l'authentification Windows pour l'authentification utilisateur. Pour plus d'informations, consultez Utilisation d'Active Directory AWS géré avec RDS pour SQL Server.	23 mars 2016
Nouvelle fonction	La supervision améliorée est disponible dans la région Asie-Pacifique (Séoul). Pour plus d'informations, consultez Surveillance des métriques du système d'exploitation à l'aide de la Surveillance améliorée.	16 mars 2016
Nouvelle fonction	Vous pouvez désormais personnaliser l'ordre dans lequel les réplicas Aurora sont promus sur l'instance principale lors d'un basculement. Pour plus d'informations, consultez Tolérance aux pannes d'un cluster de base de données Aurora.	14 mars 2016

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge le chiffrement lors de la migration vers un cluster DB Aurora. Pour plus d'informations, consultez Migration de données à partir d'un cluster de base de données Aurora.	2 mars 2016
Nouvelle fonction	Mise à jour pour prendre en charge le fuseau horaire local pour les clusters de bases de données Aurora. Pour plus d'informations, consultez Fuseau horaire local pour les clusters de base de données Aurora.	1 mars 2016
Nouvelle fonction	Mise à jour pour ajouter la prise en charge pour MySQL version 5.7 pour les classes d'instance de base de données Amazon RDS de la génération actuelle.	22 février 2016
Nouvelle fonctionnalité	Mise à jour pour prendre en charge les classes d'instance de base de données db.r3 et db.t2 dans la AWS GovCloud région (ouest des États-Unis).	11 février 2016
Nouvelle fonction	Mise à jour pour prendre en charge le chiffrement des copies de snapshots DB et le partage des snapshots DB chiffrés. Pour plus d'informations, consultez Copier un instantané de base de données pour Amazon RDS et Partage d'un instantané de base de données pour Amazon RDS.	11 février 2016
Nouvelle fonction	Mise à jour pour prendre en charge Amazon Aurora dans la région Asie-Pacifique (Sydney). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora MySQL.	11 février 2016
Nouvelle fonction	Mise à jour pour prendre en charge SSL pour les instances de base de données Oracle. Pour plus d'informations, consultez Utilisation de SSL avec une instance de base de données RDS for Oracle.	9 février 2016

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge le fuseau horaire local pour les instances de bases de données MySQL et MariaDB. Pour plus d'informations, consultez Fuseau horaire local pour les instances My SQL DB et Fuseau horaire local pour les instances de base de données MariaDB.	21 décembre 2015
Nouvelle fonction	Mise à jour pour prendre en charge la surveillance améliorée des métriques du système d'exploitation pour les instances de bases de données MySQL et MariaDB et les clusters de bases de données Aurora. Pour plus d'informations, consultez Afficher les métriques dans la RDS console Amazon.	18 décembre 2015
Nouvelle fonctionnalité	Mis à jour pour prendre en charge les classes d'instance de base de données db.t2, db.r3 et db.m4 pour la version 5.5 de MySQL. Pour plus d'informations, consultez Classes d'instances de base de données .	4 décembre 2015
Nouvelle fonction	Mise à jour pour prendre en charge la modification du port de base de données pour une instance de base de données existante.	3 décembre 2015
Nouvelle fonctionnalité	Mise à jour pour prendre en charge les mises à niveau de version majeure du moteur de base de données pour les instances PostgreSQL. Pour plus d'informations, consultez Mises à niveau du moteur de base de données RDS pour PostgreSQL.	19 novembre 2015
Nouvelle fonction	Mise à jour pour prendre en charge la modification de l'accessibilité publique d'une instance de base de données existante. Mise à jour pour prendre en charge les classes d'instance de base de données standard db.m4.	11 novembre 2015

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge le partage Guide d'un instantané de base de données. Pour plus d'informations, consultez Partage d'un instantané de base de données pour Amazon RDS.	28 octobre 2015
Nouvelle fonction	Mise à jour pour prendre en charge Microsoft SQL Server 2014 pour les éditions Web, Express et Standard.	26 octobre 2015
Nouvelle fonction	Mise à jour pour prendre en charge le moteur de base de données MariaDB basé sur MySQL. Pour plus d'informations, consultez Amazon RDS for MariaDB.	7 octobre 2015
Nouvelle fonction	Mise à jour pour prendre en charge Amazon Aurora dans la région Asie-Pacifique (Tokyo). Pour plus d'informations, consultez Disponibilité pour Amazon Aurora MySQL.	7 octobre 2015
Nouvelle fonction	Mise à jour pour prendre en charge les classes d'instance de base de données avec capacité de transmission en rafales db.t2 pour tous les moteurs DB et l'ajout de la classe d'instance de base de données db.t2.large. Pour plus d'informations, consultez Classes d'instances de base de données .	25 septembre 2015
Nouvelle fonction	Mise à jour pour prendre en charge les instances de bases de données Oracle sur les classes d'instance de base de données R3 et T2. Pour plus d'informations, consultez Classes d'instances de base de données .	5 août 2015
Nouvelle fonctionnalité	Microsoft SQL Server Enterprise Edition est désormais disponible avec le modèle de service « License Included » (licence incluse). Pour plus d'informations, consultez Gestion des licences Microsoft SQL Server sur Amazon RDS.	29 juillet 2015

Modification	Description	Date de modification
Nouvelle fonction	Amazon Aurora est officiellement disponible. Le moteur DB Amazon Aurora prend en charge plusieurs instances de bases de données dans un cluster DB. Pour plus d'informations, consultez Qu'est-ce qu'Amazon Aurora ?	27 juillet 2015
Nouvelle fonction	Mise à jour pour prendre en charge la copie des balises vers des snapshots DB.	20 juillet 2015
Nouvelle fonction	Mise à jour pour prendre en charge les augmentations du volume de stockage pour tous les moteurs DB et une hausse des IOPS provisionnées pour SQL Server.	18 juin 2015
Nouvelle fonction	Mise à jour des options pour les instances de bases de données réservées.	15 juin 2015
Nouvelle fonctionnalité	Mise à jour pour prendre en charge l'utilisation d'Amazon CloudHSM avec les instances de bases de données Oracle grâce à TDE.	8 janvier 2015
Nouvelle fonction	Mise à jour pour prendre en charge le chiffrement des données au repos et la nouvelle version de l'API 2014-10-31.	6 janvier 2015
Nouvelle fonctionnalité	Mise à jour pour inclure le nouveau moteur DB Amazon : Aurora. Le moteur DB Amazon Aurora prend en charge plusieurs instances de bases de données dans un cluster DB. Amazon Aurora est actuellement en version préliminaire et peut être modifié. Pour plus d'informations, consultez Qu'est-ce qu'Amazon Aurora ?	12 novembre 2014
Nouvelle fonction	Mise à jour pour prendre en charge les réplicas en lecture PostgreSQL.	10 novembre 2014

Modification	Description	Date de modification
Nouvelles API et fonctions	Mise à jour pour prendre en charge le type de GP2 stockage et nouvelle version de l'API 2014-09-01. Mise à jour pour prendre en charge la copie d'une option existante ou d'un groupe de paramètres pour créer une nouvelle option ou un groupe de paramètres.	7 octobre 2014
Nouvelle fonction	Mise à jour pour prendre en charge la préparation du cache InnoDB pour les instances de bases de données exécutant MySQL version 5.6.19 et ultérieure.	3 septembre 2014
Nouvelle fonction	Mise à jour pour prendre en charge la vérification du certificat SSL lors de la connexion aux moteurs de base de données MySQL (version 5.6), SQL Server et PostgreSQL.	5 août 2014
Nouvelle fonctionnalité	Mise à jour pour prendre en charge les classes d'instance de base de données avec capacité de transmission en rafales db.t2.	4 août 2014
Nouvelle fonctionnalité	Mise à jour pour prendre en charge les classes d'instance de base de données à mémoire optimisée db.r3 à utiliser avec les moteurs de base de données MySQL (version 5.6), SQL Server et PostgreSQL.	28 mai 2014
Nouvelle fonction	Mise à jour pour prendre en charge les déploiements multi-AZ SQL Server grâce à la mise en miroir SQL Server.	19 mai 2014
Nouvelle fonction	Mise à jour pour prendre en charge les mises à niveau MySQL version 5.5 à 5.6.	23 avril 2014
Nouvelle fonctionnalité	Mis à jour pour prendre en charge Oracle GoldenGate.	3 avril 2014

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge les classes d'instances de bases de données M3.	20 février 2014
Nouvelle fonction	Mise à jour pour prendre en charge l'option Oracle Timezone.	13 janvier 2014
Nouvelle fonction	Mise à jour pour prendre en charge la réplication entre des instances de bases de données MySQL dans différentes régions.	26 novembre 2013
Nouvelle fonction	Mise à jour pour prendre en charge le moteur DB PostgreSQL.	14 novembre 2013
Nouvelle fonction	Mise à jour pour prendre en charge le chiffrement TDE (Transparent Data Encryption) SQL Server.	7 novembre 2013
Nouvelles API et fonction	Mise à jour pour prendre en charge les copies d'instantané de base de données entre régions ; la nouvelle version de l'API 2013-09-09.	31 octobre 2013
Nouvelles fonctions	Mise à jour pour prendre en charge Oracle Statspack.	26 septembre 2013
Nouvelles fonctionnalités	Mise à jour pour prendre en charge l'utilisation de la réplication pour importer ou exporter des données entre des instances de MySQL exécutées sur Amazon RDS et des instances de MySQL exécutées sur site ou sur Amazon. EC2	5 septembre 2013
Nouvelles fonctions	Mise à jour pour prendre en charge la classe d'instance de base de données db.cr1.8xlarge pour MySQL 5.6.	4 septembre 2013
Nouvelle fonction	Mise à jour pour prendre en charge la réplication des réplicas en lecture.	28 août 2013

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge la création parallèle de réplicas en lecture.	22 juillet 2013
Nouvelle fonction	Mise à jour pour prendre en charge les autorisations et le balisage précis pour toutes les ressources Amazon RDS.	8 juillet 2013
Nouvelle fonction	Mise à jour pour prendre en charge MySQL 5.6 pour de nouvelles instances, dont la prise en charge pour l'interface memcached MySQL 5.6 et l'accès au journal binaire.	1 juillet 2013
Nouvelle fonction	Mise à jour pour prendre en charge les mises à niveau de version majeure MySQL 5.1 à 5.5.	20 juin 2013
Nouvelle fonction	Mise à jour des groupes de paramètres de base de données pour autoriser des expressions pour des valeurs de paramètres.	20 juin 2013
Nouvelles API et fonction	Mise à jour pour prendre en charge le statut du réplica en lecture, et nouvelle version d'API, 2013-05-15.	23 mai 2013
Nouvelles fonctions	Mise à jour pour prendre en charge les fonctions Oracle Advanced Security pour le chiffrement réseau natif et Oracle TDE.	18 avril 2013
Nouvelles fonctions	Mise à jour pour prendre en charge les mises à niveau de version majeure pour SQL Server et une fonctionnalité supplémentaire pour les IOPS provisionnées.	13 mars 2013
Nouvelle fonction	Mise à jour pour prendre en charge VPC par défaut pour RDS.	11 mars 2013
Nouvelles API et fonction	Mise à jour pour prendre en charge l'accès au journal ; la nouvelle version de l'API 2013-02-12	4 mars 2013

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge les abonnements aux notifications d'événements RDS.	4 février 2013
Nouvelles API et fonction	Mise à jour pour prendre en charge le renommage d'une instance de base de données et la migration des membres du groupe de sécurité DB d'un VPC vers un groupe de sécurité VPC.	14 janvier 2013
Nouvelle fonctionnalité	Mis à jour pour le support AWS GovCloud (ouest des États-Unis).	17 décembre 2012
Nouvelle fonction	Mise à jour pour prendre en charge les classes d'instances de base de données m1.medium et m1.xlarge.	6 novembre 2012
Nouvelle fonction	Mise à jour pour prendre en charge la promotion des réplicas en lecture.	11 octobre 2012
Nouvelle fonction	Mise à jour pour prendre en charge SSL dans les instances de base de données Microsoft SQL Server.	10 octobre 2012
Nouvelle fonction	Mise à jour pour prendre en charge les micro-instances de base de données Oracle.	27 septembre 2012
Nouvelle fonction	Mise à jour pour prendre en charge SQL Server 2012.	26 septembre 2012
Nouvelles API et fonction	Mise à jour pour prendre en charge les IOPS provisionnées. Version de l'API 2012-09-17.	25 septembre 2012
Nouvelles fonctions	Mise à jour pour prendre en charge SQL Server pour les instances de base de données dans le VPC et pour prendre en charge Oracle pour Data Pump.	13 septembre 2012
Nouvelle fonction	Mise à jour pour prendre en charge SQL Server Agent.	22 août 2012

Modification	Description	Date de modification
Nouvelle fonction	Mise à jour pour prendre en charge le balisage des instances de base de données.	21 août 2012
Nouvelles fonctions	Mise à jour pour prendre en charge Oracle APEX et XML DB, les fuseaux horaires Oracle et les instances de base de données Oracle dans un VPC.	16 août 2012
Nouvelles fonctions	Mise à jour pour prendre en charge SQL Server Database Engine Tuning Advisor et les instances de base de données Oracle dans un VPC.	18 juillet 2012
Nouvelle fonction	Mise à jour pour prendre en charge les groupes d'options et la première option, Oracle Enterprise Manager Database Control.	29 mai 2012
Nouvelle fonction	Mise à jour pour prendre en charge les réplicas en lecture dans Amazon Virtual Private Cloud.	17 mai 2012
Nouvelle fonction	Mise à jour pour prendre en charge Microsoft SQL Server.	8 mai 2012
Nouvelles fonctions	Mise à jour pour prendre en charge le basculement forcé, le déploiement multi-AZ des instances de base de données Oracle et les jeux de caractères personnalisés pour les instances de base de données Oracle.	2 mai 2012
Nouvelle fonction	Mise à jour pour prendre en charge Amazon Virtual Private Cloud (VPC).	13 février 2012
Contenu mis à jour	Mise à jour pour de nouveaux types d'instance réservée.	19 décembre 2011
Nouvelle fonction	Mise à jour pour prendre en charge le moteur Oracle.	23 mai 2011
Contenu mis à jour	Mises à jour de la console.	13 mai 2011

Modification	Description	Date de modification
Contenu mis à jour	Contenu modifié pour une sauvegarde et des fenêtres de maintenance plus courtes.	28 février 2011
Nouvelle fonction	Ajout de la prise en charge pour MySQL 5.5.	31 janvier 2011
Nouvelle fonction	Ajout de la prise en charge des réplicas en lecture.	4 octobre 2010
Nouvelle fonctionnalité	Ajout du support pour AWS Identity and Access Management (IAM).	2 septembre 2010
Nouvelle fonction	Ajout de la gestion des versions du moteur de base de données.	16 août 2010
Nouvelle fonction	Ajout d'instances de base de données réservées.	16 août 2010
Nouvelle fonction	Amazon RDS prend désormais en charge les connexions SSL vers vos instances de base de données.	28 juin 2010
Nouveau Guide	Il s'agit de la première version du Guide de l'utilisateur Amazon RDS.	7 juin 2010

AWS Glossaire

Pour la AWS terminologie la plus récente, consultez le AWS glossaire dans la Glossaire AWS référence.

Les traductions sont fournies par des outils de traduction automatique. En cas de conflit entre le contenu d'une traduction et celui de la version originale en anglais, la version anglaise prévaudra.