aws

ユーザーガイド

# Research and Engineering Studio

# Research and Engineering Studio: ユーザーガイド

# Table of Contents

# 概要

Research and Engineering Studio (RES) は、 AWS サポートされているオープンソース製品です。これにより、IT 管理者は、サイエンティストやエンジニアが でテクニカルコンピューティングワークロードを実行するためのウェブポータルを提供できます AWS。RES は、ユーザーが安全な仮想デスクトップを起動して、科学研究、製品設計、エンジニアリングシミュレーション、データ分析のワークロードを実行できる単一のペインを提供します。ユーザーは、既存の企業認証情報を使用して RES ポータルに接続し、個々のプロジェクトまたは共同プロジェクトに取り組むことができます。

管理者は、特定のユーザーのセットに対してプロジェクトと呼ばれる仮想コラボレーションスペースを作成して、共有リソースにアクセスし、共同作業を行うことができます。管理者は、独自のアプリケーションソフトウェアスタック (AMIsを構築し、RES ユーザーが Windows または Linux 仮想デスクトップを起動できるようにし、共有ファイルシステムを介してプロジェクトデータにアクセスできるようにします。管理者は、ソフトウェアスタックとファイルシステムを割り当て、それらのプロジェクトユーザーのみにアクセスを制限できます。管理者は、組み込みテレメトリを使用して環境の使用状況をモニタリングし、ユーザーの問題をトラブルシューティングできます。また、リソースの過剰消費を防ぐために、個々のプロジェクトの予算を設定することもできます。製品はオープンソースであるため、お客様は自分のニーズに合わせて RES ポータルのユーザーエクスペリエンスをカスタマイズすることもできます。

RES は追加料金なしで利用でき、アプリケーションの実行に必要な AWS リソースに対してのみ料金が発生します。

このガイドでは、 での Research and Engineering Studio の概要 AWS、そのリファレンスアーキテクチャとコンポーネント、デプロイを計画する際の考慮事項、および RES を Amazon Web Services (AWS) クラウドにデプロイするための設定手順について説明します。

# 特徴と利点

の Research and Engineering Studio AWS には、次の機能があります。

ウェブベースのユーザーインターフェイス

RES は、管理者、研究者、エンジニアが研究およびエンジニアリングワークスペースにアクセスして管理するために使用できるウェブベースのポータルを提供します。科学者やエンジニアは、RES を使用するために AWS アカウント やクラウドの専門知識を持っている必要はありません。

プロジェクトベースの設定

プロジェクトを使用して、一連のタスクまたはアクティビティのアクセス許可の定義、リソースの割り当て、予算の管理を行います。整合性とコンプライアンスを確保するために、特定のソフトウェアスタック (オペレーティングシステムと承認済みアプリケーション) とストレージリソースをプロジェクトに割り当てます。プロジェクトごとに支出を監視および管理します。

コラボレーションツール

サイエンティストやエンジニアは、プロジェクトの他のメンバーを招待してコラボレーションし、それらの同僚に求めるアクセス許可レベルを設定できます。これらの個人は RES にサインインして、それらのデスクトップに接続できます。

既存の ID 管理インフラストラクチャとの統合

既存の ID 管理およびディレクトリサービスインフラストラクチャと統合して、ユーザーの既存の企業 ID を使用して RES ポータルに接続し、既存のユーザーおよびグループメンバーシップを使用してプロジェクトにアクセス許可を割り当てることができます。

永続的なストレージと共有データへのアクセス

仮想デスクトップセッション間で共有データへのアクセスをユーザーに許可するには、既存のファイルシステムに接続するか、RES 内に新しいファイルシステムを作成します。サポートされているストレージサービスには、Linux デスクトップ用の Amazon Elastic File System と、Windows および Linux デスクトップ用の NetApp ONTAP 用の Amazon FSx が含まれます。

モニタリングとレポート

分析ダッシュボードを使用して、インスタンスタイプ、ソフトウェアスタック、オペレーティングシステムタイプのリソース使用状況をモニタリングします。ダッシュボードには、レポート用のプロジェクト別のリソース使用状況の内訳も表示されます。

予算とコスト管理

RES プロジェクト AWS Budgets にリンクして、各プロジェクトのコストをモニタリングします。予算を超えた場合は、VDI セッションの起動を制限できます。

# 概念と定義

このセクションでは、主要な概念について説明し、この製品に固有の用語を定義します。

ファイルブラウザ

ファイルブラウザは、現在ログインしているユーザーがファイルシステムを表示できる RES ユーザーインターフェイスの一部です。

ファイルシステム

ファイルシステムは、プロジェクトデータのコンテナとして機能し、データセットとも呼ばれます。データセットは、プロジェクトの境界内でストレージソリューションを提供し、コラボレーションとデータアクセスコントロールを向上させます。

プロジェクト

プロジェクトは、データとコンピューティングリソースの明確な境界として機能するアプリケーション内の論理パーティションであり、データフローのガバナンスを確保し、プロジェクト間でのデータと VDI ホストの共有を防止します。

プロジェクトベースのアクセス許可

プロジェクトベースのアクセス許可は、複数のプロジェクトが存在する可能性があるシステム内のデータと VDI ホストの両方の論理パーティションを記述します。プロジェクト内のデータと VDI ホストへのユーザーのアクセスは、関連するロールによって決まります (複数可 ) 。ユーザーには、アクセスが必要なプロジェクトごとにアクセス (またはプロジェクトメンバーシップ) を割り当てる必要があります。それ以外の場合、ユーザーはメンバーシップが付与されていないとプロジェクトデータと VDIs にアクセスできません。

ソフトウェアスタック

ソフトウェアスタックは、ユーザーが VDI ホスト用にプロビジョニングするために選択したオペレーティングシステムに基づいて、RES 固有のメタデータを持つ Amazon マシンイメージ (AMI) です。

VDI ホスト

仮想デスクトップインスタンス (VDI) ホストを使用すると、プロジェクトメンバーはプロジェクト固有のデータとコンピューティング環境にアクセスでき、安全で隔離されたワークスペースを確保できます。

AWS 用語の一般的なリファレンスについては、「AWS 全般のAWS リファレンス」の「 用語集」を参照してください。

# アーキテクチャの概要

このセクションでは、この製品でデプロイされるコンポーネントのアーキテクチャ図を示します。

## アーキテクチャ図

デフォルトのパラメータを使用してこの製品をデプロイすると、 に次のコンポーネントがデプロイされますAWS アカウント。

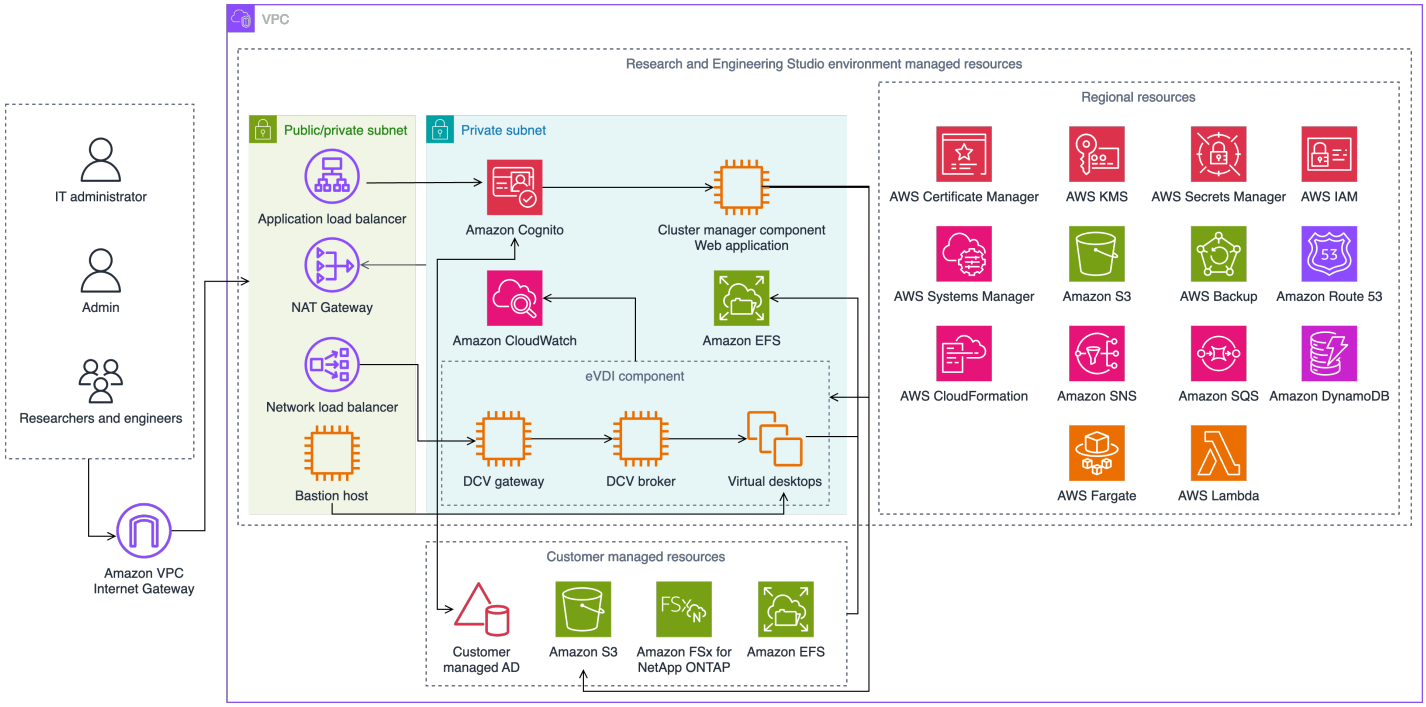



図 1: AWSアーキテクチャに関する調査とエンジニアリングスタジオ

> **Note**
>
> AWS CloudFormation リソースは AWS Cloud Development Kit (AWS CDK) コンストラクトから作成されます。

AWS CloudFormation テンプレートでデプロイされる製品コンポーネントの大まかなプロセスフローは次のとおりです。

1. RES は、ウェブポータルのコンポーネントと以下のコンポーネントをインストールします。
   a. インタラクティブワークロード用のエンジニアリング仮想デスクトップ (eVDI) コンポーネント

b. メトリクスコンポーネント

Amazon CloudWatch は eVDI コンポーネントからメトリクスを受け取ります。

c. 踏み台ホストコンポーネント

管理者は、SSH を使用して踏み台ホストコンポーネントに接続し、基盤となるインフラストラクチャを管理できます。

2. RES は、NAT ゲートウェイの背後にあるプライベートサブネットにコンポーネントをインストールします。管理者は、Application Load Balancer (ALB) または踏み台ホストコンポーネントを介してプライベートサブネットにアクセスします。
3. Amazon DynamoDB は環境設定を保存します。
4. AWS Certificate Manager (ACM) は、Application Load Balancer (ALB) のパブリック証明書を生成して保存します。

> **Note**
>
> を使用してAWS Certificate Manager、ドメインの信頼された証明書を生成することをお勧めします。

5. Amazon Elastic File System (EFS ) は、該当するすべてのインフラストラクチャホストと eVDI Linux セッションにマウントされたデフォルトの/homeファイルシステムをホストします。
6. RES は Amazon Cognito を使用して、 内に clusteradmin という名前の初期ブートストラップユーザーを作成し、インストール時に提供された E メールアドレスに一時的な認証情報を送信します。clusteradmin は、初回ログイン時にパスワードを変更する必要があります。
7. Amazon Cognito は、アクセス許可の管理のために組織の Active Directory およびユーザー ID と統合します。
8. セキュリティゾーンにより、管理者はアクセス許可に基づいて製品内の特定のコンポーネントへのアクセスを制限できます。

# AWS この製品の サービス

| AWS サービス | 説明 |
| --- | --- |
| Amazon Elastic Compute Cloud | コア。選択したオペレーティングシステムとソフトウェアスタックで仮想デスクトップを作成 |

| AWS サービス | 説明 |
| --- | --- |
| | するための基盤となるコンピューティングサービスを提供します。 |
| Elastic Load Balancing | コア。踏み台、クラスターマネージャー、VDI ホストは、ロードバランサーの背後にある Auto Scaling グループに作成されます。ELB は、ウェブポータルからのトラフィックを RES ホスト間で分散します。 |
| Amazon Virtual Private Cloud | コア。すべてのコア製品コンポーネントは VPC 内に作成されます。 |
| Amazon Cognito | コア。ユーザー ID と認証を管理します。Active Directory ユーザーは、アクセスレベルを認証するために Amazon Cognito ユーザーとグループにマッピングされます。 |
| Amazon Elastic File System | コア。/home ファイルブラウザと VDI ホスト用のファイルシステム、および共有外部ファイルシステムを提供します。 |
| 「Amazon DynamoDB」 | コア。ユーザー、グループ、プロジェクト、ファイルシステム、コンポーネント設定などの設定データを保存します。 |
| AWS Systems Manager | コア。VDI セッション管理のコマンドを実行するためのドキュメントを保存します。 |
| AWS Lambda | コア。DynamoDB テーブル内の設定の更新、Active Directory 同期ワークフローの開始、プレフィックスリストの更新などの製品機能をサポートします。 |
| Amazon CloudWatch | サポート。すべての Amazon EC2 ホストと Lambda 関数のメトリクスとアクティビティログを提供します。 |

| AWS サービス | 説明 |
| --- | --- |
| Amazon Simple Storage Service | サポート。ホストブートストラップと設定用のアプリケーションバイナリを保存します。 |
| AWS Key Management Service | サポート。Amazon SQS キュー、DynamoDB テーブル、および Amazon SNS トピックを使用した保管時の暗号化に使用されます。 |
| AWS Secrets Manager | サポート。サービスアカウントの認証情報を Active Directory に保存し、VDIs の自己署名証明書を保存します。 |
| AWS CloudFormation | サポート。製品のデプロイメカニズムを提供します。 |
| AWS Identity and Access Management | サポート。ホストのアクセスレベルを制限します。 |
| Amazon Route 53 | サポート。内部ロードバランサーと踏み台ホスト名を解決するためのプライベートホストゾーンを作成します。 |
| Amazon Simple Queue Service | サポート。非同期実行をサポートするタスクキューを作成します。 |
| Amazon Simple Notification Service | サポート。コントローラーやホストなどの VDI コンポーネント間のパブリケーションサブスクライバーモデルをサポートします。 |
| AWS Fargate | サポート。Fargate タスクを使用して環境をインストール、更新、削除します。 |
| Amazon FSx ファイルゲートウェイ | 省略可能。外部共有ファイルシステムを提供します。 |
| Amazon FSx for NetApp ONTAP | 省略可能。外部共有ファイルシステムを提供します。 |

| AWS サービス | 説明 |
| --- | --- |
| AWS Certificate Manager | 省略可能。カスタムドメインの信頼された証明書を生成します。 |
| AWS Backup | 省略可能。Amazon EC2 ホスト、ファイルシステム、DynamoDB のバックアップ機能を提供します。 |

# デプロイを計画する

## コスト

の Research and Engineering Studio AWS は追加料金なしで利用でき、アプリケーションの実行に必要なリソースに対して AWS のみ料金が発生します。詳細については、「AWS この製品の サービス」を参照してください。

> **Note**
>
> この製品の実行中に使用される AWS サービスのコストは、お客様の負担となります。コスト管理AWS Cost Explorerに役立つ予算を作成することをお勧めします。価格は変更されることがあります。詳細については、この製品で使用される各 AWS サービスの料金ウェブページを参照してください。

## セキュリティ

AWS インフラストラクチャ上にシステムを構築する場合、セキュリティ責任はお客様と の間で共有されます AWS。この責任共有モデルは、ホストオペレーティングシステム、仮想化レイヤー、サービス AWS が動作する施設の物理的なセキュリティなどのコンポーネントを運用、管理、制御するため、運用上の負担を軽減します。 AWS セキュリティの詳細については、「 セキュリティAWS クラウド」を参照してください。

## IAM ロール

AWS Identity and Access Management (IAM) ロールを使用すると、 のサービスおよびユーザーにきめ細かなアクセスポリシーとアクセス許可を割り当てることができます AWS クラウド。この製品は、製品の AWS Lambda 関数と Amazon EC2 インスタンスにリージョンリソースを作成するためのアクセス権を付与する IAM ロールを作成します。

RES は IAM 内のアイデンティティベースのポリシーをサポートします。デプロイされると、RES は管理者のアクセス許可とアクセスを定義するポリシーを作成します。製品を実装する管理者は、RES と統合された既存のカスタマー Active Directory 内でエンドユーザーとプロジェクトリーダーを作成および管理します。詳細については、「 Identity and Access Management ユーザーガイド」の「IAM ポリシーの作成」を参照してください。 AWS

組織の管理者は、アクティブディレクトリを使用してユーザーアクセスを管理できます。エンドユーザーが RES ユーザーインターフェイスにアクセスすると、RES は Amazon Cognito。

## セキュリティグループ

この製品で作成されたセキュリティグループは、Lambda 関数、EC2 インスタンス、ファイルシステム CSR インスタンス、リモート VPN エンドポイント間のネットワークトラフィックを制御および分離するように設計されています。セキュリティグループを確認し、製品のデプロイ後に必要に応じてアクセスをさらに制限することをお勧めします。

## データ暗号化

デフォルトでは、 AWS (RES) の Research and Engineering Studio は、RES が所有するキーを使用して、保管中および転送中の顧客データを暗号化します。RES をデプロイするときに、 を指定できます AWS KMS key。RES は、認証情報を使用してキーアクセスを付与します。カスタマー所有および管理の を指定すると AWS KMS key、保管中のカスタマーデータはそのキーを使用して暗号化されます。

RES は、SSL/TLS を使用して転送中の顧客データを暗号化します。TLS 1.2 が必要ですが、TLS 1.3 をお勧めします。

# サポート対象 AWS リージョン

この製品は、現在すべての で利用できないサービスを使用しています AWS リージョン。この製品は、すべてのサービス AWS リージョン が利用可能な で起動する必要があります。リージョン AWS 別のサービスの最新の可用性については、AWS リージョン「 サービスリスト」を参照してください。

の Research and Engineering Studio AWS は、次の でサポートされています AWS リージョン。

| リージョン名 | |
|---|---|
| 米国東部 ( オハイオ ) | カナダ (中部) |
| 米国東部 (バージニア北部) | 欧州 (フランクフルト) |
| 米国西部 ( 北カリフォルニア ) | 欧州 (アイルランド) |
| 米国西部 ( オレゴン ) | 欧州 (ロンドン) |

| リージョン名 | |
|---|---|
| アジアパシフィック ( ムンバイ ) | 欧州 (ミラノ) |
| アジアパシフィック (ソウル) | 欧州 (パリ) |
| アジアパシフィック (シンガポール) | イスラエル (テルアビブ) |
| アジアパシフィック (シドニー) | AWS GovCloud （ 米国西部 ） |
| アジアパシフィック (東京) | |

# クォータ

サービスクォータ (制限とも呼ばれます) は、 AWS アカウントのサービスリソースまたはオペレーションの最大数です。

## この製品の AWS サービスのクォータ

この製品 に実装されている各サービスに十分なクォータがあることを確認してください。詳細については、「AWS のサービスクォータ」を参照してください。

この製品では、以下のサービスのクォータを引き上げることをお勧めします。

- Amazon Virtual Private Cloud
- Amazon EC2

クォータの引き上げをリクエストするには、Service Quotas ユーザーガイド の「クォータ引き上げリクエスト」を参照してください。Service Quotas でクォータがまだ利用できない場合は、[上限引き上げ] フォームを使用してください。

## AWS CloudFormation クォータ

AWS アカウント には、この製品でスタックを起動するときに注意すべき AWS CloudFormation クォータがあります。これらのクォータを理解することで、この製品を正常にデプロイできないような制限エラーを回避できます。詳細については、「 ユーザーガイド」の「 のAWS CloudFormation クォータAWS CloudFormation 」を参照してください。

# レジリエンスの計画

製品は、Amazon EC2 インスタンスの最小数とサイズでデフォルトのインフラストラクチャをデプロイして、システムを運用します。大規模な本番環境の耐障害性を向上させるには、インフラストラクチャの Auto Scaling グループ (ASG) 内のデフォルトの最小容量設定を増やすことをお勧めします。値を 1 つのインスタンスから 2 つのインスタンスに増やすと、複数のアベイラビリティーゾーン (AZ) の利点が得られ、予期しないデータ損失が発生した場合にシステム機能を復元する時間を短縮できます。

ASG 設定は、https://console.aws.amazon.com/ec2/ の Amazon EC2 コンソールでカスタマイズできます。製品はデフォルトで 4 つの ASGs を作成し、各名前は で終わります-asg。最小値と希望の値は、本番環境に適した量に変更できます。変更するグループを選択し、アクションと編集 を選択します。ASGs「Amazon EC2 Auto Scaling ユーザーガイド」の「Auto Scaling グループのサイズをスケールする」を参照してください。 Amazon EC2 Auto Scaling

# 製品をデプロイする

> **Note**
>
> この製品は、 AWS CloudFormation テンプレートとスタックを使用してデプロイを自動化します。テンプレート CloudFormationは、この製品に含まれる AWS リソースとそのプロパティを記述します (複数可 )。 CloudFormation スタックは、テンプレートで説明されているリソースをプロビジョニングします (複数可 )。

製品を起動する前に、コスト、アーキテクチャ、ネットワークセキュリティ、およびこのガイドで前述したその他の考慮事項を確認してください。

トピック

- 前提条件
- デモ環境を作成する
- ステップ 1: 製品を起動する
- ステップ 2: 初めてサインインする

## 前提条件

トピック

- 管理ユーザー AWS アカウント を使用して を作成する
- Amazon EC2 SSH キーペアを作成する
- サービスクォータを増やす
- パブリックドメインを作成する (オプション )
- ドメインの作成 (GovCloud のみ )
- 外部リソースの提供
- 環境で LDAPS を設定する (オプション )
- プライベート VPC を設定する (オプション )

# 管理ユーザー AWS アカウント を使用して を作成する

管理ユーザー AWS アカウント を持つ が必要です。

1. https://portal.aws.amazon.com/billing/signup を開きます。
2. オンラインの手順に従います。

   サインアップ手順の一環として、通話呼び出しを受け取り、電話キーパッドで検証コードを入力するように求められます。

   にサインアップすると AWS アカウント、 AWS アカウントのルートユーザーが作成されます。ルートユーザーには、アカウントのすべての AWS のサービス とリソースへのアクセス権があります。セキュリティのベストプラクティスとして、ユーザーに管理アクセスを割り当て、ルートユーザーのみを使用してルートユーザーアクセスが必要なタスクを実行してください。

## Amazon EC2 SSH キーペアを作成する

Amazon EC2 SSH キーペアがない場合は、作成する必要があります。詳細については、Amazon EC2 ユーザーガイド」の「Amazon EC2 を使用してキーペアを作成するAmazon EC2」を参照してください。

## サービスクォータを増やす

以下のサービスクォータを増やすことをお勧めします。

- Amazon VPC
  - NAT ゲートウェイあたりの Elastic IP アドレスクォータを 5 から 8 に増やす
  - アベイラビリティーゾーンあたりの NAT ゲートウェイを 5 から 10 に増やす
- 「Amazon EC2」
  - EC2-VPC Elastic IPs 5 から 10 に増やす

AWS アカウントには、以前は制限と呼ばれていたデフォルトのクォータが AWS サービスごとにあります。特に明記されていない限り、クォータは地域固有です。一部のクォータについては引き上げをリクエストできますが、その他のクォータについては引き上げることはできません。詳細については、「the section called “この製品の AWS サービスのクォータ”」を参照してください。

# パブリックドメインを作成する (オプション )

ユーザーフレンドリーな URL を持つには、製品のカスタムドメインを使用することをお勧めします。Amazon Route 53 または別のプロバイダーを使用してドメインを登録し、 を使用してドメインの証明書をインポートする必要があります AWS Certificate Manager。パブリックドメインと証明書が既にある場合は、このステップをスキップできます。

1. 指示に従って、Route53 にドメインを登録します。確認メールが届きます。
2. ドメインのホストゾーンを取得します。これは Route53 によって自動的に作成されます。
    a. Route53 コンソールを開きます。
    b. 左側のナビゲーションからホストゾーンを選択します。
    c. ドメイン名用に作成されたホストゾーンを開き、ホストゾーン ID をコピーします。
3. を開き AWS Certificate Manager 、以下の手順に従ってドメイン証明書 をリクエストします。ソリューションをデプロイする予定のリージョンにいることを確認します。
4. ナビゲーションから証明書を一覧表示を選択し、証明書リクエストを見つけます。リクエストは保留中である必要があります。
5. 証明書 ID を選択してリクエストを開きます。
6. ドメインセクションで、Route53 でレコードの作成を選択します。リクエストの処理には約 10 分かかります。
7. 証明書が発行されたら、証明書のステータスセクションから ARN をコピーします。

# ドメインの作成 (GovCloud のみ )

AWS GovCloud （ 米国西部) リージョンに をデプロイする場合は、以下の前提条件のステップを完了する必要があります。

1. パブリックホストドメインが作成された商用パーティション AWS アカウントに証明書 AWS CloudFormation スタックをデプロイします。
2. 証明書 CloudFormation 出力 から、 と を見つけ`CertificateARN`てメモします`PrivateKeySecretARN`。
3. GovCloud パーティションアカウントで、`CertificateARN`出力の値を持つシークレットを作成します。がシークレット値にアクセスできるように、新しいシークレット ARN を書き留めて、シークレットに 2 `vdc-gateway` つのタグを追加します。
    a. res:ModuleName = virtual-desktop-controller

b. res:EnvironmentName = [環境名] (再デモの可能性があります )

4. GovCloud パーティションアカウントで、`PrivateKeySecretArn`出力の値を持つシークレットを作成します。がシークレット値にアクセスできるように、新しいシークレット ARN を書き留めて、シークレットに 2 `vdc-gateway` つのタグを追加します。

   a. res:ModuleName = virtual-desktop-controller

   b. res:EnvironmentName = [環境名] (再デモの可能性があります )

## 外部リソースの提供

Research and Engineering Studio を にデプロイする場合 AWS、必要な製品で使用される外部リソースがあります。RES は、デプロイ時にこれらのリソースが存在することを想定しています。

- ネットワーク (VPC、パブリック、プライベートサブネット )

  ここでは、環境、Active Directory (AD)、共有ストレージのホストに使用される EC2 インスタンスを実行します。

- ストレージ (Amazon EFS )

  ストレージボリュームには、仮想デスクトップインフラストラクチャ (VDI) に必要なファイルとデータが含まれています。

- ディレクトリサービス (AWS Directory Service for Microsoft Active Directory )

  ディレクトリサービスは、環境ページに対してユーザーを認証します。

> **Tip**
>
> デモ環境をデプロイしていて、これらの外部リソースが利用できない場合は、 AWS ハイパフォーマンスコンピューティングレシピを使用してデモ環境のリソースを生成できます。アカウントにリソースをデプロイするには、次のセクションthe section called “デモ環境を作成する”「」を参照してください。
> AWS GovCloud （ 米国西部) リージョンでのデモデプロイでは、「」の前提条件ステップを完了する必要がありますドメインの作成 (GovCloud のみ )。

# 環境で LDAPS を設定する (オプション )

環境で LDAPS 通信を使用する場合は、これらのステップを実行して、証明書を作成して AWS Managed Microsoft AD (AD) ドメインコントローラーにアタッチし、AD と RES 間の通信を提供する必要があります。

1. 「 のサーバー側の LDAPS を有効にする方法」に記載されているステップに従います AWS Managed Microsoft AD。LDAPS を既に有効にしている場合は、このステップをスキップできます。
2. LDAPS が AD に設定されていることを確認したら、AD 証明書をエクスポートします。
   a. Active Directory サーバーに移動します。
   b. 管理者 PowerShell として を開きます。
   c. `certmgr.msc` を実行して証明書リストを開きます。
   d. 最初に信頼されたルート認証機関を開き、次に証明書を開いて、証明書リストを開きます。
   e. AD サーバーと同じ名前の証明書を選択および保持 (または右クリック) し、すべてのタスクを選択し、 をエクスポートします。
   f. Base-64 でエンコードされた X.509 (.CER) を選択し、次へ を選択します。
   g. ディレクトリを選択し、次へ を選択します。
3. でシークレットを作成します AWS Secrets Manager。

   シークレットマネージャーでシークレットを作成する場合は、[シークレットのタイプ] で [その他のシークレット] を選択し、[プレーンテキスト] フィールドに PEM エンコードの証明書を貼り付けます。
4. 作成された ARN をメモし、 の `DomainTLSCertificateSecretARN`パラメータとして入力しますthe section called “ステップ 1: 製品を起動する”。

# プライベート VPC を設定する (オプション )

Research and Engineering Studio を分離された VPC にデプロイすると、組織のコンプライアンスとガバナンス要件を満たすためのセキュリティが強化されます。ただし、標準の RES デプロイは、依存関係のインストールにインターネットアクセスに依存しています。プライベート VPC に RES をインストールするには、次の前提条件を満たす必要があります。

トピック

- Amazon マシンイメージ (AMIsを準備する

- VPC エンドポイントをセットアップする
- VPC エンドポイントなしで サービスに接続する
- プライベート VPC デプロイパラメータを設定する

## Amazon マシンイメージ (AMIsを準備する

1. 依存関係 をダウンロードします。分離された VPC にデプロイするには、RES インフラストラクチャでパブリックインターネットアクセスなしで依存関係を利用できる必要があります。
2. Amazon S3 読み取り専用アクセスと Amazon EC2 としての信頼できる ID を持つ IAM ロールを作成します。 Amazon EC2
   a. IAM コンソール (https://console.aws.amazon.com/iam/) を開きます。
   b. ロール から、ロールの作成 を選択します。
   c. 信頼されたエンティティの選択ページで、次の操作を行います。
      - [Trusted entity type] (信頼できるエンティティタイプ) で、 AWS のサービス[] を選択します。
      - 「サービス」または「ユースケース」の「ECEC2」を選択し、「次へ」を選択します。
   d. アクセス許可の追加 で、次のアクセス許可ポリシーを選択し、次へ を選択します。
      - AmazonS3ReadOnlyAccess
      - AmazonSSMManagedInstanceCore
      - EC2InstanceProfileForImageBuilder
   e. ロール名 と説明 を追加し、ロールの作成 を選択します。
3. EC2 Image Builder コンポーネントを作成します。
   a. で EC2 Image Builder コンソールを開きますhttps://console.aws.amazon.com/imagebuilder。
   b. 「保存されたリソース」で、「コンポーネント」を選択し、「コンポーネントの作成」を選択します。
   c. 「コンポーネントの作成」ページで、次の詳細を入力します。
      - コンポーネントタイプ で、ビルド を選択します。
      - コンポーネントの詳細 で以下を選択します。

| パラメータ | ユーザーエントリ |
| --- | --- |
| イメージオペレーティングシステム (OS) | Linux |
| 互換性のある OS バージョン | Amazon Linux 2 |
| コンポーネント名 | <*research-and-engineering-studio-infrastructure*> などの名前を選択します。 |
| コンポーネントのバージョン | 1.0.0 から開始することをお勧めします。 |
| 説明 | オプションのユーザーエントリ。 |

d. 「コンポーネントの作成」ページで、「ドキュメントコンテンツの定義」を選択します。

i. 定義ドキュメントの内容を入力する前に、tar.gz ファイルのファイル URI が必要です。RES が提供する tar.gz ファイルを Amazon S3 バケットにアップロードし、バケットプロパティからファイルの URI をコピーします。

ii. 次のように入力します。

> **Note**
>
> `AddEnvironmentVariables` はオプションであり、インフラストラクチャホストにカスタム環境変数が必要ない場合は削除できます。
> `http_proxy` および `https_proxy`環境変数を設定する場合、インスタンスがプロキシを使用して localhost、インスタンスメタデータ IP アドレス、および VPC エンドポイントをサポートするサービスをクエリしないようにするには、`no_proxy`パラメータが必要です。

```
#  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
#  Licensed under the Apache License, Version 2.0 (the "License"). You may
 not use this file except in compliance
#  with the License. A copy of the License is located at
#
#      http://www.apache.org/licenses/LICENSE-2.0
#
```

```
#  or in the 'license' file accompanying this file. This file is
 distributed on an 'AS IS' BASIS, WITHOUT WARRANTIES
#  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
 specific language governing permissions
#  and limitations under the License.
name: research-and-engineering-studio-infrastructure
description: An RES EC2 Image Builder component to install required RES
 software dependencies for infrastructure hosts.
schemaVersion: 1.0

parameters:
  - AWSAccountID:
      type: string
      description: RES Environment AWS Account ID
  - AWSRegion:
      type: string
      description: RES Environment AWS Region
phases:
  - name: build
    steps:
       - name: DownloadRESInstallScripts
         action: S3Download
         onFailure: Abort
         maxAttempts: 3
         inputs:
            - source: '<s3 tar.gz file uri>'
              destination: '/root/bootstrap/res_dependencies/
res_dependencies.tar.gz'
              expectedBucketOwner: '{{ AWSAccountID }}'
       - name: RunInstallScript
         action: ExecuteBash
         onFailure: Abort
         maxAttempts: 3
         inputs:
            commands:
                - 'cd /root/bootstrap/res_dependencies'
                - 'tar -xf res_dependencies.tar.gz'
                - 'cd all_dependencies'
                - '/bin/bash install.sh'
       - name: AddEnvironmentVariables
         action: ExecuteBash
         onFailure: Abort
         maxAttempts: 3
         inputs:
```

```
            commands:
                - |
                  echo -e "
                  http_proxy=http://<ip>:<port>
                  https_proxy=http://<ip>:<port>

 no_proxy=127.0.0.1,169.254.169.254,169.254.170.2,localhost,
{{ AWSRegion }}.local,{{ AWSRegion }}.vpce.amazonaws.com,
{{ AWSRegion }}.elb.amazonaws.com,s3.
{{ AWSRegion }}.amazonaws.com,s3.dualstack.
{{ AWSRegion }}.amazonaws.com,ec2.{{ AWSRegion }}.amazonaws.com,ec2.
{{ AWSRegion }}.api.aws,ec2messages.{{ AWSRegion }}.amazonaws.com,ssm.
{{ AWSRegion }}.amazonaws.com,ssmmessages.
{{ AWSRegion }}.amazonaws.com,kms.
{{ AWSRegion }}.amazonaws.com,secretsmanager.
{{ AWSRegion }}.amazonaws.com,sqs.
{{ AWSRegion }}.amazonaws.com,elasticloadbalancing.
{{ AWSRegion }}.amazonaws.com,sns.{{ AWSRegion }}.amazonaws.com,logs.
{{ AWSRegion }}.amazonaws.com,logs.
{{ AWSRegion }}.api.aws,elasticfilesystem.
{{ AWSRegion }}.amazonaws.com,fsx.{{ AWSRegion }}.amazonaws.com,dynamodb.
{{ AWSRegion }}.amazonaws.com,api.ecr.
{{ AWSRegion }}.amazonaws.com,.dkr.ecr.
{{ AWSRegion }}.amazonaws.com,kinesis.{{ AWSRegion }}.amazonaws.com,.data-
kinesis.{{ AWSRegion }}.amazonaws.com,.control-
kinesis.{{ AWSRegion }}.amazonaws.com,events.
{{ AWSRegion }}.amazonaws.com,cloudformation.
{{ AWSRegion }}.amazonaws.com,sts.
{{ AWSRegion }}.amazonaws.com,application-autoscaling.
{{ AWSRegion }}.amazonaws.com,monitoring.{{ AWSRegion }}.amazonaws.com
                  " > /etc/environment
```

e. [コンポーネントを作成] を選択します。

4. Image Builder イメージレシピを作成します。

   a. レシピの作成ページで、次のように入力します。

| セクション | パラメータ | ユーザーエントリ |
| --- | --- | --- |
| レシピの詳細 | 名前 | res-recipe-linux-x86 などの適切な名前を入力します。 |
| | バージョン | バージョンを入力します。通常は 1.0.0 から開始します。 |
| | 説明 | オプションの説明を追加します。 |
| ベースイメージ | イメージの選択 | マネージドイメージを選択します。 |
| | OS | Amazon Linux |
| | イメージオリジン | クイックスタート (Amazon マネージド ) |
| | [イメージ名] | Amazon Linux 2 x86 |
| | 自動バージョニングオプション | 利用可能な最新の OS バージョンを使用します。 |
| インスタンス設定 | – | すべてをデフォルト設定のままにし、パイプラインの実行が選択されていない後に SSM エージェントを削除してください。 |
| 作業ディレクトリ | 作業ディレクトリパス | /root/bootstrap/res_dependencies |

| セクション | パラメータ | ユーザーエントリ |
| --- | --- | --- |
| コンポーネント | ビルドコンポーネント | 以下を検索して選択します。<br>• Amazon マネージド: aws-cli-version-2-linux<br>• Amazon マネージド: amazon-cloudwatch-agent-linux<br>• 所有: 以前に作成された Amazon EC2 コンポーネント。フィールドに AWS アカウント ID と現在の AWS リージョン を入力します。 |
| | テストコンポーネント | 以下を検索して選択します。<br>• Amazon マネージド : simple-boot-test-linux |

b. [レシピを作成する] を選択します。

5. Image Builder インフラストラクチャ設定を作成します。

a. 「保存されたリソース」で、「インフラストラクチャ設定」を選択します。

b. インフラストラクチャー構成の作成 を選択します。

c. 「インフラストラクチャ設定の作成」ページで、次のように入力します。

| セクション | パラメータ | ユーザーエントリ |
| --- | --- | --- |
| 全般 | 名前 | res-infra-linux-x86 などの適切な名前を入力します。 |
| | 説明 | オプションの説明を追加します。 |

| セクション | パラメータ | ユーザーエントリ |
| --- | --- | --- |
| | IAM ロール | 前に作成した IAM ロールを選択します。 |
| AWS インフラストラクチャ | インスタンスタイプ | t3.medium を選択します。 |
| | VPC、サブネット、セキュリティグループ | Amazon S3 バケットへのインターネットアクセスとアクセスを許可するオプションを選択します。セキュリティグループを作成する必要がある場合は、次の入力を使用して Amazon EC2 コンソールから作成できます。<br>• VPC: インフラストラクチャ設定に使用されているのと同じ VPC を選択します。この VPC にはインターネットアクセスが必要です。<br>• インバウンドルール：<br>  • タイプ: SSH<br>  • [Source]: Custom<br>  • CIDR ブロック: 0.0.0.0/0 |

d. インフラストラクチャー構成の作成 を選択します。

6. 新しい EC2 Image Builder パイプラインを作成します。

a. 「イメージパイプライン」に移動し、「イメージパイプラインの作成」を選択します。

b. パイプラインの詳細の指定ページで、次のように入力し、次へ を選択します。

- パイプライン名とオプションの説明

- ビルドスケジュール でスケジュールを設定するか、AMI ベーキングプロセスを手動で開始する場合は手動 を選択します。

c. 「レシピの選択」ページで「既存のレシピを使用」を選択し、前に作成したレシピ名を入力します。[次へ] をクリックします。

d. 画像プロセスの定義ページで、デフォルトのワークフローを選択し、次へ を選択します。

e. 「インフラストラクチャ設定の定義」ページで「既存のインフラストラクチャ設定を使用する」を選択し、以前に作成したインフラストラクチャ設定の名前を入力します。[次へ] をクリックします。

f. 「ディストリビューション設定の定義」ページで、選択について次の点を考慮してください。

- RES がそこからインフラストラクチャホストインスタンスを適切に起動できるように、出力イメージはデプロイされた RES 環境と同じリージョンに存在する必要があります。サービスのデフォルトを使用すると、EC2 Image Builder サービスが使用されているリージョンに出力イメージが作成されます。
- 複数のリージョンに RES をデプロイする場合は、新しいディストリビューション設定を作成し、そこにリージョンを追加できます。

g. 選択内容を確認し、パイプラインの作成 を選択します。

7. EC2 Image Builder パイプラインを実行します。

a. イメージパイプライン から、作成したパイプラインを見つけて選択します。

b. アクション を選択し、パイプラインの実行 を選択します。

パイプラインは、AMI イメージの作成に約 45 分から 1 時間かかる場合があります。

8. 生成された AMI の AMI ID をメモし、 の InfrastructureHostAMI パラメータの入力として使用しますthe section called “ステップ 1: 製品を起動する”。

## VPC エンドポイントをセットアップする

RES をデプロイして仮想デスクトップを起動するには、プライベートサブネットへのアクセス AWS のサービス が必要です。必要なアクセスを提供するように VPC エンドポイントを設定する必要があります。また、エンドポイントごとにこれらのステップを繰り返す必要があります。

1. エンドポイントが以前に設定されていない場合は、「インターフェイス VPC エンドポイント AWS のサービス を使用して にアクセスする」に記載されている手順に従ってください。
2. 2 つのアベイラビリティーゾーンのそれぞれで 1 つのプライベートサブネットを選択します。

| AWS のサービス | サービス名 |
| --- | --- |
| Application Auto Scaling | com.amazonaws.region.application-autoscaling |
| AWS CloudFormation | com.amazonaws.region.cloudformation |
| Amazon CloudWatch | com.amazonaws.region.monitoring |
| Amazon CloudWatch Logs | com.amazonaws.region.logs |
| Amazon DynamoDB | com.amazonaws.*region* .dynamodb (ゲートウェイエンドポイントが必要 ) |
| 「Amazon EC2」 | com.amazonaws.region.ec2 |
| Amazon ECR | com.amazonaws.region.ecr.api |
| | com.amazonaws.region.ecr.dkr |
| Amazon Elastic File System | com.amazonaws.region.elasticfilesystem |
| Elastic Load Balancing | com.amazonaws.region.elasticloadbalancing |
| Amazon EventBridge | com.amazonaws.region.events |
| Amazon FSx | com.amazonaws.region.fsx |
| AWS Key Management Service | com.amazonaws.region.kms |
| Amazon Kinesis Data Streams | com.amazonaws.region.kinesis-streams |
| Amazon S3 | com.amazonaws.*region* .s3 (RES でデフォルトで作成されるゲートウェイエンドポイントが必要です ) 。 |
| AWS Secrets Manager | com.amazonaws.*region*.secretsmanager |
| Amazon SES | com.amazonaws.*region* .email-smtp (次のアベイラビリティーゾーンではサポートされていません。use-1-az2、use1-az3、use1-az5、usw1-az2、usw2-az4、apne2-az4、cac1-az3、および cac1-az4) |

| AWS のサービス | サービス名 |
| --- | --- |
| AWS Security Token Service | com.amazonaws.*region*.sts |
| Amazon SNS | com.amazonaws.*region*.sns |
| Amazon SQS | com.amazonaws.*region*.sqs |
| AWS Systems Manager | com.amazonaws.*region*.ec2messages |
| | com.amazonaws.*region*.ssm |
| | com.amazonaws.*region*.ssmmessages |

## VPC エンドポイントなしで サービスに接続する

VPC エンドポイントをサポートしていないサービスと統合するには、VPC のパブリックサブネットにプロキシサーバーを設定できます。 AWS Identity Center を ID プロバイダーとして使用して、Research and Engineering Studio のデプロイに必要な最小限のアクセス権を持つプロキシサーバーを作成するには、次の手順に従います。

1. RES デプロイに使用する VPC のパブリックサブネットで Linux インスタンスを起動します。
   - Linux ファミリー – Amazon Linux 2 または Amazon Linux 3
   - アーキテクチャ – x86
   - インスタンスタイプ – t2.micro 以上
   - セキュリティグループ – 0.0.0.0/0 からのポート 3128 の TCP
2. インスタンスに接続してプロキシサーバーを設定します。
   a. http 接続を開きます。
   b. 関連するすべてのサブネットから次のドメインへの接続を許可します。
      - .amazonaws.com (汎用 AWS サービス用 )
      - .amazoncognito.com (Amazon Cognito 用 )
      - .awsapps.com (Identity Center 用 )
      - .signin.aws (Identity Center 用 )
      - amazonaws-us-gov..com ( Gov クラウド用 )

c. 他のすべての接続を拒否します。

d. プロキシサーバーをアクティブ化して起動します。

e. プロキシサーバーがリッスンする PORT を書き留めます。

3. プロキシサーバーへのアクセスを許可するようにルートテーブルを設定します。

   a. VPC コンソールに移動し、インフラストラクチャホストと VDI ホストに使用するサブネットのルートテーブルを特定します。

   b. ルートテーブルを編集して、前のステップで作成したプロキシサーバーインスタンスへのすべての受信接続を許可します。

   c. これは、インフラストラクチャ/VDIs に使用するすべてのサブネット (インターネットアクセスなし) のルートテーブルに対して行います。

4. プロキシサーバー EC2 インスタンスのセキュリティグループを変更し、プロキシサーバーがリッスンしている PORT でインバウンド TCP 接続が許可されていることを確認します。

## プライベート VPC デプロイパラメータを設定する

ではthe section called “ステップ 1: 製品を起動する”、 AWS CloudFormation テンプレートに特定のパラメータを入力することが期待されます。設定したプライベート VPC に正常にデプロイするには、次のパラメータを必ず指定してください。

| パラメータ | 入力 |
| --- | --- |
| InfrastructureHostAMI | で作成したインフラストラクチャ AMI ID を使用しますthe section called “Amazon マシンイメージ (AMIsを準備する”。 |
| IsLoadBalancerInternetFacing | false に設定します。 |
| LoadBalancerSubnets | インターネットにアクセスできないプライベートサブネットを選択します。 |
| InfrastructureHostSubnets | インターネットにアクセスできないプライベートサブネットを選択します。 |
| VdiSubnets | インターネットにアクセスできないプライベートサブネットを選択します。 |

| パラメータ | 入力 |
| --- | --- |
| ClientIP | VPC CIDR を選択して、すべての VPC IP アドレスへのアクセスを許可できます。 |

# デモ環境を作成する

非本番環境にデプロイしていて、外部リソースが利用できない場合は、HPC recipes スタックのデプロイから開始できます。本番環境にデプロイしていて、外部リソースが利用可能な場合は、「」にスキップできますthe section called “ステップ 1: 製品を起動する”。

外部リソースをデプロイした後、オプションで「」の手順に従ってthe section called “環境で LDAPS を設定する (オプション ) ”、デモ環境で Secure Lightweight Directory Access Protocol (LDAPS) 通信をテストできます。

## 外部リソースを作成する

この CloudFormation スタックは、ネットワーク、ストレージ、アクティブディレクトリ、ドメイン証明書 ( PortalDomainName が指定されている場合) を作成します。製品をデプロイするには、これらの外部リソースが使用可能である必要があります。

デプロイ前に recipes テンプレートをダウンロードできます。

デプロイ時間： 約 40 ~ 90 分

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。

   > **Note**
   >
   > 管理者アカウントにいることを確認します。

2. コンソールでテンプレートを起動します。

   AWS GovCloud （ 米国西部) リージョンにデプロイする場合は、 GovCloud パーティションアカウントでテンプレートを起動します。

3. テンプレートパラメータを入力します。

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| DomainName | `corp.res.com` | アクティブディレクトリに使用されるドメイン。デフォルト値は、ブートストラップユーザーを設定する LDIF ファイルで指定されます。デフォルトユーザーを使用する場合は、値をデフォルトのままにしておきます。値を変更するには、 を更新して別のLDIFファイルを指定します。これは、アクティブディレクトリに使用されるドメインと一致する必要はありません。 |
| SubDomain （ GovCloud のみ ） | | このパラメータは商用リージョンではオプションですが、 GovCloud リージョンでは必須です。<br><br>を指定すると SubDomain、パラメータには DomainName 指定された のプレフィックスが付けられます。指定された Active Directory ドメイン名はサブドメインになります。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| AdminPassword | | アクティブディレクトリ管理者のパスワード (ユーザー名 `Admin` )。このユーザーは、最初のブートストラップフェーズのアクティブディレクトリに作成され、その後は使用されません。<br><br>注：このユーザーのパスワードは、アクティブディレクトリ のパスワードの複雑さの要件を満たしている必要があります。 |
| ServiceAccountPassword | | サービスアカウントの作成に使用されるパスワード (`ReadOnlyUser` )。このアカウントは同期に使用されます。<br>注： このユーザーのパスワードは、アクティブディレクトリ のパスワードの複雑さの要件を満たしている必要があります。 |
| キーペア | | SSH クライアントを使用して管理インスタンスを接続します。<br><br>注: AWS Systems Manager Session Manager はインスタンスへの接続にも使用できます。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| LDIFS3Path | `aws-hpc-recipes/main/recipes/res/res_demo_env/assets/res.ldif` | アクティブディレクトリセットアップのブートストラップフェーズ中にインポートされた LDIF ファイルへの Amazon S3 パス。詳細については、「LDIF サポート」を参照してください。パラメータには、アクティブディレクトリに多数のユーザーを作成するファイルが事前に入力されています。<br><br>ファイルを表示するには、で利用可能な res.ldif ファイルを参照してください GitHub。 |
| ClientIpCidr | | サイトにアクセスする IP アドレス。例えば、IP アドレスを選択し、`[IPADDRESS]/32` を使用してホストからのアクセスのみを許可できます。このデプロイ後に更新できます。 |
| ClientPrefixList | | プレフィックスリストを入力して、アクティブディレクトリ管理ノードへのアクセスを提供します。マネージドプレフィックスリストの作成については、「カスタマーマネージドプレフィックスリストの操作」を参照してください。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| EnvironmentName | `res-[environment name]` | `PortalDomainName` が指定されている場合、このパラメータを使用して生成されたシークレットにタグを追加し、環境内で使用できるようにします。これは、RES スタックの作成時に使用する `EnvironmentName` パラメータと一致する必要があります。アカウントに複数の環境をデプロイする場合、これは一意である必要があります。 |
| PortalDomainName | | GovCloud デプロイの場合は、このパラメータを入力しないでください。証明書とシークレットは、前提条件に従って手動で作成されました。<br>アカウントの Amazon Route 53 のドメイン名。これを指定すると、パブリック証明書とキーファイルが生成され、 にアップロードされます AWS Secrets Manager。独自のドメインと証明書がある場合は、このパラメータ とを空白のままに`EnvironmentName` することができます。 |

4. 機能 のすべてのチェックボックスを確認し、スタックの作成 を選択します。

# ステップ 1: 製品を起動する

このセクション step-by-step の指示に従って、製品を設定してアカウントにデプロイします。

デプロイ時間： 約 60 分

この製品の CloudFormation テンプレートは、デプロイする前にダウンロードできます。

AWS GovCloud （ 米国西部) にデプロイする場合は、このテンプレート を使用します。

res-stack - このテンプレートを使用して、製品および関連するすべてのコンポーネントを起動します。デフォルト設定では、RES メインスタックと認証、フロントエンド、バックエンドリソースがデプロイされます。

> **Note**
>
> AWS CloudFormation リソースは AWS Cloud Development Kit (AWS CDK) （ AWS CDK) コンストラクトから作成されます。

AWS CloudFormation テンプレートは、 の AWS に Research and Engineering Studio をデプロイします AWS クラウド。スタックを起動する前に、前提条件を満たす必要があります。

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。
2. テンプレート を起動します。

   AWS GovCloud （ 米国西部) にデプロイするには、このテンプレート を起動します。
3. テンプレートはデフォルトで米国東部 (バージニア北部) リージョンで起動します。別の でソリューションを起動するには AWS リージョン、コンソールナビゲーションバーのリージョンセレクターを使用します。

   > **Note**
   >
   > この製品は Amazon Cognito サービスを使用していますが、現在すべての で利用できるわけではありません AWS リージョン。この製品は、Amazon Cognito AWS リージョンが利用可能な で起動する必要があります。リージョン別の最新の可用性については、「AWS リージョンサービスリスト」を参照してください。

4. パラメータ で、この製品テンプレートのパラメータを確認し、必要に応じて変更します。自動外部リソースをデプロイした場合、これらのパラメータは外部リソーススタックの出力タブにあります。

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| EnvironmentName | *#res-demo >* | res- で始まり、11 文字以下の RES 環境に与えられる一意の名前。 |
| AdministratorEmail | | 製品のセットアップを完了するユーザーの E メールアドレス。さらに、このユーザーは、Active Directory のシングルサインオン統合に障害が発生した場合に、Break Glass ユーザーとして機能します。 |
| InfrastructureHostAMI | ami-*########## | （オプション）すべてのインフラストラクチャホストに使用するカスタム AMI ID を指定できます。現在サポートされているベース OS は Amazon Linux 2 です。詳細については、「the section called “RES 対応 AMIs の設定”」を参照してください。 |
| SSHKeyPair | | インフラストラクチャホストへの接続に使用されるキーペア。 |
| ClientIP | *x.x.x* .0/24 または *x.x.x* .0/32 | システムへの接続を制限する IP アドレスフィルター。デプロイ ClientIpCidr 後に を更新できます。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| ClientPrefixList | | （オプション）ウェブ UI に直接アクセスし、踏み台ホストに SSH 接続できる IPs のマネージドプレフィックスリストを指定します。 |
| IAMPermissionBoundary | | （オプション）RES で作成されたすべてのロールにアクセス許可の境界としてアタッチされる管理ポリシー ARN を指定できます。詳細については、「the section called “カスタムアクセス許可の境界の設定”」を参照してください。 |
| VpcId | | インスタンスが起動する VPC の IP。 |
| IsLoadBalancerInternetFacing | | インターネット向けロードバランサーをデプロイするには true を選択します (ロードバランサーにはパブリックサブネットが必要です）。制限されたインターネットアクセスを必要とするデプロイの場合は、false を選択します。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| LoadBalancerSubnets | | ロードバランサーが起動する異なるアベイラビリティーゾーンで、少なくとも 2 つのサブネットを選択します。制限されたインターネットアクセスを必要とするデプロイでは、プライベートサブネットを選択します。インターネットアクセスが必要なデプロイの場合は、パブリックサブネットを選択します。外部ネットワークスタックによって 3 つ以上作成された場合は、作成されたすべての を選択します。 |
| InfrastructureHostSubnets | | インフラストラクチャホストが起動する異なるアベイラビリティーゾーンで、少なくとも 2 つのプライベートサブネットを選択します。外部ネットワークスタックによって 3 つ以上作成された場合は、作成されたすべての を選択します。 |
| VdiSubnets | | VDI インスタンスが起動する異なるアベイラビリティーゾーンで、少なくとも 2 つのプライベートサブネットを選択します。外部ネットワークスタックによって 3 つ以上作成された場合は、作成されたすべての を選択します。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| ActiveDirectoryName | *corp.res.com* | アクティブディレクトリのドメイン。ポータルドメイン名と一致する必要はありません。 |
| ADShortName | *corp* | アクティブディレクトリの短縮名。これは NetBIOS 名とも呼ばれます。 |
| LDAP ベース | ***DC=corp,DC=res,DC=com*** | LDAP 階層内のベースへの LDAP パス。 |
| LDAPConnectionURI | | アクティブディレクトリのホストサーバーからアクセスできる単一の ldap:// パス。デフォルトの AD ドメインで自動化された外部リソースをデプロイした場合は、ldap://corp.res.com を使用できます。 |
| ServiceAccountUserName | ServiceAccount | AD への接続に使用されるサービスアカウントのユーザー名。このアカウントには、ComputersOU 内にコンピュータを作成するためのアクセス権が必要です。 |
| ServiceAccountPassword | | 用に作成されたパスワード ServiceAccountUserName。 |
| UsersOU | | 同期するユーザーの AD 内の組織単位。 |
| GroupsOU | | 同期するグループの AD 内の組織単位。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| SudoersOU | | グローバル sudoers の AD 内の組織単位。 |
| SudoersGroupName | RESAdministrators | インストール時にインスタンスで sudoer アクセスを持つすべてのユーザーと RES で管理者アクセスを含むグループ名。 |
| ComputersOU | | インスタンスが参加する AD 内の組織単位。 |
| DomainTLSCertificateSecretARN | | （オプション）AD への TLS 通信を有効にするドメイン TLS 証明書シークレット ARN を指定します。 |
| EnableLdapIDMapping | | UID 番号と GID 番号が SSSD によって生成されるか、AD によって提供される番号を使用するかを決定します。SSSD が生成した UID と GID を使用するには True、AD が提供する UID と GID を使用するには False に設定します。 |
| DisableADJoin | False | Linux ホストがディレクトリドメインに参加しないようにするには、 を True に変更します。それ以外の場合は、デフォルト設定の False のままにします。 |

| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| ServiceAccountUserDN | | Directory のサービスアカウントユーザーの識別名 (DN) を指定します。 |
| SharedHomeFilesystemID | | Linux VDI ホストの共有ホームファイルシステムに使用する EFS ID。 |
| CustomDomainNamefo rWebApp | | （オプション）システムのウェブ部分へのリンクを提供するためにウェブポータルで使用されるサブドメイン。 |
| CustomDomainNameforVDI | | （オプション）システムの VDI 部分へのリンクを提供するためにウェブポータルで使用されるサブドメイン。 |
| ACMCertificateARNf orWebApp | | （オプション）デフォルト設定を使用する場合、製品はドメイン amazonaws.com でウェブアプリケーションをホストします。ドメインで製品サービスをホストできます。自動外部リソースをデプロイした場合、これは自動的に生成され、情報は res-bi スタックの出力にあります。ウェブアプリケーションの証明書を生成する必要がある場合は、「」を参照してください設定ガイド。 |

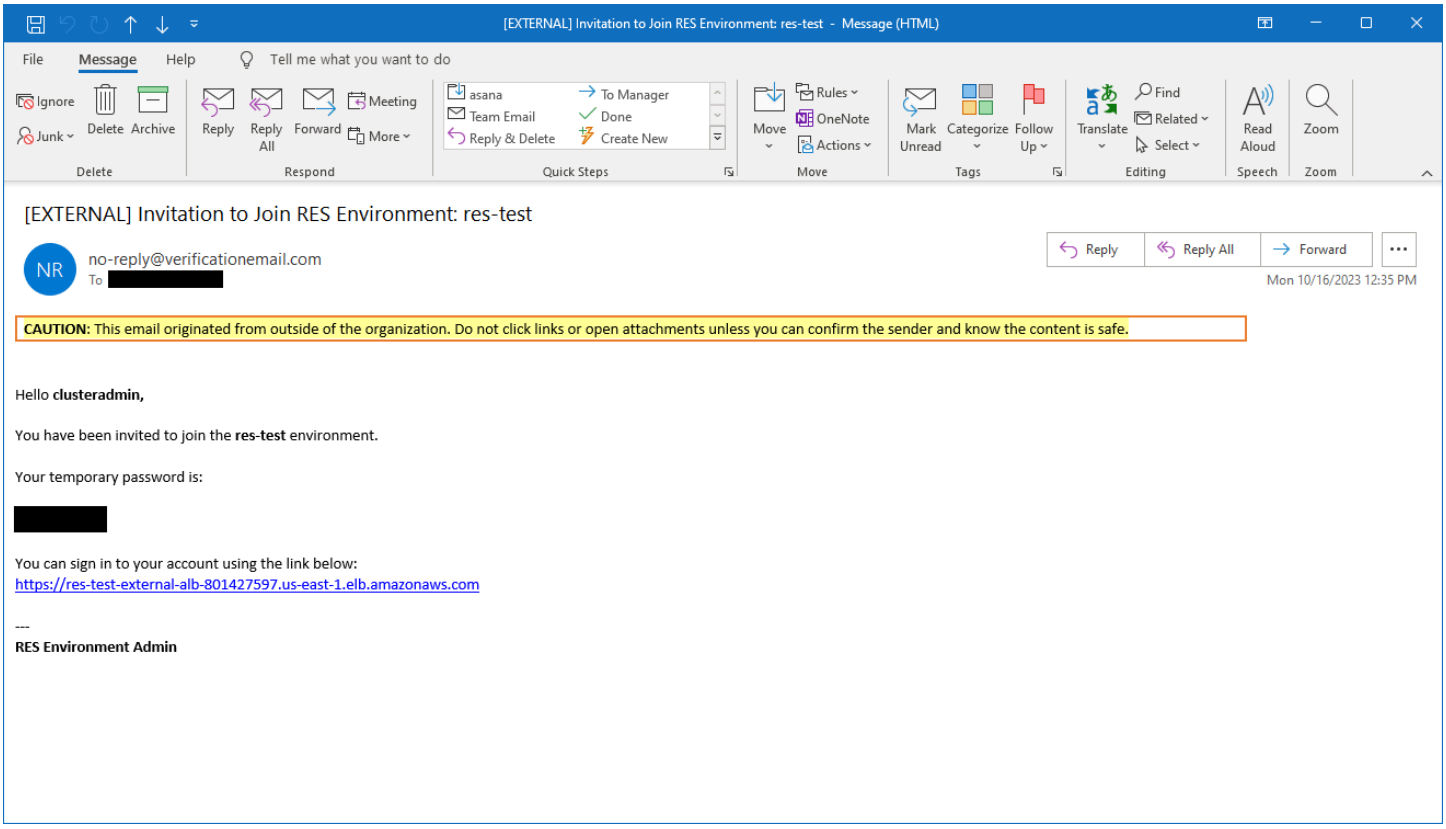
| パラメータ | デフォルト | [Description] (説明) |
| --- | --- | --- |
| CertificateSecretARNforVDI | | （オプション）この ARN シークレットは、ウェブポータルのパブリック証明書のパブリック証明書を保存します。自動外部リソースにポータルドメイン名を設定すると、res-bi スタックの出力タブにこの値が表示されます。 |
| PrivateKeySecretARNforVDI | | （オプション）この ARN シークレットは、ウェブポータルの証明書のプライベートキーを保存します。自動外部リソースにポータルドメイン名を設定すると、res-bi スタックの出力タブにこの値が表示されます。 |

5. [Create stack] (スタックの作成) を選択してスタックをデプロイします。

スタックのステータスは、 AWS CloudFormation コンソールの ステータス 列で表示できます。約 60 分後に CREATE_COMPLETE ステータスが表示されます。

# ステップ 2: 初めてサインインする

製品スタックがアカウントにデプロイされると、認証情報が記載された E メールが届きます。URL を使用してアカウントにサインインし、他のユーザーのワークスペースを設定します。




初めてサインインしたら、 SSO プロバイダーに接続するための設定をウェブポータルで設定できます。デプロイ後の設定情報については、「」を参照してください設定ガイド。

# 製品を更新する

Research and Engineering Studio (RES) には、バージョン更新がメジャーかマイナーかによって異なる 2 つの更新方法があります。

RES は日付ベースのバージョニングスキームを使用します。メジャーリリースでは年と月が使用され、マイナーリリースでは必要に応じてシーケンス番号が追加されます。例えば、バージョン 2024.01 はメジャーリリースとして 2024 年 1 月にリリースされ、バージョン 2024.01.01 はそのバージョンのマイナーリリース更新でした。

トピック



## メジャーバージョンの更新

Research and Engineering Studio は、スナップショットを使用して、環境設定を失うことなく、以前の RES 環境から最新の環境への移行をサポートします。このプロセスを使用して、ユーザーをオンボーディングする前に環境の更新をテストおよび検証することもできます。

環境を最新バージョンの RES で更新するには：

1. 現在の環境のスナップショットを作成します。the section called “スナップショットを作成する” を参照してください。
2. 新しいバージョンで RES を再デプロイします。the section called “ステップ 1: 製品を起動する” を参照してください。
3. スナップショットを更新された環境に適用します。the section called “スナップショットの適用” を参照してください。
4. すべてのデータが新しい環境に正常に移行されたことを確認します。

## マイナーバージョンの更新

RES のマイナーバージョン更新の場合、新しいインストールは必要ありません。テンプレートを更新することで、既存の RES スタックを更新できます AWS CloudFormation 。更新をデプロイ AWS CloudFormation する前に、 で現在の RES 環境のバージョンを確認してください。バージョン番号はテンプレートの先頭にあります。

例："Description": "RES_2024.1"

マイナーバージョンを更新するには：

1. 最新の AWS CloudFormation テンプレートを にダウンロードしますthe section called “ステップ 1: 製品を起動する”。
2. https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。
3. スタック から、プライマリスタックを検索して選択します。として表示されます*<stack-name>*。
4. [更新] を選択します。
5. 現在のテンプレートを置き換える を選択します。
6. [テンプレートソース] で、[テンプレートファイルのアップロード] を選択します。
7. ファイルの選択を選択し、ダウンロードしたテンプレートをアップロードします。
8. スタックの詳細を指定する で、次へ を選択します。パラメータを更新する必要はありません。
9. スタックオプションの設定 で、次へ を選択します。
10. 「<stack-name>の確認」で、「送信」を選択します。

# 製品をアンインストールする

AWS 製品上の Research and Engineering Studio は、 AWS Management Console またはを使用してアンインストールできます AWS Command Line Interface。この製品によって作成された Amazon Simple Storage Service (Amazon S3) バケットを手動で削除する必要があります。保持するデータを保存している場合でも、shared-storage-security-group この製品では < EnvironmentName >-が自動的に削除されることはありません。

## を使用する AWS Management Console

1. AWS CloudFormation コンソール にサインインします。
2. Stacks ページで、この製品のインストールスタックを選択します。
3. [削除] をクリックします。

## を使用する AWS Command Line Interface

AWS Command Line Interface (AWS CLI) がご使用の環境で使用できるかどうかを確認します。インストール手順については、『AWS CLI ユーザーガイド』の AWS Command Line Interface「What Is the」を参照してください。 AWS CLI 製品がデプロイされたリージョンの管理者アカウントでが使用可能であり、設定されていることを確認したら、次のコマンドを実行します。

```
$ aws cloudformation delete-stack --stack-name
<RES-stack-name>
```

## を削除します。 shared-storage-security-group

> **Warning**
>
> 意図しないデータ損失を防ぐため、製品はこのファイルシステムをデフォルトで保持します。セキュリティグループと関連するファイルシステムを削除すると、それらのシステム内に保持されているデータはすべて完全に削除されます。データをバックアップするか、新しいセキュリティグループにデータを再割り当てすることをお勧めします。

1. AWS Management Console にサインインし、https://console.aws.amazon.com/efs/ にある Amazon EFS コンソールを開きます。
2. <RES-stack-name>-shared-storage-security-group に関連するすべてのファイルシステムを削除します。あるいは、これらのファイルシステムを別のセキュリティグループに再割り当てしてデータを管理することもできます。
3. AWS Management Console にサインインし、https://console.aws.amazon.com/ec2/ にある Amazon EC2 コンソールを開きます。
4. <RES-stack-name>-を削除しますshared-storage-security-group。

## Amazon S3 バケットを削除する

この製品は、 AWS CloudFormation 偶発的なデータ損失を防ぐためにスタックを削除する場合でも、製品が作成した Amazon S3 バケット (オプトインリージョンでのデプロイ用) を保持するように設定されています。製品をアンインストールした後、データを保持する必要がない場合は、この S3 バケットを手動で削除できます。Amazon S3 バケットを削除するには、次の手順に従います。

1. AWS Management Console にサインインし、https://console.aws.amazon.com/s3/ にある Amazon S3 コンソールを開きます。
2. ナビゲーションペインから [バケット] を選択します。
3. `stack-name`S3 バケットを探します。
4. Amazon S3 バケットをそれぞれ選択し、[空にする] を選択します。各バケットを空にする必要があります。
5. S3 バケットを選択し、[Delete] を選択します。

を使用して S3 バケットを削除するには AWS CLI、以下のコマンドを実行します。

```
$ aws s3 rb s3://<bucket-name> --force
```

> **Note**
>
> `--force`このコマンドはバケットの内容を空にします。

# 設定ガイド

この設定ガイドでは、 AWS 製品で Research and Engineering Studio をさらにカスタマイズして統合する方法に関するデプロイ後の手順を、技術担当者に提供します。

トピック



# ユーザーとグループの管理

Research and Engineering Studio は、SAML 2.0 準拠の任意の ID プロバイダーを使用できます。外部リソースを使用して RES をデプロイした場合、または IAM Identity Center を使用する予定がある場合は、「」を参照してくださいthe section called “IAM Identity Center で SSO を設定する”。独自の SAML 2.0 準拠の ID プロバイダーがある場合は、「」を参照してくださいthe section called “シングルサインオン (SSO) 用の ID プロバイダーの設定”。

トピック



## IAM Identity Center で SSO を設定する

マネージドアクティブディレクトリに接続しているアイデンティティセンターがまだない場合は、から始めますthe section called “アイデンティティセンターをセットアップする”。マネージドアクティブディレクトリに接続されたアイデンティティセンターが既にある場合は、 から始めますthe section called “アイデンティティセンターに接続する”。

> **Note**
>
> AWS GovCloud （米国西部) リージョンにデプロイする場合は、Research and Engineering Studio をデプロイした AWS GovCloud (US) パーティションアカウントに SSO を設定します。

## ステップ 1: アイデンティティセンターを設定する

### ID センターの有効化

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/iam/ で IAM コンソールを開きます。
2. Identity Center を開きます。
3. [Enable（有効化）] を選択します。
4. で を有効にする AWS Organizationsを選択します。
5. [Continue] を選択します。

> **Note**
>
> マネージドアクティブディレクトリがあるのと同じリージョンにいることを確認します。

### ID センターをマネージドアクティブディレクトリに接続する

ID センターを有効にしたら、以下の推奨セットアップ手順を完了します。

1. ナビゲーションから、設定 を選択します。
2. ID ソース で、アクション を選択し、ID ソース の変更 を選択します。
3. 既存のディレクトリ で、ディレクトリを選択します。
4. [次へ] をクリックします。
5. 変更を確認し、確認ボックスに **ACCEPT** と入力します。
6. [Change identity source] (ID ソースの変更) を選択します。

### ユーザーとグループのアイデンティティセンターへの同期

変更the section called “ID センターをマネージドアクティブディレクトリに接続する”が完了すると、緑色のバナーが表示されます。

1. 確認バナーで、ガイド付きセットアップの開始 を選択します。
2. 属性マッピングの設定 から、次へ を選択します。
3. ユーザー セクションで、同期するユーザーを入力します。
4. [追加] を選択します。
5. [Next (次へ) ] を選択します。
6. 変更を確認し、設定の保存 を選択します。
7. 同期プロセスには数分かかる場合があります。ユーザーが同期していないという警告メッセージが表示された場合は、同期を再開 を選択します。

### ユーザーの有効化

1. メニューから、ユーザー を選択します。
2. アクセスを有効にするユーザー (複数可) を選択します。
3. ユーザーアクセスを有効にする を選択します。

## ステップ 2: アイデンティティセンターに接続する

### Identity Center でのアプリケーションのセットアップ

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/singlesignon/ で IAM Identity Center を開きます。
2. [Applications] (アプリケーション) を選択します。
3. [アプリケーションの追加] を選択します。
4. セットアップ設定 で、 を設定するアプリケーションがあるを選択します。
5. [アプリケーションタイプ] で、[SAML 2.0] を選択します。
6. [次へ] をクリックします。
7. 使用する表示名と説明を入力します。
8. IAM Identity Center メタデータ で、IAM Identity Center SAML メタデータファイルのリンクをコピーします。これは、RES ポータルで SSO を設定するときに必要になります。

9. アプリケーションプロパティ に、アプリケーション開始 URL を入力します。例えば、 <your-portal-domain>/sso などです。
10. Application ACS URL に、RES ポータルからのリダイレクト URL を入力します。これを見つけるには：

    a. 環境管理 で、全般設定 を選択します。

    b. [Identity provider] タブを選択します。

    c. Single Sign-On の下に、SAML リダイレクト URL が表示されます。
11. アプリケーション SAML 対象者 に、Amazon Cognito URN を入力します。URL を作成するには：

    a. RES ポータルから、全般設定 を開きます。

    b. ID プロバイダータブが表示されたら、ユーザープール ID を見つけます。

    c. ユーザープール ID をこの文字列に追加します。

    ```
    urn:amazon:cognito:sp:<user_pool_id>
    ```
12. [送信] を選択します。

アプリケーションの属性マッピングの設定

1. Identity Center から、作成したアプリケーションの詳細を開きます。
2. アクション を選択し、属性マッピングの編集 を選択します。
3. 件名 に ${user:email} と入力します。
4. フォーマット で、emailAddress を選択します。
5. [新規属性マッピングの追加] を選択します。
6. アプリケーションのユーザー属性 に、E メールを入力します。
7. IAM Identity Center のこの文字列値またはユーザー属性にマップ で、${user:email} と入力します。
8. フォーマット で、未指定を入力します。
9. [変更を保存] を選択します。

Identity Center でのアプリケーションへのユーザーの追加

1. Identity Center から、作成したアプリケーションの割り当て済みユーザーを開き、ユーザーの割り当て を選択します。
2. アプリケーションアクセスを割り当てるユーザーを選択します。
3. [ユーザーの割り当て] を選択します。

RES 環境内での SSO の設定

1. Research and Engineering Studio 環境から、「環境管理」の「一般設定」を開きます。
2. ID プロバイダータブを開きます。
3. Single Sign-On で、ステータス の横にある編集ボタンを選択します。
4. フォームに次の情報を入力します。
    a. SAML を選択します。
    b. プロバイダー名 に、わかりやすい名前を入力します。
    c. メタデータドキュメントエンドポイント URL を入力 を選択します。
    d. コピーした URL を入力します。 the section called “Identity Center でのアプリケーションのセットアップ”
    e. プロバイダー E メール属性 に E メールを入力します。
    f. [送信] を選択します。
5. ページを更新し、ステータスが有効として表示されることを確認します。

## シングルサインオン (SSO) 用の ID プロバイダーの設定

Research and Engineering Studio は、任意の SAML 2.0 ID プロバイダーと統合して、RES ポータルへのユーザーアクセスを認証します。これらのステップでは、選択した SAML 2.0 ID プロバイダーと統合する手順を説明します。IAM Identity Center を使用する場合は、「」を参照してくださいthe section called “IAM Identity Center で SSO を設定する”。

> **Note**
>
> ユーザーの E メールは、IDP SAML アサーションと Active Directory で一致する必要があります。ID プロバイダーを Active Directory に接続し、ユーザーを定期的に同期する必要があります。

トピック



## ID プロバイダーを設定する

このセクションでは、RES Amazon Cognito ユーザープールからの情報を使用して ID プロバイダーを設定する手順について説明します。

1. RES は、RES ポータルとプロジェクトへのアクセスが許可されているユーザー ID を持つ AD (AWS マネージド AD またはセルフプロビジョニング AD) があることを前提としています。AD を ID サービスプロバイダーに接続し、ユーザー ID を同期します。AD を接続し、ユーザー ID を同期する方法については、ID プロバイダーのドキュメントを参照してください。例えば、 ユーザーガイドの「ID ソースとしての Active Directory の使用AWS IAM Identity Center 」を参照してください。
2. ID プロバイダー (IdP) で RES 用の SAML 2.0 アプリケーションを設定します。この設定には、次のパラメータが必要です。
   - SAML リダイレクト URL — IdP が SAML 2.0 レスポンスをサービスプロバイダーに送信するために使用する URL。

     > **Note**
     >
     > IdP によっては、SAML リダイレクト URL の名前が異なる場合があります。
     >
     > - アプリケーション URL
     > - アサーションコンシューマーサービス (ACS) URL
     > - ACS POST バインディング URL

     URL を取得するには

     1. 管理者または clusteradmin として RES にサインインします。
     2. 「環境管理」「一般設定」「ID プロバイダー」に移動します。
     3. SAML リダイレクト URL を選択します。

- SAML オーディエンス URI — サービスプロバイダー側の SAML オーディエンスエンティティの一意の ID。

  > **Note**
  >
  > IdP によっては、SAML オーディエンス URI の名前が異なる場合があります。
  >
  > - ClientID
  > - アプリケーション SAML 対象者
  > - SP エンティティ ID

  入力を次の形式で入力します。

  ```
  urn:amazon:cognito:sp:user-pool-id
  ```

  SAML オーディエンス URI を検索するには

  1. 管理者または clusteradmin として RES にサインインします。
  2. 「環境管理」「一般設定」「ID プロバイダー」に移動します。
  3. ユーザープール ID を選択します。

3. RES に投稿される SAML アサーションには、次のフィールド/クレームがユーザーの E メールアドレスに設定されている必要があります。
   - SAML サブジェクトまたは NameID
   - SAML E メール
4. IdP は、設定に基づいて SAML アサーションにフィールド/クレームを追加します。RES にはこれらのフィールドが必要です。ほとんどのプロバイダーは、デフォルトでこれらのフィールドを自動的に入力します。設定する必要がある場合は、次のフィールドの入力と値を参照してください。

   - AudienceRestriction — を に設定しますurn:amazon:cognito:sp:*user-pool-id*。を Amazon Cognito ユーザープールの ID *user-pool-id*に置き換えます。

     ```
     <saml:AudienceRestriction>
     ```

```
    <saml:Audience> urn:amazon:cognito:sp:user-pool-id
</saml:AudienceRestriction>
```

- レスポンス — InResponseToに設定しますhttps://*user-pool-domain*/saml2/idpresponse。を Amazon Cognito ユーザープールのドメイン名*user-pool-domain*に置き換えます。

```
<saml2p:Response
  Destination="http://user-pool-domain/saml2/idpresponse"
  ID="id123"
  InResponseTo="_dd0a3436-bc64-4679-a0c2-cb4454f04184"
  IssueInstant="Date-time stamp"
  Version="2.0"
  xmlns:saml2p="urn:oasis:names:tc:SAML:2.0:protocol"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

- SubjectConfirmationData — ユーザープールsaml2/idpresponseエンドポイントRecipientに を、元の SAML リクエスト ID InResponseToに設定します。

```
<saml2:SubjectConfirmationData
  InResponseTo="_dd0a3436-bc64-4679-a0c2-cb4454f04184"
  NotOnOrAfter="Date-time stamp"
  Recipient="https://user-pool-domain/saml2/idpresponse"/>
```

- AuthnStatement — 次のように を設定します。

```
<saml2:AuthnStatement AuthnInstant="2016-10-30T13:13:28.152TZ"
  SessionIndex="32413b2e54db89c764fb96ya2k"
 SessionNotOnOrAfter="2016-10-30T13:13:28">
    <saml2:SubjectLocality />
    <saml2:AuthnContext>

 <saml2:AuthnContextClassRef>urn:oasis:names:tc:SAML:2.0:ac:classes:Password</
saml2:AuthnContextClassRef>
    </saml2:AuthnContext>
</saml2:AuthnStatement>
```

5. SAML アプリケーションにログアウト URL フィールドがある場合は、 に設定します<*domain-url*>/saml2/logout。

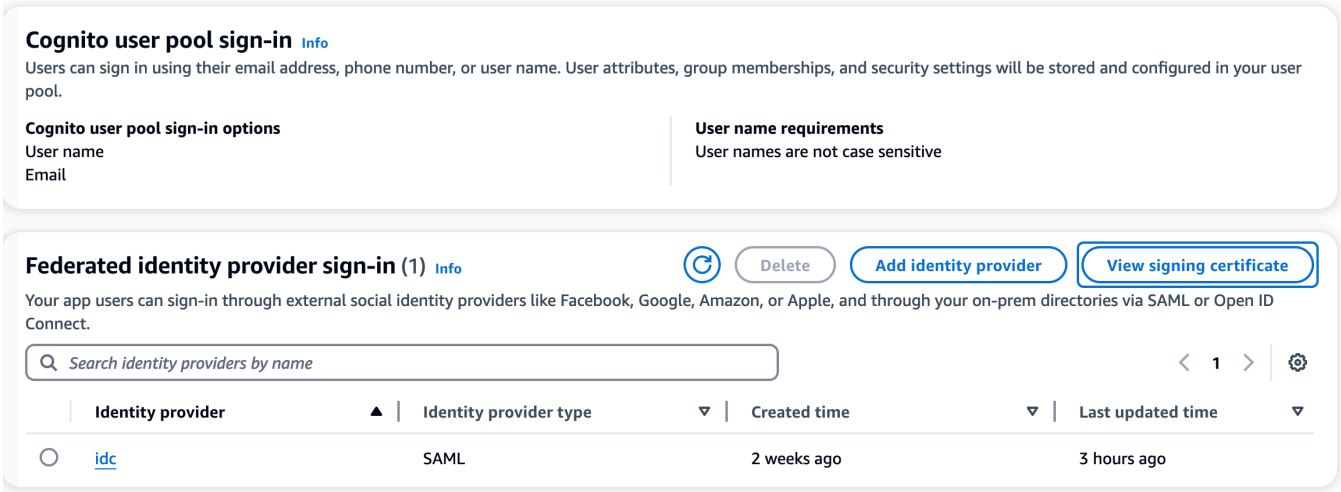
ドメイン URL を取得するには

1. 管理者または clusteradmin として RES にサインインします。
2. 「環境管理」「一般設定」「ID プロバイダー」に移動します。
3. ドメイン URL を選択します。

6. IdP が Amazon Cognito との信頼を確立するために署名証明書を受け入れる場合は、Amazon Cognito 署名証明書をダウンロードして IdP にアップロードします。

署名証明書を取得するには

1. 入門ガイドの Amazon Cognito コンソールを開きます。 AWS Management Console
2. ユーザープールを選択します。ユーザープールは である必要がありますres-`<environment name>`-user-pool。
3. [Sign-in experience] (サインインエクスペリエンス) タブを選択します。
4. フェデレーティッド ID プロバイダーのサインインセクションで、署名証明書の表示 を選択します。




この証明書を使用して、Active Directory IDP をセットアップし、 を追加しrelying party trust、この証明書利用者に対して SAML サポートを有効にできます。

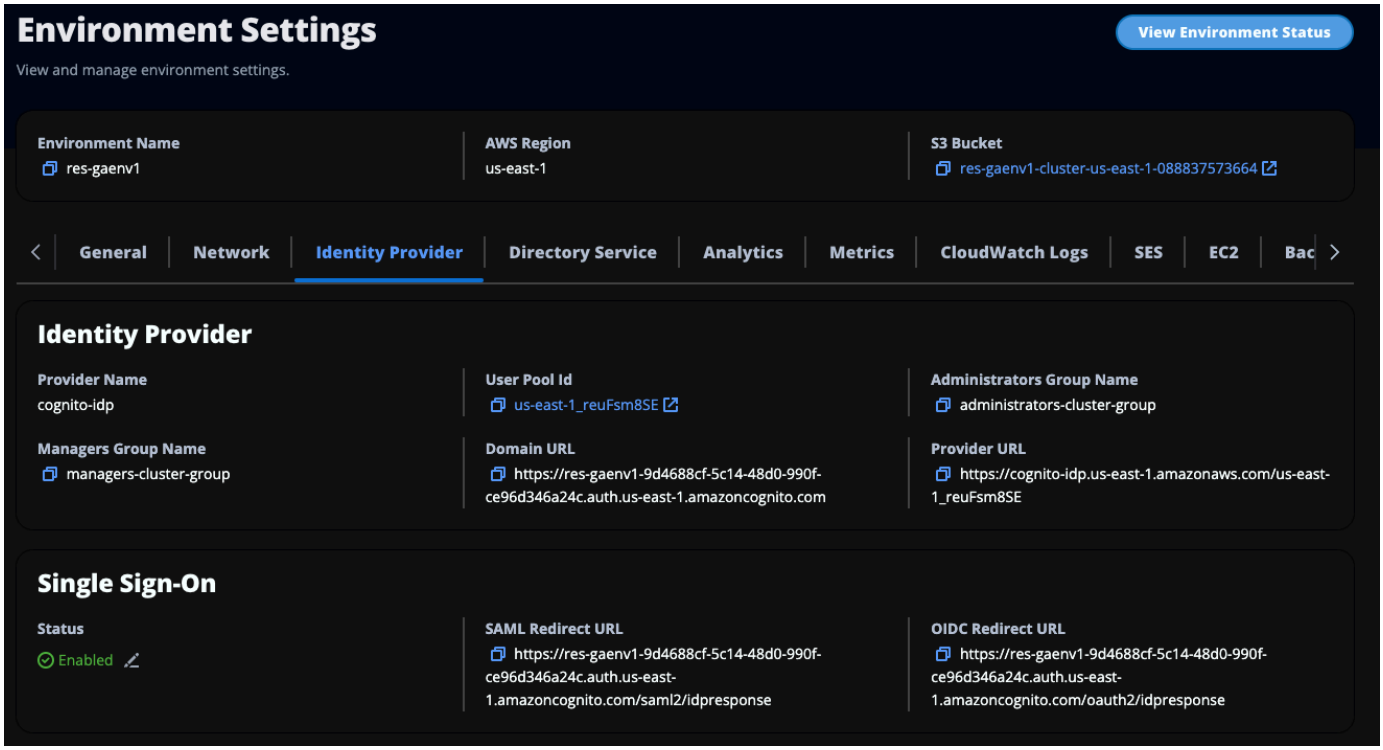
> **Note**
>
> これは Keycloak と IDC には適用されません。

5. アプリケーションのセットアップが完了したら、SAML 2.0 アプリケーションメタデータ XML または URL をダウンロードします。次のセクションで使用します。

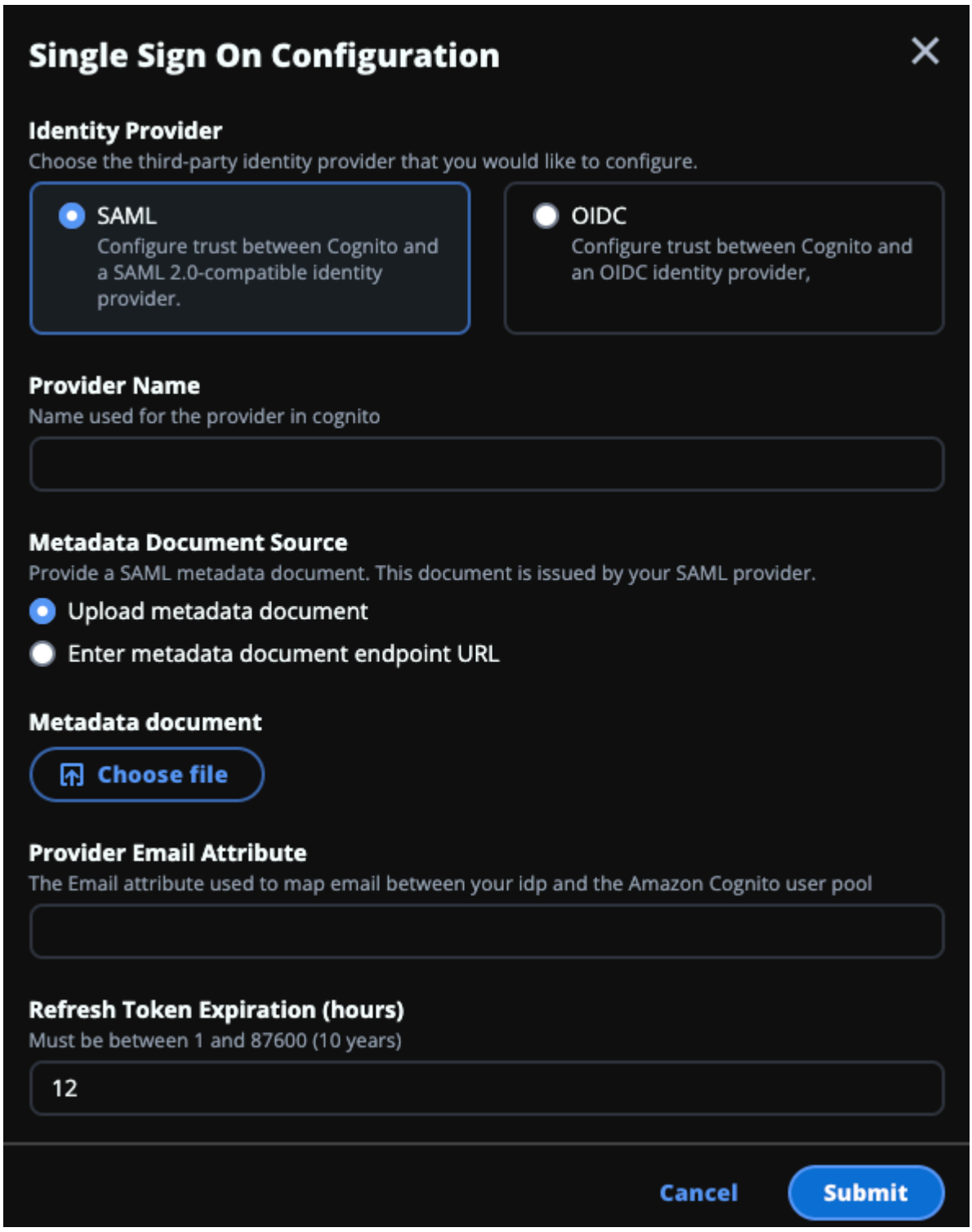
## ID プロバイダーを使用するように RES を設定する

RES のシングルサインオン設定を完了するには

1. 管理者または clusteradmin として RES にサインインします。
2. 「環境管理」「一般設定」「ID プロバイダー」に移動します。




3. Single Sign-On で、ステータスインジケータの横にある編集アイコンを選択し、 Single Sign-On 設定ページを開きます。




a. ID プロバイダー で、SAML を選択します。

b. プロバイダー名 には、ID プロバイダーの一意の名前を入力します。

> **Note**
>
> 次の名前は使用できません。
>
> - Cognito
> - IdentityCenter

c. メタデータドキュメントソース で、適切なオプションを選択し、メタデータ XML ドキュメントをアップロードするか、ID プロバイダーから URL を指定します。

d. プロバイダー E メール属性 には、テキスト値 を入力しますemail。

e. [送信] を選択します。

4. 環境設定ページを再ロードします。設定が正しい場合、シングルサインオンが有効になります。

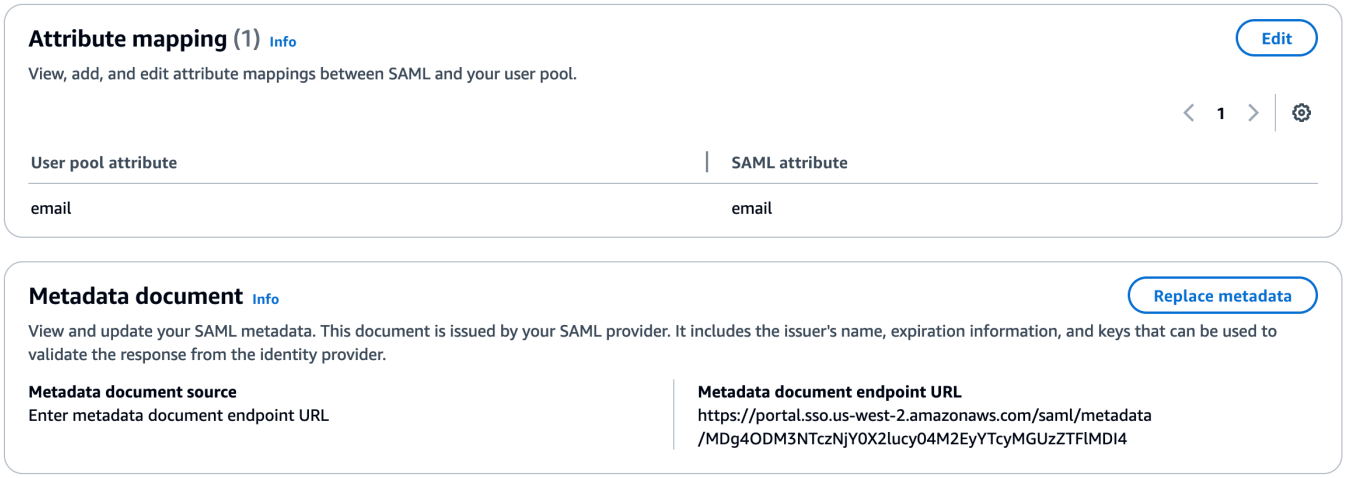
## 非本番環境での ID プロバイダーの設定

提供された外部リソースを使用して非本番環境の RES 環境を作成し、IAM Identity Center を ID プロバイダーとして設定した場合は、Okta などの別の ID プロバイダーを設定することをお勧めします。RES SSO 有効化フォームでは、次の 3 つの設定パラメータを要求します。

1. プロバイダー名 — 変更できません
2. メタデータドキュメントまたは URL — 変更可能
3. プロバイダー E メール属性 — 変更可能

メタデータドキュメントとプロバイダーの E メール属性を変更するには、次の手順を実行します。

1. Amazon Cognito コンソールに移動します。
2. ナビゲーションから、ユーザープール を選択します。
3. ユーザープールを選択すると、ユーザープールの概要が表示されます。
4. サインインエクスペリエンスタブから、フェデレーティッド ID プロバイダーのサインインに移動し、設定した ID プロバイダーを開きます。
5. 通常、メタデータを変更し、属性マッピングを変更しないだけで済みます。属性マッピング を更新するには、編集 を選択します。メタデータドキュメント を更新するには、メタデータ の置き換え を選択します。




6. 属性マッピングを編集した場合は、DynamoDB の`<environment name>.cluster-settings`テーブルを更新する必要があります。

   a. DynamoDB コンソールを開き、ナビゲーションからテーブルを選択します。

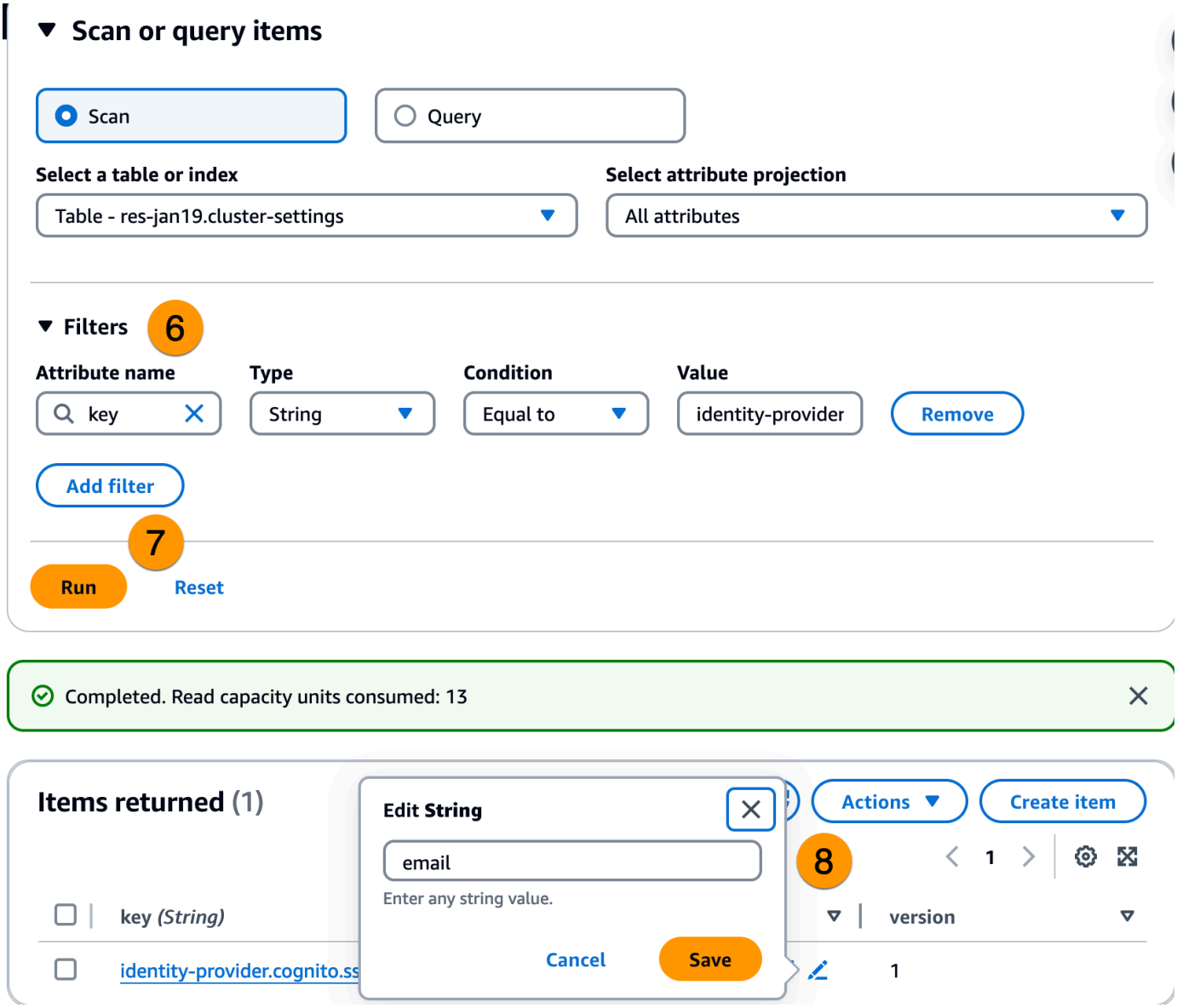
   b. `<environment name>.cluster-settings` テーブルを検索して選択し、アクションメニューから項目を探索を選択します。

   c. スキャンまたはクエリ項目 で、フィルター に移動し、次のパラメータを入力します。

      - 属性名 — `key`
      - 値 — `identity-provider.cognito.sso_idp_provider_email_attribute`

   d. [実行] を選択します。

7. 「返されたアイテム`identity-provider.cognito.sso_idp_provider_email_attribute`」で文字列を検索し、「編集」を選択して、Amazon Cognito の変更と一致するように文字列を変更します。




## SAML IdP の問題のデバッグ

SAML トレーサー — この拡張機能を Chrome ブラウザで使用して、SAML リクエストを追跡し、SAML アサーション値を確認できます。詳細については、Chrome ウェブストアの「SAML トレーサー」を参照してください。

SAML デベロッパーツール — SAML エンコードされた値をデコードし、SAML アサーションの必須フィールドをチェックするために使用できるツール OneLogin を提供します。詳細については、OneLogin ウェブサイトの「Base 64 Decode + Inflate」を参照してください。

Amazon CloudWatch Logs — RES ログは CloudWatch 、ログでエラーや警告を確認できます。ログは、 という名前のロググループにあります*res-environment-name*/cluster-manager。

Amazon Cognito ドキュメント — Amazon Amazon Cognito デベロッパーガイド」の「ユーザープールへの SAML ID プロバイダーの追加」を参照してください。 Amazon Cognito

## ユーザーのパスワードの設定

1. AWS Directory Service コンソールから、作成したスタックのディレクトリを選択します。
2. アクションメニューで、ユーザーパスワードのリセット を選択します。
3. ユーザーを選択し、新しいパスワードを入力します。
4. 「パスワードのリセット」を選択します。

# サブドメインの作成

カスタムドメインを使用している場合は、ポータルのウェブ部分と VDI 部分をサポートするようにサブドメインを設定する必要があります。

> **Note**
>
> AWS GovCloud （ 米国西部) リージョンにデプロイする場合は、ドメインパブリックホストゾーンをホストする商用パーティションアカウントでウェブアプリケーションと VDI サブドメインを設定します。

1. にサインイン AWS Management Console し、https://console.aws.amazon.com/route53/ で Route 53 コンソールを開きます。
2. 作成したドメインを検索し、レコードの作成 を選択します。
3. レコード名としてウェブを入力します。
4. レコードタイプとして CNAME を選択します。
5. 値 には、最初の E メールで受け取ったリンクを入力します。
6. [レコードを作成] を選択します。
7. のレコードを作成するには、NLB アドレスを取得します。

    a. にサインイン AWS Management Console し、https://console.aws.amazon.com/cloudformation で AWS CloudFormation コンソールを開きます。

b. [<environment-name>-vdc] を選択します。

c. リソースを選択し、 を開きます<environmentname>-vdc-external-nlb。

d. NLB から DNS 名をコピーします。

8. にサインイン AWS Management Console し、https://console.aws.amazon.com/route53/ で Route 53 コンソールを開きます。
9. ドメインを検索し、レコードの作成 を選択します。
10. レコード名 に と入力しますvdc。
11. [レコードタイプ] で、[CNAME] を選択します。
12. NLB の場合は、DNS を入力します。
13. [Create record] (レコードを作成) を選択します。

## ACM 証明書を作成する

デフォルトでは、RES はドメイン amazonaws.com を使用して、アプリケーションロードバランサーでウェブポータルをホストします。独自のドメインを使用するには、ユーザーが提供する、または AWS Certificate Manager (ACM) からリクエストされたパブリック SSL/TLS 証明書を設定する必要があります。ACM を使用する場合、クライアントとウェブサービスホスト間の SSL/TLS チャネルを暗号化するためのパラメータとして指定する必要がある AWS リソース名を受け取ります。

> **Tip**
>
> 外部リソースデモパッケージをデプロイする場合は、 に外部リソーススタックをデプロイPortalDomainNameするときに、選択したドメインを に入力する必要がありますthe section called “外部リソースを作成する”。

カスタムドメインの証明書を作成するには：

1. コンソールから AWS Certificate Managerを開き、パブリック証明書をリクエストします。AWS GovCloud （ 米国西部) にデプロイする場合は、 GovCloud パーティションアカウントに証明書を作成します。
2. 「パブリック証明書のリクエスト」を選択し、「次へ」を選択します。
3. ドメイン名 で、 *.PortalDomainNameと の両方の証明書をリクエストしますPortalDomainName。

4. 検証方法 で、DNS 検証 を選択します。
5. [リクエスト] を選択します。
6. 証明書リストから、リクエストされた証明書を開きます。各証明書のステータスは、検証保留中になります。

> **Note**
>
> 証明書が表示されない場合は、リストを更新します。

7. 次のいずれかを行います。
   - 商用デプロイ： リクエストされた各証明書の証明書の詳細から、Route 53 でレコードを作成するを選択します。証明書のステータスは発行済み に変わります。
   - GovCloud デプロイ： AWS GovCloud（米国西部) にデプロイする場合は、CNAME キーと値をコピーします。商用パーティションアカウントから、 値を使用してパブリックホストゾーンに新しいレコードを作成します。証明書のステータスは発行済み に変わります。
8. 新しい証明書 ARN をコピーして、 のパラメータとして入力しますACMCertificateARNforWebApp。

## Amazon CloudWatch Logs

Research and Engineering Studio は、インストール CloudWatch 中に以下のロググループを に作成します。デフォルトの保持については、次の表を参照してください。

| CloudWatch ロググループ | Retention |
|---|---|
| /aws/lambda/<installation-stack-name>-cluster-endpoints | 有効期限なし |
| /aws/lambda/<installation-stack-name>-cluster-manager-scheduled-ad-sync | 有効期限なし |
| /aws/lambda/<installation-stack-name>-cluster-settings | 有効期限なし |
| /aws/lambda/<installation-stack-name>-oauth-credentials | 有効期限なし |

| CloudWatch ロググループ | Retention |
| --- | --- |
| /aws/lambda/<installation-stack-name>-self-signed-certificate | 有効期限なし |
| /aws/lambda/<installation-stack-name>-update-cluster-prefix-list | 有効期限なし |
| /aws/lambda/<installation-stack-name>-vdc-scheduled-event-transformer | 有効期限なし |
| /aws/lambda/<installation-stack-name>-vdc-update-cluster-manager-client-scope | 有効期限なし |
| /<installation-stack-name>/cluster-manager | 3 か月間 |
| /<installation-stack-name>/vdc/コントローラー | 3 か月間 |
| /<installation-stack-name>/vdc/dcv-broker | 3 か月間 |
| /<installation-stack-name>/vdc/dcv-connection-gateway | 3 か月間 |

ロググループのデフォルトの保持期間を変更する場合は、https://console.aws.amazon.com/cloudwatch/ の CloudWatch コンソールに移動し、CloudWatch 「ログ のログデータ保持期間を変更する」の指示に従ってください。

# カスタムアクセス許可の境界の設定

2024 年 4 月現在、カスタムアクセス許可の境界をアタッチすることで、オプションで RES によって作成されたロールを変更できます。カスタムアクセス許可の境界は、IAM PermissionBoundary パラメータの一部としてアクセス許可の境界の ARN を指定することで、RES AWS CloudFormation インストールの一部として定義できます。このパラメータを空のままにした場合、どの RES ロールにもアクセス許可の境界は設定されません。以下は、RES ロールが動作するために必要なアクションのリストです。使用する予定のアクセス許可の境界で、以下のアクションが明示的に許可されていることを確認してください。

```
[
```

```
    {
        "Effect": "Allow",
        "Resource": "*",
        "Sid": "ResRequiredActions",
        "Action": [
            "access-analyzer:*",
            "account:GetAccountInformation",
            "account:ListRegions",
            "acm:*",
            "airflow:*",
            "amplify:*",
            "amplifybackend:*",
            "amplifyuibuilder:*",
            "aoss:*",
            "apigateway:*",
            "appflow:*",
            "application-autoscaling:*",
            "appmesh:*",
            "apprunner:*",
            "aps:*",
            "athena:*",
            "auditmanager:*",
            "autoscaling-plans:*",
            "autoscaling:*",
            "backup-gateway:*",
            "backup-storage:*",
            "backup:*",
            "batch:*",
            "bedrock:*",
            "budgets:*",
            "ce:*",
            "cloud9:*",
            "cloudformation:*",
            "cloudfront:*",
            "cloudtrail-data:*",
            "cloudtrail:*",
            "cloudwatch:*",
            "codeartifact:*",
            "codebuild:*",
            "codeguru-profiler:*",
            "codeguru-reviewer:*",
            "codepipeline:*",
            "codestar-connections:*",
            "codestar-notifications:*",
```

```
            "codestar:*",
            "cognito-identity:*",
            "cognito-idp:*",
            "cognito-sync:*",
            "comprehend:*",
            "compute-optimizer:*",
            "cur:*",
            "databrew:*",
            "datapipeline:*",
            "datasync:*",
            "dax:*",
            "detective:*",
            "devops-guru:*",
            "dlm:*",
            "dms:*",
            "drs:*",
            "dynamodb:*",
            "ebs:*",
            "ec2-instance-connect:*",
            "ec2:*",
            "ec2messages:*",
            "ecr:*",
            "ecs:*",
            "eks:*",
            "elastic-inference:*",
            "elasticache:*",
            "elasticbeanstalk:*",
            "elasticfilesystem:*",
            "elasticloadbalancing:*",
            "elasticmapreduce:*",
            "elastictranscoder:*",
            "es:*",
            "events:*",
            "firehose:*",
            "fis:*",
            "fms:*",
            "forecast:*",
            "fsx:*",
            "geo:*",
            "glacier:*",
            "glue:*",
            "grafana:*",
            "guardduty:*",
            "health:*",
```

```
            "iam:*",
            "identitystore:*",
            "imagebuilder:*",
            "inspector2:*",
            "inspector:*",
            "internetmonitor:*",
            "iot:*",
            "iotanalytics:*",
            "kafka:*",
            "kafkaconnect:*",
            "kinesis:*",
            "kinesisanalytics:*",
            "kms:*",
            "lambda:*",
            "lightsail:*",
            "logs:*",
            "memorydb:*",
            "mgh:*",
            "mobiletargeting:*",
            "mq:*",
            "neptune-db:*",
            "organizations:DescribeOrganization",
            "osis:*",
            "personalize:*",
            "pi:*",
            "pipes:*",
            "polly:*",
            "qldb:*",
            "quicksight:*",
            "rds-data:*",
            "rds:*",
            "redshift-data:*",
            "redshift-serverless:*",
            "redshift:*",
            "rekognition:*",
            "resiliencehub:*",
            "resource-groups:*",
            "route53:*",
            "route53domains:*",
            "route53resolver:*",
            "rum:*",
            "s3:*",
            "sagemaker:*",
            "scheduler:*",
```

```
            "schemas:*",
            "sdb:*",
            "secretsmanager:*",
            "securityhub:*",
            "serverlessrepo:*",
            "servicecatalog:*",
            "servicequotas:*",
            "ses:*",
            "signer:*",
            "sns:*",
            "sqs:*",
            "ssm:*",
            "ssmmessages:*",
            "states:*",
            "storagegateway:*",
            "sts:*",
            "support:*",
            "tag:GetResources",
            "tag:GetTagKeys",
            "tag:GetTagValues",
            "textract:*",
            "timestream:*",
            "transcribe:*",
            "transfer:*",
            "translate:*",
            "vpc-lattice:*",
            "waf-regional:*",
            "waf:*",
            "wafv2:*",
            "wellarchitected:*",
            "wisdom:*",
            "xray:*"
        ]
    }
]
```

# RES 対応 AMIs の設定

RES 対応 AMIsを使用すると、仮想デスクトップインスタンス (VDIs) の RES 依存関係をカスタム AMIs にプリインストールできます。RES 対応 AMIs を使用すると、事前にベイクされたイメージを使用する VDI インスタンスの起動時間が短縮されます。EC2 Image Builder を使用する

と、AMIs を新しいソフトウェアスタックとして構築して登録できます。Image Builder の詳細については、「Image Builder ユーザーガイド」を参照してください。

開始する前に、最新バージョンの RES をデプロイする必要があります。

トピック

- RES 環境にアクセスするための IAM ロールを準備する
- EC2 Image Builder コンポーネントを作成する
- EC2 Image Builder レシピを準備する
- EC2 Image Builder インフラストラクチャを設定する
- Image Builder イメージパイプラインを設定する
- Image Builder イメージパイプラインを実行する
- RES に新しいソフトウェアスタックを登録する

## RES 環境にアクセスするための IAM ロールを準備する

EC2 Image Builder から RES 環境サービスにアクセスするには、RES-EC2InstanceProfileForImageBuilder という IAM ロールを作成または変更する必要があります。Image Builder で使用する IAM ロールの設定については、Image Builder ユーザーガイドのAWS Identity and Access Management 「(IAM)」を参照してください。

ロールには以下が必要です。

- 信頼関係には Amazon EC2 サービスが含まれます
- AmazonSSMManagedInstanceCore および EC2InstanceProfileForImageBuilder ポリシー
- デプロイされた RES 環境への DynamoDB および Amazon S3 アクセスが制限されたカスタム RES ポリシー

  （このポリシーは、カスタマー管理ポリシードキュメントまたはカスタマーインラインポリシードキュメントのいずれかになります）。

信頼関係エンティティ：

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
```

```
            "Effect": "Allow",
            "Principal": {
                "Service": "ec2.amazonaws.com"
            }
            "Action": "sts:AssumeRole"
        }
    ]
}
```

RES ポリシー :

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "RESDynamoDBAccess",
            "Effect": "Allow",
            "Action": "dynamodb:GetItem",
            "Resource": "arn:aws:dynamodb:{AWS-Region}:{AWS-Account-ID}:table/{RES-
EnvironmentName}.cluster-settings",
            "Condition": {
                "ForAllValues:StringLike": {
                    "dynamodb:LeadingKeys": [
                        "global-settings.gpu_settings.*",
                        "global-settings.package_config.*"
                    ]
                }
            }
        },
        {
            "Sid": "RESS3Access",
            "Effect": "Allow",
            "Action": "s3:GetObject",
            "Resource": "arn:aws:s3:::{RES-EnvironmentName}-cluster-{AWS-Region}-{AWS-
Account-ID}/idea/vdc/res-ready-install-script-packages/*"
        }
    ]
}
```

# EC2 Image Builder コンポーネントを作成する

Image Builder ユーザーガイドの「Image Builder コンソールを使用してコンポーネントを作成する」の指示に従ってください。

コンポーネントの詳細を入力します。

1. タイプ で、ビルド を選択します。
2. イメージオペレーティングシステム (OS) で、Linux または Windows を選択します。
3. コンポーネント名 には、 などのわかりやすい名前を入力します**research-and-engineering-studio-vdi-<operating-system>**。
4. コンポーネントのバージョン番号を入力し、オプションで説明を追加します。
5. 定義ドキュメント には、次の定義ファイルを入力します。エラーが発生した場合、YAML ファイルはスペースに敏感であり、最も可能性の高い原因です。

   Linux

```
#  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
#  Licensed under the Apache License, Version 2.0 (the "License"). You may not
 use this file except in compliance
#  with the License. A copy of the License is located at
#
#      http://www.apache.org/licenses/LICENSE-2.0
#
#  or in the 'license' file accompanying this file. This file is distributed on
 an 'AS IS' BASIS, WITHOUT WARRANTIES
#  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
 specific language governing permissions
#  and limitations under the License.
name: research-and-engineering-studio-vdi-linux
description: An RES EC2 Image Builder component to install required RES software
 dependencies for Linux VDI.
schemaVersion: 1.0
parameters:
  - AWSAccountID:
      type: string
      description: RES Environment AWS Account ID
  - RESEnvName:
      type: string
      description: RES Environment Name
  - RESEnvRegion:
      type: string
      description: RES Environment Region
  - RESEnvReleaseVersion:
      type: string
```

```
      description: RES Release Version

phases:
  - name: build
    steps:
       - name: PrepareRESBootstrap
         action: ExecuteBash
         onFailure: Abort
         maxAttempts: 3
         inputs:
            commands:
                - 'mkdir -p /root/bootstrap/logs'
                - 'mkdir -p /root/bootstrap/latest'
       - name: DownloadRESLinuxInstallPackage
         action: S3Download
         onFailure: Abort
         maxAttempts: 3
         inputs:
            - source: 's3://{{ RESEnvName }}-cluster-{{ RESEnvRegion }}-
{{ AWSAccountID }}/idea/vdc/res-ready-install-script-packages/linux/
res_linux_install_{{ RESEnvReleaseVersion }}.tar.gz'
              destination: '/root/bootstrap/
res_linux_install_{{ RESEnvReleaseVersion }}.tar.gz'
              expectedBucketOwner: '{{ AWSAccountID }}'
       - name: RunInstallScript
         action: ExecuteBash
         onFailure: Abort
         maxAttempts: 3
         inputs:
            commands:
                - 'tar -xvf
 {{ build.DownloadRESLinuxInstallPackage.inputs[0].destination }} -C /root/
bootstrap/latest'
                - '/bin/bash /root/bootstrap/latest/virtual-desktop-host-linux/
install.sh -r {{ RESEnvRegion }} -n {{ RESEnvName }} -g NONE'
       - name: FirstReboot
         action: Reboot
         onFailure: Abort
         maxAttempts: 3
         inputs:
            delaySeconds: 0
       - name: RunInstallPostRebootScript
         action: ExecuteBash
         onFailure: Abort
```

```
         maxAttempts: 3
         inputs:
            commands:
                - '/bin/bash /root/bootstrap/latest/virtual-desktop-host-linux/
install_post_reboot.sh'
       - name: SecondReboot
         action: Reboot
         onFailure: Abort
         maxAttempts: 3
         inputs:
            delaySeconds: 0
```

### Windows

```
#  Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
#  Licensed under the Apache License, Version 2.0 (the "License"). You may not
 use this file except in compliance
#  with the License. A copy of the License is located at
#
#      http://www.apache.org/licenses/LICENSE-2.0
#
#  or in the 'license' file accompanying this file. This file is distributed on
 an 'AS IS' BASIS, WITHOUT WARRANTIES
#  OR CONDITIONS OF ANY KIND, express or implied. See the License for the
 specific language governing permissions
#  and limitations under the License.
name: research-and-engineering-studio-vdi-windows
description: An RES EC2 Image Builder component to install required RES software
 dependencies for Windows VDI.
schemaVersion: 1.0
parameters:
  - AWSAccountID:
      type: string
      description: RES Environment AWS Account ID
  - RESEnvName:
      type: string
      description: RES Environment Name
  - RESEnvRegion:
      type: string
      description: RES Environment Region
  - RESEnvReleaseVersion:
      type: string
```

```
      description: RES Release Version

phases:
  - name: build
    steps:
       - name: CreateRESBootstrapFolder
         action: CreateFolder
         onFailure: Abort
         maxAttempts: 3
         inputs:
            - path: 'C:\Users\Administrator\RES\Bootstrap'
              overwrite: true
       - name: DownloadRESWindowsInstallPackage
         action: S3Download
         onFailure: Abort
         maxAttempts: 3
         inputs:
            - source: 's3://{{ RESEnvName }}-cluster-{{ RESEnvRegion }}-
{{ AWSAccountID }}/idea/vdc/res-ready-install-script-packages/windows/
res_windows_install_{{ RESEnvReleaseVersion }}.tar.gz'
              destination:
 '{{ build.CreateRESBootstrapFolder.inputs[0].path }}\res_windows_install_{{ RESEnvRelea
              expectedBucketOwner: '{{ AWSAccountID }}'
       - name: RunInstallScript
         action: ExecutePowerShell
         onFailure: Abort
         maxAttempts: 3
         inputs:
            commands:
                - 'cd {{ build.CreateRESBootstrapFolder.inputs[0].path }}'
                - 'Tar -xf
 res_windows_install_{{ RESEnvReleaseVersion }}.tar.gz'
                - 'Import-Module .\virtual-desktop-host-windows\Install.ps1'
                - 'Install-WindowsEC2Instance'
       - name: Reboot
         action: Reboot
         onFailure: Abort
         maxAttempts: 3
         inputs:
            delaySeconds: 0
```

6. オプションのタグを作成し、コンポーネントの作成 を選択します。

# EC2 Image Builder レシピを準備する

EC2 Image Builder レシピでは、新しいイメージを作成するための開始点として使用するベースイメージと、イメージをカスタマイズしてすべてが期待どおりに動作することを確認するために追加する一連のコンポーネントを定義します。レシピを作成または変更して、必要な RES ソフトウェアの依存関係を持つターゲット AMI を構築する必要があります。レシピの詳細については、「レシピの管理」を参照してください。

RES は、次のイメージオペレーティングシステムをサポートしています。

- Amazon Linux 2 (x86 および ARM64)
- CentOS 7 (x86 および ARM64)
- RHEL 7 (x86)、8 (x86)、および 9 (x86)
- Windows 2019、2022 (x86)

Create a new recipe

1. で EC2 Image Builder コンソールを開きますhttps://console.aws.amazon.com/imagebuilder。
2. 保存済みリソース で、イメージレシピ を選択します。
3. [イメージレシピの作成] を選択します。
4. 一意の名前とバージョン番号を入力します。
5. RES でサポートされているベースイメージを選択します。
6. インスタンス設定 で、SSM エージェントがプリインストールされていない場合は、インストールします。ユーザーデータおよびその他の必要なユーザーデータに情報を入力します。

   > **Note**
   >
   > SSM エージェントをインストールする方法については、以下を参照してください。
   >
   > - Linux 用 EC2 インスタンスに SSM エージェントを手動でインストールする
   > - Windows Server 用の EC2 インスタンスに SSM エージェントを手動でインストールおよびアンインストールする

7. Linux ベースのレシピの場合は、Amazon が管理するaws-cli-version-2-linuxビルドコンポーネントをレシピに追加します。RES インストールスクリプトは AWS CLI 、 を使用し

て、DynamoDB クラスター設定の構成値への VDI アクセスを提供します。Windows では、このコンポーネントは必要ありません。

8. Linux または Windows 環境用に作成された EC2 Image Builder コンポーネントを追加し、必要なパラメータ値を入力します。次のパラメータは必須入力です: AWSAccountID、RES EnvName、RES EnvRegion、および RES EnvReleaseVersion。

> **Important**
>
> Linux 環境では、`aws-cli-version-2-linux`ビルドコンポーネントを最初に追加した状態で、これらのコンポーネントを追加する必要があります。

9. ( 推奨) Amazon が管理する`simple-boot-test-<linux-or-windows>`テストコンポーネントを追加して、AMI を起動できることを確認します。これは最小限の推奨事項です。要件を満たす他のテストコンポーネントを選択できます。
10. 必要に応じてオプションのセクションを完了し、他の必要なコンポーネントを追加して、レシピの作成 を選択します。

### Modify a recipe

既存の EC2 Image Builder レシピがある場合は、次のコンポーネントを追加して使用できます。

1. Linux ベースのレシピの場合は、Amazon が管理する`aws-cli-version-2-linux`ビルドコンポーネントをレシピに追加します。RES インストールスクリプトは AWS CLI 、 を使用して、DynamoDB クラスター設定の構成値への VDI アクセスを提供します。Windows では、このコンポーネントは必要ありません。
2. Linux または Windows 環境用に作成された EC2 Image Builder コンポーネントを追加し、必要なパラメータ値を入力します。次のパラメータは必須入力です: AWSAccountID、RES EnvName、RES EnvRegion、および RES EnvReleaseVersion。

> **Important**
>
> Linux 環境では、`aws-cli-version-2-linux`ビルドコンポーネントを最初に追加した状態で、これらのコンポーネントを追加する必要があります。

3. 必要に応じてオプションのセクションを完了し、その他の必要なコンポーネントを追加して、レシピの作成 を選択します。

# EC2 Image Builder インフラストラクチャを設定する

インフラストラクチャ設定を使用して、Image Builder が Image Builder イメージの構築とテストに使用する Amazon EC2 インフラストラクチャを指定できます。RES で使用するには、新しいインフラストラクチャ設定を作成するか、既存のインフラストラクチャ設定を使用するかを選択できます。

- 新しいインフラストラクチャ設定を作成するには、「インフラストラクチャ設定の作成」を参照してください。
- 既存のインフラストラクチャ設定を使用するには、インフラストラクチャ設定 を更新します。

Image Builder インフラストラクチャを設定するには：

1. IAM ロール には、 で以前に設定したロールを入力しますthe section called “RES 環境にアクセスするための IAM ロールを準備する”。
2. インスタンスタイプ では、4 GB 以上のメモリを持つタイプを選択し、選択したベース AMI アーキテクチャをサポートします。Amazon EC2 インスタンスタイプ」を参照してください。
3. VPC、サブネット、およびセキュリティグループ の場合、ソフトウェアパッケージをダウンロードするためのインターネットアクセスを許可する必要があります。RES 環境の `cluster-settings` DynamoDB テーブルと Amazon S3 クラスターバケットへのアクセスも許可する必要があります。

# Image Builder イメージパイプラインを設定する

Image Builder イメージパイプラインは、ベースイメージ、構築とテスト用のコンポーネント、インフラストラクチャ設定、およびディストリビューション設定をアセンブルします。RES 対応 AMIs 用にイメージパイプラインを設定するには、新しいパイプラインを作成するか、既存のパイプラインを使用するかを選択できます。詳細については、Image Builder ユーザーガイドの「AMI イメージパイプラインの作成と更新」を参照してください。

Create a new Image Builder pipeline

1. で Image Builder コンソールを開きますhttps://console.aws.amazon.com/imagebuilder。
2. ナビゲーションから、イメージパイプライン を選択します。
3. 「イメージパイプラインの作成」を選択します。
4. 一意の名前、オプションの説明、スケジュール、頻度を入力して、パイプラインの詳細を指定します。

5. レシピの選択 で、既存のレシピを使用 を選択し、 で作成されたレシピを選択しますthe section called “EC2 Image Builder レシピを準備する”。レシピの詳細が正しいことを確認します。
6. イメージ作成プロセスの定義 では、ユースケースに応じてデフォルトワークフローまたはカスタムワークフローを選択します。ほとんどの場合、デフォルトのワークフローで十分です。詳細については、EC2 Image Builder パイプラインのイメージワークフローを設定する」を参照してください。
7. 「インフラストラクチャ設定の定義」で、「既存のインフラストラクチャ設定の選択」を選択し、「」で作成したインフラストラクチャ設定を選択しますthe section called “EC2 Image Builder インフラストラクチャを設定する”。インフラストラクチャの詳細が正しいことを確認します。
8. ディストリビューション設定の定義 で、サービスのデフォルト を使用してディストリビューション設定を作成する を選択します。出力イメージは、RES 環境 AWS リージョンと同じ に存在する必要があります。サービスのデフォルトを使用すると、Image Builder が使用されているリージョンにイメージが作成されます。
9. パイプラインの詳細を確認し、パイプラインの作成 を選択します。

Modify an existing Image Builder pipeline

1. 既存のパイプラインを使用するには、 で作成されたレシピを使用するように詳細を変更しますthe section called “EC2 Image Builder レシピを準備する”。
2. [変更を保存] を選択します。

## Image Builder イメージパイプラインを実行する

設定された出力イメージを生成するには、イメージパイプラインを開始する必要があります。イメージレシピのコンポーネント数によっては、構築プロセスに最大 1 時間かかる場合があります。

イメージパイプラインを実行するには：

1. イメージパイプライン から、 で作成されたパイプラインを選択しますthe section called “Image Builder イメージパイプラインを設定する”。
2. アクション から、パイプラインの実行 を選択します。

# RES に新しいソフトウェアスタックを登録する

1. 「」の指示に従ってthe section called “ソフトウェアスタック (AMIs)”、ソフトウェアスタックを登録します。
2. AMI ID には、 で構築された出力イメージの AMI ID を入力しますthe section called “Image Builder イメージパイプラインを実行する”。

# 管理者ガイド

この管理者ガイドでは、 AWS 製品で Research and Engineering Studio をさらにカスタマイズして統合する方法に関する追加の手順を、技術的な視聴者に説明します。

トピック

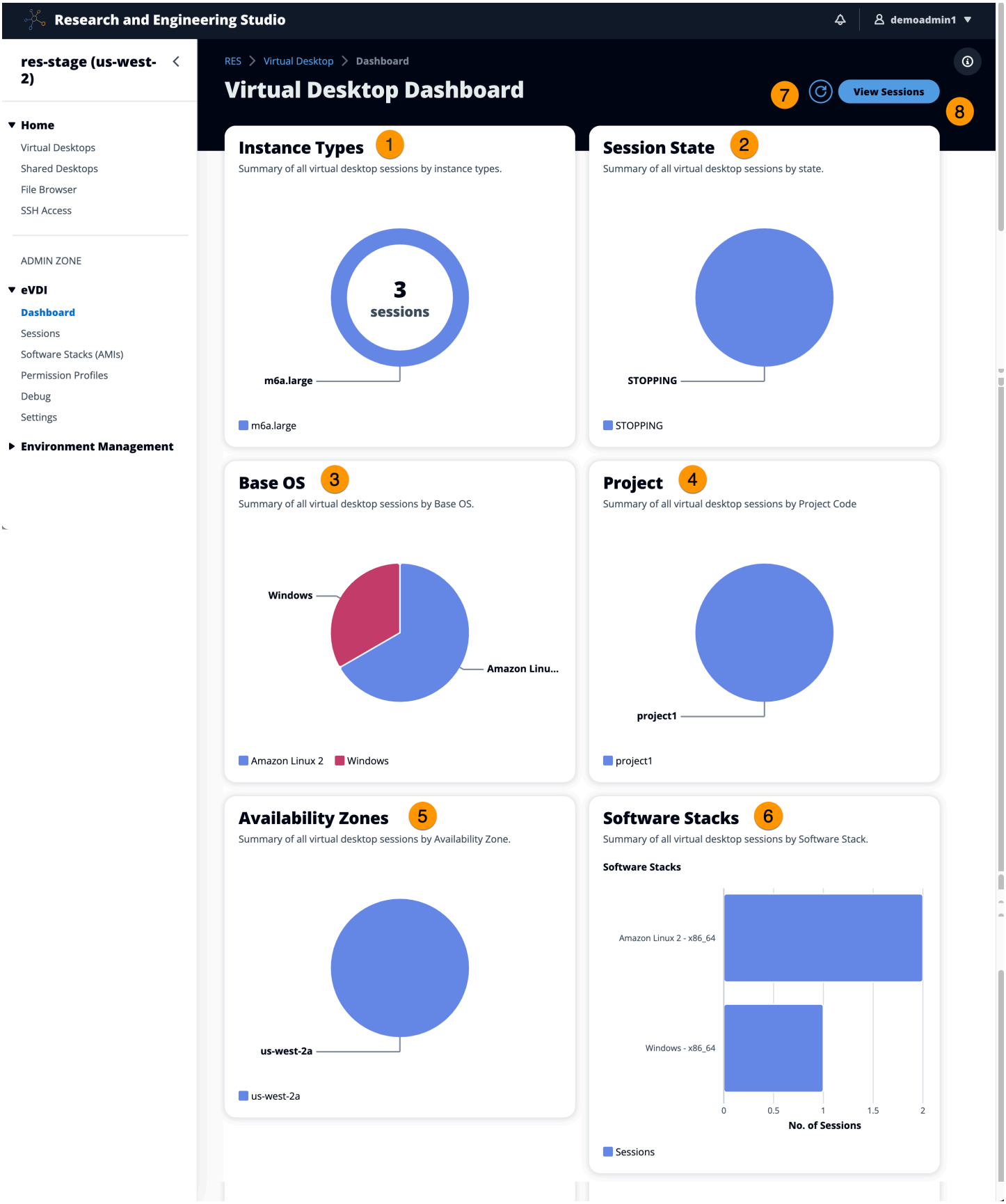


## セッション管理

セッション管理は、セッションを開発およびテストするための柔軟でインタラクティブな環境を提供します。管理ユーザーとして、プロジェクト環境内でインタラクティブセッションを作成および管理することをユーザーに許可できます。

トピック

# ダッシュボード


Research and Engineering Studio
demoadmin1
res-stage (us-west-2)
Home
Virtual Desktops
Shared Desktops
File Browser
SSH Access
ADMIN ZONE
eVDI
Dashboard
Sessions
Software Stacks (AMIs)
Permission Profiles
Debug
Settings
Environment Management
RES > Virtual Desktop > Dashboard
Virtual Desktop Dashboard
7
View Sessions
8
Instance Types
1
Summary of all virtual desktop sessions by instance types.
3
sessions
m6a.large
m6a.large
Session State
2
Summary of all virtual desktop sessions by state.
STOPPING
STOPPING
Base OS
3
Summary of all virtual desktop sessions by Base OS.
Windows
Amazon Linu...
Amazon Linux 2
Windows
Project
4
Summary of all virtual desktop sessions by Project Code
project1
project1
Availability Zones
5
Summary of all virtual desktop sessions by Availability Zone.
us-west-2a
us-west-2a
Software Stacks
6
Summary of all virtual desktop sessions by Software Stack.
Software Stacks
Amazon Linux 2 - x86_64
Windows - x86_64
0
0.5
1
1.5
2
No. of Sessions
Sessions

セッション管理ダッシュボードでは、管理者は以下をすばやく確認できます。

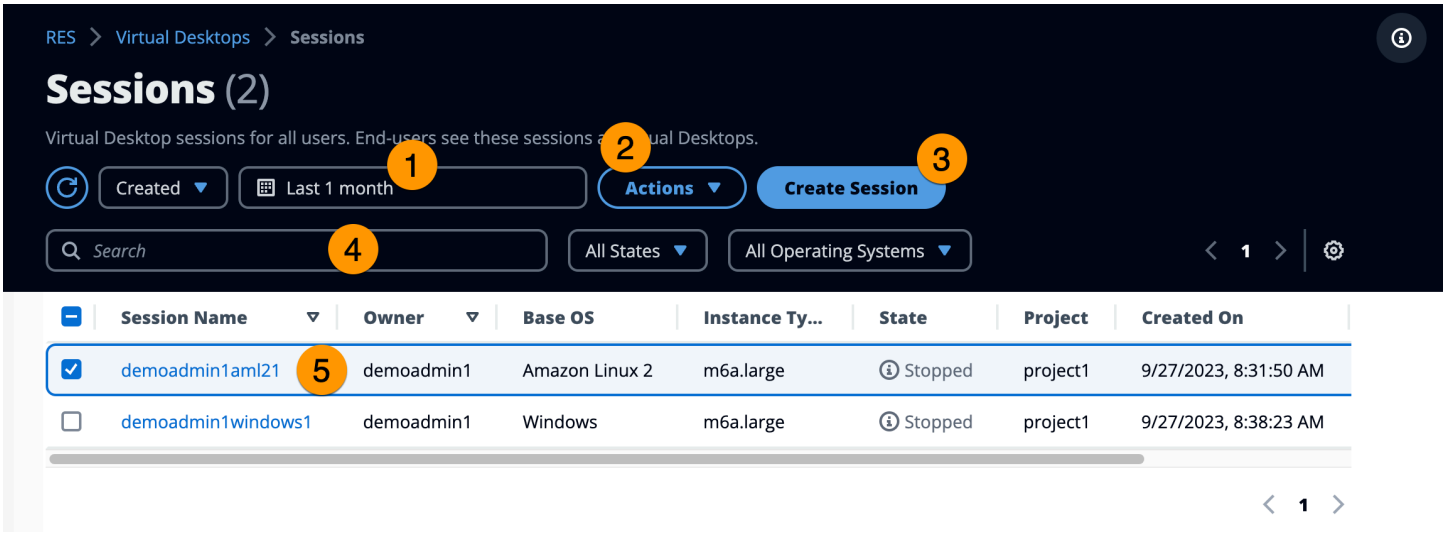
1. インスタンスのタイプ
2. セッション状態
3. ベース OS
4. プロジェクト
5. アベイラビリティーゾーン
6. ソフトウェアスタック

さらに、管理者は次のことができます。

7. ダッシュボードを更新して情報を更新します。
8. セッションの表示を選択してセッションに移動します。

## セッション

セッションには、Research and Engineering Studio 内で作成されたすべての仮想デスクトップが表示されます。セッションページから、セッション情報をフィルタリングして表示したり、新しいセッションを作成したりできます。




1. メニューを使用して、指定した期間内に作成または更新されたセッションで結果をフィルタリングします。
2. セッションを選択し、アクションメニューを使用して次の操作を行います。
   a. セッションを再開する (複数可 )

b. セッションの停止/休止 (複数可 )

c. 強制停止/休止セッション (複数可 )

d. セッションを終了する (複数可 )

e. セッションを強制終了する (複数可 )

f. セッションの正常性 (複数可 )

g. ソフトウェアスタックの作成

3. セッションの作成 を選択して、新しいセッションを作成します。
4. 名前でセッションを検索し、状態とオペレーティングシステムでフィルタリングします。
5. セッション名を選択すると、詳細が表示されます。

## セッションを作成する

1. セッションの作成 を選択します。新しい仮想デスクトップの起動モーダルが開きます。
2. 新しいセッションの詳細を入力します。
3. (オプション) 詳細オプションを表示をオンにして、サブネット ID や DCV セッションタイプなどの追加の詳細を提供します。
4. [送信] を選択します。

# Launch New Virtual Desktop

**Session Name**
Enter a name for the virtual desktop

Session Name is required. Use any characters and form a name of length between 3 and 24 characters, inclusive.

**User**
Select the user to create the session for

**Project**
Select the project under which the session will get created

**Operating System**
Select the operating system for the virtual desktop

Amazon Linux 2

**Software Stack**
Select the software stack for your virtual desktop

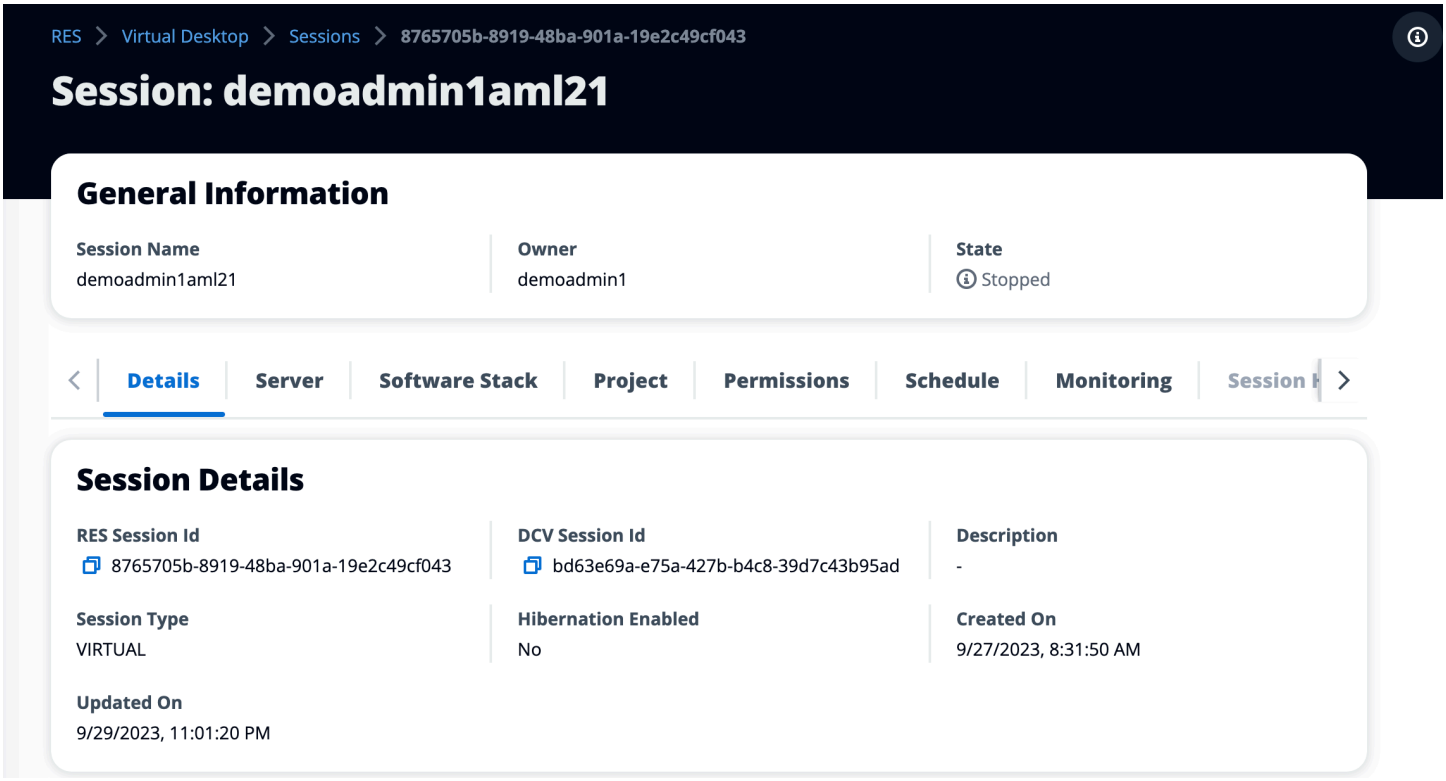
**Enable Instance Hibernation**
Hibernation saves the contents from the instance memory (RAM) to your Amazon Elastic Block Store (Amazon EBS) root volume. You can not change instance type if you enable this option.

**Virtual Desktop Size**
Select a virtual desktop instance type

**Storage Size (GB)**
Enter the storage size for your virtual desktop in GBs

### セッションの詳細

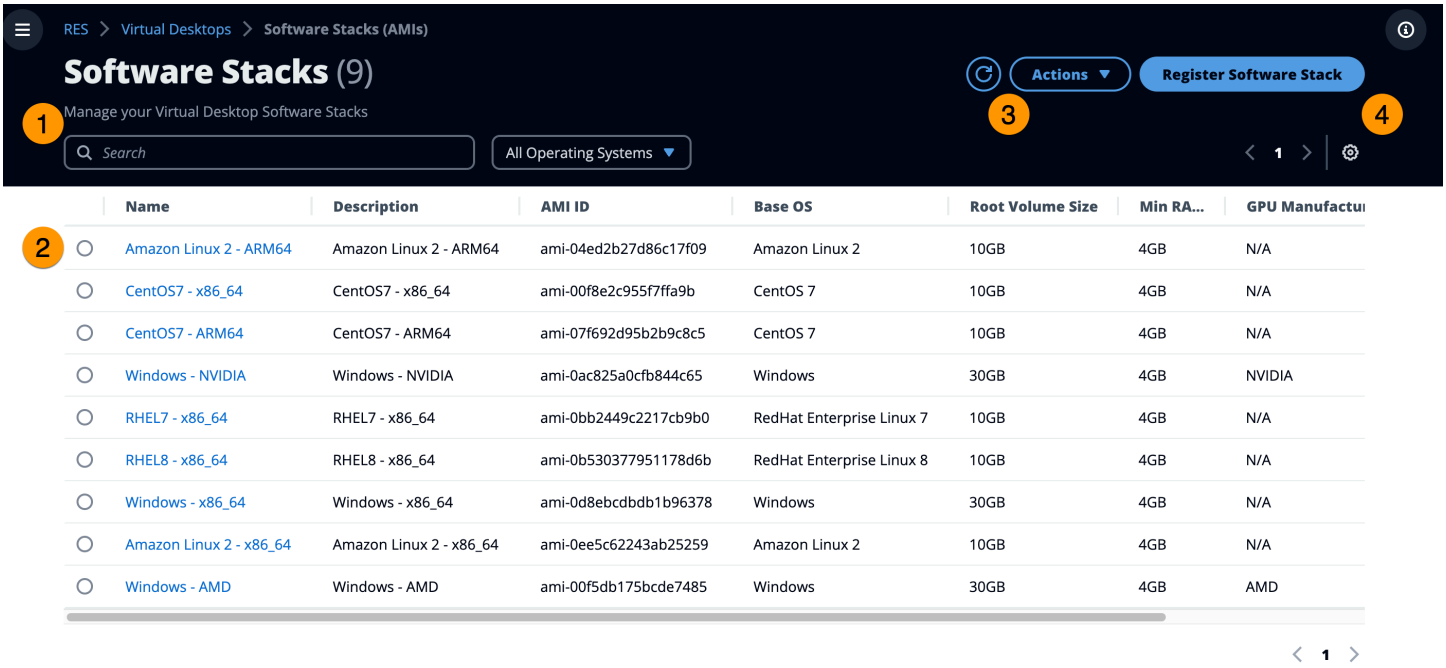
セッションリストからセッション名を選択して、セッションの詳細を表示します。




## ソフトウェアスタック (AMIs)

ソフトウェアスタックページから、Amazon マシンイメージ (AMIsを設定し、既存の AMIs を管理できます。

> **Note**
>
> で提供されている CentSO7 ソフトウェアスタックを実行するには AWS GovCloud (US)、リンクされた標準アカウント AWS Marketplace を使用して 内の AMI をサブスクライブする必要があります。




1. 既存のソフトウェアスタックを検索します。OS でフィルタリングするには、オペレーティングシステムのドロップダウンを使用します。
2. ソフトウェアスタックの名前を選択して、スタックの詳細を表示します。
3. ソフトウェアスタックを選択する場合は、アクションメニューを使用してスタックを編集し、スタックをプロジェクトに割り当てます。
4. ソフトウェアスタックの登録 を選択して、新しいスタックを作成します。

## ソフトウェアスタックを登録する

1. 「ソフトウェアスタックの登録」を選択します。
2. 新しいソフトウェアスタックの詳細を入力します。
3. [送信] を選択します。

# Register new Software Stack

**Name**
Enter a name for the software stack

Use any characters and form a name of length between 3 and 24 characters, inclusive.

**Description**
Enter a user friendly description for the software stack

**AMI Id**
Enter the AMI Id

AMI Id must start with ami-xxx

**Operating System**
Select the operating system for the software stack

Amazon Linux 2

**GPU Manufacturer**
Select the GPU Manufacturer for the software stack

N/A

**Min. Storage Size (GB)**
Enter the min. storage size for your virtual desktop in GBs

10

**Min. RAM (GB)**
Enter the min. ram for your virtual desktop in GBs

10

**Projects**
Select applicable projects for the software stack

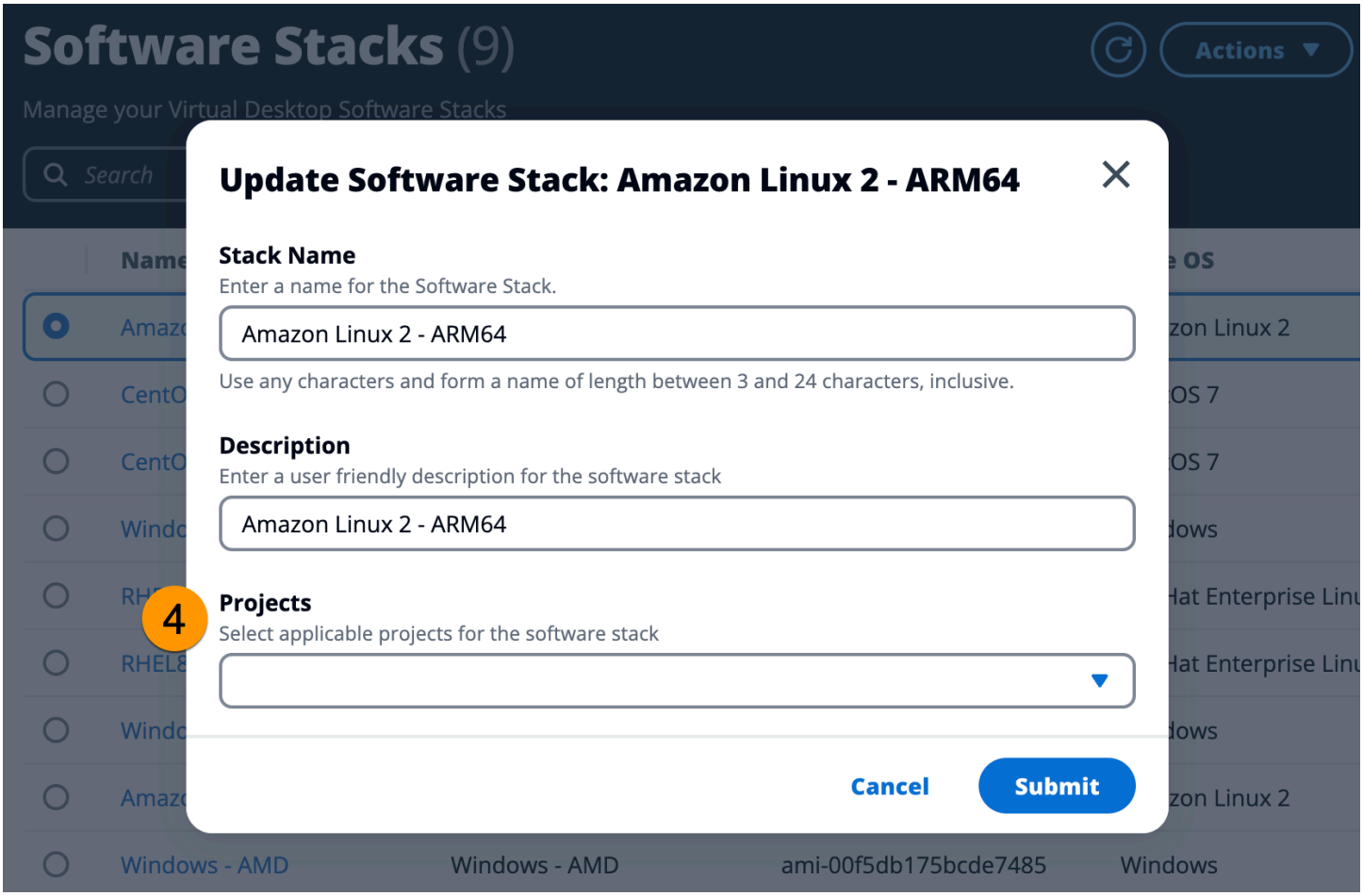
# プロジェクトにソフトウェアスタックを割り当てる

新しいソフトウェアスタックを作成するときに、スタックをプロジェクトに割り当てることができます。最初の作成後にスタックをプロジェクトに追加する必要がある場合は、次の手順を実行します。

> **Note**
>
> ソフトウェアスタックは、自分がメンバーであるプロジェクトにのみ割り当てることができます。

1. 「ソフトウェアスタック」ページから、プロジェクトに追加する必要があるソフトウェアスタックを選択します。
2. [アクション] を選択します。
3. [編集] を選択します。
4. プロジェクトドロップダウンを使用してプロジェクトを選択します。
5. [送信] を選択します。

スタックの詳細ページからソフトウェアスタックを編集することもできます。

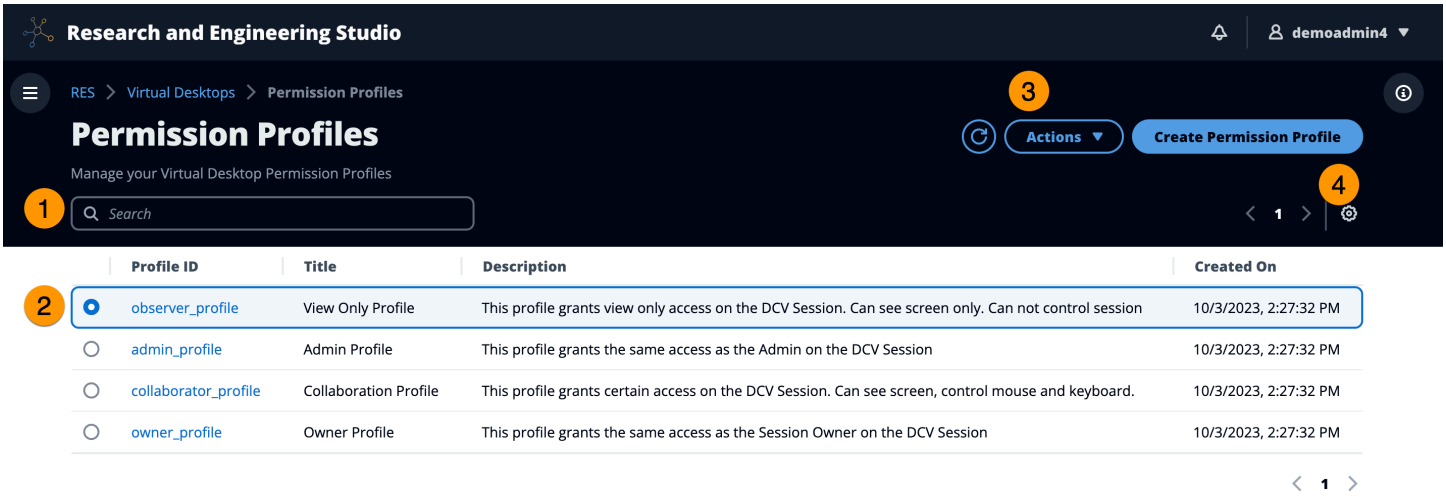



### ソフトウェアスタックの詳細を表示する

ソフトウェアスタックリストからソフトウェアスタック名を選択して詳細を表示します。詳細ページから、編集を選択してソフトウェアスタックを編集することもできます。

## アクセス許可プロファイル

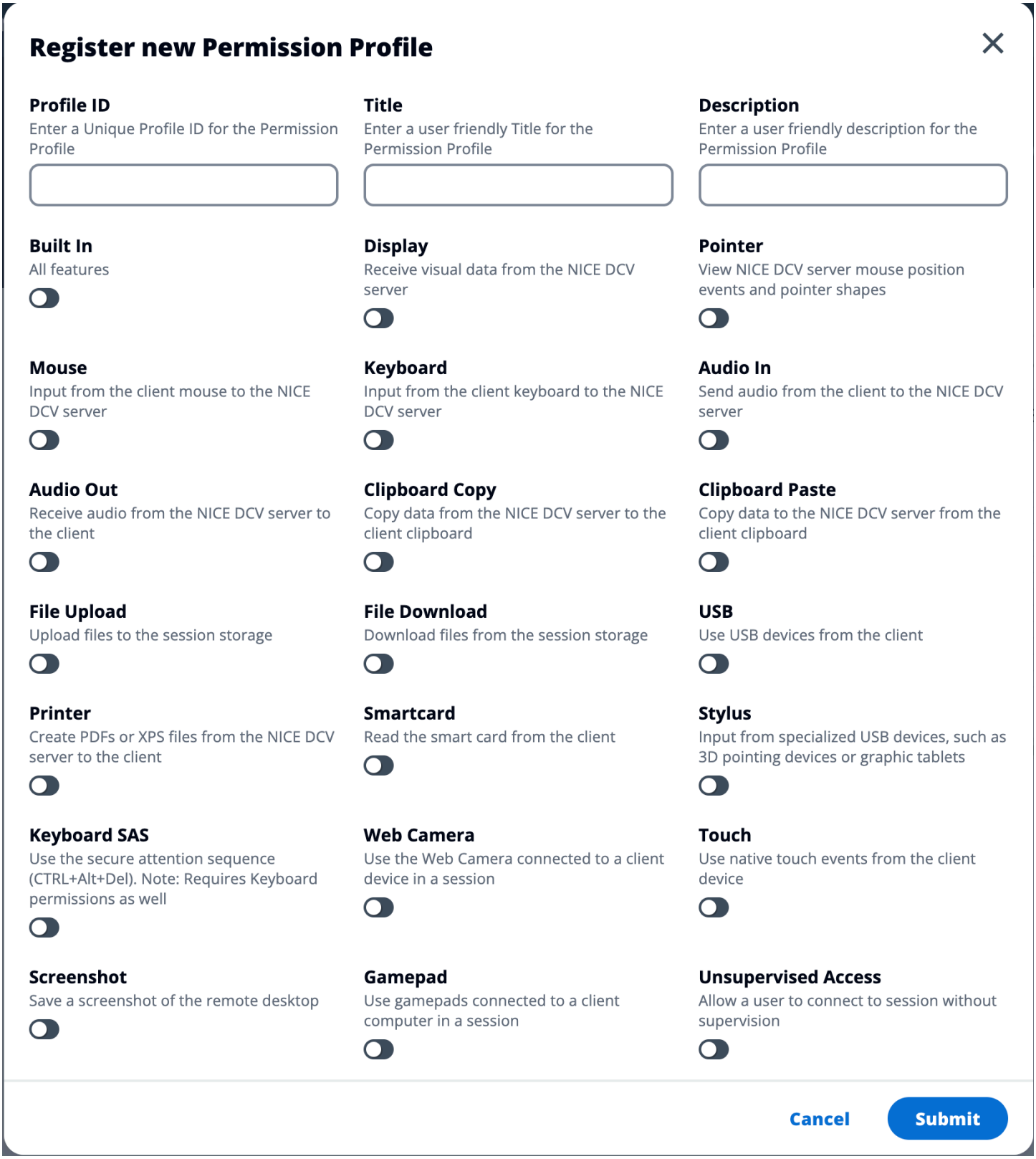
アクセス許可プロファイルを使用して、アクセス許可の再利用可能なプロファイルを作成および管理します。




1. アクセス許可プロファイルを検索します。
2. 詳細を表示するには、プロファイル ID を選択します。
3. プロファイルを選択したら、アクションメニューを使用してプロファイルを編集します。
4. アクセス許可プロファイルの作成 を選択して、新しいプロファイルを作成します。

## アクセス許可プロファイルを作成する

1. アクセス許可プロファイルの作成 を選択します。
2. 新しいプロファイルの詳細を入力し、アクセス許可の切り替えを使用してプロファイルのアクセス許可を選択します。
3. [送信] を選択します。


Register new Permission Profile
Profile ID
Enter a Unique Profile ID for the Permission Profile
Title
Enter a user friendly Title for the Permission Profile
Description
Enter a user friendly description for the Permission Profile
Built In
All features
Display
Receive visual data from the NICE DCV server
Pointer
View NICE DCV server mouse position events and pointer shapes
Mouse
Input from the client mouse to the NICE DCV server
Keyboard
Input from the client keyboard to the NICE DCV server
Audio In
Send audio from the client to the NICE DCV server
Audio Out
Receive audio from the NICE DCV server to the client
Clipboard Copy
Copy data from the NICE DCV server to the client clipboard
Clipboard Paste
Copy data to the NICE DCV server from the client clipboard
File Upload
Upload files to the session storage
File Download
Download files from the session storage
USB
Use USB devices from the client
Printer
Create PDFs or XPS files from the NICE DCV server to the client
Smartcard
Read the smart card from the client
Stylus
Input from specialized USB devices, such as 3D pointing devices or graphic tablets
Keyboard SAS
Use the secure attention sequence (CTRL+Alt+Del). Note: Requires Keyboard permissions as well
Web Camera
Use the Web Camera connected to a client device in a session
Touch
Use native touch events from the client device
Screenshot
Save a screenshot of the remote desktop
Gamepad
Use gamepads connected to a client computer in a session
Unsupervised Access
Allow a user to connect to session without supervision
Cancel
Submit

## アクセス許可プロファイルを編集する

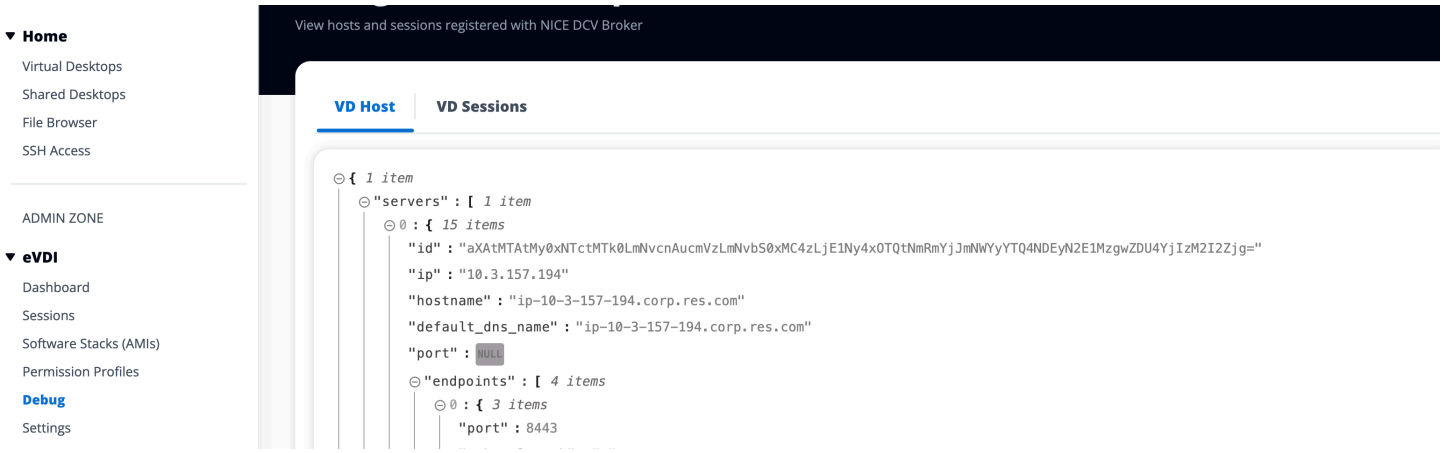
1. 「アクセス許可プロファイル」ページから編集する必要があるアクセス許可プロファイルを選択します。
2. [アクション] を選択します。
3. アクセス許可プロファイルの編集 を選択します。
4. プロファイルを編集します。
5. [送信] を選択します。

## アクセス許可プロファイルの詳細を表示する

アクセス許可プロファイル リストから、詳細を表示するプロファイル ID を選択します。詳細ページから、編集を選択してアクセス許可プロファイルを編集することもできます。

# デバッグ

デバッグパネルには、仮想デスクトップに関連付けられたメッセージトラフィックが表示されます。このパネルを使用して、ホスト間のアクティビティを監視できます。VD ホストタブにはインスタンス固有のアクティビティが表示され、VD セッションタブには進行中のセッションアクティビティが表示されます。

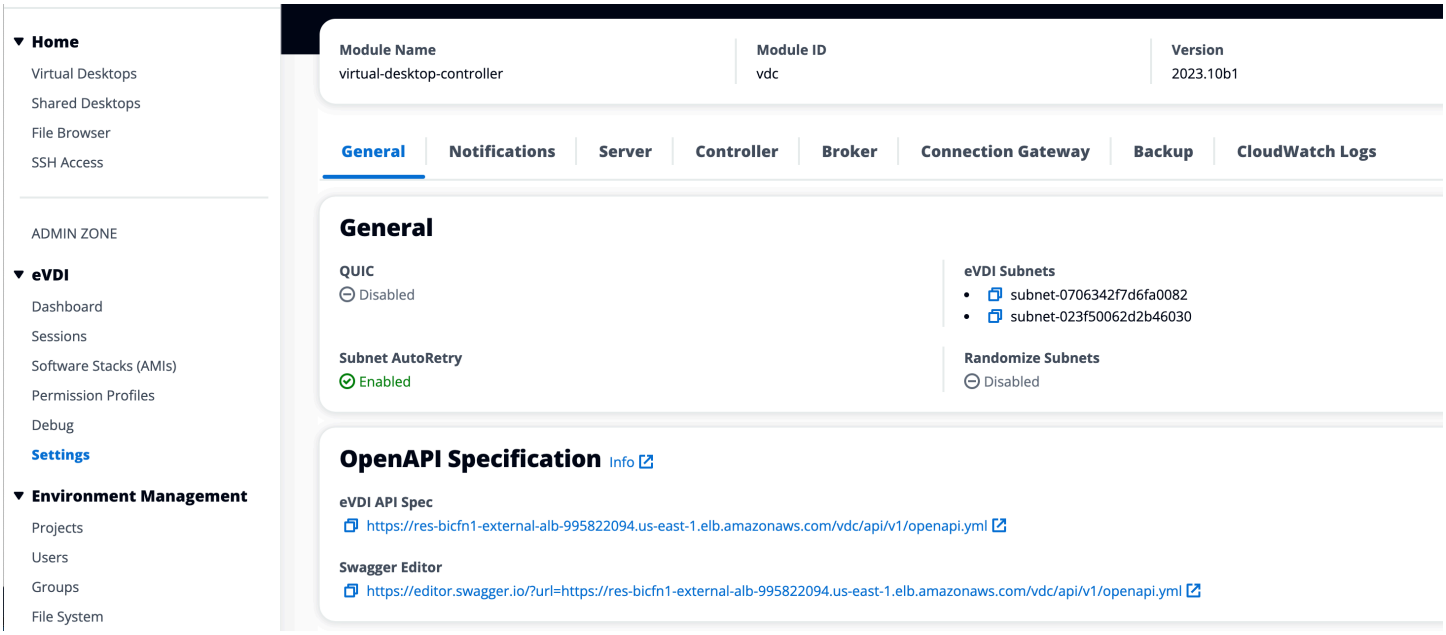



# デスクトップ設定

デスクトップ設定ページを使用して、仮想デスクトップに関連付けられたリソースを設定できます。サーバータブでは、次のような設定にアクセスできます。

- DCV セッションアイドルタイムアウト

- アイドルタイムアウトの警告
- CPU 使用率のしきい値
- ユーザーあたりの許可されたセッション




# 環境管理

RES の「環境管理」セクションから、管理ユーザーは研究およびエンジニアリングプロジェクト用に分離された環境を作成および管理できます。これらの環境には、コンピューティングリソース、ストレージ、その他の必要なコンポーネントがすべて安全な環境内に含まれる場合があります。ユーザーは、プロジェクトの特定の要件を満たすようにこれらの環境を設定およびカスタマイズできるため、他のプロジェクトや環境に影響を与えることなく、ソリューションの実験、テスト、反復が容易になります。

トピック

- プロジェクト
- [ユーザー]
- グループ
- ファイルシステム
- 環境ステータス
- スナップショット管理

- 環境設定

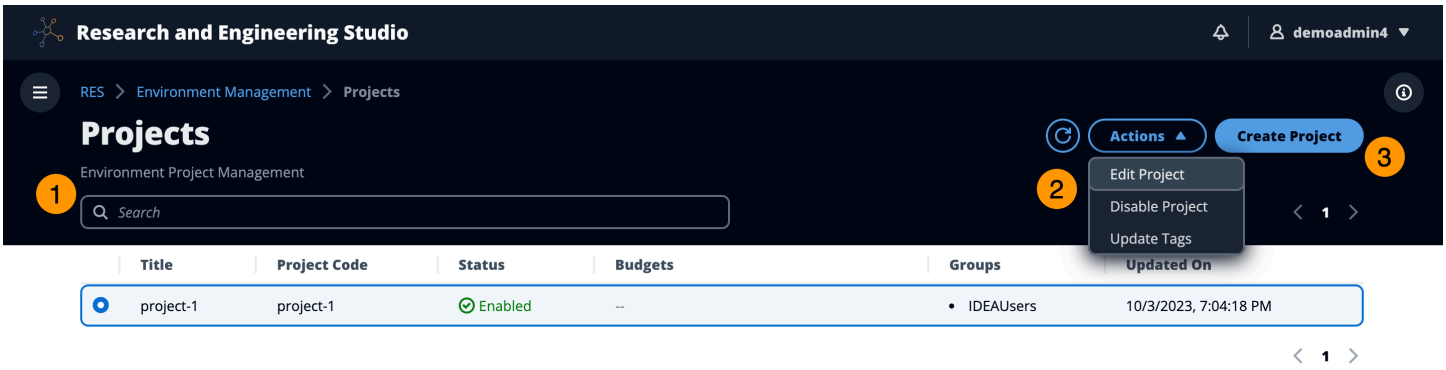
# プロジェクト

プロジェクトは、仮想デスクトップ、チーム、予算の境界を形成します。プロジェクトを作成するときは、名前、説明、環境設定などの設定を定義します。プロジェクトには通常、コンピューティングリソースのタイプとサイズ、ソフトウェアスタック、ネットワーク設定など、プロジェクトの特定の要件を満たすようにカスタマイズできる 1 つ以上の環境が含まれます。

トピック

- プロジェクトを表示する
- プロジェクトを作成する
- プロジェクトを編集する
- プロジェクトへのタグの追加または削除
- プロジェクトに関連付けられたファイルシステムを表示する
- 起動テンプレートを追加する

## プロジェクトを表示する




プロジェクトダッシュボードには、利用可能なプロジェクトのリストが表示されます。プロジェクトダッシュボードから、次のことができます。

1. 検索フィールドを使用してプロジェクトを検索できます。
2. プロジェクトを選択すると、アクションメニューを使用して次のことができます。
   a. プロジェクトを編集する
   b. プロジェクトの無効化または有効化

c. プロジェクトタグの更新

3. プロジェクトの作成を選択して、新しいプロジェクトを作成できます。

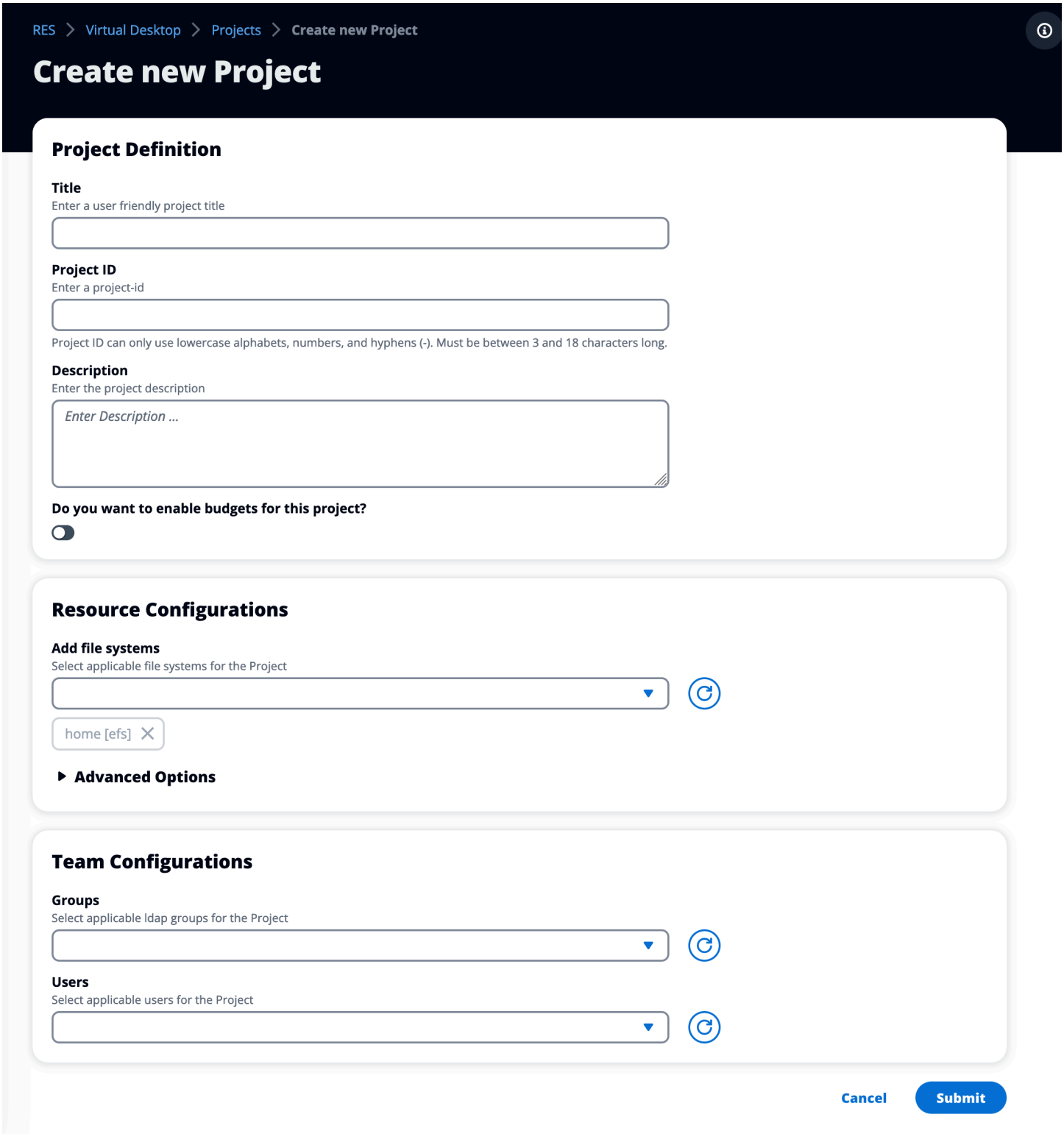
## プロジェクトを作成する

1. [プロジェクトを作成] を選択します。
2. プロジェクトの詳細を入力します。

   - プロジェクト ID は、 でコスト配分を追跡するために使用できるリソースタグです AWS Cost Explorer Service。詳細については、「ユーザー定義のコスト配分タグのアクティブ化」を参照してください。

     > **Important**
     >
     > 作成後にプロジェクト ID を変更することはできません。

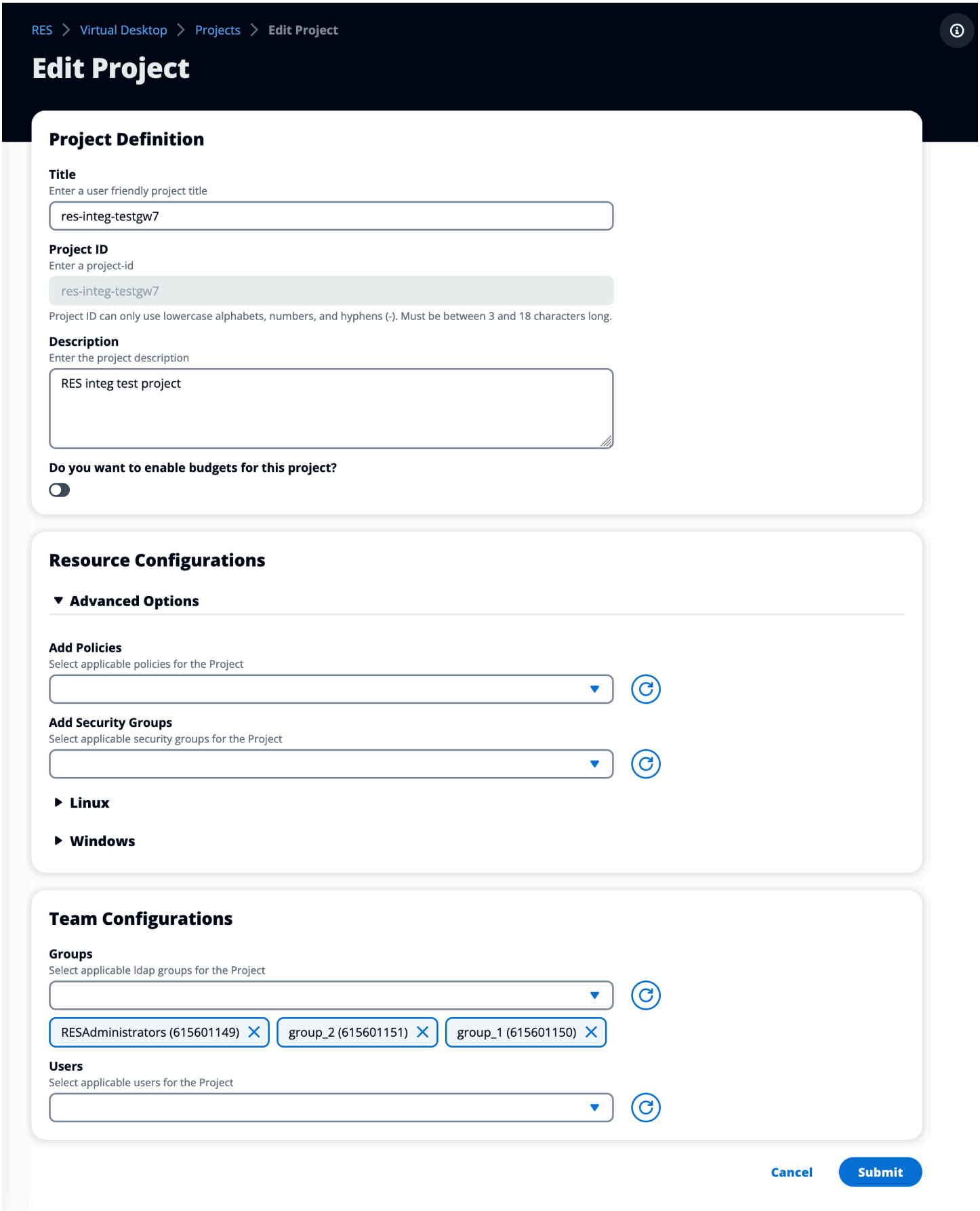
   - 詳細オプション の詳細については、「」を参照してくださいthe section called “起動テンプレートを追加する”。
3. ( オプション) プロジェクトの予算を有効にします。予算の詳細については、「」を参照してくださいthe section called “コストのモニタリングと制御”。
4. [送信] を選択します。




## プロジェクトを編集する

1. プロジェクトリストでプロジェクトを選択します。

2. アクションメニューから、プロジェクトの編集 を選択します。
3. 更新を入力します。予算を有効にする場合は、the section called “コストのモニタリングと制御”「」で詳細を確認してください。詳細オプション の詳細については、「」を参照してくださいthe section called “起動テンプレートを追加する”。
4. [送信] を選択します。


RES > Virtual Desktop > Projects > Edit Project
Edit Project
Project Definition
Title
Enter a user friendly project title
res-integ-testgw7
Project ID
Enter a project-id
res-integ-testgw7
Project ID can only use lowercase alphabets, numbers, and hyphens (-). Must be between 3 and 18 characters long.
Description
Enter the project description
RES integ test project
Do you want to enable budgets for this project?
Resource Configurations
Advanced Options
Add Policies
Select applicable policies for the Project
Add Security Groups
Select applicable security groups for the Project
Linux
Windows
Team Configurations
Groups
Select applicable ldap groups for the Project
RESAdministrators (615601149)
group_2 (615601151)
group_1 (615601150)
Users
Select applicable users for the Project
Cancel
Submit

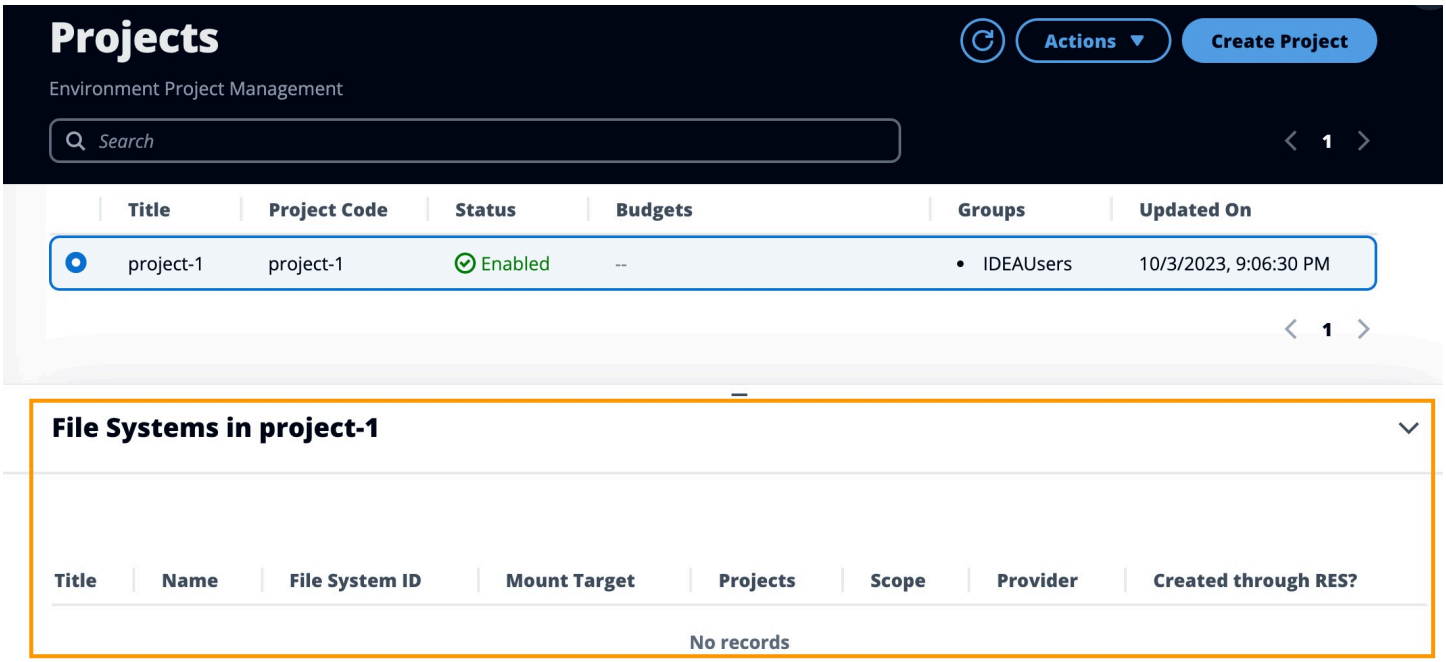
## プロジェクトへのタグの追加または削除

プロジェクトタグは、そのプロジェクトで作成されたすべてのインスタンスにタグを割り当てます。

1. プロジェクトリストでプロジェクトを選択します。
2. アクションメニューから、タグの更新 を選択します。
3. タグの追加 を選択し、キー の値を入力します。
4. タグを削除するには、削除するタグの横にある削除を選択します。

## プロジェクトに関連付けられたファイルシステムを表示する

プロジェクトを選択すると、画面の下部にあるファイルシステムペインを展開して、プロジェクトに関連付けられたファイルシステムを表示できます。




## 起動テンプレートを追加する

プロジェクトを作成または編集するときは、プロジェクト設定内の詳細オプションを使用して起動テンプレートを追加できます。起動テンプレートは、セキュリティグループ、IAM ポリシー、起動スクリプトなどの追加設定をプロジェクト内のすべての VDI インスタンスに提供します。

ポリシーの追加

IAM ポリシーを追加して、プロジェクトの下にデプロイされたすべてのインスタンスの VDI アクセスを制御できます。ポリシーをオンボードするには、ポリシーに次のキーと値のペアをタグ付けします。

```
res:Resource/vdi-host-policy
```

IAM ロールの詳細については、「IAM のポリシーとアクセス許可」を参照してください。

セキュリティグループの追加

セキュリティグループを追加して、プロジェクト内のすべての VDI インスタンスの出力データと入力データを制御できます。セキュリティグループをオンボードするには、セキュリティグループに次のキーと値のペアをタグ付けします。

```
res:Resource/vdi-security-group
```

セキュリティグループの詳細については、「Amazon VPC ユーザーガイド」の「セキュリティグループを使用して AWS リソースへのトラフィックを制御する」を参照してください。

起動スクリプトの追加

プロジェクト内のすべての VDI セッションで開始する起動スクリプトを追加できます。RES は Linux および Windows のスクリプト開始をサポートしています。スクリプトを開始するには、次のいずれかを選択できます。

VDI の開始時にスクリプトを実行する

このオプションは、RES 設定またはインストールを実行する前に、VDI インスタンスの先頭でスクリプトを開始します。

VDI が設定されている場合にスクリプトを実行する

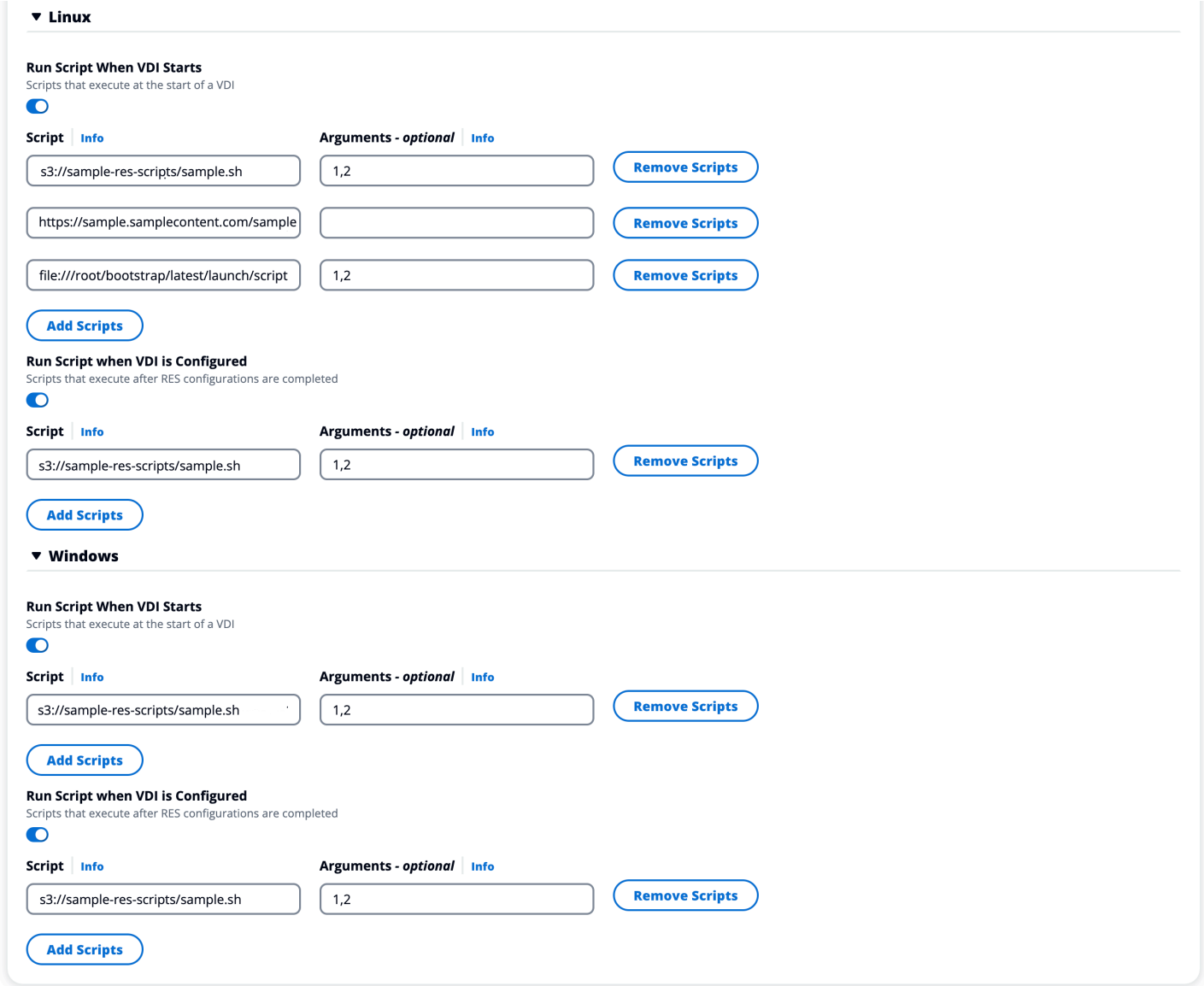
このオプションは、RES 設定が完了した後にスクリプトを開始します。

スクリプトは、次のオプションをサポートしています。

| スクリプト設定 | 例 |
| --- | --- |
| S3 URI | s3://bucketname/script.sh |

| スクリプト設定 | 例 |
| --- | --- |
| HTTPS URL | https://sample.samplecontent.com/sample |
| ローカルファイル | file:///user/scripts/example.sh |

引数 には、カンマで区切られた引数を指定します。

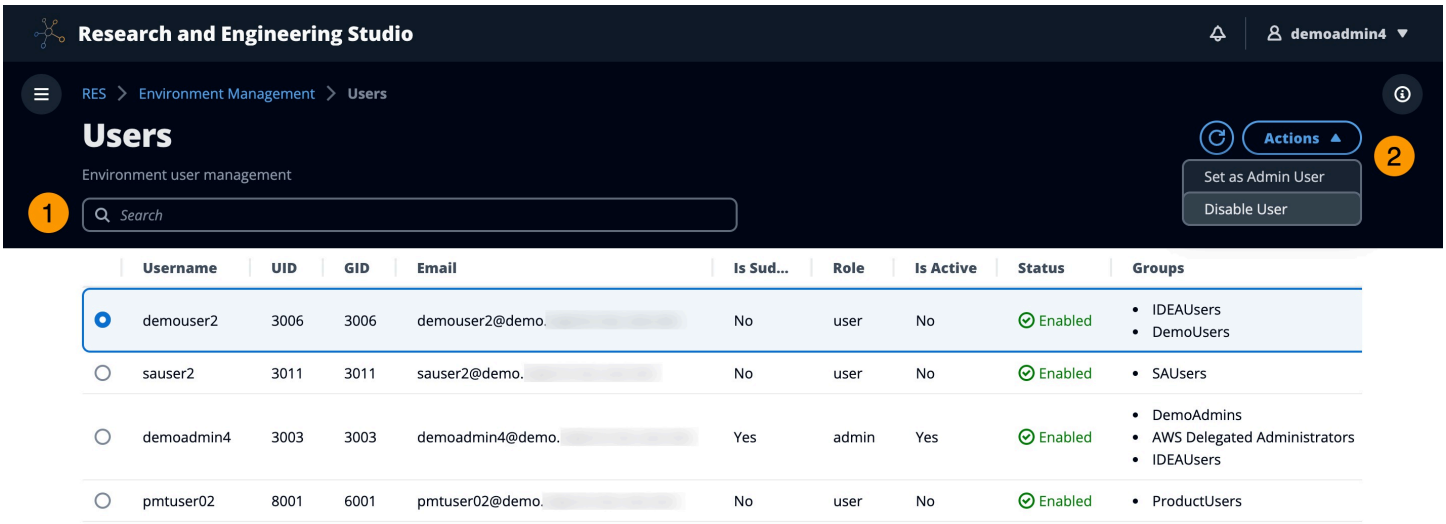



プロジェクト設定の例

# [ユーザー]

アクティブディレクトリから同期されたすべてのユーザーがユーザーページに表示されます。ユーザーは、製品の設定中に cluster-admin ユーザーによって同期されます。初期設定の詳細については、「」を参照してください設定ガイド。

> **Note**
>
> 管理者は、アクティブなユーザーのセッションのみを作成できます。デフォルトでは、すべてのユーザーは製品環境にサインインするまで非アクティブ状態になります。ユーザーが非アクティブの場合は、セッションを作成する前にサインインするようユーザーに依頼します。

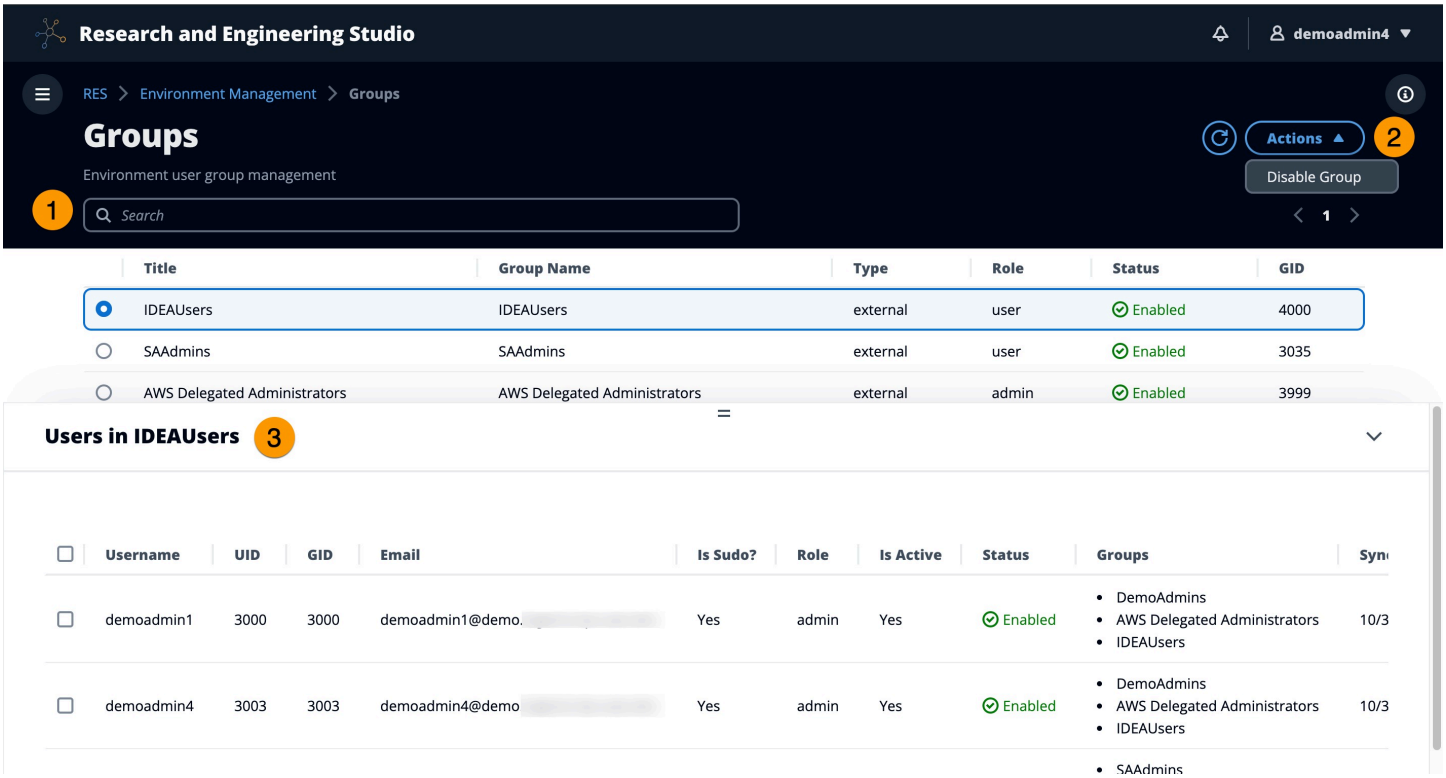



ユーザーページから、次のことができます。

1. ユーザーを検索します。
2. ユーザー名を選択したら、アクションメニューを使用して次の操作を行います。
   a. 管理者ユーザーとして設定する
   b. ユーザーを無効にする

# グループ

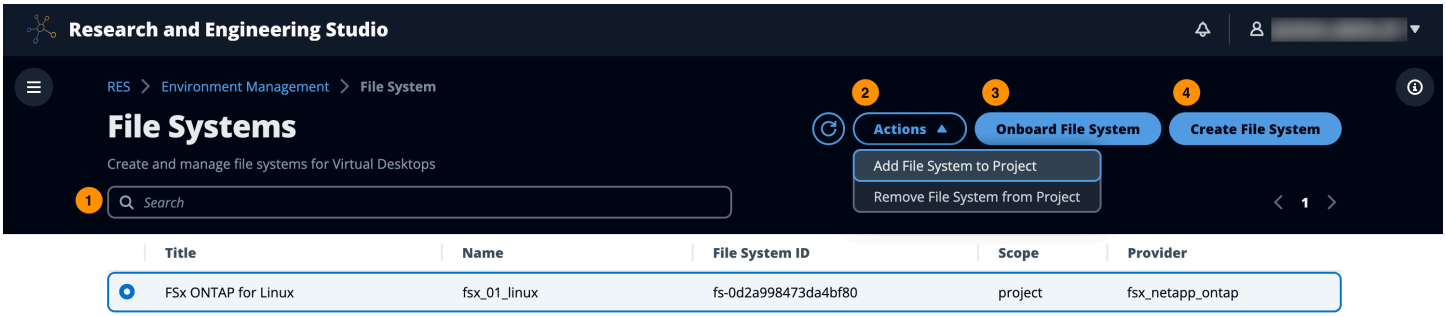
アクティブディレクトリから同期されたすべてのグループは、グループページに表示されます。グループの設定と管理の詳細については、「」を参照してください設定ガイド。




グループページから、次のことができます。

1. ユーザーグループを検索します。
2. ユーザーグループを選択したら、アクションメニューを使用してグループを無効または有効にします。
3. ユーザーグループを選択すると、画面の下部にあるユーザーペインを展開して、グループ内のユーザーを表示できます。

# ファイルシステム




ファイルシステムページから、次のことができます。

1. ファイルシステムを検索します。
2. ファイルシステムを選択したら、アクションメニューを使用して次の操作を行います。
   a. ファイルシステムをプロジェクトに追加する
   b. プロジェクトからファイルシステムを削除する
3. 新しいファイルシステムをオンボードします。
4. ファイルシステムを作成します。
5. ファイルシステムを選択すると、画面の下部にあるペインを展開して、ファイルシステムの詳細を表示できます。

## ファイルシステムを作成する

1. [ファイルシステムの作成] を選択します。
2. 新しいファイルシステムの詳細を入力します。
3. VPC からサブネット IDs を指定します。IDs は、環境管理 > 設定 > ネットワークタブにあります。
4. [送信] を選択します。

# Create new File System

**Title**

Enter a user friendly file system title

Eg. EFS 01

**Name**

Enter a file system name

File System name can only use lowercase alphabets, numbers and underscore (_). Must be between 3 and 18 characters long.

**File System Provider**

Select applicable file system type

EFS

**Projects**

Select applicable project

**Subnet ID 1**

Enter subnet id to create mount target

**Subnet ID 2**

Enter second subnet to create mount target

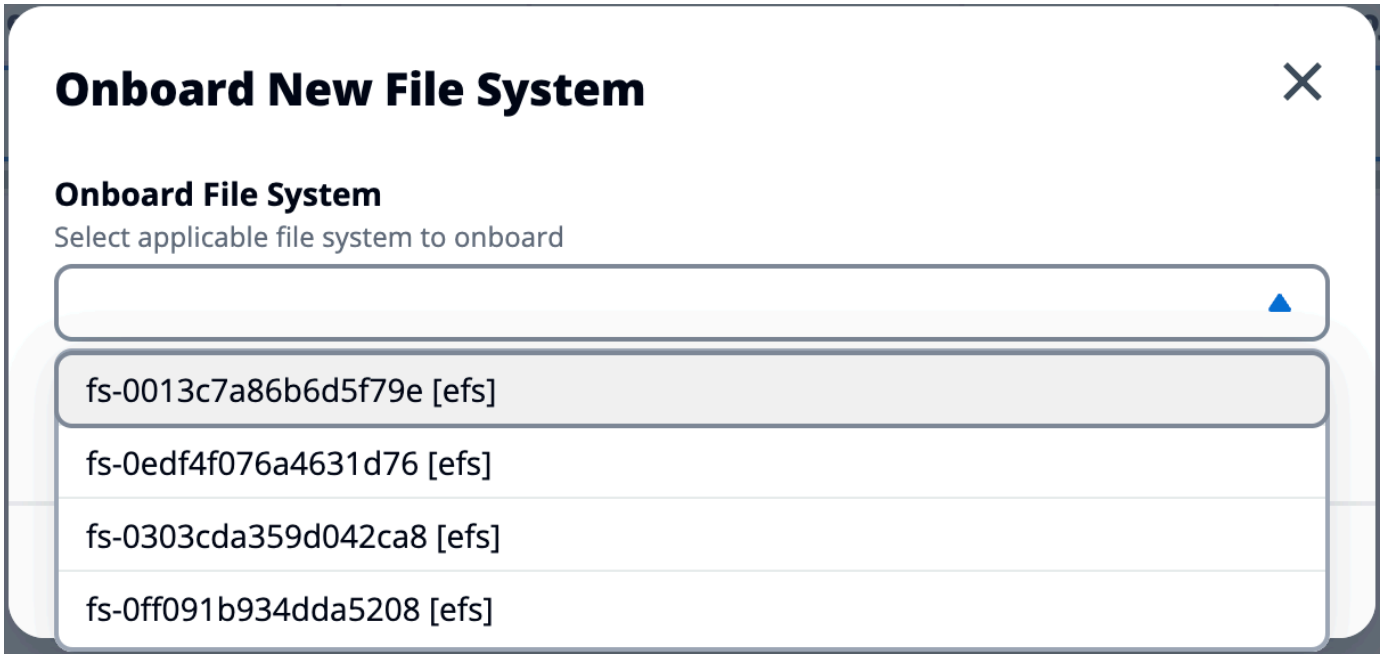
Subnet ID 1 and Subnet ID 2 should be in two different AZs

**Mount Directory**

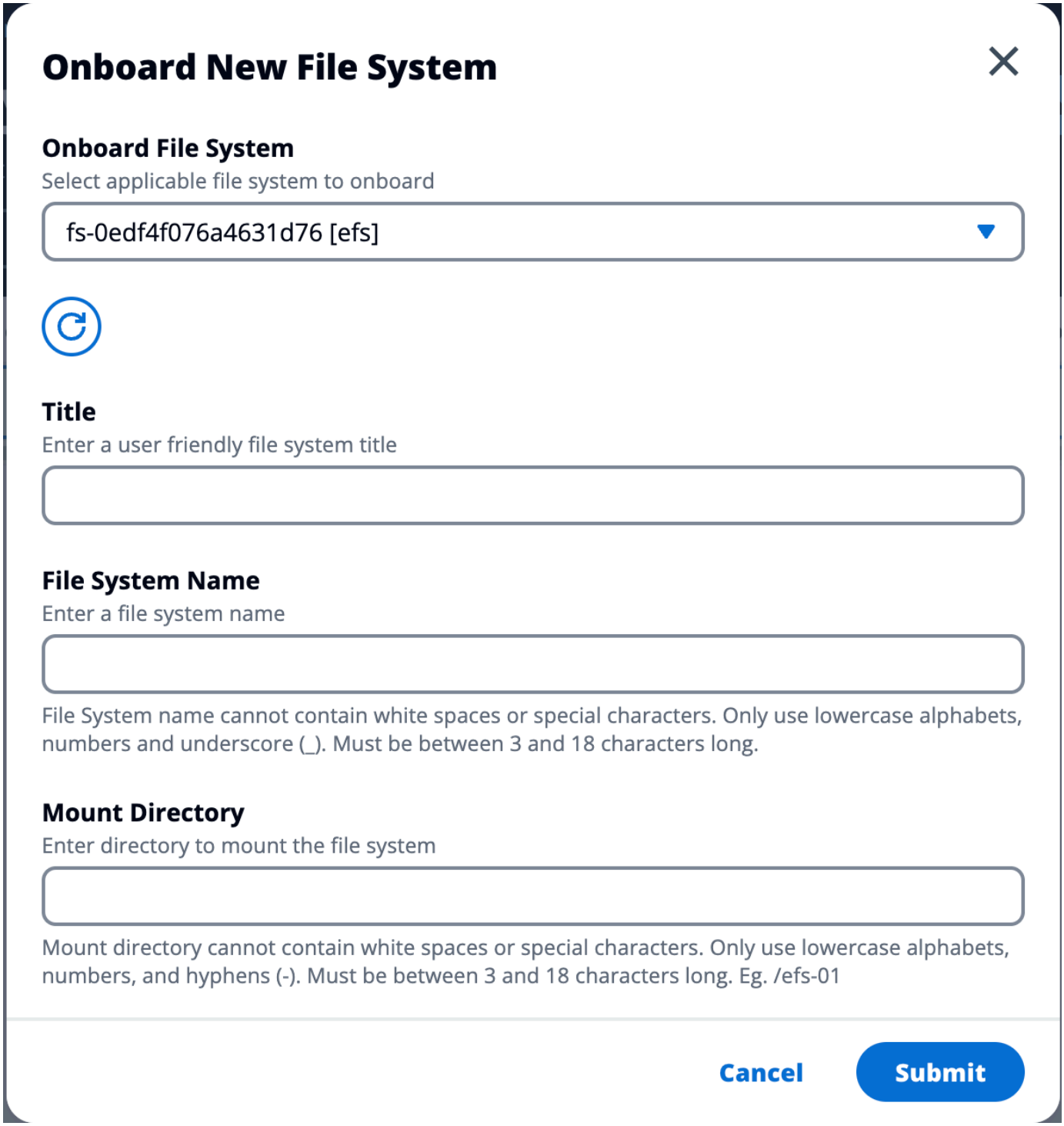
Enter directory to mount the file system

# ファイルシステムのオンボード

1. ファイルシステムのオンボード を選択します。
2. ドロップダウンからファイルシステムを選択します。モーダルが拡張され、追加の詳細エントリが表示されます。

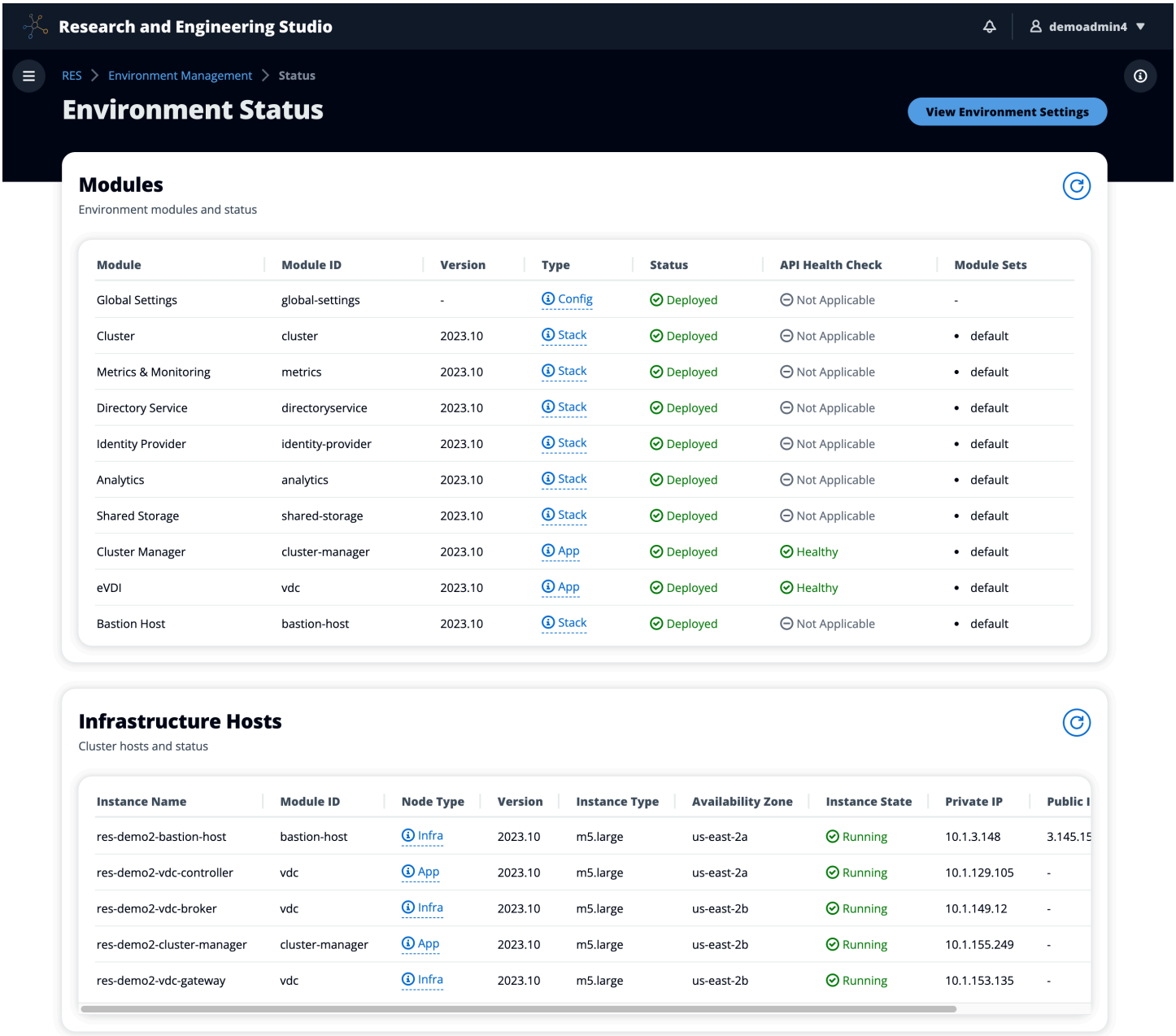



3. ファイルシステムの詳細を入力します。
4. [送信] を選択します。



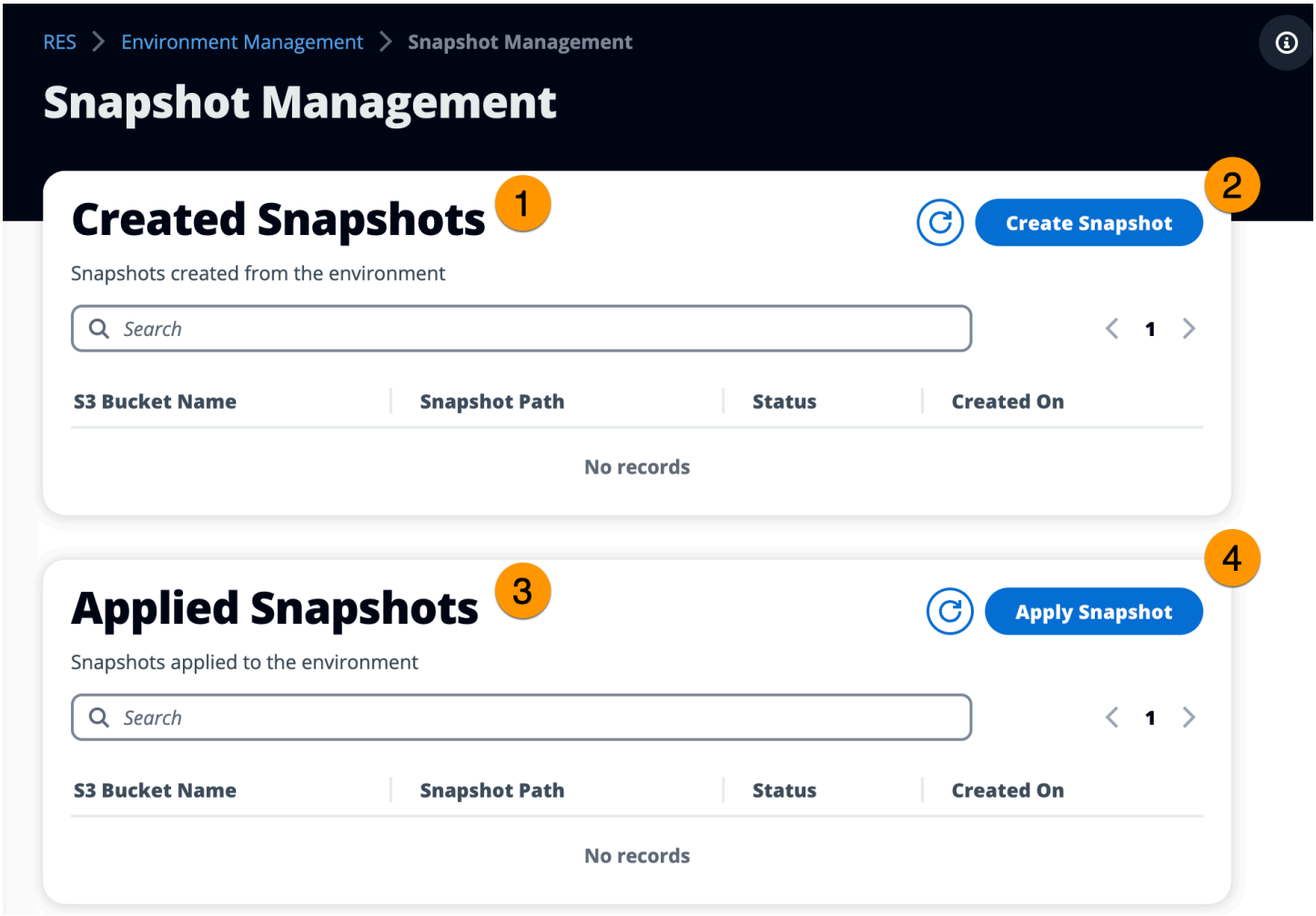

## 環境ステータス

環境ステータスページには、製品内にデプロイされたソフトウェアとホストが表示されます。これには、ソフトウェアバージョン、モジュール名、その他のシステム情報などの情報が含まれます。




# スナップショット管理

スナップショット管理は、環境間でのデータの保存と移行のプロセスを簡素化し、一貫性と正確性を確保します。スナップショットを使用すると、環境の状態を保存し、同じ状態の新しい環境にデータを移行できます。




スナップショット管理ページから、次のことができます。

1. 作成されたすべてのスナップショットとそのステータスを表示します。
2. スナップショットを作成します。スナップショットを作成する前に、適切なアクセス許可を持つバケットを作成する必要があります。
3. 適用されたすべてのスナップショットとそのステータスを表示します。
4. スナップショットを適用します。

## スナップショットを作成する

スナップショットを作成する前に、必要なアクセス許可を Amazon S3 バケットに提供する必要があります。 バケットの作成については、「バケットを作成する」を参照してください。バケットのバージョニングとサーバーアクセスのログ記録を有効にすることをお勧めします。これらの設定は、プロビジョニング後にバケットのプロパティタブから有効にできます。

> **Note**
>
> この Amazon S3 バケットのライフサイクルは、製品内で管理されません。バケットのライフサイクルはコンソールから管理する必要があります。

バケットにアクセス許可を追加するには：

1. バケットリストから作成したバケットを選択します。
2. [アクセス許可] タブを選択します。
3. [バケットポリシー] で [編集] を選択します。
4. バケットポリシーに次のステートメントを追加します。以下の値を自分の値に置き換えてください。

   - AWS_ACCOUNT_ID
   - RES_ENVIRONMENT_NAME
   - AWS_REGION
   - S3_BUCKET_NAME

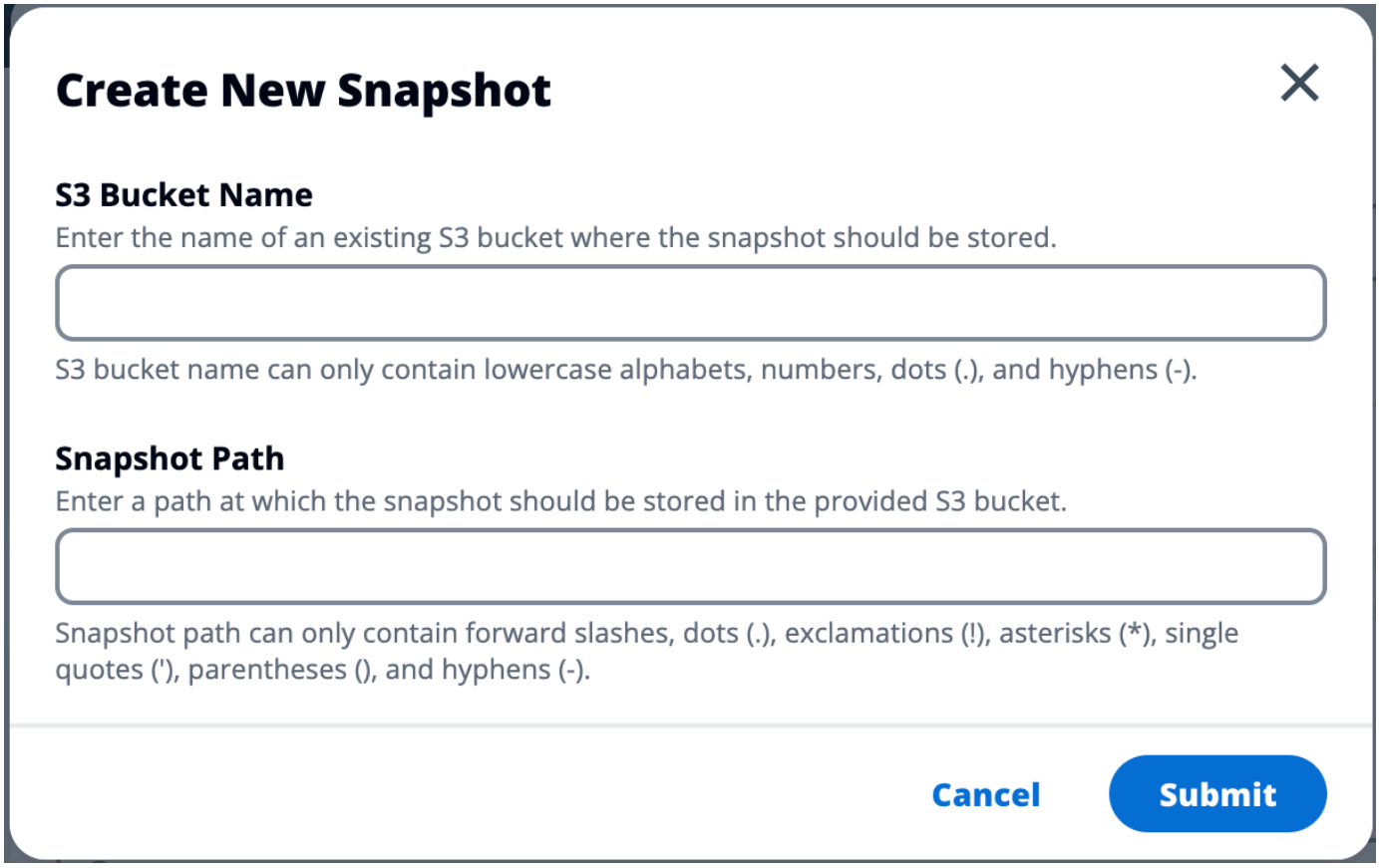
> **Important**
>
> でサポートされる限定バージョンの文字列があります AWS。詳細については、「https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_version.html」を参照してください。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Export-Snapshot-Policy",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{AWS_ACCOUNT_ID}:role/{RES_ENVIRONMENT_NAME}-
cluster-manager-role-{AWS_REGION}}"
            },
            "Action": [
```

```
                "s3:GetObject",
                "s3:ListBucket",
                "s3:AbortMultipartUpload",
                "s3:PutObject",
                "s3:PutObjectAcl"
            ],
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
            ]
        },
        {
            "Sid": "AllowSSLRequestsOnly",
            "Action": "s3:*",
            "Effect": "Deny",
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
            ],
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            },
            "Principal": "*"
        }
    ]
}
```

スナップショットを作成するには：

1. [スナップショットの作成] を選択します。
2. 作成した Amazon S3 バケットの名前を入力します。
3. バケット内にスナップショットを保存するパスを入力します。例えば **october2023/23** です。
4. [送信] を選択します。




5. 5～10 分後、スナップショットページで更新を選択してステータスを確認します。スナップショットは、ステータスが IN_PROGRESS から COMPLETED に変わるまで有効になりません。

## スナップショットの適用

環境のスナップショットを作成したら、そのスナップショットを新しい環境に適用してデータを移行できます。環境がスナップショットを読み取れるように、バケットに新しいポリシーを追加する必要があります。

スナップショットを適用すると、ユーザーアクセス許可、プロジェクト、ソフトウェアスタック、アクセス許可プロファイル、ファイルシステムなどのデータが新しい環境にコピーされます。ユーザーセッションはレプリケートされません。スナップショットが適用されると、各リソースレコードの基本情報をチェックして、そのスナップショットが既に存在するかどうかを確認します。レコードが重複している場合、スナップショットは新しい環境でのリソースの作成をスキップします。名前やキーを共有するなど、似たようなレコードの場合、他の基本的なリソース情報はさまざまですが、次の規則を使用して名前とキーを変更した新しいレコードが

作成されます: RecordName_SnapshotRESVersion_ApplySnapshotID。はタイムスタンプのApplySnapshotIDように見えるため、スナップショットの適用を試みるたびに識別されます。

スナップショットアプリケーション中、スナップショットはリソースの可用性をチェックします。新しい環境で使用できないリソースは作成されません。依存リソースを持つリソースの場合、スナップショットは依存リソースの可用性をチェックします。依存リソースが使用できない場合、依存リソースなしでメインリソースが作成されます。

新しい環境が想定どおりにないか、失敗した場合、 CloudWatch ロググループで見つかったログ/res-<env-name>/cluster-managerの詳細を確認できます。各ログには [apply snapshot] タグがあります。スナップショットを適用したら、ページからそのステータスを確認できますthe section called “スナップショット管理”。

バケットにアクセス許可を追加するには：

1. バケットリストから作成したバケットを選択します。
2. [アクセス許可] タブを選択します。
3. [バケットポリシー] で [編集] を選択します。
4. バケットポリシーに次のステートメントを追加します。以下の値を自分の値に置き換えてください。

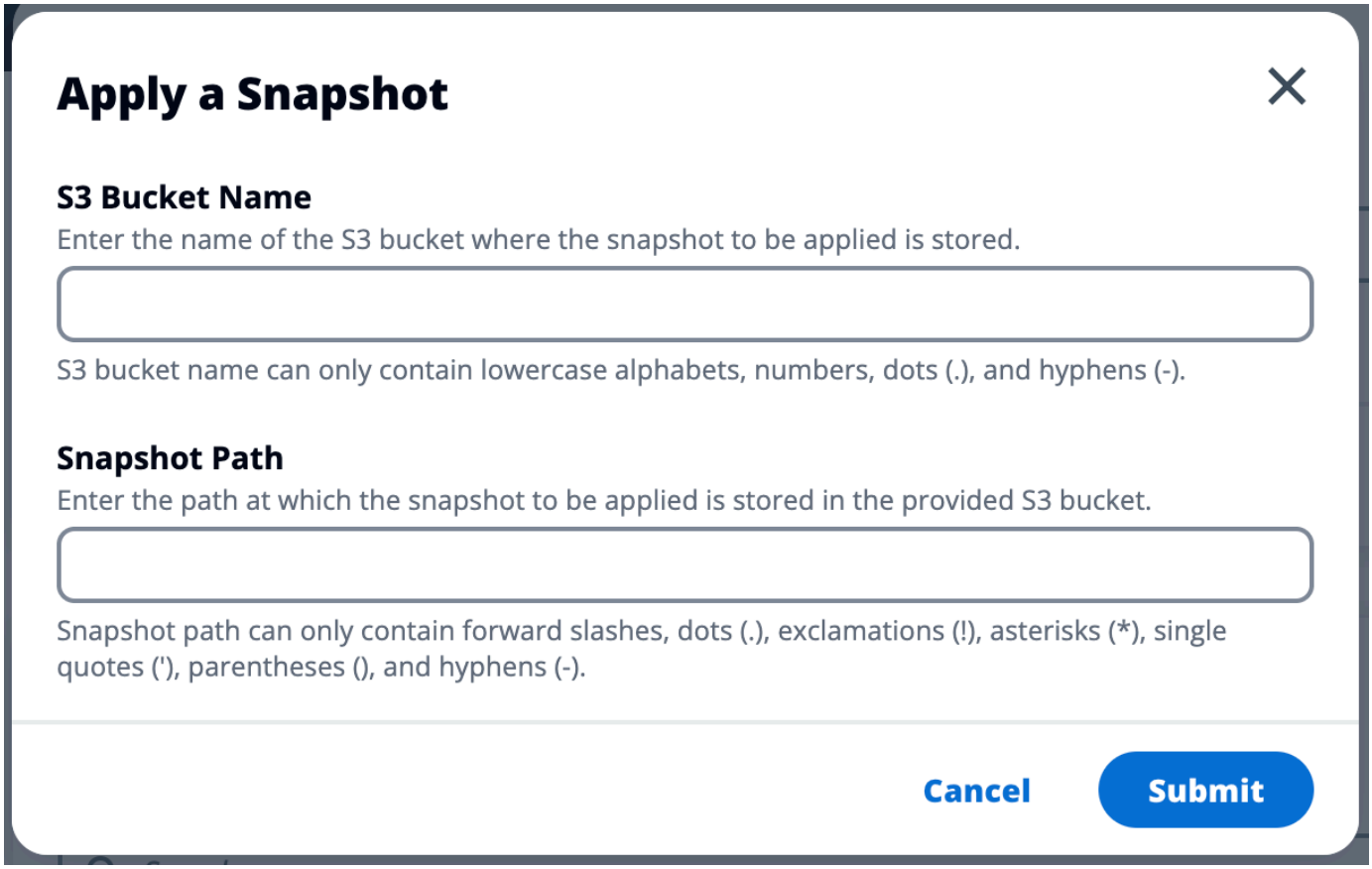
   - AWS_ACCOUNT_ID
   - RES_ENVIRONMENT_NAME
   - AWS_REGION
   - S3_BUCKET_NAME

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Export-Snapshot-Policy",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::{AWS_ACCOUNT_ID}:role/{RES_ENVIRONMENT_NAME}-
cluster-manager-role-{AWS_REGION}}"
            },
            "Action": [
                "s3:GetObject",
```

```
                "s3:ListBucket"
            ],
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
            ]
        },
        {
            "Sid": "AllowSSLRequestsOnly",
            "Action": "s3:*",
            "Effect": "Deny",
            "Resource": [
                "arn:aws:s3:::{S3_BUCKET_NAME}",
                "arn:aws:s3:::{S3_BUCKET_NAME}/*"
            ],
            "Condition": {
                "Bool": {
                    "aws:SecureTransport": "false"
                }
            },
            "Principal": "*"
        }
    ]
}
```

スナップショットを適用するには：

1. スナップショットを適用 を選択します。
2. スナップショットを含む Amazon S3 バケットの名前を入力します。
3. バケット内のスナップショットへのファイルパスを入力します。
4. [送信] を選択します。




5. 5～10 分後、スナップショット管理ページで更新を選択してステータスを確認します。

## 環境設定

環境設定には、次のような製品設定の詳細が表示されます。

- 全般

  製品をプロビジョニングしたユーザーの管理者ユーザー名や E メールなどの情報を表示します。ウェブポータルのタイトルと著作権テキストを編集できます。

- ID プロバイダー

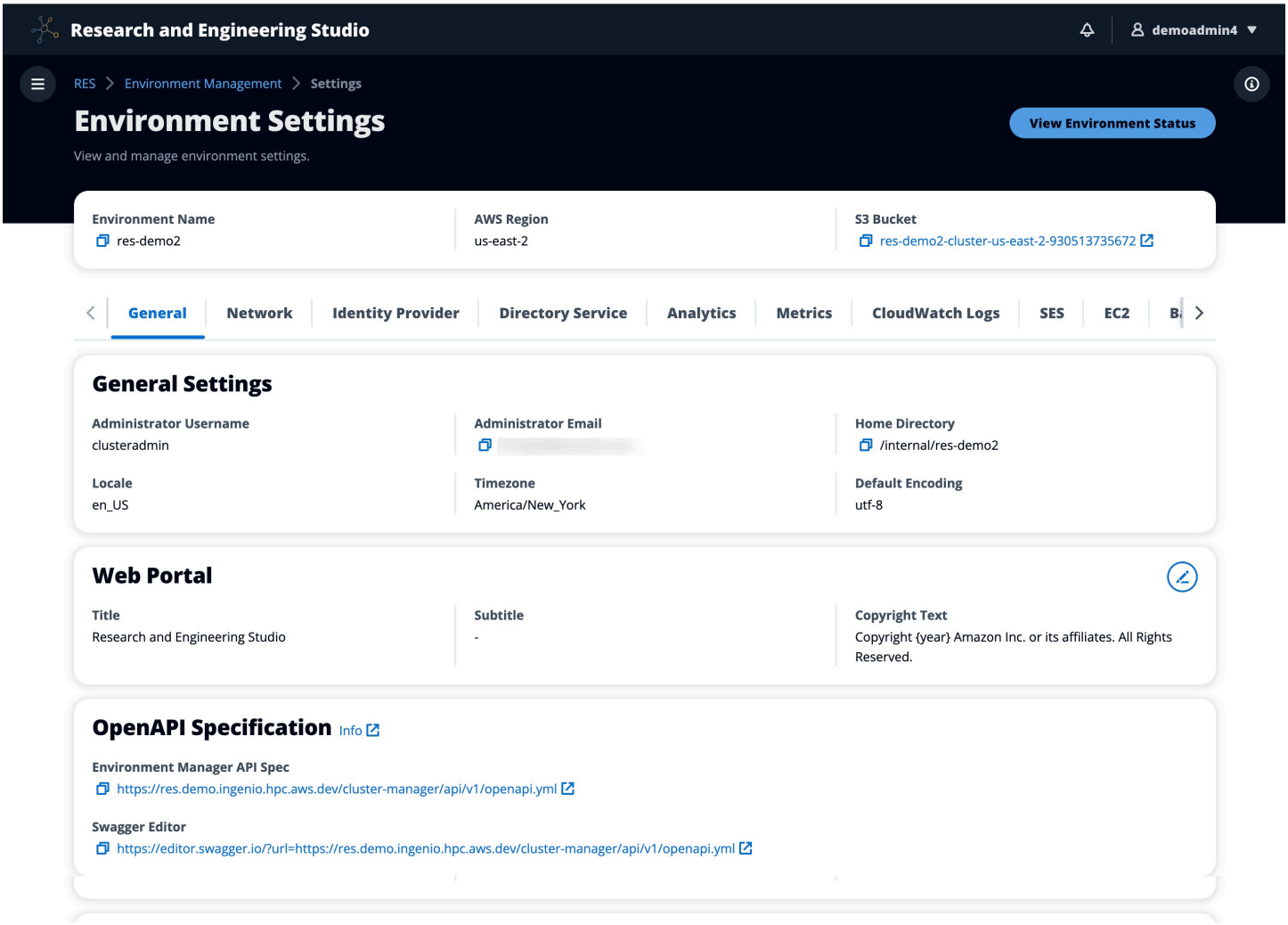
  Single Sign-On ステータスなどの情報を表示します。

- ネットワーク

  アクセス用の VPC ID、プレフィックスリスト IDsを表示します。

- Directory Service

ユーザー名とパスワードのアクティブディレクトリ設定とサービスアカウントシークレットマネージャー ARN を表示します。




# シークレットの管理

Research and Engineering Studio は、 を使用して以下のシークレットを保持します AWS Secrets Manager。RES は、環境の作成時にシークレットを自動的に作成します。環境の作成時に管理者が入力したシークレットは、パラメータとして入力されます。

| シークレット名 | 説明 | 生成された RES | 入力された管理者 |
|---|---|---|---|
| <envname>-sso-client-secret | 環境用の Single Sign-On OAuth2 クライアントシークレット | ✓ | |
| <envname>-vdc-client-secret | VDC ClientSecret | ✓ | |
| <envname>-vdc-client-id | VDC ClientId | ✓ | |
| <envname>-vdc-gateway-certificate-private-key | ドメインの自己署名証明書プライベートキー | ✓ | |
| <envname>-vdc-gateway-certificate-certificate | ドメインの自己署名証明書 | ✓ | |
| <envname>-cluster-manager-client-secret | クラスターマネージャー ClientSecret | ✓ | |
| <envname>-cluster-manager-client-id | クラスターマネージャー ClientId | ✓ | |
| <envname>-external-private-key | ドメインの自己署名証明書プライベートキー | ✓ | |
| <envname>-external-certificate | ドメインの自己署名証明書 | ✓ | |
| <envname>-internal-private-key | ドメインの自己署名証明書プライベートキー | ✓ | |
| <envname>-internal-certificate | ドメインの自己署名証明書 | ✓ | |

| シークレット名 | 説明 | 生成された RES | 入力された管理者 |
|---|---|---|---|
| <envname>-directoryservice-ServiceAccountUsername | | | ✓ |
| <envname>-directoryservice-ServiceAccountPassword | | | ✓ |

次のシークレット ARN 値は、DynamoDB の <envname>-cluster-settings テーブルに含まれています。

| キー | ソース |
|---|---|
| identity-provider.cognito.sso_client_secret | |
| vdc.dcv_connection_gateway.certificate.certificate_secret_arn | スタック |
| vdc.dcv_connection_gateway.certificate.private_key_secret_arn | スタック |
| cluster.load_balancers.internal_alb.certificates.private_key_secret_arn | スタック |
| directoryservice.root_username_secret_arn | |
| vdc.client_secret | スタック |
| cluster.load_balancers.external_alb.certificates.certificate_secret_arn | スタック |
| cluster.load_balancers.internal_alb.certificates.certificate_secret_arn | スタック |
| directoryservice.root_password_secret_arn | |
| cluster.secretsmanager.kms_key_id | |

| キー | ソース |
| --- | --- |
| cluster.load_balancers.external_alb.certificates.private_key_secret_arn | スタック |
| cluster-manager.client_secret | |

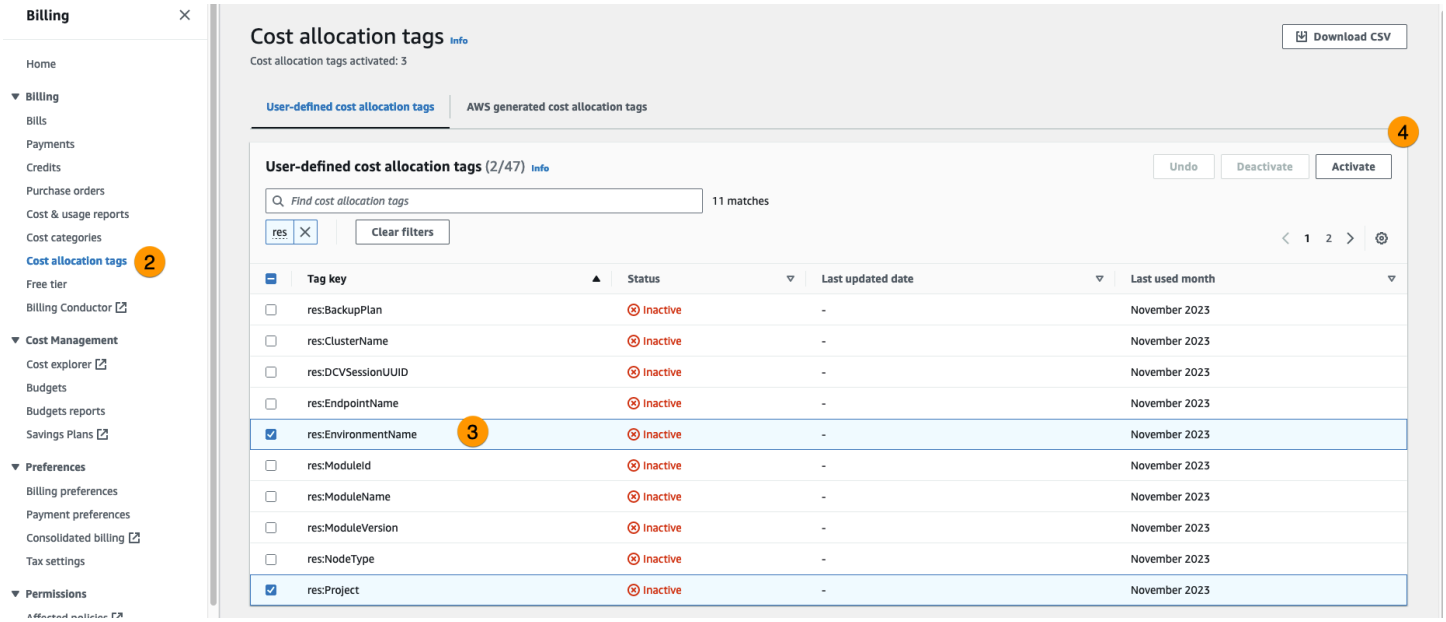
# コストのモニタリングと制御

> **Note**
>
> Research and Engineering Studio プロジェクトを に関連付けることは AWS Budgets 、 ではサポートされていません AWS GovCloud (US)。

AWS Cost Explorer を使用して予算を作成し、コストを管理することをお勧めします。価格は変更されることがあります。詳細については、各 の料金ウェブページを参照してくださいthe section called “AWS この製品の サービス”。

コスト追跡を支援するために、RES プロジェクトを 内で作成された予算に関連付けることができます AWS Budgets。まず、請求コスト配分タグ内で環境タグをアクティブ化する必要があります。

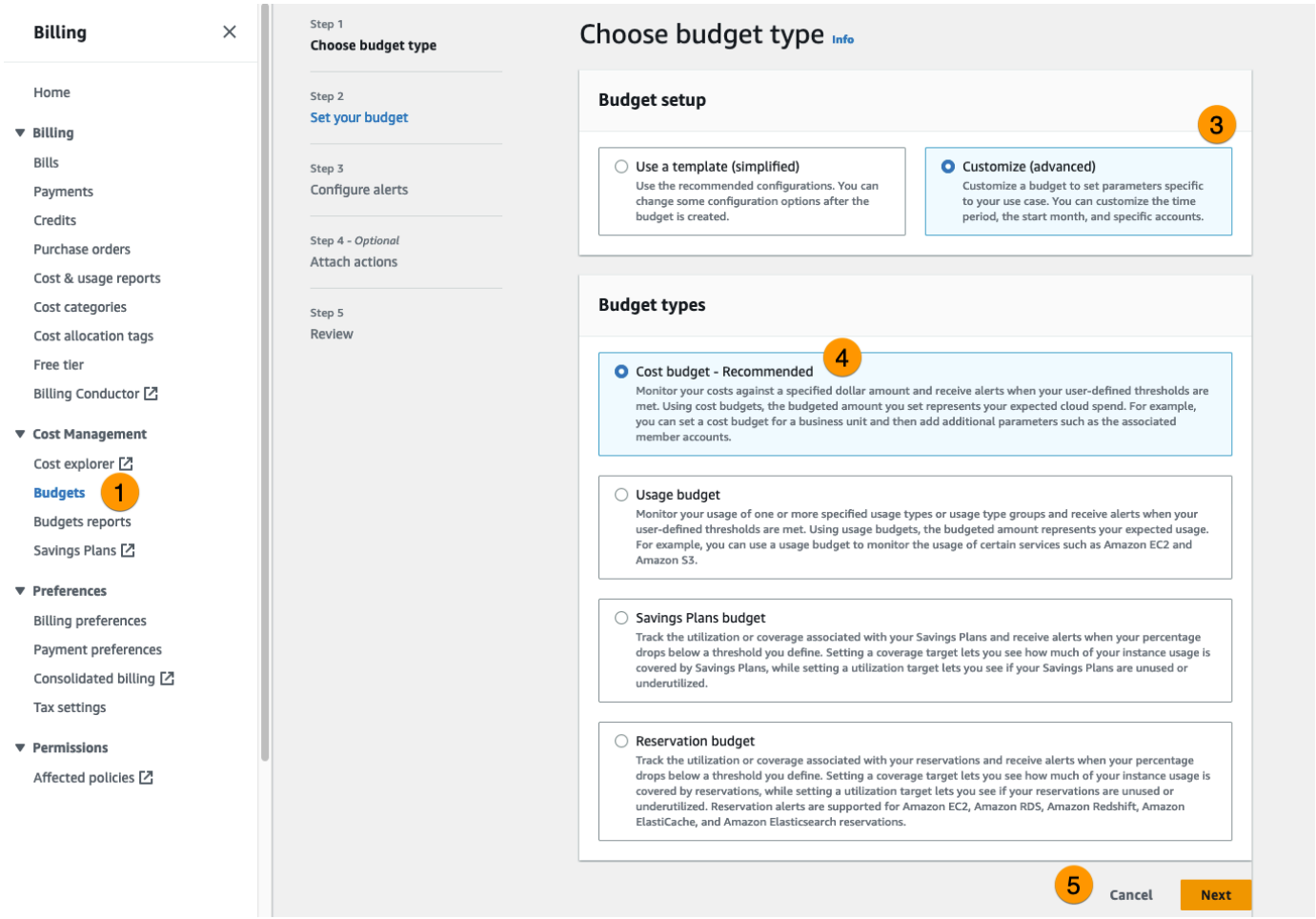
1. にサインイン AWS Management Console し、https://console.aws.amazon.com/billing/ で AWS Billing コンソールを開きます。
2. コスト配分タグ を選択します。
3. および `res:EnvironmentName` タグを検索`res:Project`して選択します。
4. [アクティブ化] を選択します。




> **Note**
>
> RES タグがデプロイ後に表示されるまでに最大 1 日かかる場合があります。

RES リソースの予算を作成するには：

1. 請求コンソールから、予算 を選択します。
2. 予算の作成 を選択します。
3. [Budget setup] (予算の設定) で、[Customize (advanced)] (カスタマイズ (高度)) を選択します。
4. 予算タイプ で、コスト予算 - 推奨 を選択します。
5. [次へ] をクリックします。

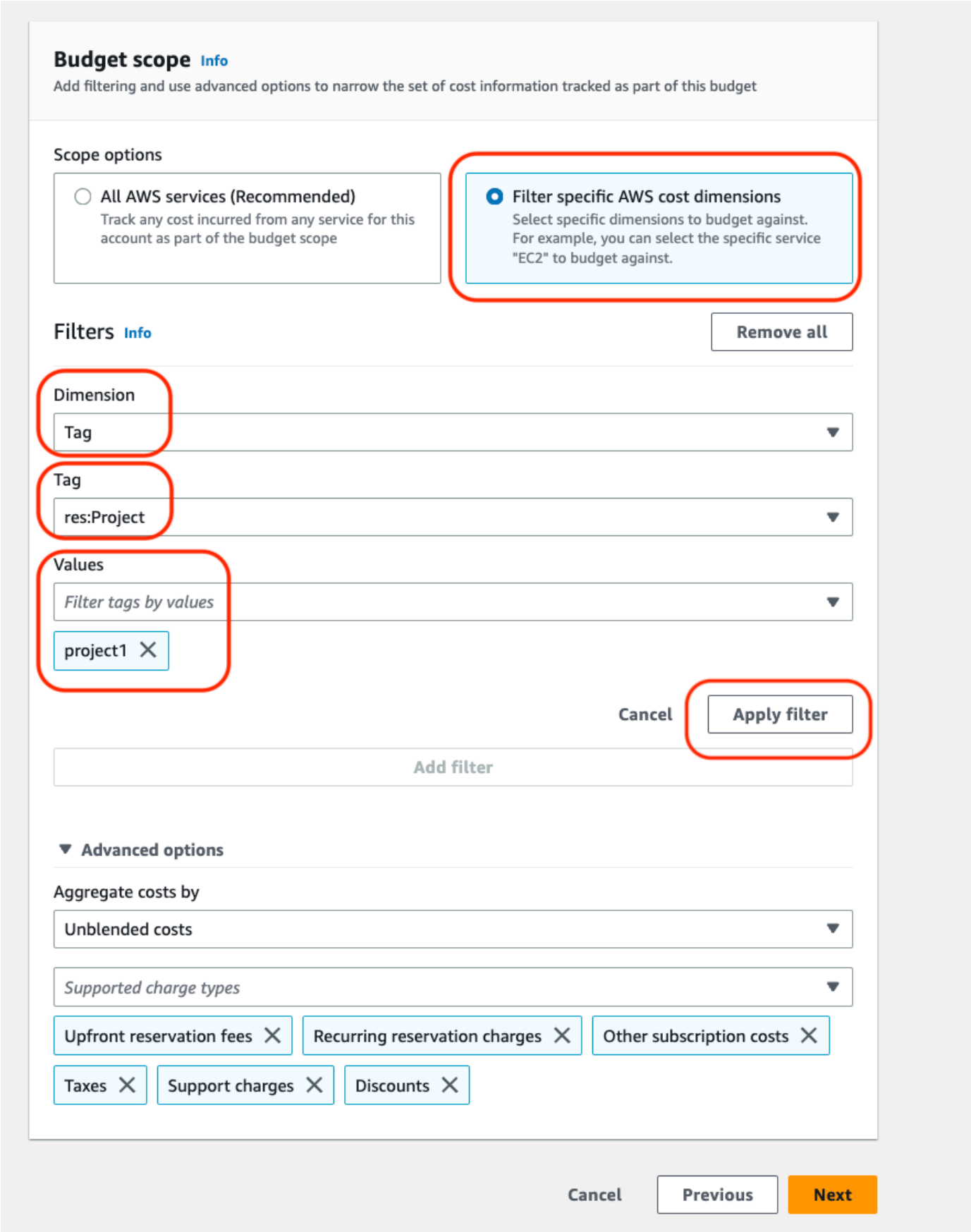



6. 詳細 で、予算のわかりやすい Budget 名を入力して、アカウントの他の予算と区別します。例えば、〔EnvironmentName]-[ProjectName]-[] などBudgetNameです。
7. 予算額の設定 で、プロジェクトに予算された金額を入力します。
8. 予算範囲 で、フィルター固有の AWS コストディメンション を選択します。
9. [Add filter] (フィルターを追加) を選択します。
10. ディメンション で、タグ を選択します。
11. タグ で、res:Project を選択します。

> **Note**
>
> タグと値が使用可能になるまでに最大 2 日かかる場合があります。プロジェクト名が使用可能になったら、予算を作成できます。

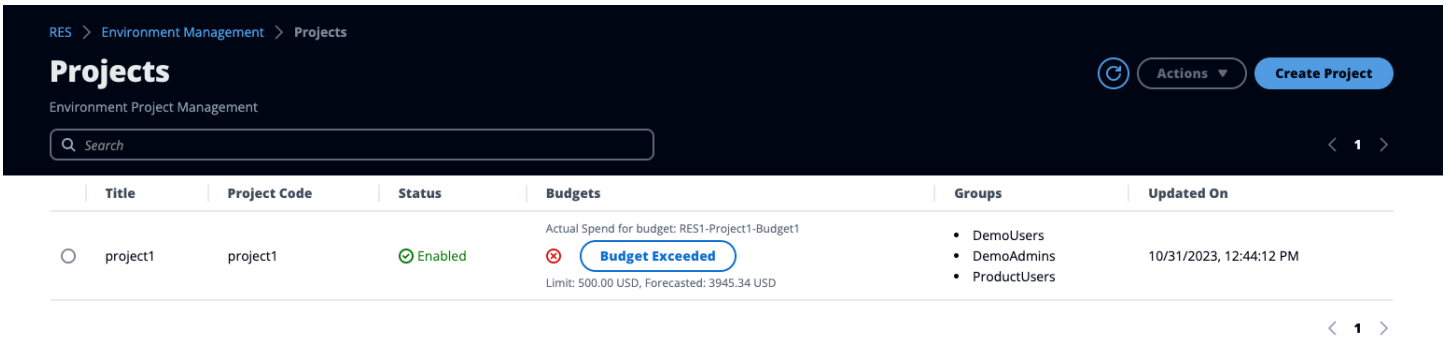
12. 値 で、プロジェクト名を選択します。
13. フィルターを適用 を選択して、プロジェクトフィルターを予算にアタッチします。

14. [次へ] をクリックします。


Budget scope Info
Add filtering and use advanced options to narrow the set of cost information tracked as part of this budget
Scope options
All AWS services (Recommended)
Track any cost incurred from any service for this account as part of the budget scope
Filter specific AWS cost dimensions
Select specific dimensions to budget against. For example, you can select the specific service "EC2" to budget against.
Filters Info
Remove all
Dimension
Tag
Tag
res:Project
Values
Filter tags by values
project1
Cancel
Apply filter
Add filter
Advanced options
Aggregate costs by
Unblended costs
Supported charge types
Upfront reservation fees
Recurring reservation charges
Other subscription costs
Taxes
Support charges
Discounts
Cancel
Previous
Next

15. (オプション) アラートのしきい値を追加します。
16. [次へ] をクリックします。
17. (オプション) アラートが設定されている場合は、アタッチアクションを使用して、アラートで目的のアクションを設定します。
18. [次へ] をクリックします。
19. 予算設定を確認し、「追加の予算パラメータ」で正しいタグが設定されていることを確認します。
20. [予算を作成] をクリックします。

予算が作成されたら、プロジェクトの予算を有効にできます。プロジェクトの予算を有効にするには、「」を参照してくださいthe section called “プロジェクトを編集する”。予算を超えると、仮想デスクトップの起動がブロックされます。デスクトップの起動中に予算を超えた場合、デスクトップは引き続き動作します。




予算を変更する必要がある場合は、コンソールに戻って予算額を編集します。RES 内で変更が有効になるまでに最大 15 分かかる場合があります。または、プロジェクトを編集して予算を無効にすることもできます。

# 製品を使用する

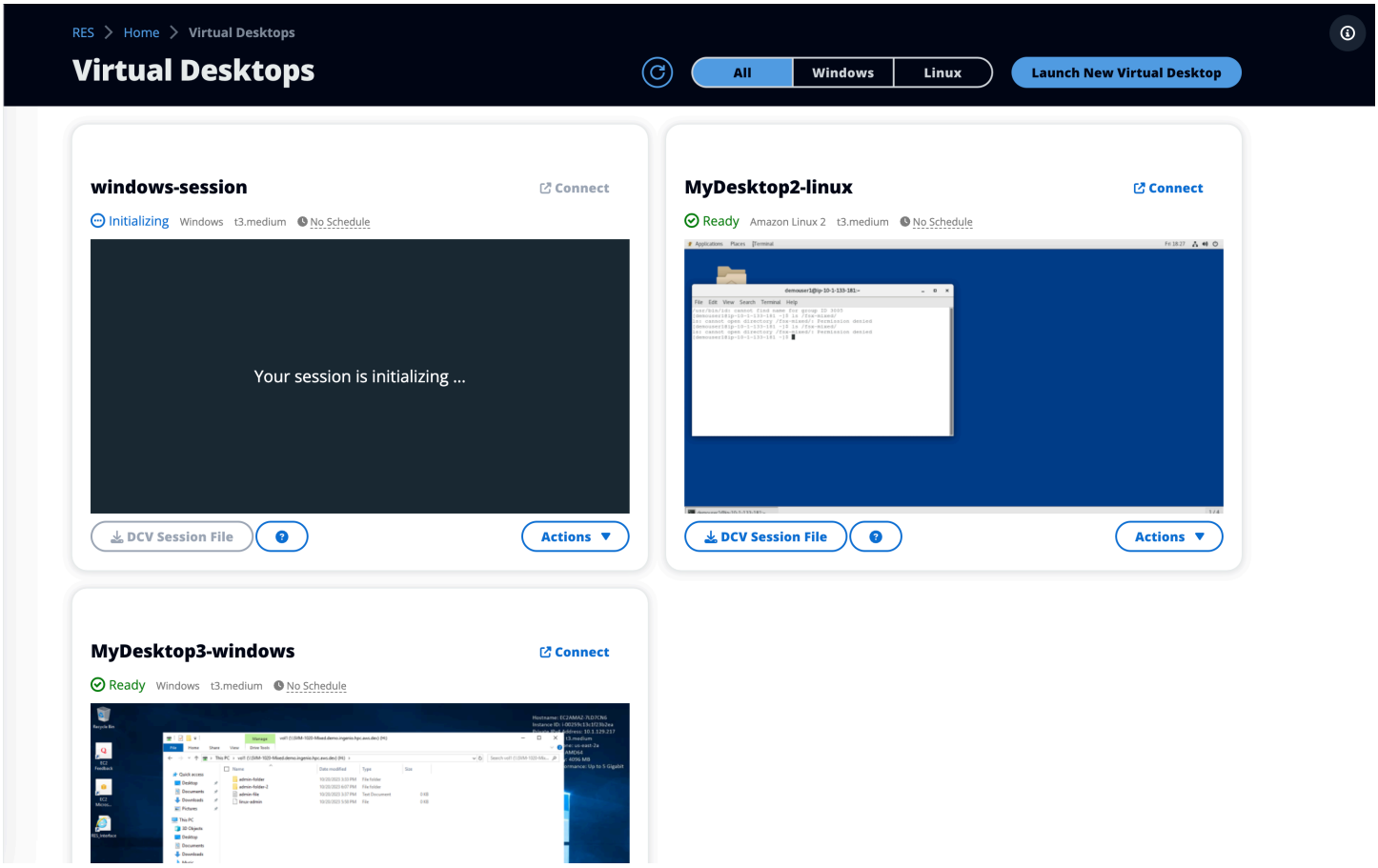
このセクションでは、仮想デスクトップを使用して他のユーザーとコラボレーションするためのガイダンスをユーザーに提供します。

トピック



## 仮想デスクトップ

仮想デスクトップインターフェイス (VDI) モジュールを使用すると、ユーザーは で Windows または Linux 仮想デスクトップを作成および管理できます AWS。ユーザーは、プリインストールおよび設定されたお気に入りのツールとアプリケーションで Amazon EC2 インスタンスを起動できます。



## 新しいデスクトップを起動する

1. メニューから、My Virtual Desktops を選択します。
2. 新しい仮想デスクトップの起動 を選択します。
3. 新しいデスクトップの詳細を入力します。
4. [送信] を選択します。

デスクトップ情報を含む新しいカードがすぐに表示され、デスクトップは 10～15 分以内に使用できるようになります。起動時間は、選択したイメージによって異なります。RES は GPU インスタンスを検出し、関連するドライバーをインストールします。

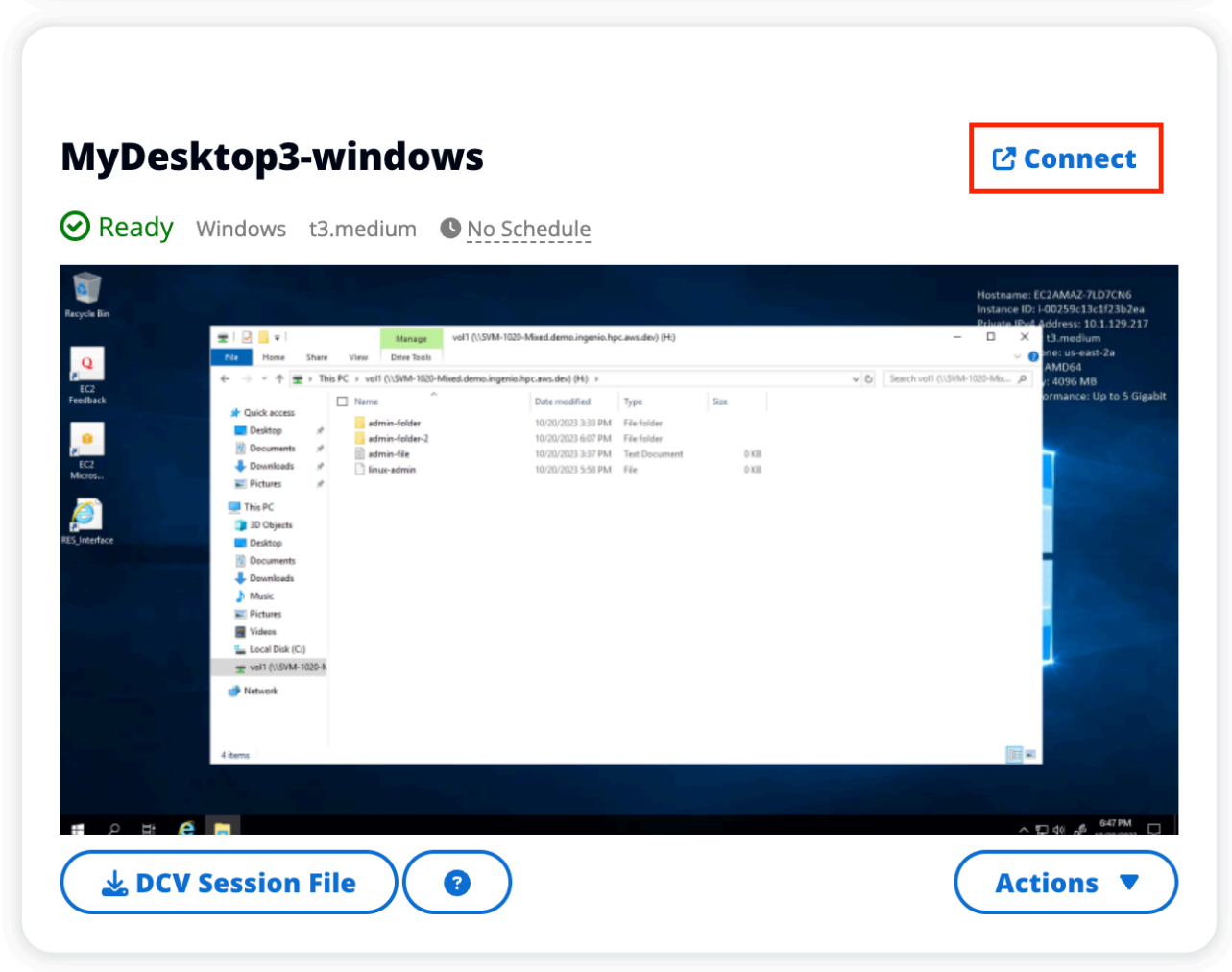
## デスクトップにアクセスする

仮想デスクトップにアクセスするには、デスクトップのカードを選択し、ウェブクライアントまたは DCV クライアントを使用して接続します。

Web connection

ウェブブラウザからデスクトップにアクセスするのが最も簡単な接続方法です。

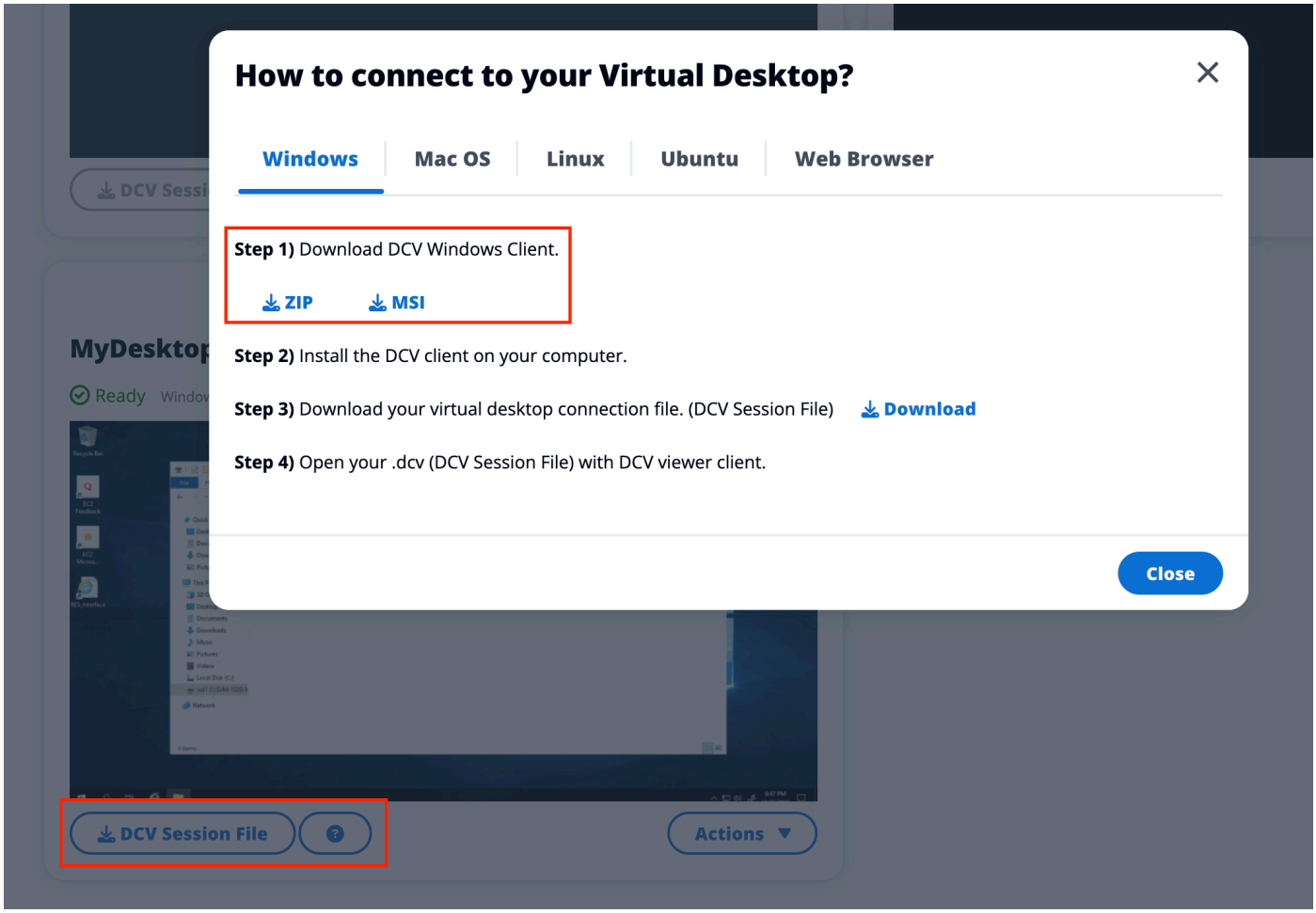
- Connect を選択するか、サムネイルを選択してブラウザから直接デスクトップにアクセスします。




DCV connection

DCV クライアント経由でデスクトップにアクセスすると、最高のパフォーマンスが得られます。DCV 経由で にアクセスするには：

1. DCV セッションファイルを選択して、.dcv ファイルをダウンロードします。DCV クライアントがシステムにインストールされている必要があります。
2. インストール手順については、？ アイコンを選択します。




# デスクトップの状態を制御する

デスクトップの状態を制御するには：

1. [アクション] を選択します。
2. 仮想デスクトップの状態 を選択します。次の 4 つの州から選択できます。

   - [Stop] (停止)

     停止したセッションではデータが失われることはなく、停止したセッションはいつでも再開できます。

   - 再起動

     現在のセッションを再起動します。

   - 終了

セッションを完全に終了します。エフェメラルストレージを使用している場合、セッションを終了するとデータが失われる可能性があります。終了する前に、データを RES ファイルシステムにバックアップする必要があります。

- 休止

  デスクトップの状態はメモリに保存されます。デスクトップを再起動すると、アプリケーションは再開されますが、リモート接続が失われる可能性があります。すべてのインスタンスが休止をサポートしているわけではなく、 オプションはインスタンスの作成時に有効になっている場合にのみ使用できます。インスタンスがこの状態をサポートしているかどうかを確認するには、「休止の前提条件」を参照してください。

## 仮想デスクトップの変更

仮想デスクトップのハードウェアを更新するか、セッション名を変更できます。

1. インスタンスサイズを変更する前に、セッションを停止する必要があります。

   a. [アクション] を選択します。

   b. 仮想デスクトップの状態 を選択します。

   c. [Stop] (停止) を選択します。

   > **Note**
   >
   > 休止したセッションのデスクトップサイズは更新できません。

2. デスクトップが停止したことを確認したら、アクション を選択し、セッションの更新 を選択します。
3. セッション名を変更するか、必要なデスクトップサイズを選択します。
4. [送信] を選択します。
5. インスタンスが更新されたら、デスクトップを再起動します。

   a. [アクション] を選択します。

   b. 仮想デスクトップの状態 を選択します。

   c. [開始] を選択します。

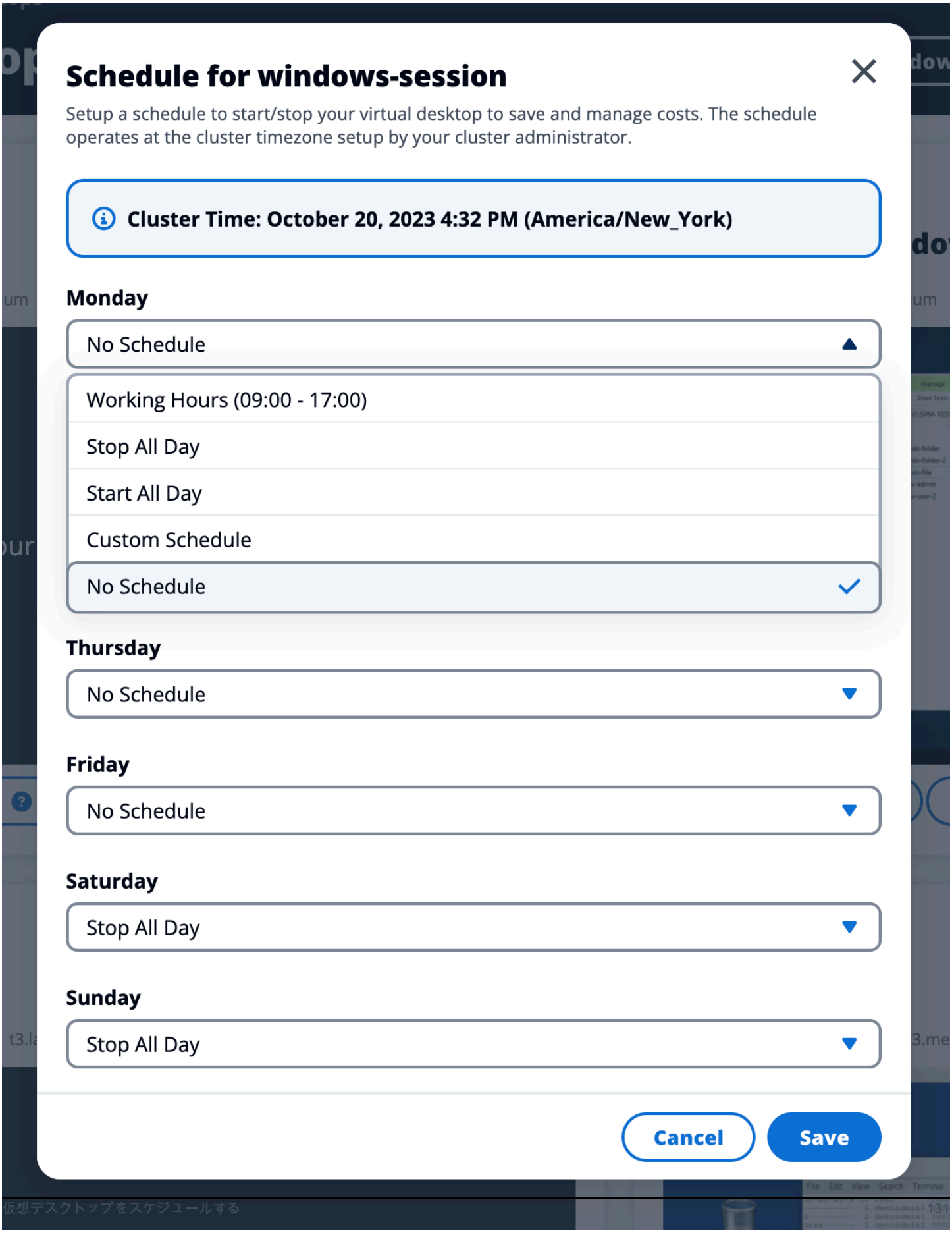
## セッション情報を取得する

1. [アクション] を選択します。
2. 情報を表示 を選択します。

## 仮想デスクトップをスケジュールする

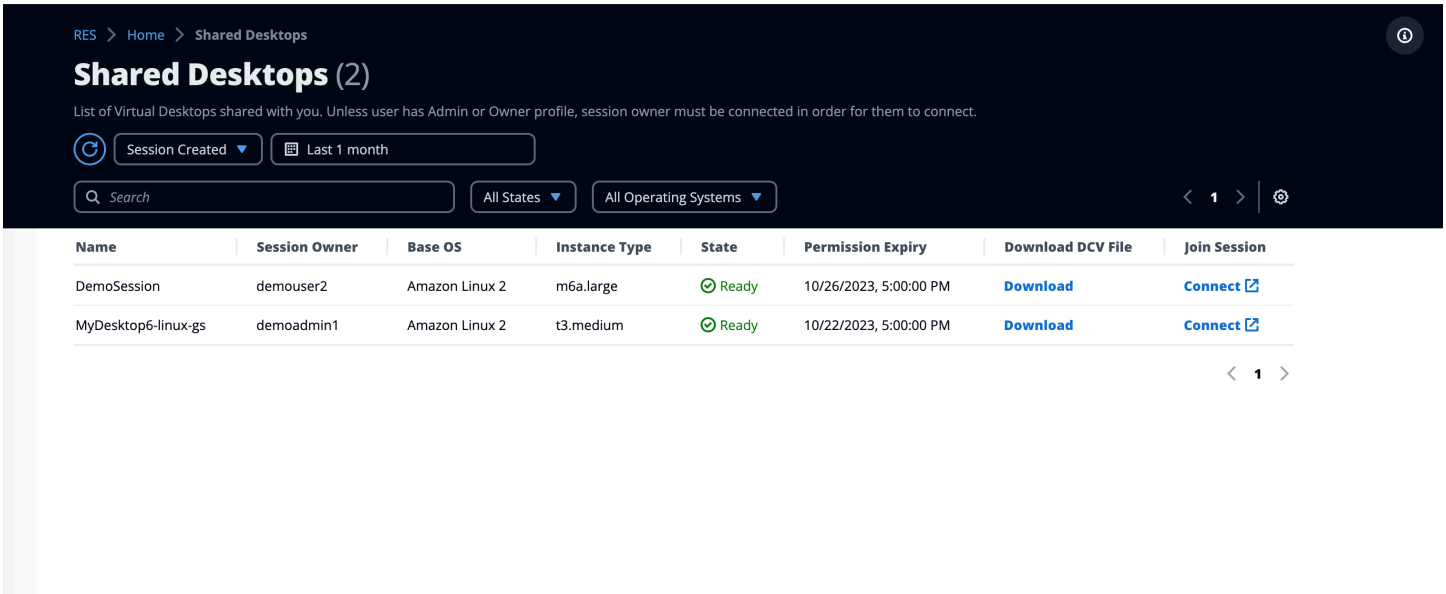
デフォルトでは、仮想デスクトップにはスケジュールがなく、セッションを停止または終了するまでアクティブのままになります。また、デスクトップはアイドル状態になると停止し、誤って停止しないようにします。アイドル状態は、アクティブな接続がなく、CPU 使用率が少なくとも 15 分間 15% 未満であることによって決まります。デスクトップを自動的に起動および停止するようにスケジュールを設定できます。

1. [アクション] を選択します。
2. [スケジュール] を選択します。
3. 日ごとにスケジュールを設定します。
4. [保存] を選択します。


Schedule for windows-session
Setup a schedule to start/stop your virtual desktop to save and manage costs. The schedule operates at the cluster timezone setup by your cluster administrator.
Cluster Time: October 20, 2023 4:32 PM (America/New_York)
Monday
No Schedule
Working Hours (09:00 - 17:00)
Stop All Day
Start All Day
Custom Schedule
No Schedule
Thursday
No Schedule
Friday
No Schedule
Saturday
Stop All Day
Sunday
Stop All Day
Cancel
Save

# 共有デスクトップ

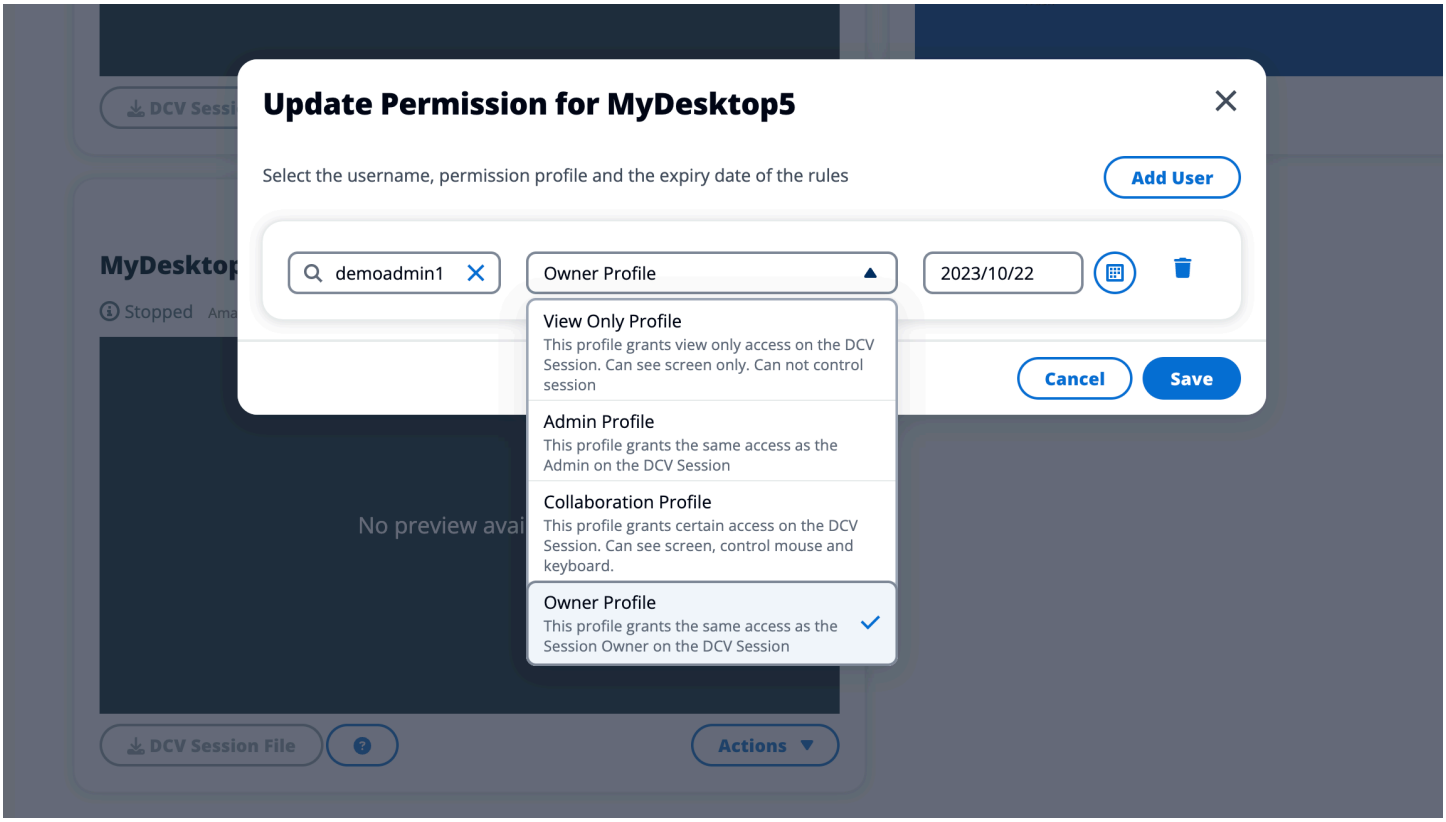
共有デスクトップでは、共有されているデスクトップを確認できます。デスクトップに接続するには、管理者または所有者でない限り、セッション所有者も接続されている必要があります。




セッションを共有するときに、共同作業者のアクセス許可を設定できます。例えば、コラボレーションしているチームメイトに読み取り専用アクセス権を付与できます。

## デスクトップを共有する

1. デスクトップセッションから、アクション を選択します。
2. セッションアクセス許可 を選択します。
3. ユーザーとアクセス許可レベルを選択します。有効期限を設定することもできます。
4. [保存] を選択します。



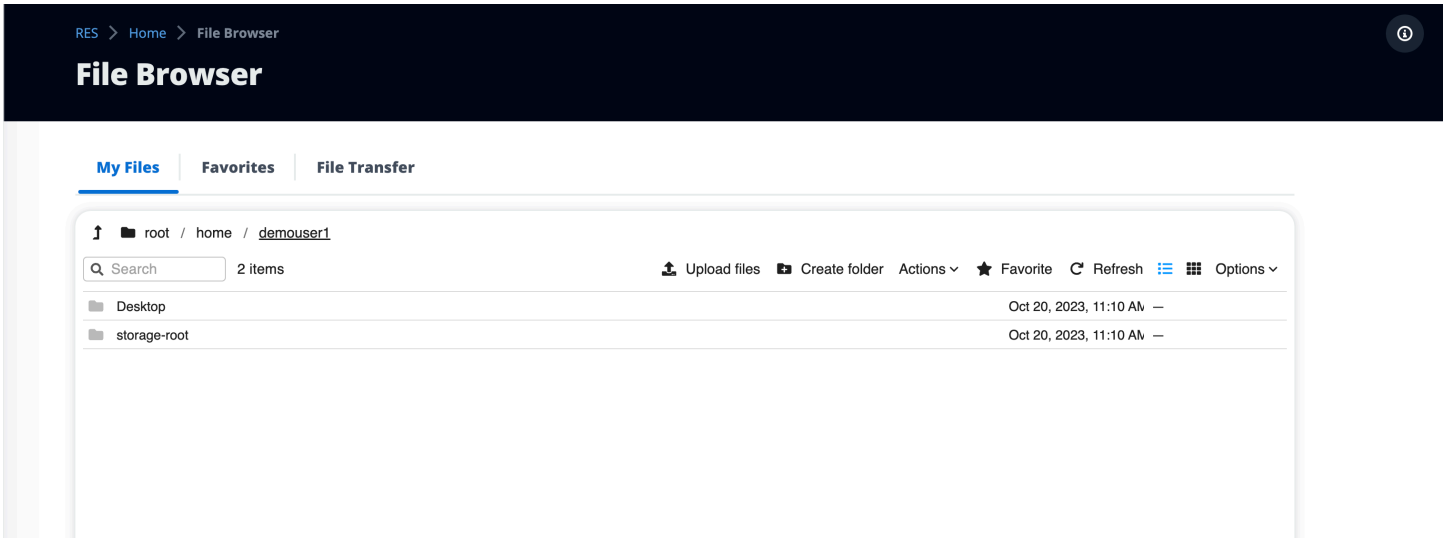

アクセス許可の詳細については、「」を参照してくださいthe section called “アクセス許可プロファイル”。

## 共有デスクトップにアクセスする

共有デスクトップから、共有されているデスクトップを表示し、インスタンスに接続できます。ウェブブラウザまたは DCV で参加できます。接続するには、「」の指示に従ってくださいthe section called “デスクトップにアクセスする”。

# ファイルブラウザ

ファイルブラウザを使用すると、ウェブポータルからファイルシステムにアクセスできます。基盤となるファイルシステム上でアクセス許可を持つ使用可能なすべてのファイルを管理できます。バックエンドストレージ (Amazon EFS ) は、すべての Linux ノードで使用できます。Linux および Windows ノードでは、FSx for ONTAP を使用できます。仮想デスクトップ上のファイルの更新は、ターミナルまたはウェブベースのファイルブラウザを介したファイルの更新と同じです。




## ファイルのアップロード (複数可 )

1. [ファイルをアップロード] を選択します。
2. ファイルを削除するか、アップロードするファイルを参照します。
3. アップロード (n) ファイル を選択します。

## ファイルを削除する (複数可 )

1. 削除するファイル (複数可) を選択します。
2. [アクション] を選択します。
3. ファイルの削除 を選択します。

または、任意のファイルまたはフォルダを右クリックし、ファイルの削除 を選択することもできます。

## お気に入りを管理する

重要なファイルやフォルダを固定するには、それらをお気に入りに追加します。

1. ファイルまたはフォルダを選択します。
2. お気に入り を選択します。

または、任意のファイルまたはフォルダを右クリックして、お気に入り を選択することもできます。

> **Note**
>
> お気に入りはローカルブラウザに保存されます。ブラウザを変更したり、キャッシュをクリアしたりする場合は、お気に入りを再ピン留めする必要があります。

## ファイルの編集

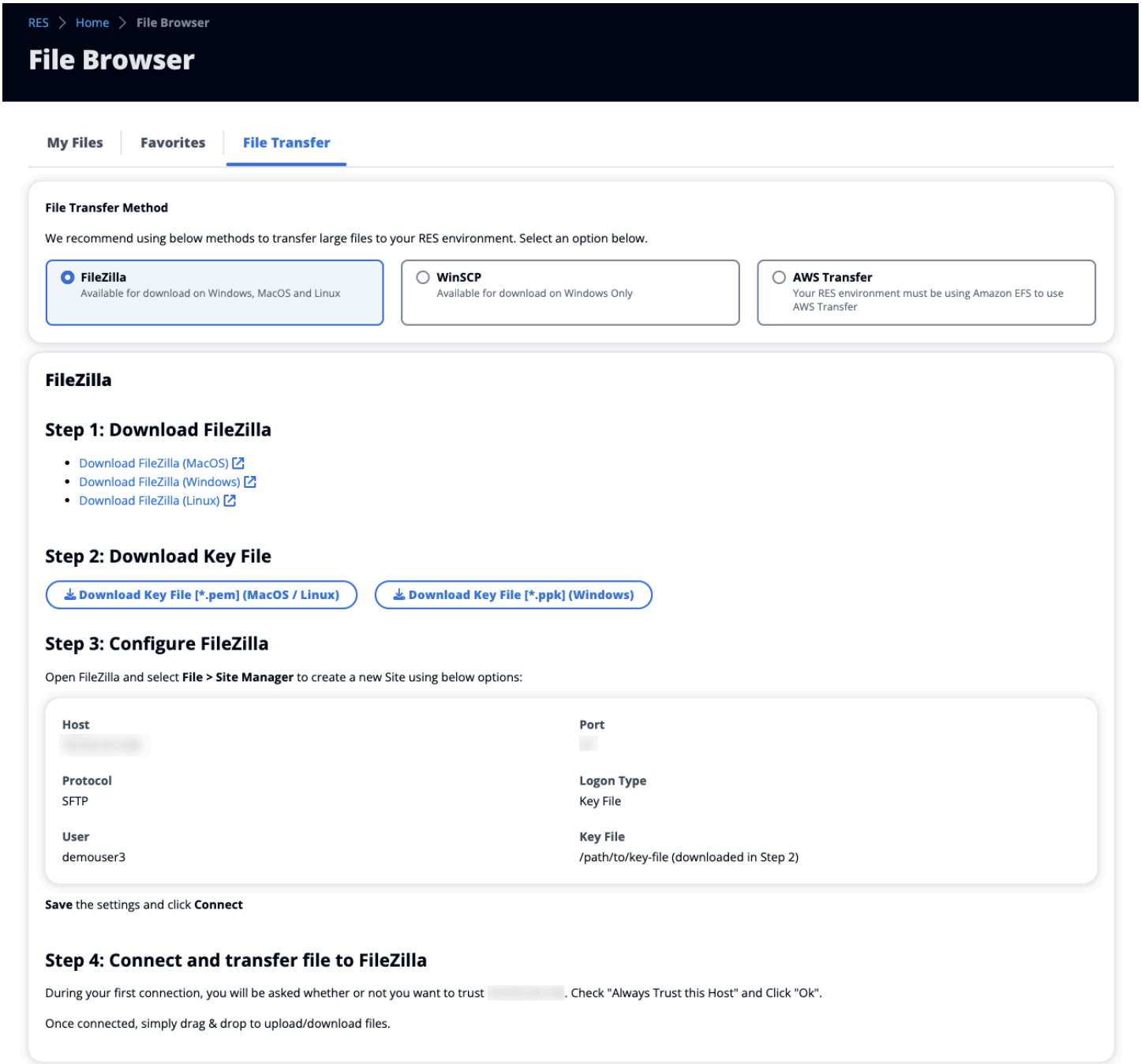
ウェブポータル内のテキストベースのファイルのコンテンツを編集できます。

1. 更新するファイルを選択します。モーダルが開き、ファイルの内容が表示されます。
2. 更新を行い、保存 を選択します。

## ファイルの転送

ファイル転送を使用して、外部ファイル転送アプリケーションを使用してファイルを転送します。次のアプリケーションから選択し、画面の指示に従ってファイルを転送できます。

- FileZilla (Windows、MacOS、Linux)
- WinSCP (Windows)
- AWS Transfer for FTP (Amazon EFS )




# SSH アクセス

SSH を使用して踏み台ホストにアクセスするには：

1. RES メニューから SSH アクセス を選択します。
2. アクセスに SSH または PuTTY を使用するには、画面の指示に従ってください。

# トラブルシューティング

このドキュメントには、システムをモニタリングする方法と、発生する可能性のある特定の問題のトラブルシューティング方法に関する情報が含まれています。問題の解決策が見つからない場合は、で追加のトラブルシューティングトピック GitHubが見つかる場合があります。

トピック

- インストールの問題
- ID 管理の問題

## インストールの問題

トピック

- AWS CloudFormation スタックはWaitCondition 「失敗したメッセージを受信しました」というメッセージで を作成できません。エラー：ステータスTaskFailed。"
- AWS CloudFormation スタックが正常に作成された後に E メール通知が受信されない
- インスタンスのサイクルまたは vdc コントローラーが失敗状態
- 依存オブジェクトエラーにより環境 CloudFormation スタックの削除に失敗する
- 環境の作成中に CIDR ブロックパラメータでエラーが発生しました
- CloudFormation 環境作成中の スタック作成の失敗
- 外部リソース (デモ) スタックの作成が AdDomainAdminNode CREATE_FAILED で失敗する

### AWS CloudFormation スタックはWaitCondition 「失敗したメッセージを受信しました」というメッセージで を作成できません。エラー：ステータスTaskFailed。"

問題を特定するには、 という名前の Amazon CloudWatch ロググループを調べます<stack-name>-InstallerTasksCreateTaskDefCreateContainerLogGroup<nonce>-<nonce>。同じ名前のロググループが複数ある場合は、最初に使用可能なロググループを調べます。ログ内のエラーメッセージには、問題に関する詳細情報が表示されます。

> **Note**
>
> パラメータ値にスペースがないことを確認します。

## AWS CloudFormation スタックが正常に作成された後に E メール通知が受信されない

が正常に AWS CloudFormation 作成された後に E メールの招待を受信しなかった場合は、以下を確認します。

1. E メールアドレスパラメータが正しく入力されたことを確認します。

   E メールアドレスが正しくないか、アクセスできない場合は、Research and Engineering Studio 環境を削除して再デプロイします。

2. Amazon EC2 コンソールでサイクルインスタンスの証拠を確認します。

   <envname> プレフィックス が付いた Amazon EC2 インスタンスが終了済みとして表示され、新しいインスタンスに置き換えられた場合、ネットワークまたは Active Directory の設定に問題がある可能性があります。

3. High Performance Compute AWS レシピをデプロイして外部リソースを作成する場合は、VPC、プライベートサブネットとパブリックサブネット、およびその他の選択したパラメータがスタックによって作成されたことを確認します。

   パラメータのいずれかが正しくない場合は、RES 環境を削除して再デプロイする必要がある場合があります。詳細については、「製品をアンインストールします。」を参照してください。

4. 独自の外部リソースを使用して製品をデプロイした場合は、ネットワークと Active Directory が想定された設定と一致していることを確認します。

   インフラストラクチャインスタンスが Active Directory に正常に参加したことを確認することが重要です。のステップを試the section called “インスタンスのサイクルまたは vdc コントローラーが失敗状態”して問題を解決してください。

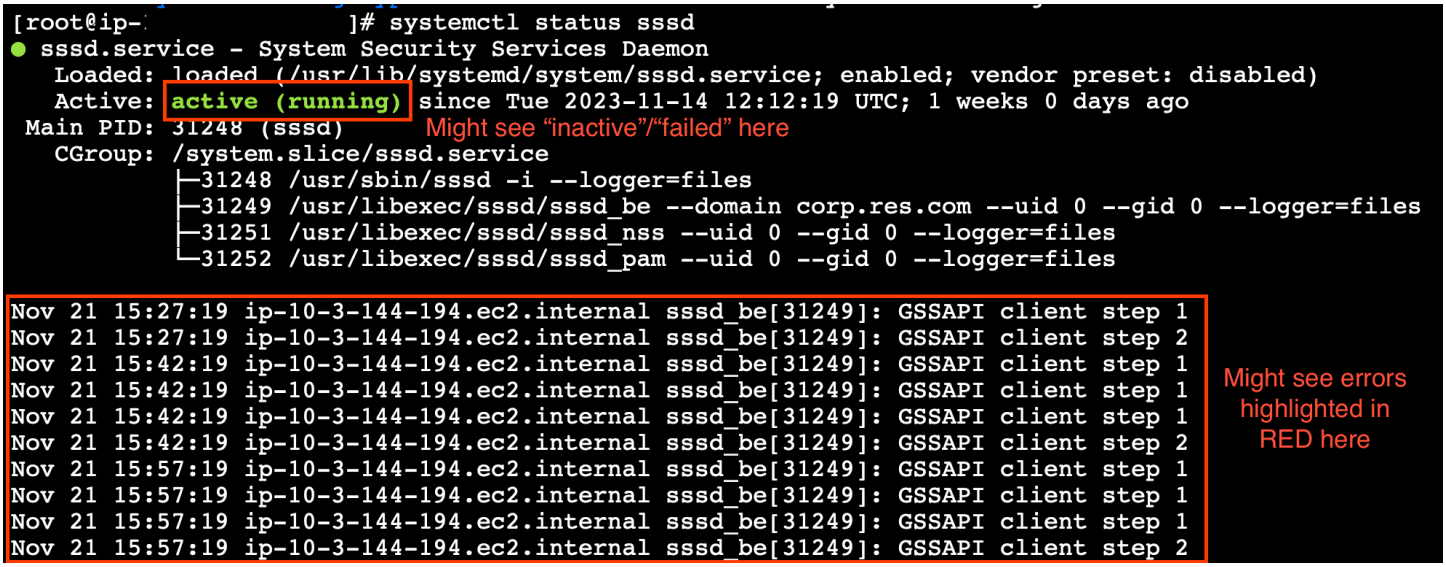
## インスタンスのサイクルまたは vdc コントローラーが失敗状態

この問題の最も可能性の高い原因は、リソースが Active Directory に接続または参加できないことです。

問題を検証するには：

1. コマンドラインから、vdc-controller の実行中のインスタンスで SSM とのセッションを開始します。
2. `sudo su -` を実行します。
3. `systemctl status sssd` を実行します。

ステータスが非アクティブ、失敗、またはログにエラーが表示される場合、インスタンスは Active Directory に参加できませんでした。




SSM エラーログ

問題を解決するには：

- 同じコマンドラインインスタンスから、 `cat /root/bootstrap/logs/userdata.log`を実行してログを調査します。

この問題は、考えられる 3 つの根本原因のいずれかである可能性があります。

根本原因 1: 不正な ldap 接続の詳細が入力されました

ログを見直します。以下が複数回繰り返される場合、インスタンスは Active Directory に参加できませんでした。

```
+ local AD_AUTHORIZATION_ENTRY=
```

```
+ [[ -z '' ]]
+ [[ 0 -le 180 ]]
+ local SLEEP_TIME=34
+ log_info '(0 of 180) waiting for AD authorization, retrying in 34 seconds ...'
++ date '+%Y-%m-%d %H:%M:%S,%3N'
+ echo '[2024-01-16 22:02:19,802] [INFO] (0 of 180) waiting for AD authorization,
 retrying in 34 seconds ...'
[2024-01-16 22:02:19,802] [INFO] (0 of 180) waiting for AD authorization, retrying in
 34 seconds ...
+ sleep 34
+ (( ATTEMPT_COUNT++ ))
```

1. RES スタックの作成中に、以下のパラメータ値が正しく入力されたことを確認します。
   - directoryservice.ldap_connection_uri
   - directoryservice.ldap_base
   - directoryservice.users.ou
   - directoryservice.groups.ou
   - directoryservice.sudoers.ou
   - directoryservice.computers.ou
   - directoryservice.name
2. DynamoDB テーブルの誤った値を更新します。テーブルは、DynamoDB コンソールのテーブルにあります。テーブル名は である必要があります**[stack name]**.cluster-settings。
3. テーブルを更新したら、現在環境インスタンスを実行している cluster-manager と vdc-controller を削除します。Auto Scaling は、DynamoDB テーブルの最新の値を使用して新しいインスタンスを起動します。

根本原因 2: 入力された ServiceAccount ユーザー名が正しくない

ログが を返す場合`Insufficient permissions to modify computer account`、スタックの作成時に入力した ServiceAccount 名前が正しくない可能性があります。

1. AWS コンソールから Secrets Manager を開きます。
2. `directoryserviceServiceAccountUsername` を検索します。シークレットは である必要があります**[stack name]**-directoryservice-ServiceAccountUsername。
3. シークレットを開いて詳細ページを表示します。シークレット値 で、シークレット値の取得 を選択し、プレーンテキスト を選択します。

4. 値が更新された場合は、現在実行中の環境の cluster-manager インスタンスと vdc-controller インスタンスを削除します。Auto Scaling は、Secrets Manager の最新の値を使用して新しいインスタンスを起動します。

根本原因 3: 入力された ServiceAccount パスワードが正しくない

ログに と表示されている場合`Invalid credentials`、スタックの作成時に入力した ServiceAccount パスワードが正しくない可能性があります。

1. AWS コンソールから Secrets Manager を開きます。
2. `directoryserviceServiceAccountPassword` を検索します。シークレットは である必要があります`[stack name]-directoryservice-ServiceAccountPassword`。
3. シークレットを開いて詳細ページを表示します。シークレット値 で、シークレット値の取得 を選択し、プレーンテキスト を選択します。
4. パスワードを忘れた場合、または入力したパスワードが正しいかどうかわからない場合は、Active Directory と Secrets Manager でパスワードをリセットできます。

   a. でパスワードをリセットするには AWS Managed Microsoft AD：

      i. AWS コンソールを開き、 に移動します AWS Directory Service。

      ii. RES ディレクトリのディレクトリ ID を選択し、アクション を選択します。

      iii. [Reset user password] (ユーザーパスワードをリセットする) を選択します。

      iv. ServiceAccount ユーザー名を入力します。

      v. 新しいパスワードを入力し、パスワードのリセット を選択します。

   b. Secrets Manager でパスワードをリセットするには：

      i. AWS コンソールを開き、Secrets Manager に移動します。

      ii. `directoryserviceServiceAccountPassword` を検索します。シークレットは である必要があります`[stack name]-directoryservice-ServiceAccountPassword`。

      iii. シークレットを開いて詳細ページを表示します。シークレット値 で、シークレット値の取得 を選択し、プレーンテキスト を選択します。

      iv. [編集] を選択します。

      v. ServiceAccount ユーザーの新しいパスワードを設定し、保存 を選択します。

5. 値が更新された場合は、現在実行中の環境の cluster-manager インスタンスと vdc-controller インスタンスを削除します。Auto Scaling は、最新の値を使用して新しいインスタンスを起動します。

## 依存オブジェクトエラーにより環境 CloudFormation スタックの削除に失敗する

などの依存オブジェクトエラーが原因で**[env-name]**-vdc CloudFormationスタックの削除が失敗した場合vdcdcvhostsecuritygroup、コンソールを使用して AWS RES が作成したサブネットまたはセキュリティグループに起動された Amazon EC2 インスタンスが原因である可能性があります。

この問題を解決するには、この方法で起動されたすべての Amazon EC2 インスタンスを検索して終了します。その後、環境の削除を再開できます。

## 環境の作成中に CIDR ブロックパラメータでエラーが発生しました

環境を作成すると、レスポンスステータスが [FAILED] の CIDR ブロックパラメータにエラーが表示されます。

エラーの例：

```
Failed to update cluster prefix list:
        An error occurred (InvalidParameterValue) when calling the
 ModifyManagedPrefixList operation:
        The specified CIDR (52.94.133.132/24) is not valid. For example, specify a CIDR
 in the following form: 10.0.0.0/16.
```

この問題を解決するために想定される形式は x.x.x.0/24 または x.x.x.0/32 です。

## CloudFormation 環境作成中の スタック作成の失敗

環境の作成には、一連のリソース作成オペレーションが含まれます。リージョンによっては、キャパシティの問題が発生し、 CloudFormation スタックの作成が失敗することがあります。

この場合、環境を削除し、作成を再試行してください。または、別のリージョンで作成を再試行することもできます。

# 外部リソース (デモ) スタックの作成が AdDomainAdminNode CREATE_FAILED で失敗する

デモ環境スタックの作成が次のエラーで失敗した場合、インスタンスの起動後のプロビジョニング中に Amazon EC2 のパッチ適用が予期せず発生したことが原因である可能性があります。

```
AdDomainAdminNode CREATE_FAILED  Failed to receive 1 resource signal(s) within the
 specified duration
```

失敗の原因を特定するには：

1. SSM ステートマネージャーで、パッチ適用が設定されているかどうか、およびすべてのインスタンスに対して設定されているかどうかを確認します。
2. SSM RunCommand/オートメーションの実行履歴で、パッチ適用関連の SSM ドキュメントの実行がインスタンスの起動と一致するかどうかを確認します。
3. 環境の Amazon EC2 インスタンスのログファイルで、ローカルインスタンスのログ記録を確認して、プロビジョニング中にインスタンスが再起動されたかどうかを確認します。

パッチ適用が原因で問題が発生した場合は、起動から少なくとも 15 分後に RES インスタンスのパッチ適用を遅らせます。

# ID 管理の問題

シングルサインオン (SSO) と ID 管理のほとんどの問題は、設定ミスが原因で発生します。SSO 設定の設定については、以下を参照してください。

- the section called “IAM Identity Center で SSO を設定する”
- the section called “シングルサインオン (SSO) 用の ID プロバイダーの設定”

ID 管理に関連するその他の問題のトラブルシューティングについては、以下のトラブルシューティングトピックを参照してください。

トピック

- 環境にログインすると、すぐにログインページに戻ります。
- ログインしようとしたときに「ユーザーが見つかりません」というエラーが表示される

- ユーザーが Active Directory に追加されましたが、RES に がありません
- セッションの作成時にユーザーが使用できない
- CloudWatch クラスターマネージャーログのサイズ制限超過エラー

## 環境にログインすると、すぐにログインページに戻ります。

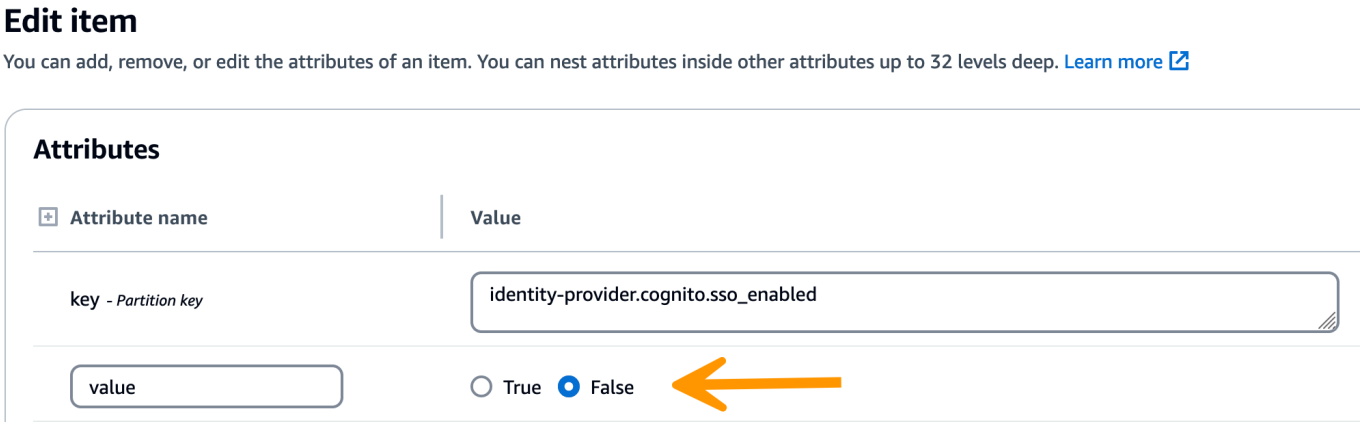
この問題は、 SSO 統合の設定が間違っている場合に発生します。問題を特定するには、コントローラーインスタンスのログを確認し、設定でエラーがないか確認します。

ログを確認するには：

1. https://console.aws.amazon.com/cloudwatch/ で CloudWatch コンソールを開きます。
2. ロググループ から、 という名前のグループを見つけます/*<environment-name>*/cluster-manager。
3. ロググループを開いて、ログストリーム内のエラーを検索します。

設定を確認するには：

1. DynamoDB コンソール (https://console.aws.amazon.com/dynamodb/) を開きます。
2. テーブル から、 という名前のテーブルを見つけます*<environment-name>*.cluster-settings。
3. テーブルを開き、テーブル項目の探索を選択します。
4. フィルターセクションを展開し、以下の変数を入力します。
   - 属性名 – キー
   - 条件 - を含む
   - 値 – sso
5. [実行] を選択します。
6. 返された文字列で、 SSO 設定値が正しいことを確認します。正しくない場合は、sso_enabled キーの値を False に変更します。




7. RES ユーザーインターフェイスに戻り、 SSO を再設定します。

## ログインしようとしたときに「ユーザーが見つかりません」というエラーが表示される

RES インターフェイスにログインするときに「User not found」というエラーが表示される場合、そのユーザーは Active Directory に存在しますが、RES には存在しません。最近 AD にユーザーを追加した場合、そのユーザーは RES と同期されない可能性があります。RES は 1 時間ごとに同期するため、次の同期後にユーザーが追加されたことを待機して確認する必要がある場合があります。すぐに同期するには、「」のステップに従いますthe section called “ユーザーが Active Directory に追加されましたが、RES に がありません”。

ユーザーが RES に存在する場合：

1. 属性マッピングが正しく設定されていることを確認します。詳細については、「the section called “シングルサインオン (SSO) 用の ID プロバイダーの設定”」を参照してください。
2. SAML 件名と SAML E メールの両方がユーザーの E メールアドレスにマッピングされていることを確認します。

## ユーザーが Active Directory に追加されましたが、RES に がありません

ユーザーを Active Directory に追加したが、RES にない場合、AD 同期をトリガーする必要があります。AD 同期は、AD エントリを RES 環境にインポートするために Lambda 関数によって 1 時間ごとに実行されます。場合によっては、新しいユーザーまたはグループを追加した後に次の同期プロセスが実行されるまでに遅延が生じることがあります。Amazon Simple Queue Service から手動で同期を開始できます。

同期プロセスを手動で開始します。

1. Amazon SQS コンソール (https://console.aws.amazon.com/sqs/) を開きます。
2. キュー から、 を選択します<environment-name>-cluster-manager-tasks.fifo。
3. [メッセージの送信と受信] を選択します。
4. メッセージ本文 には、次のように入力します。

   **{ "name": "adsync.sync-from-ad", "payload": {} }**
5. メッセージグループ ID には、次のように入力します。 **adsync.sync-from-ad**
6. メッセージ重複排除 ID には、ランダムな英数字の文字列を入力します。このエントリは、5 分以内にすべての呼び出しと異なる必要があります。そうしないと、リクエストは無視されます。

## セッションの作成時にユーザーが使用できない

セッションを作成する管理者が、セッションの作成時に Active Directory に属しているユーザーが使用できない場合は、ユーザーが初めてログインする必要がある場合があります。セッションは、アクティブなユーザーに対してのみ作成できます。アクティブなユーザーは、少なくとも 1 回は環境にログインする必要があります。

## CloudWatch クラスターマネージャーログのサイズ制限超過エラー

```
2023-10-31T18:03:12.942-07:00 ldap.SIZELIMIT_EXCEEDED: {'msgtype': 100, 'msgid': 11,
 'result': 4, 'desc': 'Size limit exceeded', 'ctrls': []}
```

CloudWatch クラスターマネージャーログにこのエラーが表示された場合、ldap 検索で返されるユーザーレコードが多すぎる可能性があります。この問題を解決するには、IDP の ldap 検索結果の制限を増やします。

# 注意

各 Amazon EC2 インスタンスには、管理目的で 2 つのリモートデスクトップサービス (ターミナルサービス) ライセンスが付属しています。この情報は、管理者にこれらのライセンスをプロビジョニングするのに役立ちます。を使用することもできます。これによりAWS Systems Manager Session Manager、RDP や RDP ライセンスを必要とせずに Amazon EC2 インスタンスにリモート接続できます。追加のリモートデスクトップサービスライセンスが必要な場合は、Microsoft または Microsoft ライセンスリセラーからリモートデスクトップユーザー CALs を購入する必要があります。アクティブなソフトウェアアシュアランスを持つリモートデスクトップユーザーの CALs にはライセンスモビリティのメリットがあり、デフォルトの (共有) テナント環境に移行 AWS できます。ソフトウェアアシュアランスまたはライセンスモビリティのメリットなしでライセンスを持ち込む方法については、 よくある質問のこのセクションを参照してください。

お客様は、本書に記載されている情報を独自に評価する責任を負うものとします。このドキュメント： (a) は情報提供のみを目的としています。 (b) は、現在の製品提供とプラクティスを表 AWS し、 予告なしに変更される可能性があるもの。 および (c) は、 AWS およびその関連会社からいかなる約束または保証も生じません。 サプライヤーまたは licensors. AWS products またはサービスは、保証なしで「現状有姿」で提供されます。 表現、 またはあらゆる種類の条件、 明示的か黙示的かにかかわらず、顧客に対する AWS 責任と責任は AWS 契約によって管理されます。 このドキュメントは の一部ではありません。 も変更されません。 AWS とその顧客との間の契約。

の Research and Engineering Studio AWS は、「Apache Software Foundation」で入手可能な Apache License Version 2.0 の条項に基づいてライセンスされています。

# リビジョン

詳細については、リポジトリ内の CHANGELOG.md ファイルを参照してください。 GitHub

| 日付 | 変更 |
| --- | --- |
| 2023 年 11 月 | 初回リリース |
| 2023 年 12 月 | GovCloud ルート案内とテンプレートが追加されました。 |
| 2024 年 1 月 | リリースバージョン 2024.01 |
| 2024 年 2 月 | リリースバージョン 2024.01.01 — デプロイテンプレートが更新されました |
| 2024 年 3 月 | その他のトラブルシューティングトピック、CloudWatch ログの保存、マイナーバージョンのアンインストール |
| 2024 年 4 月 | リリースバージョン 2024.04 — RES対応 AMI とプロジェクト開始テンプレート |

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。